A FREQUENCY SYNTHESIZER WITH FREQUENCY DIVIDER AND FREQUENCY MULTIPLIER FOR SPUR REDUCTION

By

SINISA MILICEVIC

A Ph.D. thesis
presented to Carleton University
in fulfilment of the
thesis requirement for the degree of
DOCTOR OF PHILOSOPHY
in
ELECTRICAL ENGINEERING

Ottawa, Ontario, Canada

Your file Votre référence
ISBN: 978-0-494-43900-5
Our file Notre référence
ISBN: 978-0-494-43900-5

Abstract

A novel frequency synthesizer, incorporating subsystem of frequency dividers and frequency multipliers in the feedback loop, is presented in this thesis. The frequency dividers and the frequency multipliers are programmable allowing programmable frequency resolution, smaller than the frequency of the reference signal.

There are many possible implementations of the proposed system architecture. This thesis discusses three implementations denoted as divide-multiply implementation, multiply-divide implementation, and divide-multiply-divide implementation. This thesis is written with an emphasis put on the divide-multiply implementation, while the multiply-divide implementation and the divide-multiply-divide implementation are discussed at the end of the thesis as a separate appendix.

The divide-multiply implementation is illustrated in a 0.13μm CMOS technology through a 10GHz frequency synthesizer, operating from 1V supply, with a reference signal of 20MHz and channel spacing of 500kHz. Using differential cells with resistor tails allowed operation at 10GHz with a reduced power supply. The custom differential cells are demonstrated as building components of the frequency dividers and the phase frequency detector (PFD).

The primary goal of this thesis work is to reduce the spurious tones that can appear

on the output signal of the frequency synthesizer. To accomplish this objective, a PFD with a linear tracking characteristic and a novel charge pump (CP) are used with the divide-multiply implementation. The advantages and the disadvantages of the three implementations are discussed and compared to the state-of-the-art frequency synthesizers found in the literature.

The theoretical contribution of this thesis includes the transient and the phase noise analysis of the new system. A formula to predict the frequency of oscillation of a cross-coupled voltage-controlled oscillator (VCO) is also derived. In addition to the aforementioned topics, a mathematical model for the voltage control signal used to tune a VCO is derived for the transient lock-in process and included as an appendix at the end of the thesis.

Acknowledgements

I would like to express my gratitude to my supervisor, Dr. Leonard MacEachern, for his support and assistance during my studies.

Special gratitude to my wife, Simka, and daughter, Daniela, for their love, support, encouragement and motivation which were my driving force throughout my studies at Carleton University and internships in California.

I would like to thank my parents along with my sisters, and my in-law family for their unlimited support and encouragements.

Finally, I would like to say thank you to all of my friends that I have met at Carleton University and especially to the members of the OMIC group for their camaraderie over the course of my studies.

to Simka and Daniela

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ω_n	Natural frequency
$\phi_{\mathtt{LO}}$	Phase of the LO signal
$\phi_{\mathtt{Ref}}$	Phase of the reference signal
$\phi_{\mathtt{VCO}}$	Phase of the VCO output signal
τ	Timing constant
$\mathtt{A}_{\mathtt{LO}}$	Amplitude of the LO signal
$\mathtt{A}_{\mathtt{Ref}}$	Amplitude of the reference signal
$\mathtt{A}_{\mathtt{VCO}}$	Amplitude of the output signal from the VCO
ξ	Damping constant
i_n	Output noise current
$v_{\mathtt{LO}}(\mathtt{t})$	LO signal in time domain
$v_{\mathtt{Ref}}(\mathtt{t})$	Reference signal in time domain
LO	Signal from a local oscillator
AHDL	Analog Hardware Description Language
B	Bulk
BiCMOS	Bipolar Complementary Metal Oxide Semiconductor
C	Capacitor
CML	Current Mode Logic
CMOS	Complementary Metal Oxide Semiconductor
CP	Charge Pump
D	Drain

DAC	Digital to Analog Converter
DDS	Direct Digital Synthesis
DLL	Delay Locked Loop
DUT	Device Under Test
FF	Flip-Flop
FOM	Figure of Merit
G	Gate
I_{CP}	Charge Pump Current
K_{phase}	Gain of the phase frequency detector
K_{vco}	Gain of the voltage-controlled oscillator
KVL	Kirchhoff's Voltage Law
L	Inductor
LBW	Loop Bandwidth
LC	Inductor Capacitor
LPF	Low Pass Filter
M	Multiplication ratio of the frequency multiplier
N	Division ratio of the frequency divider
NG	Not Given
PCB	Printed Circuit Board
PFD	Phase Frequency Detector
PLL	Phase Locked Loop
PN	Phase Noise
R	Resistor
RF	Radio Frequency
S	Source
SFDR	Spurious Free Dynamic Range
VCO	Voltage Controlled Oscillator

Chapter 1

Introduction

This thesis discusses a novel system and its building blocks for a phase locked loop (PLL) based frequency synthesizers. Additional readings about this topic can be found in [1–3].

1.1 Motivation

Frequency synthesizers are building components of the communication systems. For example, Figure 1.1 shows an implementation of a particular type of radio transceiver[1] [4]. In this particular example, each of the blocks denoted as "RF Oscillator" and "IF Oscillator" represent a frequency synthesizer.

Based on a number of research papers prevalent in the literature, the frequency synthesizer implementing a $\Delta\Sigma$ modulator is attractive for many researchers. A primary disadvantage of this type of a fractional frequency synthesizer is the appearance of spurious tones in the signal output spectrum. Figure 1.2 illustrates the appearance of the fractional spurs of the $\Sigma\Delta$ based frequency synthesizers as shown in [5]. These spurious tones limit the performance of the frequency synthesizer. For example, in

[1]The reader should understand that there are many variations of transceivers and the one shown in Figure 1.1 is used only as an example to illustrate a possible placement of the frequency synthesizers.

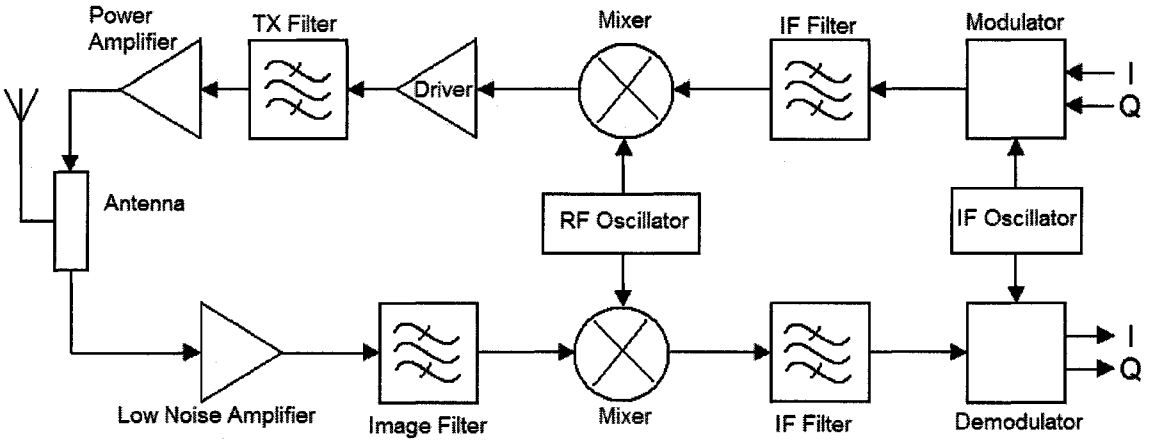


Figure 1.1: RF section of a transceiver wireless system [4].

the receiver side, the spurious tones can mix with the undesired signal and produce noise in the channel of the interest. This can reduce the sensitivity and the selectivity of the receiver. Similarly, in the transmitter side, the spurious tones can mix with the modulated baseband signal and produce undesired spectral emissions. The direct consequence is the reduced modulation accuracy and the increased channel interference.

These unwanted tones appear because the voltage controlled oscillator output frequency is divided by a time-varying fractional number [4] within the synthesizer loop. While techniques have been developed to reduce the output spurs (e.g. higher order $\Delta\Sigma$ modulators may whiten and shape the spurious energy to high-frequency noise, which can be removed by a low-pass loop filter [6]), these techniques typically increase chip complexity, power consumption, and layout area requirements [7, 8].

To reduce spurious tones is the primary motivation for this Ph.D. thesis work. The thesis proposes a system architecture and custom blocks in order to design a frequency synthesizer with reduced spurious tones.

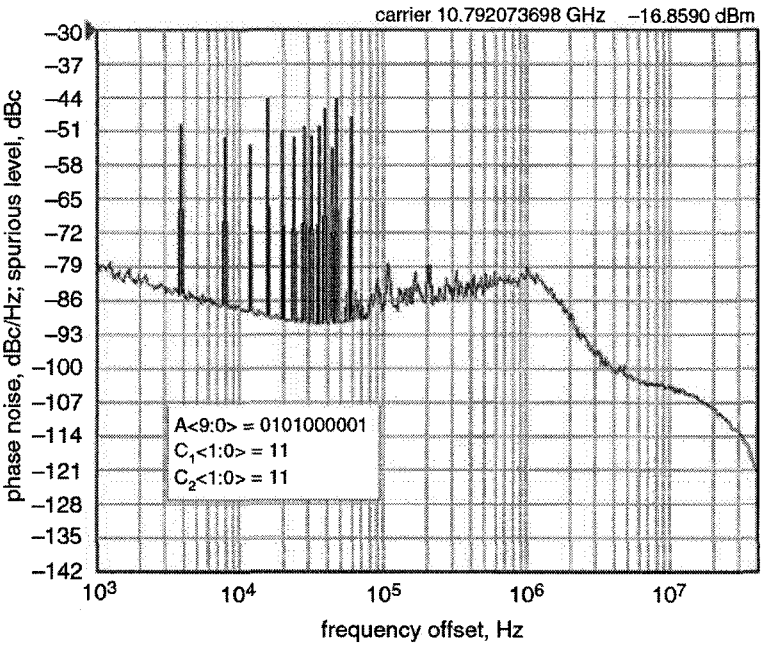


Figure 1.2: Illustration of the fractional spurs of the $\Sigma\Delta$ fractional frequency synthesizers found in [5].

The frequency of the signal from the frequency divider of the proposed synthesizer, while in a locked state, is not time varying as in a conventional fractional frequency synthesizer. Even though non-idealities may cause ripple on the control, causing spurious tones in the output signal, appropriate selection of the loop filters as well as a proper design of the PLL blocks reduce the apparent fractional spurious tones to the point of being imperceptible. Additional fractional spurs cancellation techniques are not needed.

Additional motivation is the proposed frequency synthesizer to have no negative impact of the phase noise and the switching speed over other architectures. Indeed, the analytical analysis and experimental results indicate that the new architecture for a fractional frequency synthesizer is not a limiting factor for the phase noise

performance itself. The phase noise performance predominantly depends on the phase noise performance of the individual PLL blocks.

1.2 Document Outline

This Ph.D. thesis is organized as follows:

Chapter 2 describes the existing techniques for frequency synthesis; they fall in three categories: direct analog, direct digital and indirect analog frequency synthesizers. Advantages and disadvantages of each technique are discussed.

Chapter 3 describes the proposed concepts for frequency synthesizer. The transient analysis of the PLL response to various disturbances under linear conditions is included. To illustrate the proposed system, three possible implementations are identified. The first implementation, denoted as divide-multiply implementation, is discussed in this chapter, as well as in the main body of this thesis. The other two implementations, denoted as multiply-divide implementation and divide-multiply-divide implementation, are discussed in a separate appendix of this thesis.

Chapter 4 depicts the implemented architecture of the frequency divider used with the divide-multiply implementation. Replacing the tail current source of the current-mode logic (CML) gates with a common-source resistor resulted in novel differential gates. The advantages and disadvantages of these gates are investigated and reported.

Chapter 5 describes the phase frequency detector and the novel charge pump derived for the divide-multiply implementation. The phase frequency detector implemented the differential gates as described in the previous chapter. The charge

pump utilized two complementary current sources, and two complementary differential pairs for the UP and Down signals. Phase noise contribution from this blocks is also discussed.

Chapter 6 describes the topology of the voltage controlled oscillator (VCO) operating at 10GHz and used with the divide-multiply implementation. A method to improve the phase noise performance from a reduced power supply by using a resistor tail bias technique and a proper selection of a CMOS varactor topology is discussed. A formula to predict the frequency of oscillation of the VCO is derived and a comparison between the calculated and the simulated frequency of oscillation is reported. In addition, the simulated results of the VCO are compared to state-of-the-art designs found in the literature.

Chapter 7 describes the test setup and the equipment for measuring the fabricated chip. This chapter discusses the measured results and investigates the cause of any discrepancy between the measured and the simulated results. The simulated and the measured results of the divide-multiply implementation are compared to the experimental results of the integer-N and the fractional-N frequency synthesizer found in the literature. The advantages and the disadvantages of the divide-multiply implementation are discussed.

Chapter 8 concludes this work and discusses future work.

Three additional chapters (appendixes) are added to the thesis.

Appendix A discusses an example of the multiply-divide implementation. A comparison based on the simulated results between the divide-multiply and the multiply-divide implementation is discussed. The simulated results of the multiply-divide

implementation are compared to the experimental results of the $\Sigma\Delta$ based frequency synthesizers. In addition, the phase noise performance of the multiply-divide implementation is discussed based on the numerical specification for the phase noise according to the GSM standard.

Appendix B discusses the divide-multiply-divide implementation. A comparison, based on the simulated results, between the divide-multiply-divide implementation and the divide-multiply and the multiply-divide implementation is included.

Appendix C discusses a mathematical formulation of the transient portion of the voltage-controlled signal. The comparison between the calculated, simulated and measured results is reported.

1.3 Thesis Contributions and Publications

The contributions to the art from this thesis work are:

1. a system architecture for a frequency synthesizer;

2. a charge pump design; and

3. differential gates.

Chapter 3 describes how an integer frequency synthesizer may be transformed into a frequency synthesizer that retains the attractive features of the integer frequency synthesizers while enabling a programmable channel resolution smaller than the frequency of the reference signal. Most importantly, fractional spur cancellation techniques employed when using $\Delta\Sigma$ frequency synthesizers are not needed in the new architecture.

Chapter 5 describes a new charge pump design with differential inputs and a single-ended output. In order to architect a frequency synthesizer with reduced spurious tones, the new charge pump helps in reduction of the ripples on the controlled line, and thus reduces the spurious energy in the output spectrum of the VCO.

Finally, the third contribution from this work is the resistive tail biasing technique described in Chapter 4. Although this technique is reported by others for the VCO design and the differential buffer (amplifier), it is this work that reports the resistor tail bias technique for the first time for the differential gates where the current mode logic was dominant at lower frequencies. The resistive technique allows operation from a reduced power supply and improves the phase noise performance.

This thesis work produced the following conference papers:

1. "A Phase-Frequency Detector and a Charge Pump Design for PLL Applications", presented at the 2008 IEEE International Symposium on Circuits and Systems, ISCAS 2008, May 18-21 2008, Seattle, USA.

2. "Frequency Dividers Implementing Custom Cells with Resistor Tail Bias", presented at the International Symposium on Signals, Systems and Electronics 2007, ISSSE 2007, July 30 to August 2 2007, Montreal, Quebec, Canada.

3. "Frequency of Oscillation of a Cross-Coupled CMOS VCO with Resistor Tail Biasing", presented at the 50th IEEE International Midwest Symposium on Circuits and Systems, MWSCAS 2007, August 5-8, 2007, Montreal, Quebec, Canada.

Chapter 2

Techniques for Frequency Synthesis

There are several different frequency synthesis techniques prevalent in the literature. In general, the frequency synthesizers can be divided into three categories:

- direct analog frequency synthesizers (systems without feedback) [4, 9, 10],
- direct digital frequency synthesizers (systems without feedback) [11–13],
- indirect analog frequency synthesizers (systems with feedback) [6–8, 14–18].

2.1 Direct Analog Frequency Synthesizer

Figure 2.1 shows an example of a direct analog synthesizer [4]. This type of synthesizer works as follows:
The selection bit denoted as "a_0" selects one of the applied oscillators and mixes it with the 10kHz signal. The up-converted signal is sent through a bandpass filter to a frequency divider that divides the input signal by 10. This can be repeated (note the other selection bits denoted as "a_1", "a_2", ...) until the required frequency resolution is accomplished.

Advantages of the direct analog frequency synthesizer include a fast switching

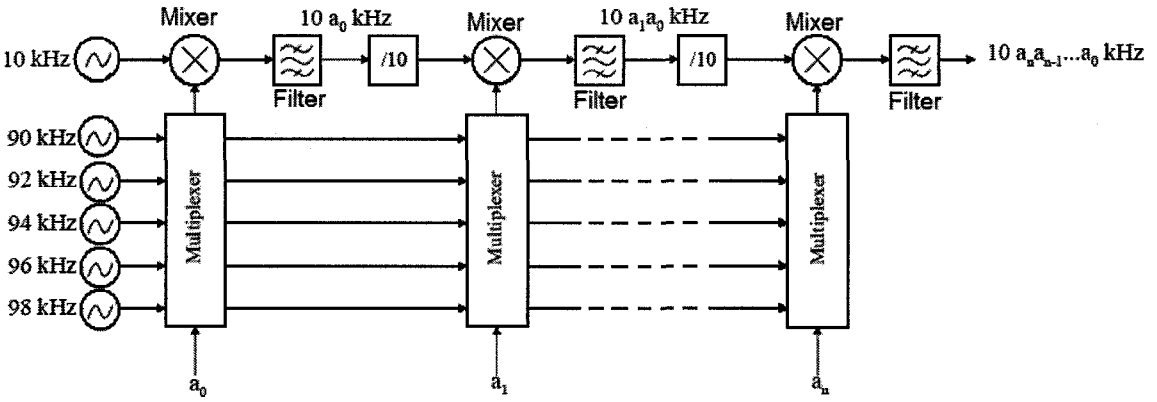


Figure 2.1: Block diagram of a direct analog synthesizer [4].

time and a fine frequency resolution.

There are several disadvantages of the direct analog frequency synthesizer. First, this type of a frequency synthesizer is not recommended for high frequency and low phase noise synthesis in a conventional CMOS (or BiCMOS) technology [4]. In fact, when frequency multiplication of a high factor is involved and the phase noise characteristics of the oscillators is critical, crystal oscillators are preferable [10]. However, multiple crystal oscillators increase the overall system complexity and cost. Finally, to accomplish a fine frequency resolution, the signal should go through multiple mixers, filters and dividers. Each of these components introduce additional noise to the signal.

Examples of direct analog frequency synthesizers can be found in [9, 10].

2.2 Direct Digital Frequency Synthesizer

Figure 2.2 shows a block diagram of a direct digital frequency synthesizer [11]. The following is the functional description of this synthesis technique.

The desired output signal is brought to the input of the phase accumulator as a digital

word (digital bits). Based on the digital code, the phase accumulator increments its output value every clock cycle. Once the full scale is reached, the phase accumulator goes to its starting point. Consequently, the output of the phase accumulator is a digital ramp with period equal to that of the desired output signal.

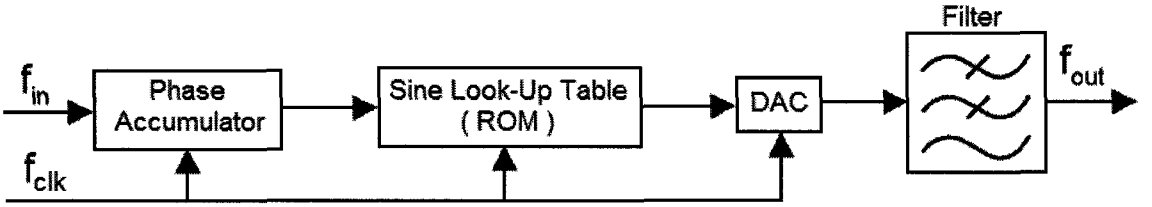


Figure 2.2: Block diagram of a direct digital synthesizer [11].

The output of the phase accumulator is fed to a read-only-memory (ROM) block. The output of the ROM encodes the desired instantaneous amplitude of the synthesized signal.

The output of the ROM goes to a digital-to-analog converter (DAC). The DAC converts a digital input signal to an analog signal. This analog signal is filtered through a low-pass filter (LPF) to eliminate any undesired signals such as spurious tones and harmonics.

Compared to the direct analog synthesis, the direct digital synthesis can also achieve a fine frequency resolution and a fast switching speed [4].

The DAC performance[1] can be a limiting factor when this synthesis technique is implemented in high speed and low phase noise application [4].

Examples of direct digital frequency synthesizers can be found in [11–13]. Table 2.1 shows a summary of the experimental results of the aforementioned works. The

[1]Some of the characteristics of the digital-to-analog converters prevalent in the literature are the clock frequency, the linearity, and the resolution.

Table 2.1: Examples of direct digital frequency synthesizers.

Reference	[11]	[12]	[13]
Technology	0.8μm BiCMOS	CMOS 0.35μm	0.35μm InP DHBT
Supply voltage	5V	3.3V	4.5V
Power	0.6W	0.2W	9.45W
Clock Frequency	150MHz	300MHz	32GHz
Output Frequency	45.8MHz	8MHz	125MHz
SFDR	52.5dBc	78dBc	31dBc
Resolution	0.0394Hz	4.48kHz	125MHz
Settling Time	140ns	N/A	N/A
Chip area	3.9mm^2	1.1mm^2	3.9mm^2

clock frequency utilized by [11] is 150MHz, while the work found in [12] and [13] reported a clock frequency of 300MHz and 32GHz, respectively. The 32GHz clock frequency reported by [13] resulted in 9.45W power, compared to 0.2W and 0.6W for the work found in [12] and [11], respectively. The [11] achieved a fine resolution of 0.0394Hz with settling time of 140ns.

2.3 Indirect Analog Frequency Synthesizer

Figure 2.3 shows the block diagram of an indirect analog frequency synthesizer also known as a phase locked loop[2] (PLL).

The frequency of the output signal of a tunable voltage controlled oscillator (VCO) is divided by a programmable frequency divider. The output of the frequency divider is fed to the input of a phase frequency detector (PFD). The PFD compares the phase of the divider output signal to the phase of a reference signal. The output of the PFD goes through a charge pump (CP) and a low pass filter (LPF) and as a voltage-control signal goes to the VCO. This voltage-control signal tunes the VCO to the

[2]The shown block diagram is only a particular example of a PLL for the case when a phase frequency detector is used. Another possible block diagram of a PLL is the case when a phase detector followed by a low pass filter is used.

desired frequency. Under PLL locked conditions the output of the frequency divider will have a phase and frequency equal to the phase and frequency of the reference signal.

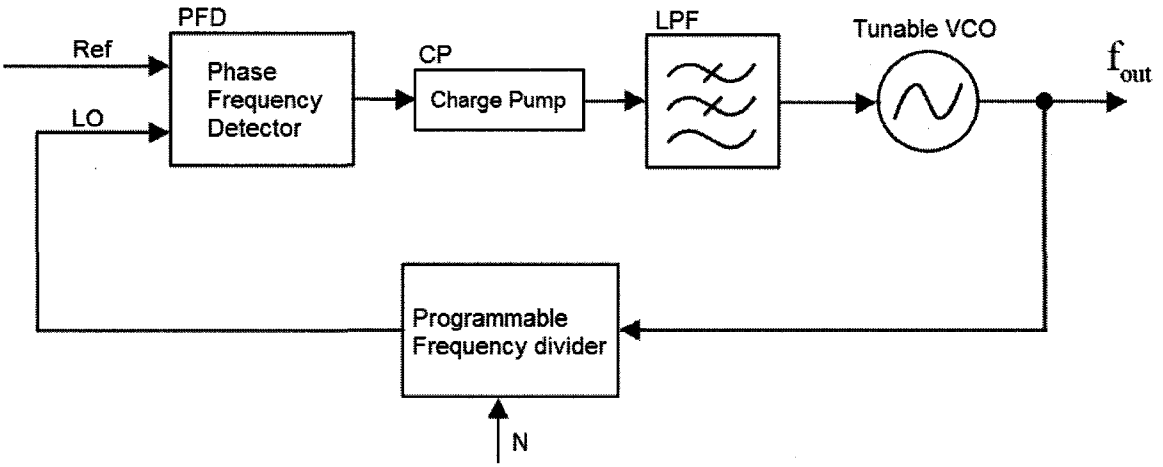


Figure 2.3: Block diagram of a phase-locked loop [14].

The frequency synthesizer based on this PLL technique is suitable for high-speed applications and integration into a CMOS or a BiCMOS technology is possible.

The main disadvantage of this type of frequency synthesizer, also known as an integer-N frequency synthesizer, is the frequency resolution. The frequency resolution equals the reference signal. If a low frequency resolution is required then a large difference between the VCO and reference signal results in a high division ratio of the frequency divider. Consequently, the phase noise of the frequency synthesizer is affected in a negative way. To illustrate the problem, the channel spacing in GSM and DCS-1800 systems is 200kHz [6]. If an integer frequency synthesizer is used then the frequency of the reference signal should be equal to or smaller than the channel spacing. This leads to very high values for the division N of the frequency divider. For DCS-1800 system, the carrier frequency is between 1710MHz and 1880MHz. If

Table 2.2: Examples of integer frequency synthesizers.

Reference	[14]	[15]	[16]
Technology	CMOS 0.4μm	0.5μ m SiGe BiCMOS	CMOS 0.18μm
Supply voltage	2.6V	2.7V	1V
Power	47mW	121mW	27.5mW
Frequency	5.2GHz	4.4GHz	5.2GHz
Phase Noise	-100dBc/Hz	-119dBc/Hz	-136dBc/Hz
Frequency Offset	10MHz	1MHz	20MHz
Resolution	23.5MHz	20MHz	20MHz
Settling Time	40μsec	N/A	51μsec
Loop Bandwidth	Not Specified	N/A	100kHz
Chip area	2mm^2	1.1mm^2	1mm^2

200kHz is used as a reference signal then the division N varies from 8550 to 9400. An N of 9400 means that the reference noise close to the carrier is increased by 80dB (20log9400). In addition, the low frequency reference signals requires a small loop bandwidth. However, a small loop bandwidth affects the switching speed and stability of the frequency synthesizer [4].

Examples of integer-N frequency synthesizers can be found in [14–16] and Table 2.2 shows a summary of the experimental results. The work found in [16] dissipated 27.5mW from 1V supply while generating frequency around 5.2GHz with 20MHz resolution. The cited work used loop bandwidth of 100kHz and achieved settling time of 51μs. The work in [14] generated frequencies around 5.2GHz as well, however with higher resolution and 47mW power from 2.6V supply. Although the work found in [15] generated frequencies around 4.4GHz with 20MHz resolution, the power dissipation of 121mW from 2.7V supply is high compared to the work found in [14] and [16].

The fractional-N frequency synthesizers are another type of indirect analog frequency synthesizer. Figure 2.4 shows a block diagram of a $\Delta\Sigma$ fractional frequency

synthesizer. To lock the VCO at a fractional multiple of the reference signal, the $\Delta\Sigma$ modulator varies the division ratio of the divider between two integer values in such a way that the average value of N is a fractional number.

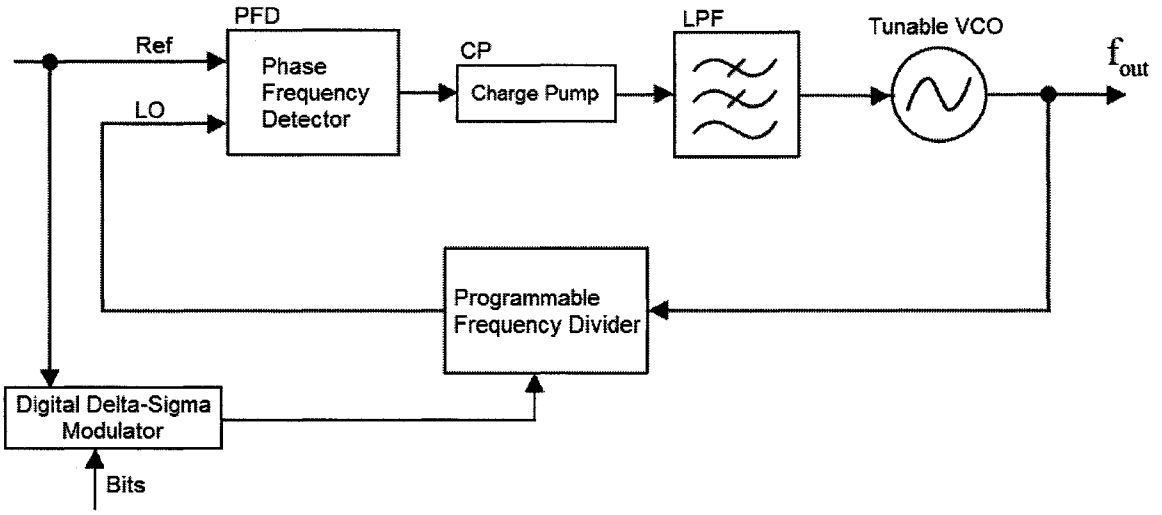


Figure 2.4: Block diagram of a $\Delta\Sigma$ phase-locked loop.

The advantages of the fractional-N over the integer-N frequency synthesizer are: higher reference signal, higher loop bandwidth, and reduced reference noise close to the carrier [4].

The main disadvantage of the fractional-N synthesizer is that the instantaneous division ratio is an integer number. That means that the frequency divider is not dividing the frequency of the VCO output signal by a fraction but by an integer number. As a consequence, the PLL is never actually locked in a way as is the case for the integer frequency synthesizer.

To understand the above problem, one could analyze the case where a first-order $\Delta\Sigma$ modulator[3] controls the frequency divider. To simplify the analysis, it can be assumed that the loop is locked and that the value of the adder is zero. In this case,

[3]A first-order $\Delta\Sigma$ modulator can be a digital adder also referred as an accumulator.

the frequency of the VCO output signal is divided by some number N_1. The phase detector compares the transition edges, either rising or falling, of the divided and reference signal. After every clock cycle of the reference signal, the value of the adder is increased. Once the accumulator reaches its full scale it goes back to its starting point. However, this time the accumulator programs the divider to divide by $N_2 > N_1$. The next time the accumulator overflows and goes to its starting point the divider is programmed to divide by N_1 again. Under locked condition this repeats and the VCO output frequency is divided by a fractional number in a statistical way. The undesired result is that the output of the phase detector will have a periodic sawtooth shape with a frequency proportional to the reference signal. This periodic sawtooth shape causes the appearance of spikes or fractional spurious tones close to the desired frequency[4].

One solution is to apply the accumulated phase error to a digital-to-analog converter (DAC) and to subtract it from the phase detector output. However, the nonlinearities of the DAC can cause spurs in the output spectrum making this method difficult to implement. A more popular[5] solution is to implement a higher order $\Delta\Sigma$ modulator. However, a higher-order modulator requires a higher order of loop filter as well. To overcome this problem, multibit single-loop modulators are used in the literature. The major problem with this method is the stability of the frequency synthesizer [6].

[4]At frequency offsets that are multiple of the frequency of the periodic sawtooth signal.
[5]Based on the number of research works prevalent in the literature.

Table 2.3: Examples of $\Sigma\Delta$ fractional frequency synthesizers.

Reference	[6]	[7]	[17]	[18]	[8]
Technology	CMOS 0.25μm	CMOS 0.13μm	SiGe 0.5μm	CMOS 0.5μm	CMOS 0.35μm
Supply voltage	2V	1.2V	2.8V	3.3V	3.3V
Current consumption	35mA	9.5mA	19.5mA	15.8mA	15mA
Output frequency	1.8(30%) GHz	2.4-2.48 GHz	1.15-1.75 GHz	1.67-1.79 GHz	2.4-2.5 GHz
Reference frequency	26MHz	19.2MHz	13MHz	20MHz	256MHz
In-band phase noise @ Offset	-60 dBc/Hz 10kHz	-80 dBc/Hz 100kHz	-80 dBc/Hz 10kHz	-80 dBc/Hz 10kHz	-80 dBc/Hz 1kHz
Out-of-band phase noise @ Offset	-120 dBc/Hz 600kHz	-125 dBc/Hz 1MHz	-129 dBc/Hz 400kHz	-118 dBc/Hz 1MHz	-97 dBc/Hz 1MHz
Resolution	400Hz	50Hz	3Hz	10Hz	62.5kHz
Settling time	226μs	70μs	150μs	50μs	not given
Loop bandwidth	35kHz	100kHz	25kHz	20kHz	4MHz
Fractional spurs @ Offset	-100dBc 600kHz	-63dBc 1MHz	-70dBc 300kHz	-70dBc 2.5MHz	-55dBc 62.5kHz
Reference spurs @ Offset	-75dBc not given	not given not given	-75dBc 13MHz	not given not given	not given not given
Chip area	4mm^2	0.9mm^2	not given	10.7mm^2	3.7mm^2

Examples of fractional frequency synthesizers can be found in [6–8,17,18] and Table 2.3 shows a summary of the experimental results. The work found in [7] reported the lowest power dissipation, compared to the work summarized in Table 2.3, while generating signals with frequency between 2.4 and 2.48GHz. Regarding the in-band phase noise, the work found in [6] reported the worst performance. Regarding the out-of-band phase noise, the worst performance is reported in the work found in [8]. The finest frequency resolution of 3Hz is reported in [17]. The fastest switching time is reported in [18] with a loop bandwidth of 20kHz. The best spurious performance

is reported in [6].

2.4 Summary

This chapter briefly described the frequency synthesis techniques prevalent in the literature. They were divided into three categories denoted as direct analog frequency synthesizers (systems without feedback), direct digital frequency synthesizers (systems without feedback), and indirect analog frequency synthesizers (systems with feedback). Advantages of the direct analog frequency synthesizer are a fast switching time and a fine frequency resolution. However, the direct analog frequency synthesizer is not recommended for high frequency and low phase noise synthesis in a conventional CMOS (or BiCMOS) technology. The direct digital synthesis can also achieve a fine frequency resolution and a fast switching speed. Nevertheless, the DAC performance can be a limiting factor when this synthesis technique is implemented in high speed and low phase noise applications.

The indirect analog frequency synthesizers can further be divided into two categories: integer-N and fractional-N frequency synthesizers. The integer-N frequency synthesizer is suitable for high-speed applications and integration into a CMOS or a BiCMOS technology is possible. The main disadvantage of this type of frequency synthesizer is the frequency resolution. The frequency resolution equals the reference signal. If a fine frequency resolution is required then a large difference between the VCO and reference signal results in a high division ratio of the frequency divider which affects the in-band phase noise of the frequency synthesizer.

The advantages of the fractional-N over the integer-N frequency synthesizer are:

higher reference signal, higher loop bandwidth, and reduced reference noise close to the carrier. The main disadvantage of the fractional-N synthesizer is that the instantaneous division ratio is an integer number. That means that the frequency divider is not dividing the frequency of the VCO output signal by a fraction but by an integer number. As a consequence, the fractional-N frequency synthesizer is never actually locked in a way as is the case for the integer-N frequency synthesizer. The undesired result is that the output of the phase detector will have a periodic sawtooth shape with a frequency proportional to the reference signal. This periodic sawtooth shape causes the appearance of spikes or fractional spurious tones close to the desired frequency. In order to reduce the fractional spurs of the fractional-N frequency synthesizers, various spur cancelation techniques are used in the literature.

The primary goal of this thesis is to architect a PLL type of a frequency synthesizer with reduced amount of spurious tones, without the need of spur cancelation techniques, and with a frequency resolution smaller than the reference signal.

Chapter 3

Proposed System Architecture

This chapter discusses the proposed architectural concept for a frequency synthesis technique.

3.1 System Description

The proposed architecture for a frequency synthesizer is based on an integer-N frequency synthesizer. The programmable frequency divider found within the feedback loop of the integer-N frequency synthesizer is replaced with subsystem implemented with one or more programmable frequency multipliers and one or more programmable frequency dividers. The input signal to the introduced subsystem is the output signal from the voltage-controlled oscillator. The frequency and the phase of the output signal from the introduced subsystem, in a lock state, are equal to the frequency and the phase of the reference signal.

Depending of the number of the programmable frequency multipliers and the programmable frequency dividers, as well as their placement, there are multiple implementations of the proposed architectural concept. To illustrate the proposed architectural concept for a frequency synthesizer, this thesis discusses three possible

implementations.

If the frequency multiplier is placed after the frequency divider, then the implementation is denoted as a divide-multiply implementation and is depicted in Figure 3.1. A multiply-divide implementation is the case when the frequency multiplier is placed immediately after the VCO, as illustrated in Figure 3.2.

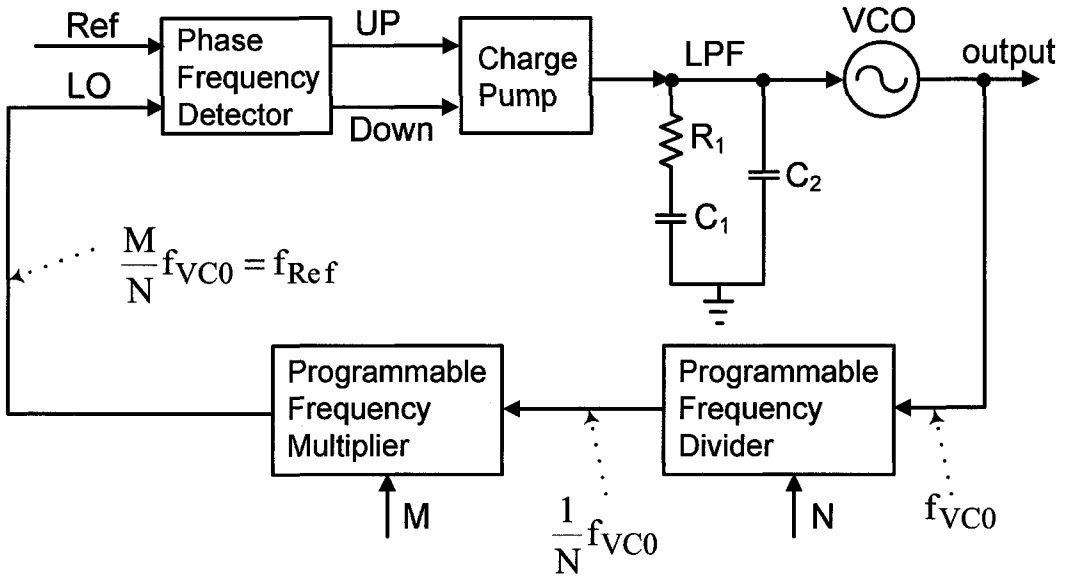


Figure 3.1: A block diagram of the proposed frequency synthesizer when the VCO signal goes first to a frequency divider, then the signal from the frequency divider is brought to a frequency multiplier and the output of the frequency multiplier is brought to the input of the phase frequency detector.

Figure 3.3 shows the block diagram of the divide-multiply-divide implementation. The feedback loop of the frequency synthesizer consists of a cascade connection of two frequency dividers and one frequency multiplier. The output signal from the VCO first is brought to a frequency divider. The output signal from the frequency divider is brought to the frequency multiplier. The output signal from the frequency divider is sent to another frequency divider. The output signal from the second frequency divider is brought to the input of the phase frequency detector.

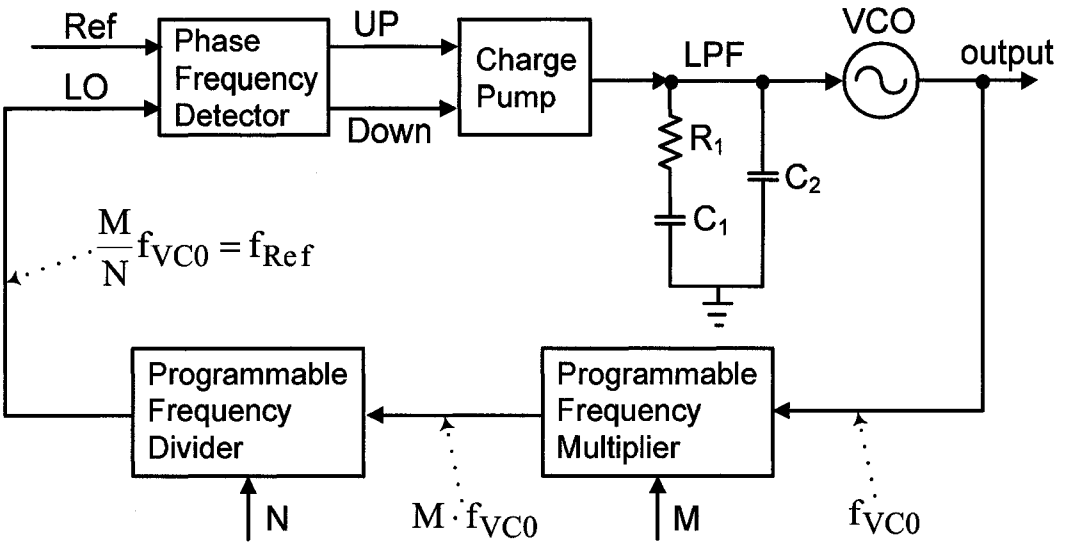


Figure 3.2: A block diagram of the proposed frequency synthesizer when the VCO signal goes first to a frequency multiplier, then the signal from the frequency multiplier is brought to a frequency divider and the output of the frequency divider is brought to the input of the phase frequency detector.

As illustrated for the integer-N frequency synthesizers (Section 2.3 from Chapter 2) the division ratio of the frequency divider affects the phase noise of the frequency synthesizer. The division ratio of the frequency divider of the proposed system architecture, implemented with one frequency divider and one frequency multiplier, can have high values. For example, the illustrated divide-multiply implementation utilizes frequency divider which division ratio have values around 20000. Does this mean that the reference noise close to the carrier is increased by 20logN or 86dB? The proposed architecture has a frequency multiplier placed in cascade with the frequency divider. This thesis shows that the placement of the frequency multiplier can reduce the influence of the frequency divider on the phase noise.

Among all advantages of the proposed system architecture (to be discussed further in the thesis), an additional advantage is that the theory built for the integer-N

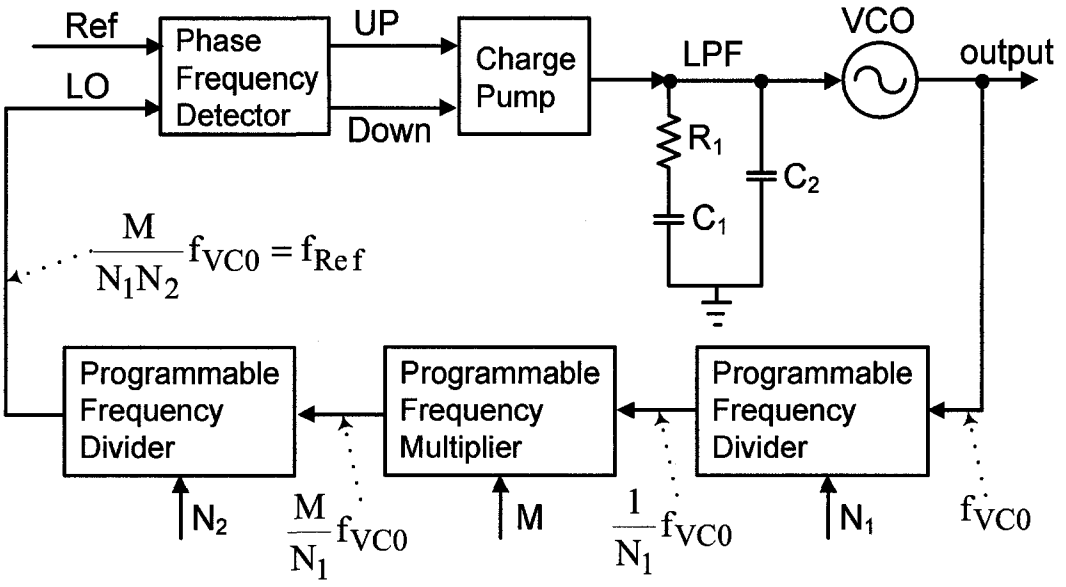


Figure 3.3: A block diagram of the divide-multiply-divide implementation.

frequency synthesizers is applicable for the proposed frequency synthesizer as well. The following subsections applies and extends that theory, as explained in [1] and [2], and discusses the transient and the phase noise analysis of the proposed loop system.

To simplify the theoretical analysis, the derivations are performed assuming that the frequency multiplier requires one clock cycle to generate the desired output frequency. However, in a case of a PLL type of a frequency multiplier, more clock cycles are required before the frequency multiplier re-acquires its lock again. This case is discussed further in this chapter.

3.1.1 Transfer Function

Assuming a sinusoidal reference signal, in order to simplify the theoretical analysis, the reference signal can be expressed as,

$$v_{\mathtt{Ref}}(t) = \mathtt{A}_{\mathtt{Ref}} \sin\left(\omega t + \phi_{\mathtt{Ref}}(t)\right) \tag{3.1.1}$$

where ω is the angular frequency, $\mathtt{A}_{\mathtt{Ref}}$ is the amplitude, and $\phi_{\mathtt{Ref}}$ is the phase of the reference signal.

Under a locked condition, the frequency of the LO signal is equal to the frequency of the reference signal. However, the amplitude, $\mathtt{A}_{\mathtt{LO}}$, and the phase of the LO signal, $\phi_{\mathtt{LO}}$, and the amplitude and the phase of the reference signal are not necessarily equal. Thus, the LO signal can be expressed as,

$$v_{\mathtt{LO}}(t) = \mathtt{A}_{\mathtt{LO}} \sin\left(\omega t + \phi_{\mathtt{LO}}(t)\right). \tag{3.1.2}$$

The output signal from the VCO can be expressed as,

$$v_{\mathtt{VCO}}(t) = \mathtt{A}_{\mathtt{VCO}} \sin\left(\omega_{\mathtt{VCO}} t + \phi_{\mathtt{VCO}}(t)\right). \tag{3.1.3}$$

The PFD generates a signal that is proportional to the phase difference between the reference and the LO signal. If the PFD has a sawtooth characteristic over 2π radians, then the output signal from the phase-frequency detector and the charge pump (PFD-CP) is,

$$v_{\mathtt{PFD-CP}}(t) = \frac{\mathtt{I}_{\mathtt{CP}}}{2\pi}\left(\phi_{\mathtt{Ref}}(t) - \phi_{\mathtt{LO}}(t)\right) \tag{3.1.4}$$

where $\mathtt{I}_{\mathtt{CP}}$ is the charge pump current.

The control signal for the VCO can be calculated as a convolution between the signal $v_{\mathtt{PFD-CP}}(t)$ and the impulse response of the implemented filter,

$$V_{\mathtt{ctrl}}(t) = v_{\mathtt{PFD-CP}}(t) \circledast f(t). \tag{3.1.5}$$

The instantaneous angular frequency of the VCO, $\omega_{\mathtt{inst}}$, is a linear function of the control signal for the VCO around the central angular frequency $\omega_{\mathtt{VCO}}$,

$$\omega_{\mathtt{inst}} = \frac{\mathtt{d}}{\mathtt{dt}}\left(\omega_{\mathtt{VCO}}t + \phi_{\mathtt{VCO}}(t)\right) = \omega_{\mathtt{VCO}} + \mathtt{K}_{\mathtt{VCO}}V_{\mathtt{ctrl}}(t). \tag{3.1.6}$$

Thus,

$$\frac{\mathtt{d}\phi_{\mathtt{VCO}}(t)}{\mathtt{dt}} = \mathtt{K}_{\mathtt{VCO}}V_{\mathtt{ctrl}}(t) \tag{3.1.7}$$

where $\mathtt{K}_{\mathtt{VCO}}$ is the modulation sensitivity of the `VCO` in `rad/s/V`.

The expressions (3.1.4), (3.1.5), and (3.1.7) give the general time domain equation for a phase locked loop with a sawtooth characteristic phase detector,

$$\frac{\mathtt{d}\phi_{\mathtt{VCO}}(t)}{\mathtt{dt}} = \mathtt{K}_{\mathtt{VCO}}\frac{\mathtt{I}_{\mathtt{CP}}}{2\pi}\left(\phi_{\mathtt{Ref}}(t) - \phi_{\mathtt{LO}}(t)\right) \circledast f(t). \tag{3.1.8}$$

If the multiplication ratio and the division ratio of the frequency multiplier and the frequency divider are `M` and `N`, respectively, then the relationship between the phase of the `LO` signal and the phase of the `VCO` signal is,

$$\phi_{\mathtt{LO}}(t) = \frac{\mathtt{M}}{\mathtt{N}}\phi_{\mathtt{VCO}}(t). \tag{3.1.9}$$

If the expression (3.1.9) is substituted into the expression (3.1.8) then a Laplace transform of the resulting equation gives,

$$\mathtt{s}\Phi_{\mathtt{VCO}}(\mathtt{s}) = \mathtt{K}_{\mathtt{VCO}}\frac{\mathtt{I}_{\mathtt{CP}}}{2\pi}\left(\Phi_{\mathtt{Ref}}(\mathtt{s}) - \frac{\mathtt{M}}{\mathtt{N}}\Phi_{\mathtt{VCO}}(\mathtt{s})\right)\mathtt{F}(\mathtt{s}) \tag{3.1.10}$$

where $\Phi_{VCO}(s)$, $\Phi_{Ref}(s)$, and $F(s)$ are the Laplace transforms of ϕ_{VCO}, ϕ_{Ref}, and $f(t)$, respectively.

The expression (3.1.10) gives the phase transfer function of the proposed system,

$$\Phi_o(s) = \frac{\Phi_{VCO}(s)}{\Phi_{Ref}(s)} = \frac{K_{VCO}\frac{I_{CP}}{2\pi}F(s)}{s + K_{VCO}\frac{I_{CP}}{2\pi\cdot\frac{N}{M}}F(s)}. \tag{3.1.11}$$

The transfer function of the implemented loop filter, shown in Figure 3.1 or Figure 3.2, is,

$$F(s) = \frac{1 + sR_1C_1}{s(C_1 + C_2) + s^2R_1C_1C_2}. \tag{3.1.12}$$

The transfer function of the phase-locked loop with the considered loop filter is,

$$\Phi_o(s) = \frac{\frac{K_{VCO}I_{CP}}{2\pi\cdot(C_1+C_2)}(1 + sR_1C_1)}{s^3\frac{R_1C_1C_2}{(C_1+C_2)} + s^2 + s\frac{R_1C_1K_{VCO}I_{CP}}{2\pi\cdot\frac{N}{M}(C_1+C_2)} + \frac{K_{VCO}I_{CP}}{2\pi\cdot\frac{N}{M}(C_1+C_2)}} \tag{3.1.13}$$

If the capacitor C_2 is sized to be at least 10-times smaller compared to the capacitor C_1 then the simplified phase transfer function of the proposed system is,

$$\frac{\Phi_{VCO}(s)}{\Phi_{Ref}(s)} = \frac{K_{VCO}I_{CP}\frac{1}{2\pi\cdot C_1}(1 + sR_1C_1)}{s^2 + s\cdot R_1K_{VCO}I_{CP}\frac{M}{2\pi\cdot N} + K_{VCO}I_{CP}\frac{M}{2\pi\cdot N\cdot C_1}}. \tag{3.1.14}$$

The transfer function of the loop system may be written as,

$$\Phi_{VCO}(s) = \Phi_{Ref}(s)\frac{\omega_n^2\frac{N}{M} + s\cdot 2\xi\omega_n\frac{N}{M}}{s^2 + s\cdot 2\xi\omega_n + \omega_n^2} \tag{3.1.15}$$

where, ξ is the damping factor, and ω_n is the natural frequency of the loop system defined as,

$$2\xi\omega_n = R_1K_{VCO}I_{CP}\frac{M}{2\pi N} \tag{3.1.16}$$

$$\omega_n^2 = K_{VCO}I_{CP}\frac{M}{2\pi NC_1}. \tag{3.1.17}$$

3.1.2 Error Function

The instantaneous phase error of the phase locked loop is given by,

$$\phi(t) = \phi_{\mathrm{Ref}}(t) - \phi_{\mathrm{LO}}(t). \tag{3.1.18}$$

Using the expression (3.1.9) the instantaneous phase error may also be written as,

$$\phi(t) = \phi_{\mathrm{Ref}}(t) - \frac{\mathrm{M}}{\mathrm{N}}\phi_{\mathrm{VCO}}(t). \tag{3.1.19}$$

In the Laplace domain,

$$\Phi(\mathrm{s}) = \Phi_{\mathrm{Ref}}(\mathrm{s}) - \frac{\mathrm{M}}{\mathrm{N}}\Phi_{\mathrm{VCO}}(\mathrm{s}) \tag{3.1.20}$$

If the expression (3.1.15) is substituted into the expression (3.1.20) then the error function can be written as,

$$\Phi(\mathrm{s}) = \Phi_{\mathrm{Ref}}(\mathrm{s})\frac{\mathrm{s}^2}{\mathrm{s}^2 + \mathrm{s}\cdot 2\xi\omega_n + \omega_n^2} \tag{3.1.21}$$

3.1.3 Transient Analysis

The following part discusses the transient and the phase noise analysis of the proposed system. The response of the loop to different disturbances are examined. The following disturbances are investigated:

- Input signal phase step θ,
- Input signal angular frequency step $\Delta\omega$,
- Linear variation of slope Δf of the input signal frequency.

The transient analysis is performed assuming that the loop system was locked before the disturbance arrived. The disturbance is assumed to be small so that the linear

regime is preserved and the frequency multiplier and the frequency divider generate signals with stable frequency.

Phase Step Response:

The phase step response can be calculated by applying a θ amplitude phase step to the input signal,

$$\phi_{\mathtt{Ref}}(t) = \theta \mathtt{U}(t) \tag{3.1.22}$$

where $\mathtt{U}(t)$ is the unit step function.

The Laplace transform of the equation (3.1.22) is,

$$\Phi_{\mathtt{Ref}}(\mathtt{s}) = \frac{\theta}{\mathtt{s}}. \tag{3.1.23}$$

The expressions (3.1.21) and (3.1.23) lead to,

$$\Phi(\mathtt{s}) = \frac{\theta \mathtt{s}}{\mathtt{s}^2 + \mathtt{s} \cdot 2\xi\omega_n + \omega_n^2}. \tag{3.1.24}$$

The instantaneous phase error of the phase locked loop, $\phi(t)$, is derived as an inverse Laplace transform of $\Phi(\mathtt{s})$. The inverse Laplace transform is easier to perform if the denominator of the expression (3.1.24) is decomposed into two products, i.e.,

$$\Phi(\mathtt{s}) = \frac{\theta \mathtt{s}}{\left(\mathtt{s} + \omega_n\left(\xi + \sqrt{\xi^2 - 1}\right)\right)\left(\mathtt{s} + \omega_n\left(\xi - \sqrt{\xi^2 - 1}\right)\right)}. \tag{3.1.25}$$

By definition, the inverse Laplace transform of the function,

$$\Gamma(\mathtt{s}) = \frac{\mathtt{A}\mathtt{s}}{(\mathtt{s} + \mathtt{a})(\mathtt{s} + \mathtt{b})} \tag{3.1.26}$$

where A is a constant, is,

$$\gamma(t) = \frac{\mathtt{A}}{\mathtt{a} - \mathtt{b}}\left(\mathtt{a} \cdot \mathtt{e}^{-\mathtt{a}t} - \mathtt{b} \cdot \mathtt{e}^{-\mathtt{b}t}\right). \tag{3.1.27}$$

Thus, the expression of $\phi(t)$ is,

$$\phi(t) = \theta \frac{\omega_n\left(\xi + \sqrt{\xi^2-1}\right)\cdot \mathrm{e}^{-\omega_n\left(\xi+\sqrt{\xi^2-1}\right)t} - \omega_n\left(\xi - \sqrt{\xi^2-1}\right)\cdot \mathrm{e}^{-\omega_n\left(\xi-\sqrt{\xi^2-1}\right)t}}{\omega_n\left(\xi + \sqrt{\xi^2-1}\right) - \omega_n\left(\xi - \sqrt{\xi^2-1}\right)}. \tag{3.1.28}$$

In order to come to an expression for the phase error $\phi(t)$ that will reflect the value of the damping factor ξ, the equation (3.1.28) can be simplified to,

$$\phi(t) = \theta e^{-\xi\omega_n t}\left(\frac{\xi}{\sqrt{\xi^2-1}}\frac{\left(e^{-\omega_n\sqrt{\xi^2-1}t} - e^{+\omega_n\sqrt{\xi^2-1}t}\right)}{2} + \frac{\left(e^{-\omega_n\sqrt{\xi^2-1}t} + e^{+\omega_n\sqrt{\xi^2-1}t}\right)}{2}\right). \tag{3.1.29}$$

Case I: $\xi < 1$

If the damping factor is smaller than one, then the term

$$\sqrt{\xi^2-1}$$

can be rewritten as,

$$\mathtt{i}\cdot\sqrt{1-\xi^2}$$

where $\mathtt{i} = \sqrt{-1}$ indicates a complex number.

If the aforementioned is substituted into the equation (3.1.29), and the following Euler's formula is implemented,

$$e^{\pm \mathtt{i}\mathtt{x}} = \cos\mathtt{x} \pm \mathtt{i}\sin\mathtt{x} \tag{3.1.30}$$

then the equation (3.1.29) can be simplified to,

$$\phi(t) = \theta e^{-\xi\omega_n t}\left(\cos\omega_n\sqrt{1-\xi^2}t - \frac{\xi}{\sqrt{1-\xi^2}}\sin\omega_n\sqrt{1-\xi^2}t\right). \tag{3.1.31}$$

Case II: $\xi = 1$

If the damping factor is equal to 1 then a simple substitution into the equation (3.1.29)

would result the first term inside the brackets to have a nominator and a denominator equal to zero,

$$\lim_{\xi \to 1} \phi(t) = \theta e^{-\omega_n t} \left(\frac{0}{0} + 1 \right) \tag{3.1.32}$$

In such a case, the L'Hospital's rule can be implemented on the aforementioned term to find the value that this term would receive if ξ equals 1, i.e.,

$$\lim_{\xi \to 1} \frac{\frac{d}{d\xi}\left(\xi\left(e^{-\omega_n\sqrt{\xi^2-1}t} - e^{+\omega_n\sqrt{\xi^2-1}t}\right)\right)}{\frac{d}{d\xi}\left(2\sqrt{\xi^2-1}\right)} = -\omega_n t \tag{3.1.33}$$

Thus, in the case when the damping factor is equal to 1, the phase error has the following expression,

$$\phi(t) = \theta e^{-\xi\omega_n t}(1 - \omega_n t). \tag{3.1.34}$$

Case III: $\xi > 1$

If the damping factor is greater than one then the term $\sqrt{\xi^2 - 1}$ is real. Then by definition,

$$\frac{e^{+x} - e^{-x}}{2} = \sinh x \tag{3.1.35}$$

$$\frac{e^{+x} + e^{-x}}{2} = \cosh x \tag{3.1.36}$$

If the above definitions are implemented then the equation (3.1.29) gives,

$$\phi(t) = \theta e^{-\xi\omega_n t}\left(\cosh\omega_n\sqrt{\xi^2-1}t - \frac{\xi}{\sqrt{\xi^2-1}}\sinh\omega_n\sqrt{\xi^2-1}t\right) \tag{3.1.37}$$

In summary, depending of the value of the damping factor, the phase error function $\phi(t)$ can have the following expressions,

$$\phi(t) = \begin{cases} \theta e^{-\xi\omega_n t}\left(\cos\omega_n\sqrt{1-\xi^2}t - \frac{\xi}{\sqrt{1-\xi^2}}\sin\omega_n\sqrt{1-\xi^2}t\right), & \xi < 1 \\ \theta e^{-\xi\omega_n t}(1 - \omega_n t), & \xi = 1 \\ \theta e^{-\xi\omega_n t}\left(\cosh\omega_n\sqrt{\xi^2-1}t - \frac{\xi}{\sqrt{\xi^2-1}}\sinh\omega_n\sqrt{\xi^2-1}t\right), & \xi > 1 \end{cases} \tag{3.1.38}$$

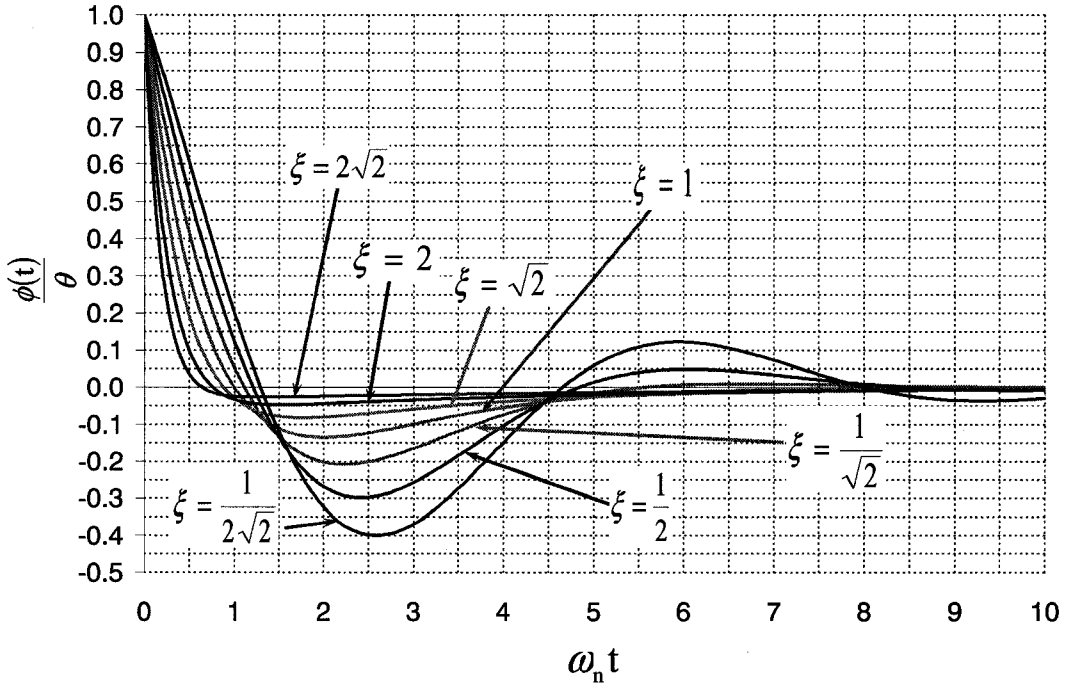


Figure 3.4: Calculated phase error of the proposed frequency synthesizer for a phase step of the input signal.

Figure 3.4 shows the curves representing the normalized phase error, $\phi(t)/\theta$, for different values of the damping factor ξ. Equation (3.1.38) and the curves shown in Figure 3.4 are valid if $\theta < \pi$ radians and the phase detector is linear over $(-\pi, \pi)$ [1]. If a normalized phase error of 10^{-3} or smaller is monitored then a close examination of the plots indicated that the fastest elimination of the phase error can be accomplished if the damping factor of the proposed frequency synthesizer is set to $\xi = \frac{1}{\sqrt{2}}$.

Frequency Step Response:

To calculate the frequency step response, a frequency step $\Delta\omega$ is applied to the input signal of the phase-locked system,

$$\phi_{\mathtt{Ref}}(t) = \Delta\omega t \mathtt{U}(t). \tag{3.1.39}$$

The Laplace transform gives,

$$\Phi_{\mathrm{Ref}}(s) = \frac{\Delta\omega}{\mathrm{s}^2}. \tag{3.1.40}$$

If equation (3.1.40) is substituted into the equation (3.1.21) then the expression for the phase error function when a frequency step is applied to the input of the loop system is,

$$\Phi(\mathrm{s}) = \frac{\Delta\omega}{\mathrm{s}^2 + \mathrm{s}\cdot 2\xi\omega_n + \omega_n^2}. \tag{3.1.41}$$

To find the inverse Laplace transform, the equation (3.1.41) can be written as,

$$\Phi(\mathrm{s}) = \frac{\Delta\omega}{\left(\mathrm{s} + \omega_n\left(\xi + \sqrt{\xi^2 - 1}\right)\right)\left(\mathrm{s} + \omega_n\left(\xi - \sqrt{\xi^2 - 1}\right)\right)}. \tag{3.1.42}$$

By definition, the inverse Laplace transform of the function,

$$\Gamma(\mathrm{s}) = \frac{\mathrm{A}}{(\mathrm{s}+\mathrm{a})(\mathrm{s}+\mathrm{b})} \tag{3.1.43}$$

where A is a constant, is,

$$\gamma(t) = \frac{\mathrm{A}}{\mathrm{b}-\mathrm{a}}\left(\mathrm{e}^{-\mathrm{a}t} - \mathrm{e}^{-\mathrm{b}t}\right). \tag{3.1.44}$$

If the above defined inverse Laplace transform is implemented and the new expression is simplified then the phase error function in time domain is,

$$\phi(t) = \frac{\Delta\omega}{\omega_n}\mathrm{e}^{-\xi\omega_n t}\left(\frac{\mathrm{e}^{\omega_n\sqrt{\xi^2-1}t} - \mathrm{e}^{-\omega_n\sqrt{\xi^2-1}t}}{2\sqrt{\xi^2-1}}\right) \tag{3.1.45}$$

If the same analogy is undertaken as for the case of the phase step, then it can be shown that the expressions for the phase error depending of the damping factor are,

$$\phi(t) = \begin{cases} \frac{\Delta\omega}{\omega_n}\mathrm{e}^{-\xi\omega_n t}\left(\frac{\sin\omega_n\sqrt{1-\xi^2}t}{\sqrt{1-\xi^2}}\right), & \xi < 1 \\ \frac{\Delta\omega}{\omega_n}\mathrm{e}^{-\xi\omega_n t}(\omega_n t), & \xi = 1 \\ \frac{\Delta\omega}{\omega_n}\mathrm{e}^{-\xi\omega_n t}\left(\frac{\sinh\omega_n\sqrt{\xi^2-1}t}{\sqrt{\xi^2-1}}\right), & \xi > 1 \end{cases} \tag{3.1.46}$$

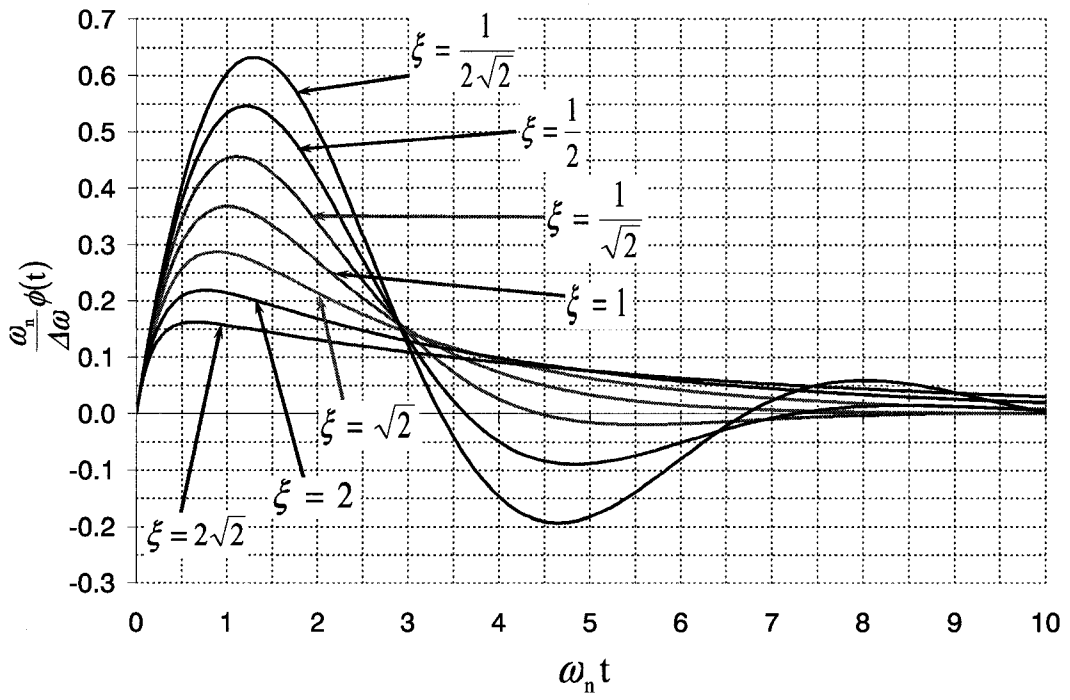


Figure 3.5: Calculated phase error of the proposed frequency synthesizer for a frequency step of the input signal.

Figure 3.5 shows the curves representing the normalized phase error, $\phi(t)\frac{\omega_n}{\Delta\omega}$, for different values of the damping factor ξ. The curves show that as the damping factor is decreasing the maximum phase error increases monitored short-after the disturbance was introduced to the system. However, for all cases the steady-state error is null after the elimination of the phase error. If a normalized phase error of 10^{-3} or smaller is monitored then a close examination of the plotted curves indicated that the choice $\xi = \frac{1}{\sqrt{2}}$ gives the fastest elimination of the normalized phase error.

Response to a Linear Frequency Variation:

The linear frequency variation can be expressed as,

$$f_{\text{Ref}}(t) = f + \Delta f t \mathtt{U}(t) \tag{3.1.47}$$

where Δf is the slope of the frequency variations expressed in Hz/s.

The time integral of the equation (3.1.47) gives the phase variation of the input signal (with respect to the normal phase $2\pi \mathtt{ft}$),

$$\phi_{\mathtt{Ref}}(t) = \Delta\omega \frac{t^2}{2} \mathtt{U}(t). \tag{3.1.48}$$

where $\Delta\omega = 2\pi\Delta f$ is the slope of the input angular frequency and is expressed in $\mathrm{rad}/\mathtt{s}^2$.

The Laplace transform gives,

$$\Phi_{\mathtt{Ref}}(s) = \frac{\Delta\omega}{\mathtt{s}^3}. \tag{3.1.49}$$

If equation (3.1.49) is substituted into the equation (3.1.21), then the expression for the phase error function when a linear frequency variation is applied to the input of the loop system is,

$$\Phi(\mathtt{s}) = \frac{\Delta\omega}{s \cdot (\mathtt{s}^2 + \mathtt{s} \cdot 2\xi\omega_n + \omega_n^2)}. \tag{3.1.50}$$

In an expanded form, the equation (3.1.50) can be written as,

$$\Phi(\mathtt{s}) = \frac{\Delta\omega}{s \cdot \left(\mathtt{s} + \omega_n\left(\xi + \sqrt{\xi^2 - 1}\right)\right)\left(\mathtt{s} + \omega_n\left(\xi - \sqrt{\xi^2 - 1}\right)\right)}. \tag{3.1.51}$$

By definition, the inverse Laplace transform of the function,

$$\Gamma(\mathtt{s}) = \frac{\mathtt{A}}{s(\mathtt{s} + \mathtt{a})(\mathtt{s} + \mathtt{b})} \tag{3.1.52}$$

where A is a constant, is,

$$\gamma(t) = \frac{\mathtt{A}}{\mathtt{ab}}\left(1 - \frac{\mathtt{b}}{\mathtt{b} - \mathtt{a}}\mathtt{e}^{-\mathtt{a}t} + \frac{\mathtt{a}}{\mathtt{b} - \mathtt{a}}\mathtt{e}^{-\mathtt{b}t}\right). \tag{3.1.53}$$

If the aforementioned definition is implemented then the expression of the phase error for the case when a linear frequency variation is applied to the input signal is,

$$\phi(t) = \frac{\Delta\omega}{\omega_n^2}\left(1 + \frac{\xi - \sqrt{\xi^2-1}}{2\sqrt{\xi^2-1}}e^{-\omega_n\left(\xi+\sqrt{\xi^2-1}\right)t} - \frac{\xi + \sqrt{\xi^2-1}}{2\sqrt{\xi^2-1}}e^{-\omega_n\left(\xi-\sqrt{\xi^2-1}\right)t}\right) \tag{3.1.54}$$

The expression (3.1.54) can also be written as,

$$\phi(t) = \begin{cases} \frac{\Delta\omega}{\omega_n^2}\left(1 - e^{-\xi\omega_n t}\left(\cos\omega_n\sqrt{1-\xi^2}t + \frac{\xi}{\sqrt{1-\xi^2}}\sin\omega_n\sqrt{1-\xi^2}t\right)\right), & \xi < 1 \\ \frac{\Delta\omega}{\omega_n^2}\left(1 - e^{-\xi\omega_n t}\left(1+\omega_n t\right)\right), & \xi = 1 \\ \frac{\Delta\omega}{\omega_n^2}\left(1 - e^{-\xi\omega_n t}\left(\cosh\omega_n\sqrt{\xi^2-1}t + \frac{\xi}{\sqrt{\xi^2-1}}\sinh\omega_n\sqrt{\xi^2-1}t\right)\right), & \xi > 1 \end{cases} \tag{3.1.55}$$

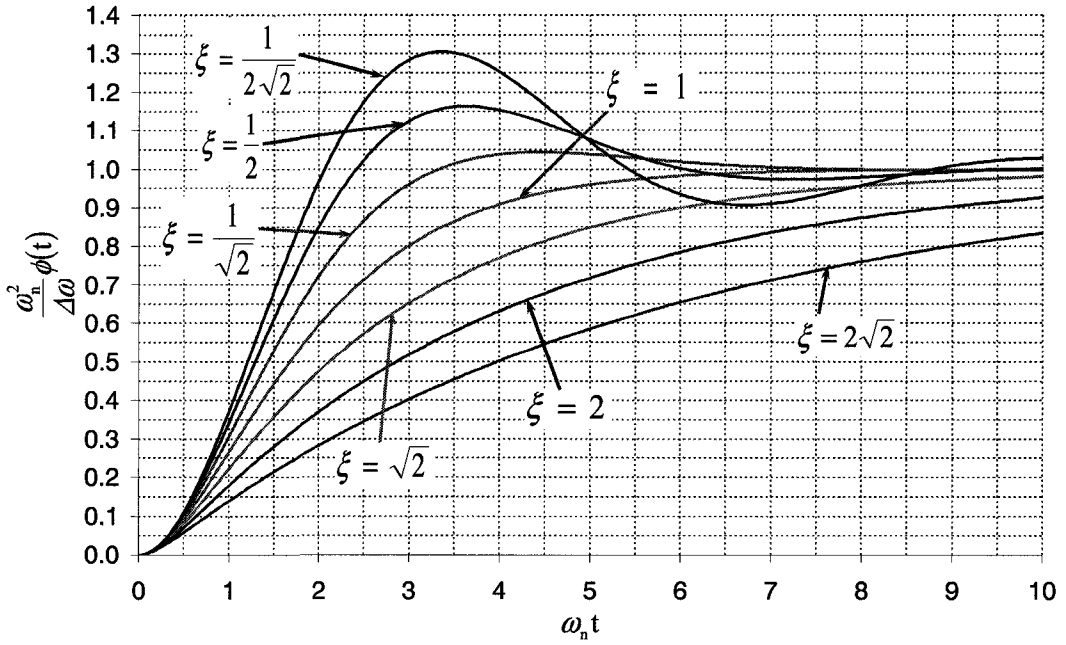


Figure 3.6: Calculated phase error of the proposed frequency synthesizer for a linear variation of the frequency of the input signal.

Figure 3.6 shows the normalized curves ($\frac{\omega_n^2}{\Delta\omega}\phi(t)$) as a function of the $\omega_n t$. The plot shows that, unlike the previous cases when the input signal faced a phase and a frequency step, the steady-state phase error is not null but equal to $\frac{\Delta\omega}{\omega_n^2}$. In order

to get a null phase error an additional integration in the loop would be required [1]. Another method is to keep $\frac{\Delta\omega}{\omega_n^2}$ small enough and the damping factor as large as possible. In that case the VCO instantaneous frequency becomes a linear function of time of slope Δf and the loop stays in lock.

3.1.4 Frequency Multiplier

A frequency multiplier is placed in the synthesizer feedback loop. The frequency multiplier is programmable, and can be implemented using available means such as a delay locked loop (DLL) architecture as discussed in [19–21], or a programmable self-adaptive frequency multiplier able to generate the output signal within one master clock period, as presented in [22]. The frequency multiplier used in the present work was of the standard integer-N frequency synthesizer variety as shown in Figure 3.7.

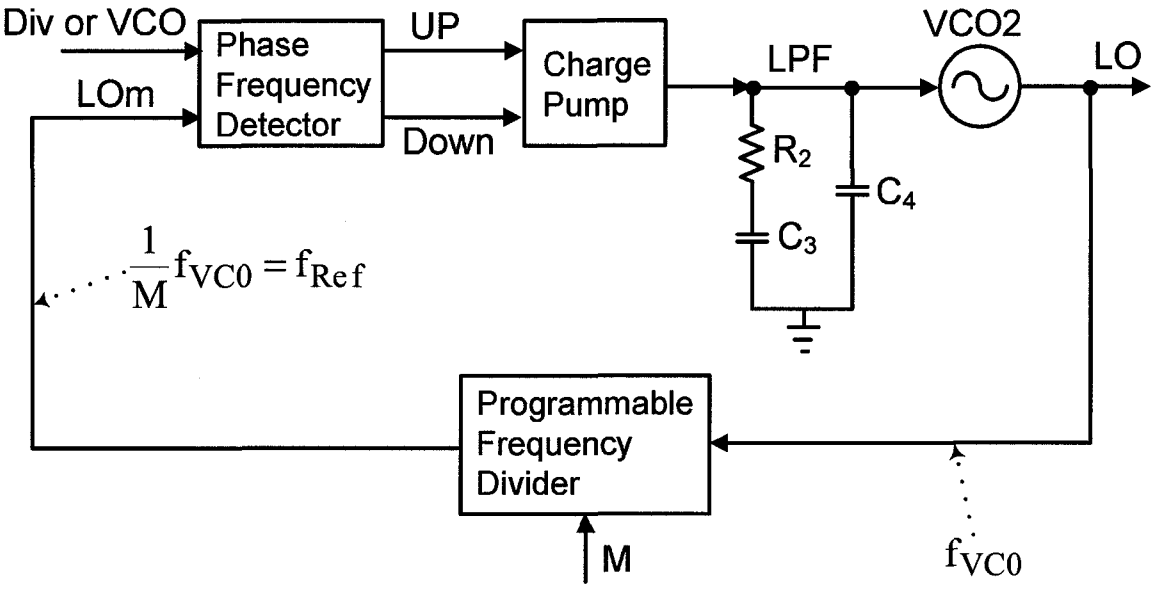


Figure 3.7: Block diagram of a PLL type frequency multiplier circuit.

Following the same analysis as discussed in section 3.1.1, the transfer function of

the frequency multiplier shown in Figure 3.7 may be written as,

$$\Phi_{\mathrm{LO}}(\mathrm{s}) = \Phi_{\mathrm{Div\ or\ VCO}}(\mathrm{s})\frac{\omega_{nM}^2\mathrm{M} + \mathrm{s}\cdot 2\xi_M\omega_{nM}\mathrm{M}}{\mathrm{s}^2 + \mathrm{s}\cdot 2\xi_M\omega_{nM} + \omega_{nM}^2} \tag{3.1.56}$$

where, ξ_M is the damping factor, and ω_{nM} is the natural frequency of the frequency multiplier defined as,

$$2\xi_M\omega_{nM} = \mathrm{R_2K_{VCO2}I_{CP}}\frac{1}{2\pi\mathrm{M}} \tag{3.1.57}$$

$$\omega_{nM}^2 = \mathrm{K_{VCO2}I_{CP}}\frac{1}{2\pi\mathrm{MC_3}}. \tag{3.1.58}$$

When the proposed system is disturbed such that the frequency multiplier is affected, then as a result of the expression (3.1.56) the error function of the proposed system can be written as,

$$\Phi(\mathrm{s}) = \Phi_{\mathrm{Ref}}(\mathrm{s})\frac{\mathrm{s}^2}{\mathrm{s}^2 + (\frac{\omega_{nM}^2+\mathrm{s}\cdot 2\xi_M\omega_{nM}}{\mathrm{s}^2+\mathrm{s}\cdot 2\xi_M\omega_{nM}+\omega_{nM}^2})(\mathrm{s}\cdot 2\xi\omega_n + \omega_n^2)} \tag{3.1.59}$$

where the damping factor, ξ, and the natural frequency, ω_n, of the proposed system are defined with the expressions (3.1.16) and (3.1.17), respectively.

In order for the proposed system to re-acquire a lock condition, the frequency multiplier should re-acquire its lock first. Once the frequency multiplier locks, then the term in the expression (3.1.59) caused by the frequency multiplier is equal to one,

$$\lim_{s\to 0}\left(\frac{\omega_{nM}^2 + \mathrm{s}\cdot 2\xi_M\omega_{nM}}{\mathrm{s}^2 + \mathrm{s}\cdot 2\xi_M\omega_{nM} + \omega_{nM}^2}\right) = 1 \tag{3.1.60}$$

and the error function of the proposed system may be written as,

$$\Phi(\mathrm{s}) = \Phi_{\mathrm{Ref}}(\mathrm{s})\frac{\mathrm{s}^2}{\mathrm{s}^2 + \mathrm{s}\cdot 2\xi\omega_n + \omega_n^2} \tag{3.1.61}$$

or equal to the expression (3.1.21).

In other words, in the case of a PLL type of a frequency multiplier, the frequency

multiplier needs to re-acquire the lock state first before the proposed system re-acquires the lock state. The transition time of the proposed system is equal to the time that is needed so the phase error of the frequency multiplier is eliminated, plus the time the proposed system returns to a lock state.

3.1.5 Phase Noise Analysis

The phase noise from each individual block of the PLL is shaped through the loop system before it affects the phase noise of the output signal. Except for the phase noise of the VCO, which is output referred, the phase noise of all other blocks is input referred. Thus, there are two transfer functions of the loop system that would shape the phase noise.

The magnitude of the transfer functions that shapes the phase noise of the reference signal, the PFD-CP, the frequency multiplier, the frequency divider and the loop filter is,

$$\left|\frac{\Phi_{\mathtt{out}}(\omega)}{\Phi_{\mathtt{input}}(\omega)}\right| = \left|\frac{\Phi_{\mathtt{VCO}}(\omega)}{\Phi_{\mathtt{Ref}}(\omega)}\right| = \left(\frac{\mathtt{N}}{\mathtt{M}}\omega_n^2\right)\sqrt{\frac{1+(\mathtt{R}_1\mathtt{C}_1\omega)^2}{(\omega_n^2-\omega^2)^2+(2\xi\omega_n)^2\omega^2}}. \tag{3.1.62}$$

where ξ and ω_n are the damping constant and the natural frequency of the loop given, respectively.

On the other hand, the magnitude of the transfer functions that shapes the phase noise of the VCO signal is,

$$\left|\frac{\Phi_{\mathtt{outVCO}}(\omega)}{\Phi_{\mathtt{inputVCO}}(\omega)}\right| = \frac{\omega^2}{\sqrt{(\omega_n^2-\omega^2)^2+(2\xi\omega_n)^2\omega^2}}. \tag{3.1.63}$$

Phase noise due to the reference signal: If the phase noise of the reference signal is denoted as $\mathtt{PNoise}_{\mathtt{Ref}}$ then the output signal would see the following phase

noise,

$$\mathtt{PN_{Ref}} = \left| \frac{\Phi_{\mathtt{out}}(\omega)}{\Phi_{\mathtt{input}}(\omega)} \right| \cdot \mathtt{PNoise_{Ref}} \tag{3.1.64}$$

expressed in dB as,

$$(\mathtt{PN_{Ref}})_{\mathtt{dB}} = 20\log_{10}\left(\left| \frac{\Phi_{\mathtt{out}}(\omega)}{\Phi_{\mathtt{input}}(\omega)} \right| \cdot \mathtt{PNoise_{Ref}} \right). \tag{3.1.65}$$

Phase noise due to the frequency multiplier: If the phase noise of the frequency multiplier (FM) is denoted as $\mathtt{PNoise_{FM}}$ then the output signal would see the following phase noise,

$$\mathtt{PN_{FM}} = \left| \frac{\Phi_{\mathtt{out}}(\omega)}{\Phi_{\mathtt{input}}(\omega)} \right| \cdot \mathtt{PNoise_{FM}} \tag{3.1.66}$$

expressed in dB as,

$$(\mathtt{PN_{FM}})_{\mathtt{dB}} = 20\log_{10}\left(\left| \frac{\Phi_{\mathtt{out}}(\omega)}{\Phi_{\mathtt{input}}(\omega)} \right| \cdot \mathtt{PNoise_{FM}} \right). \tag{3.1.67}$$

Phase noise due to the PFD and the CP: The open loop noise from the PFD and the CP can be expressed as,

$$\mathtt{Noise_{PFDCP}} = \frac{i_n}{\mathtt{K_{phase}}} \tag{3.1.68}$$

where i_n is the output noise current from the CP, while $\mathtt{K_{phase}}$ is the gain of the PFD-CP and, for a PFD with a sawtooth characteristic, can be expressed as [2],

$$\mathtt{K_{phase}} = \frac{\mathtt{I_{CP}}}{2\pi}. \tag{3.1.69}$$

The phase noise contribution from the PFD and the CP in the loop is,

$$\mathtt{PN_{PFDCP}} = \left| \frac{\Phi_{\mathtt{out}}(\omega)}{\Phi_{\mathtt{input}}(\omega)} \right| \cdot \mathtt{Noise_{PFDCP}} \tag{3.1.70}$$

expressed in dB as,

$$(\mathrm{PN_{PFDCP}})_{\mathrm{dB}} = 20\log_{10}\left(\left|\frac{\Phi_{\mathrm{out}}(\omega)}{\Phi_{\mathrm{input}}(\omega)}\right| \cdot \mathrm{Noise_{PFDCP}}\right). \tag{3.1.71}$$

Phase noise due to the VCO: If the phase noise of the VCO signal is denoted as $\mathrm{PNoise_{VCO}}$ then the output signal would see the following phase noise,

$$\mathrm{PN_{VCO}} = \left|\frac{\Phi_{\mathrm{outVCO}}(\omega)}{\Phi_{\mathrm{inputVCO}}(\omega)}\right| \cdot \mathrm{PNoise_{VCO}} \tag{3.1.72}$$

expressed in dB as,

$$(\mathrm{PN_{VCO}})_{\mathrm{dB}} = 20\log_{10}\left(\left|\frac{\Phi_{\mathrm{outVCO}}(\omega)}{\Phi_{\mathrm{inputVCO}}(\omega)}\right| \cdot \mathrm{PNoise_{VCO}}\right). \tag{3.1.73}$$

Total phase noise: Assuming that the phase noise due to the loop filter is negligible, then the total phase noise, in dB, in the proposed loop system is,

$$(\mathrm{PN_{total}})_{\mathrm{dB}} = 20\log_{10}(\mathrm{PN_{Ref}} + \mathrm{PN_{FM}} + \mathrm{PN_{PFDCP}} + \mathrm{PN_{VCO}}). \tag{3.1.74}$$

If the reference signal is generated from a high quality crystal oscillator, then the phase noise due to the reference signal can be ignored as well. In such a case, the frequency multiplier, the PFD, the CP and the VCO are the major contributors to the total phase noise in the proposed system,

$$(\mathrm{PN_{total}})_{\mathrm{dB}} = 20\log_{10}(\mathrm{PN_{FM}} + \mathrm{PN_{PFDCP}} + \mathrm{PN_{VCO}}). \tag{3.1.75}$$

3.2 Divide-Multiply Implementation

A synthesizer implementing the divide-multiply concept was designed to operate from 1V supply in a 0.13μm CMOS technology. The frequency of the reference signal was chosen as 20MHz, and the outer-loop VCO was designed to operate with a center frequency of 10GHz with a tuning range of 750MHz [23]. The charge pump was designed to generate 65μA current [24], while the PFD was designed with a sawtooth characteristic [24]. A multi-modulus frequency divider [25] was used such that the channel spacing was 500kHz. A PLL-based frequency multiplier was implemented, including a 20MHz ring oscillator within the frequency multiplier. The aforementioned values were used for illustration purposes only.

The loop bandwidth of the frequency multiplier was sized with a damping factor of 0.707 and a natural frequency of 90kHz. The loop bandwidth of the main system (Figure 3.1) was sized with a damping factor of 0.707 and a natural frequency of 30kHz. The aforementioned values (the damping factor and the natural frequency) were selected in order to accomplish a fast switching speed without compromising the stability of the loop system.

Figure 3.8 shows the acquisition time for the aforementioned case scenario. The proposed system was initially forced to acquire lock at 10GHz. The control signal for the VCO was settled at 500mV. At 50μs the system was forced to lock at the neighboring channel separated by 500kHz. After approximately 35μs ($\pm$0.02% of the final control voltage value) the control signal for the VCO settled at a new value of 500.67mV and the frequency synthesizer generated a new signal with a frequency of

10.0005GHz.

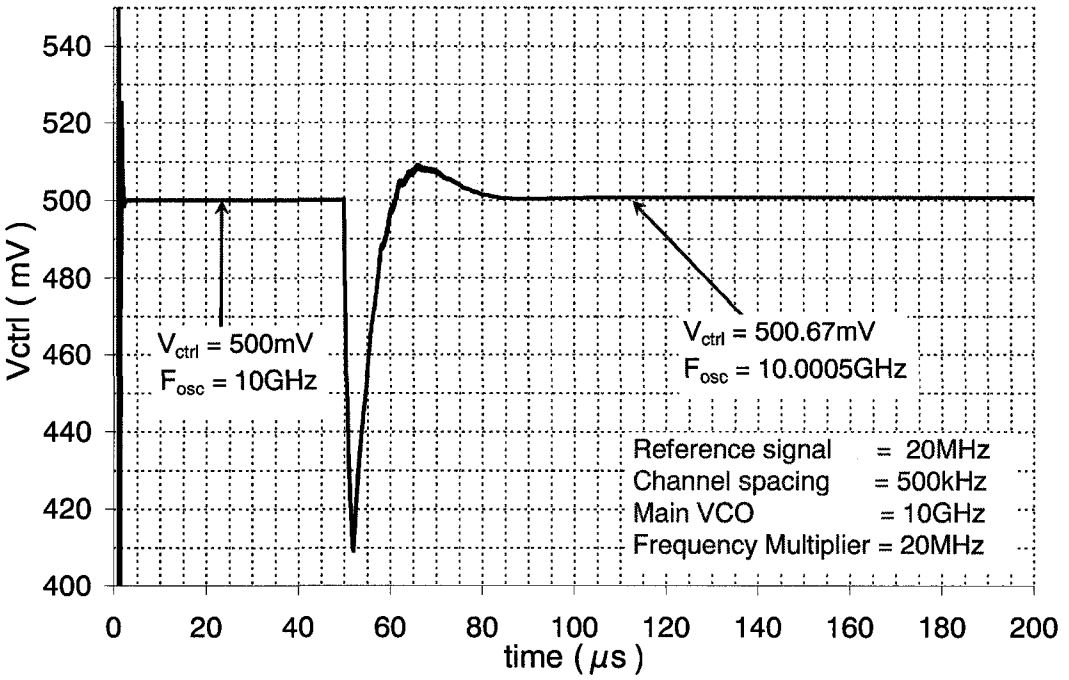


Figure 3.8: Acquisition time for the divide-multiply implementation.

The frequency multiplier acquires lock to the signal coming out from the frequency divider. In a lock state, the frequency of that signal is equal to the resolution frequency (500kHz in the particular example). As a result of the non-idealities of the PLL blocks of the frequency multiplier, ripple, caused by the resolution frequency, can appear on the loop voltage within the frequency multiplier. This ripple will cause spurious tones to appear on the output signal from the frequency multiplier, denoted as LO. The phase frequency detector of the proposed system will try to match the frequency and the phase of the LO signal and the frequency and the phase of the Ref signal. Due to the non-idealities of the PLL blocks of the main system, ripple, caused by the reference signal (20MHz in the particular example) and the resolution frequency, will appear

on the loop voltage of the proposed system. This ripple will cause reference and fractional spurs to appear in the spectrum of the output signal from the proposed system. Thus, it is expected that the divide-multiply implementation can have a problem with the fractional spurious tones. Practically, however, there are three factors that reduce the ripple due to the resolution frequency. First, the ripple that appear on the loop voltage within the frequency multiplier are reduced due to the implemented low pass filter (for the particular example the LPF of the frequency multiplier was set to 90kHz). Second, the PFD and the CP can reduce the spurs as follows. As discussed in Chapter 5, this thesis work utilizes a PFD with a linear characteristic. The function of the PFD is to match the frequency and the phase of the LO signal to the frequency and the phase of the reference signal (from the main system). Thus, if a linear type of a PFD is implemented (within the main system) then the PFD would help the proposed system to acquire lock to the desired LO signal. In addition, the CP is designed to reduce the ripple on the loop voltage. Finally, the loop filter of the main system is even smaller than the loop filter within the frequency multiplier (for the particular example the main loop filter is set to 30kHz). Consequently, the fractional spurious tones have been reduced through the loop system without the need of any known fractional spurs cancellation techniques implemented with the $\Delta\Sigma$ frequency synthesizers, for example.

To monitor the frequency spectrum of the generated signal with a frequency of 10.0005GHz, a DFT is performed on this signal within the time period where the system is locked and the frequency spectrum is shown in Figure 3.9. The noise floor of the output spectrum is around -120dB. The noise floor of the main VCO (operating

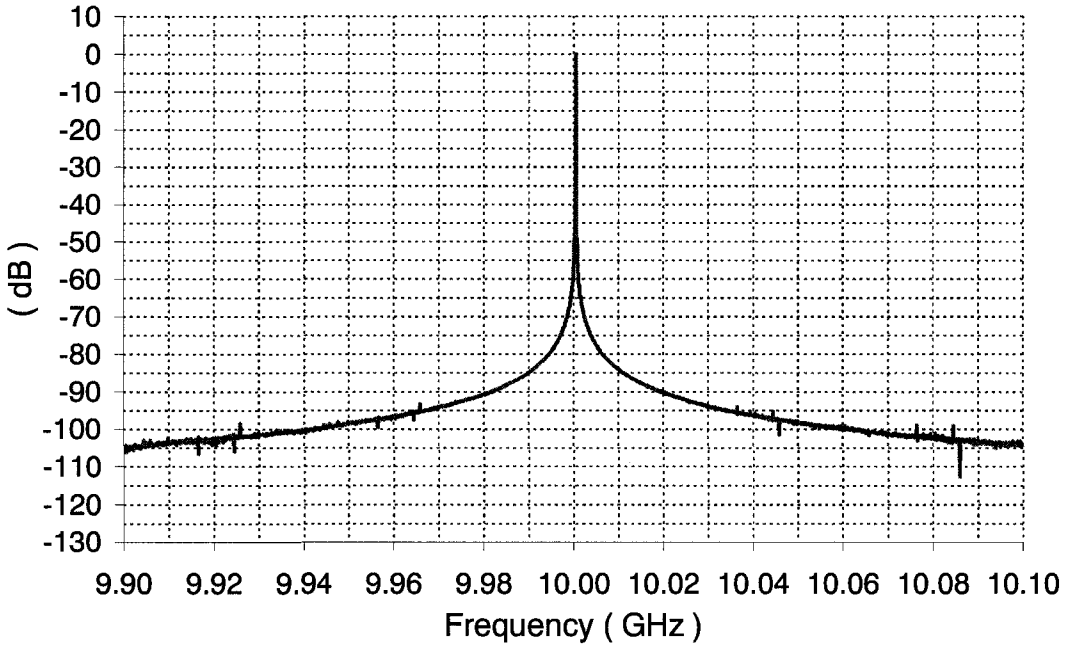


Figure 3.9: Divide-multiply implementation: DFT of the output signal from the frequency synthesizer.

at 10GHz in the particular example) is about -135dB as discussed in [23].

Figure 3.10 shows the estimated total phase noise of the generated signal from the proposed divide-multiply implementation that implements a PLL type of a frequency multiplier. The curves are drawn based on the theory discussed in section 3.1.5. It is assumed that the phase noise of the reference signal is small and can be ignored. As shown on the plot, when the phase noise due to the frequency multiplier is output-referred, then it becomes the main contributor to the total phase noise of the frequency synthesizer. Considering the frequency multiplier, an additional investigation indicated that the main contributor to the phase noise from the frequency multiplier is the phase noise due to the phase frequency detector and the charge pump (PFD-CP). Although the results are based on the design choices while

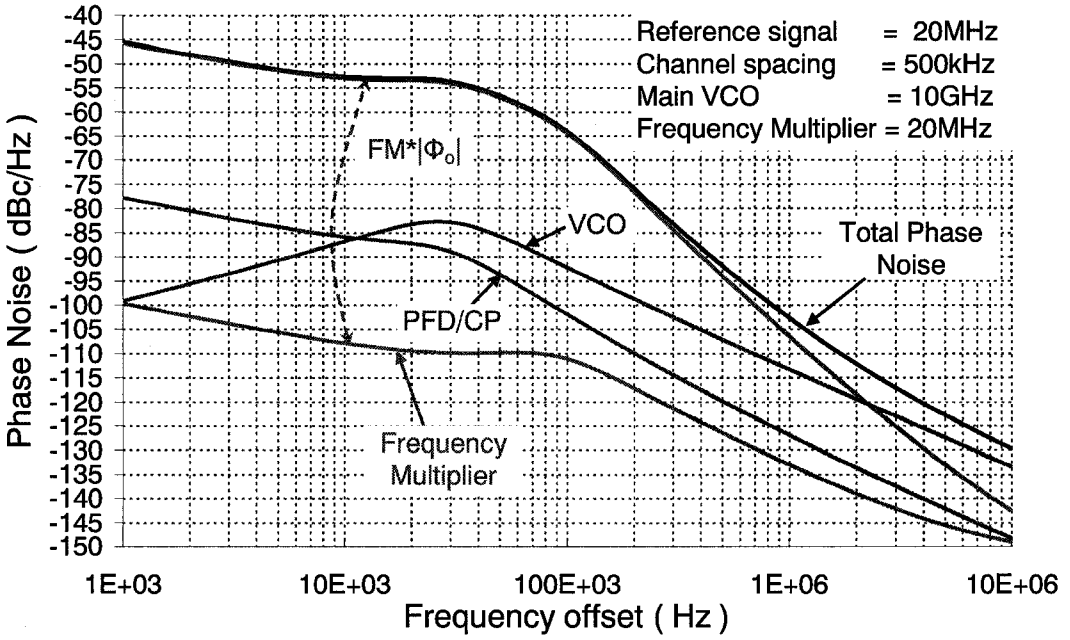


Figure 3.10: A particular example of the phase noise of the divide-multiply implementation.

implementing the proposed concept, it should give a picture that the phase noise from a PLL type of a frequency multiplier determines the overall phase noise performance of the divide-multiply implementation.

An alternative of the divide-multiply implementation is the multiply-divide implementation. The simulated results of an example of this implementation are discussed and compared to the simulated results of the divide-multiply implementation in Appendix A.

Although the multiply-divide implementation achieves better performance regarding the phase noise and the switching speed, the main disadvantage of the multiply-divide implementation is that the frequency of operation of the frequency multiplier

is higher than the application frequency. Consequently, the divide-multiply architecture was fabricated in a CMOS 0.13μm technology. The photograph of the fabricated chip, the test setup for the measurements, and screen shots of the measured results are discussed in Chapter 7. The following chapters discuss the custom cells used to design the divide-multiply implementation.

3.3 Summary

A new architectural concept for a frequency synthesizer implementing a subsystem of frequency multipliers and frequency dividers within its feedback loop was proposed and discussed in this chapter. Three possible implementations of the proposed system, including their block diagrams, were presented.

The first-half of this chapter discussed the transient and the phase noise analysis of the proposed loop system. The first advantage of the proposed system is that the theory built for the integer-N frequency synthesizers is applicable for the proposed frequency synthesizer as well. The effect of the selected topology of the frequency multiplier on the transient analysis is also discussed.

The second-half of this chapter discussed a frequency synthesizer implementing the divide-multiply concept. The frequency synthesizer was designed to operate from 1V supply in a 0.13μm CMOS technology. The frequency of the reference signal was chosen as 20MHz, and the outer-loop VCO was designed to operate with a center frequency of 10GHz and channel spacing of 500kHz. The loop bandwidth of the divide-multiply implementation, incorporating a PLL type of a frequency multiplier, is limited by the channel resolution. The small loop bandwidth, along with the proper

design of the PFD and the CP, causes the fractional spurious tones to be reduce through the loop system without the need of any known fractional spurs cancellation techniques implemented with the $\Delta\Sigma$ frequency synthesizers. The advantages and the disadvantages of the divide-multiply implementation are discussed in details in Chapter 7.

Chapter 4

Programmable Frequency Divider

This chapter depicts the implemented architecture of the frequency divider used with the divide-multiply implementation. A frequency divider based on a 2/3 divider cell found in [26] is used as a reference. Other readings regarding the frequency dividers can be found in [27–32].

4.1 Divider Architecture

The programmable frequency divider architecture is shown in Figure 4.1. The modular structure consists of a chain of 2/3 divider cells as described in [26] and [28]. The operation of the frequency divider is explained in [26].

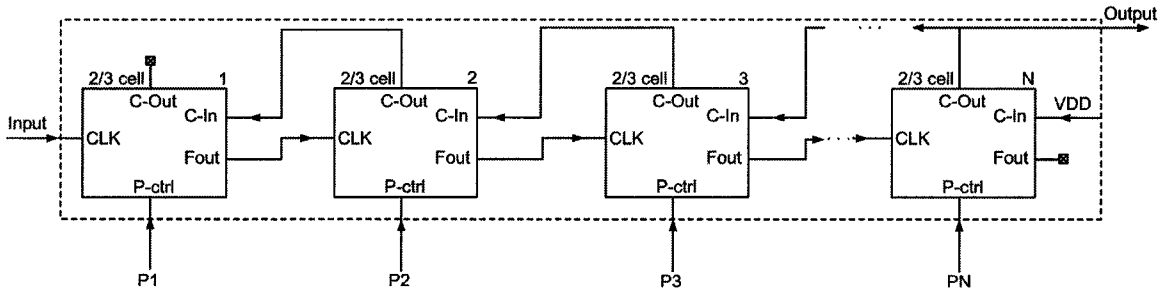


Figure 4.1: Block diagram of the frequency divider's architecture.

Figure 4.2 shows the functional blocks and the logic implementation of a 2/3

divider cell. The aforementioned divider cell is similar to the divider cell in [26]. The inclusion of clock tree buffers, used to boost the frequency of operation of the divider, and the "resistive technique" described in the following section, used to implement the functional blocks, are two differences between the referenced work and this work. The functional description of the divider cell is given in [26]. The advantages and the disadvantages of the modified CML custom cells for the circuit implementation of the 2/3 divider cells are discussed in the following section.

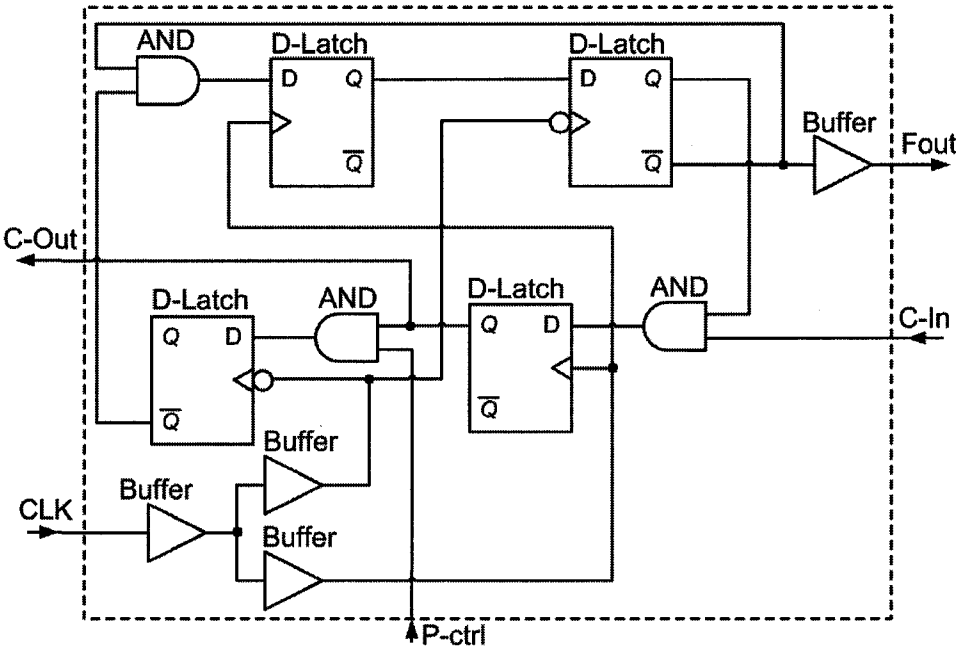
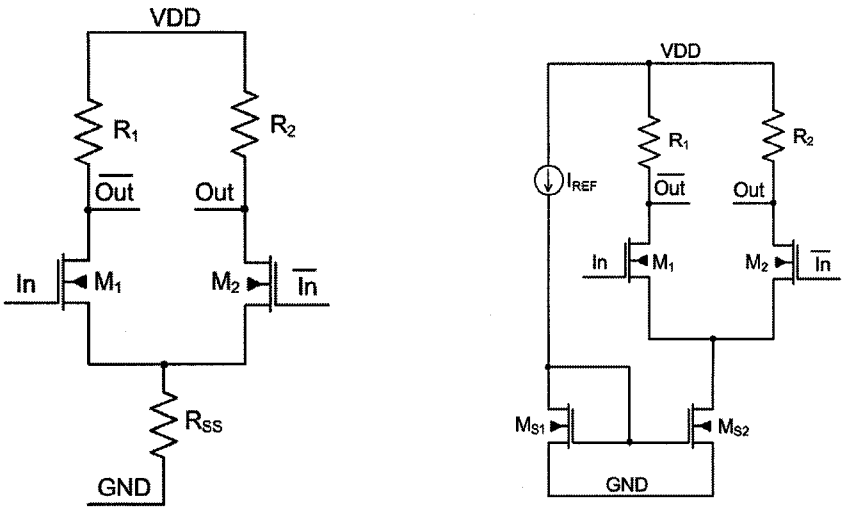


Figure 4.2: Functional blocks and logic implementation of a single divide by 2/3 cell.

4.2 Differential Gates with Resistor Tail Bias

Current mode logic (CML) is a good choice for high speed applications [33–35]. The differential operation of CML helps in improving the noise performance of the system, and CML exhibits power dissipation independent of the operating frequency. However, as the power supply is reduced the circuit design becomes a challenge. A

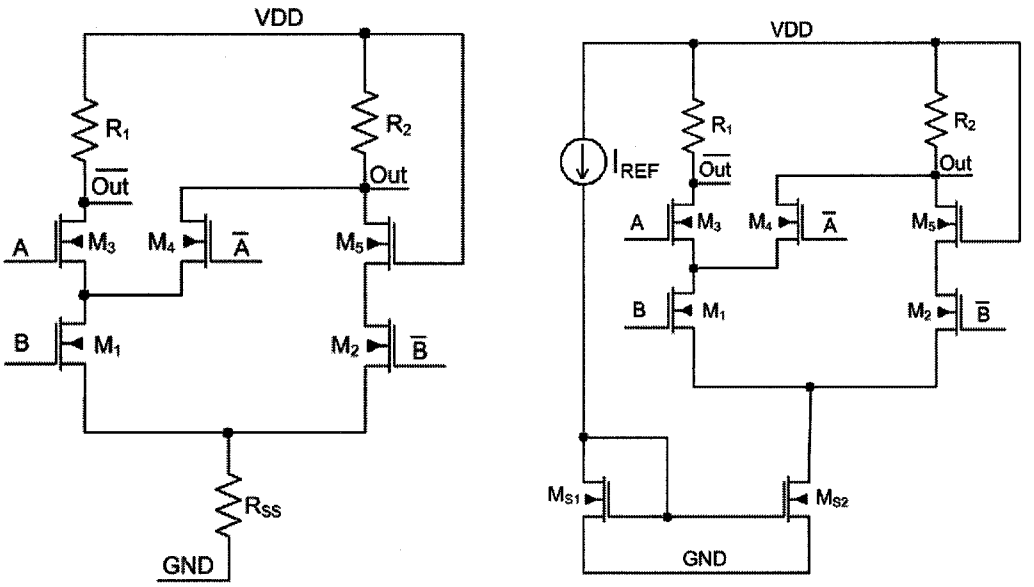


(a) with a source resistor. (b) with a current mirror.

Figure 4.3: Differential buffer gate

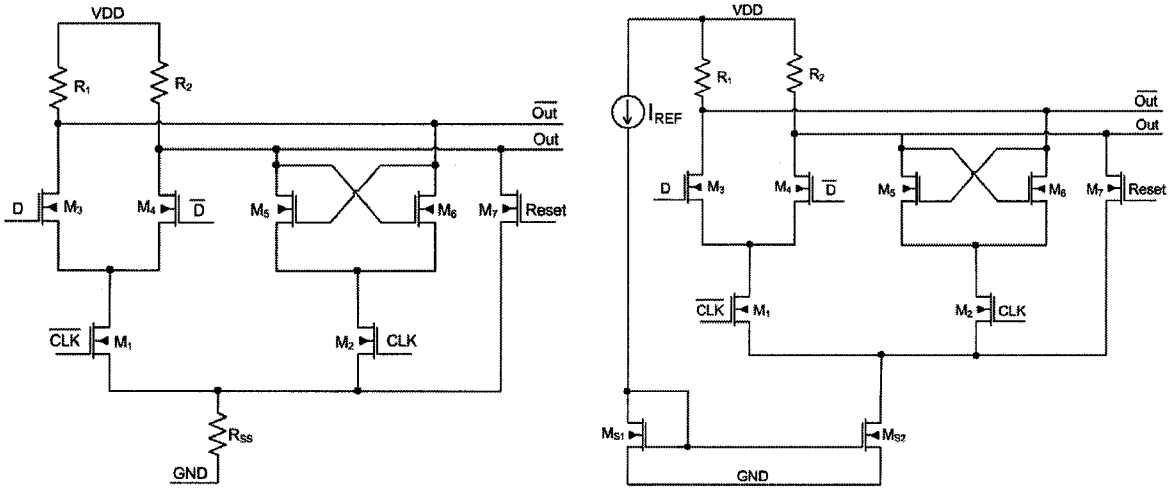
major problem while designing applications operating at several GHz is to bias and to keep the current mirrors in their active region regardless of the process variations during the fabrication. In addition, the size of the current mirrors increases with the increase of the frequency. To overcome these problems, this thesis recommends the current mirrors found in the conventional CML logic be replaced with a common source resistor.

Figures 4.3(a), 4.4(a) and 4.5(a) show the buffer gate, AND gate and D-Latch gate used to implement programmable frequency dividers. As shown, the current mirrors found in the conventional CML logic shown in Figure 4.3(b), 4.4(b), and 4.5(b) are replaced with a common source resistor.



(a) with a source resistor. (b) with a current mirror.

Figure 4.4: Differential AND gate



(a) with a source resistor. (b) with a current mirror.

Figure 4.5: Differential D-Latch gate

By replacing the current source with a resistor, the following advantages are achieved:

1. simplified design;

2. reduced power supply;

3. eliminated source of 1/f noise.

A potential disadvantage of the resistor tail bias technique could be the power supply rejection ratio (PSRR).

With the elimination of the current mirror, three or more components are replaced with only one resistor. It was estimated that the layout area occupied by the common source resistor is approximately $13.6\mu m^2$ while the layout area occupied by the current mirror is about $80\mu m^2$. Thus, the layout savings per one CML gate is estimated to be $66.4\mu m^2$. The layout savings for a 8-bit programmable frequency divider implementing 8-2/3 divider cells is estimated to be approximately $5843.2\mu m^2$. The quoted figures are for a specific implementation in a $0.13\mu m$ CMOS technology and involve a specific current ratio, and are therefore only examples. Obviously the current mirror ratio will determine the saved area.

The common source resistor allows reduction of the power supply down to 1V[1]. Moreover, the current mirror is a source of 1/f noise and a thermal noise, while the resistor is a source of a thermal noise. Since the 1/f noise is much higher at lower frequencies compared to the thermal noise, the close-in phase noise introduced by the custom cell with a common source resistor is reduced.

[1]Through the simulated results was found that operation from a 0.8V supply is also possible.

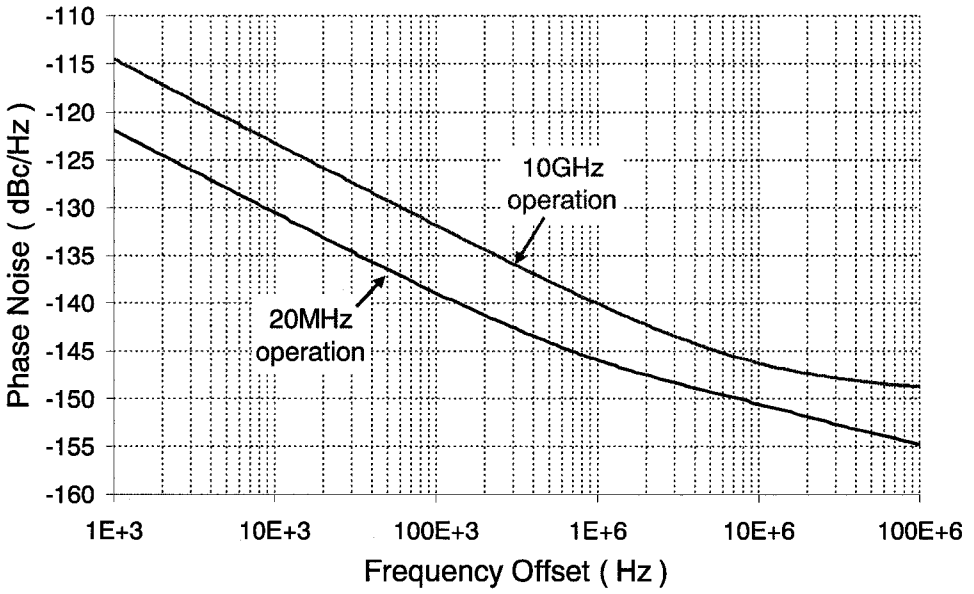


Figure 4.6: Phase noise from a single divide by 2/3 frequency divider at 10GHz and 20MHz operation.

Figure 4.6 shows the simulated phase noise of a single 2/3 divider cell operating at 10GHz and 20MHz. In both cases, the divider cell divides the input frequency by 2 resulting in two output signals with frequencies 5GHz (input 10GHz) and 10MHz (input 20MHz). The phase noise is simulated with SpectreRF, by applying an ideal sinusoidal signal at the input of the divider cell. The simulated phase noise at 1kHz offset from a 5GHz carrier is better than -115dBc/Hz, while the simulated phase noise at 1kHz offset from a 10MHz carrier is better than -122dBc/Hz. At a 100MHz offset, the simulated phase noise is -149dBc/Hz and -155dBc/Hz for the 5GHz and 10MHz carriers, respectively. The waveforms of the signals at lower frequency exhibit sharper rising edges. As a consequence, the switching time of the dividers building cells is reduced. The reduced switching time reduced the phase noise as well.

A 900MHz signal with an amplitude of 100mV was added in series to the power supply of a single 2/3 divider cell in order to simulate the power supply rejection

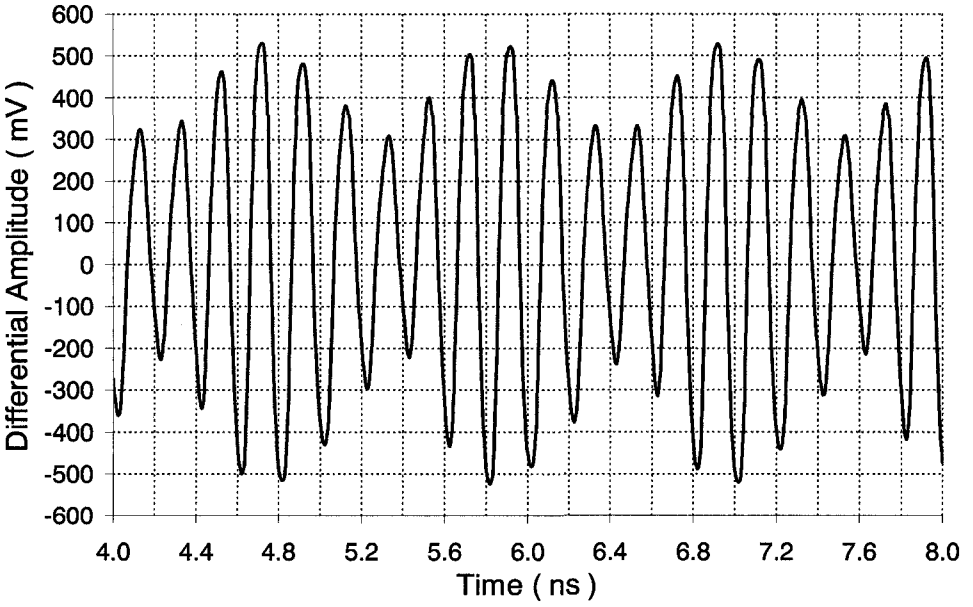


Figure 4.7: Distortion of the output signal from a single 2/3 divider cell at 10GHz due to 900MHz signal with an amplitude of 100mV coupled on the power supply line. The divider cell divides the frequency by 2 and operates from 1V supply.

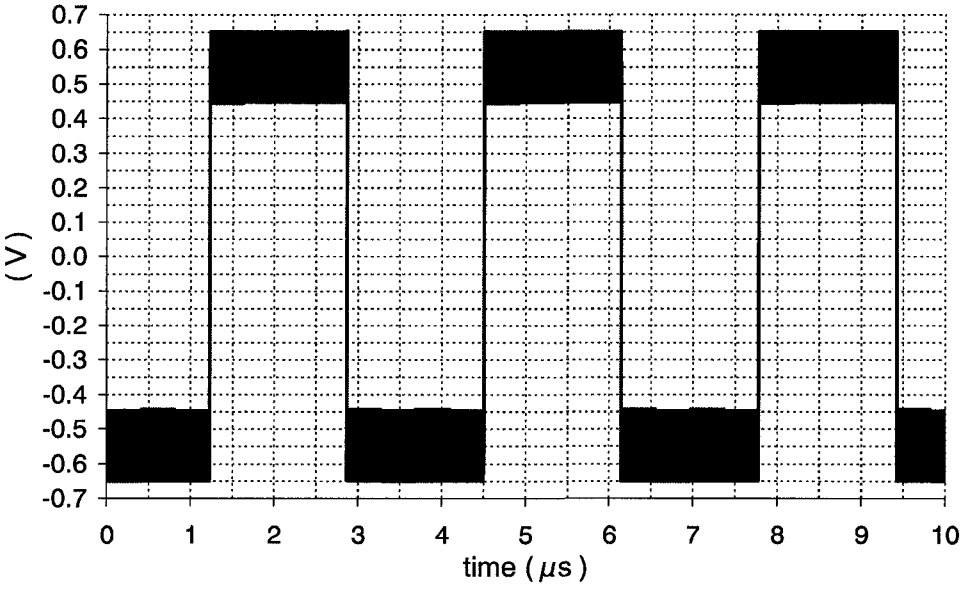


Figure 4.8: Simulated waveform of the output signal from the 14-bit frequency divider operating at 10GHz due to 900MHz signal with an amplitude of 100mV coupled on the power supply line. The programmable bits are set to logic one, and the frequency divider operates from 1V supply.

ratio (PSRR). Although the frequency of the input signal was correctly divided by 2, an amplitude modulation of the output signal was noticed as shown in Figure 4.7. A higher amplitude of the coupled signal to the power supply line would have more negative effect.

The divide-multiply implementation of the proposed architectural concept for a frequency synthesizer utilized a 14-bit frequency divider. A post-layout simulation was performed to determine the effect of a coupled signal on the power supply. Similarly to the previous simulation of a single 2/3 divider cell, a 900MHz signal with an amplitude of 100mV was added in series to the power supply. A 10GHz signal was applied at the input of the frequency divider programmed to divide by 32767. Figure 4.8 shows the output waveform from the 14-bit frequency divider. The input frequency was correctly divided and the added 900MHz signal had almost no effect on the transition edge of the waveform. This is particularly important for PLL applications because the signal from the frequency divider goes to a phase frequency detector. The phase frequency detector (used with the divide-multiply implementation) is sensitive to the rising edges of the input signals (the reference signal and the signal from the frequency divider). Thus, as long as the output signal from the frequency divider has well defined rising edges, the phase frequency detector would made a correct decision.

In conclusion to the two simulations regarding the power supply rejection ratio, if the frequency of the output signal from the frequency divider is higher compared to the frequency of the coupled signal on the power supply (for example 5GHz compared to 900MHz) then the amplitude modulation of the output signal is a problem. However,

if the frequency of the output signal from the frequency divider is lower compared to the frequency of the coupled signal (for example 305.2kHz compared to 900MHz), then the coupled signal would not modify the transition edges of the output signal resulting in a correct functionality of the phase frequency detector.

One possible solution that can be used in order to reduce the effect of the coupled signal on the power supply and to improve the waveform of the output signal from the frequency divider is to add an off-chip 10μF capacitor between the power supply and ground. On-chip power supply regulative circuit as well as a proper chip level physical design can also be considered in order to reduce the effect of the coupled signal on the power supply.

A Monte Carlo simulation was used to test the sensitivity of the implemented frequency dividers to the process and mismatch variations. The post-layout simulation was performed on a single 2/3 divider cell optimized to work at 10GHz.

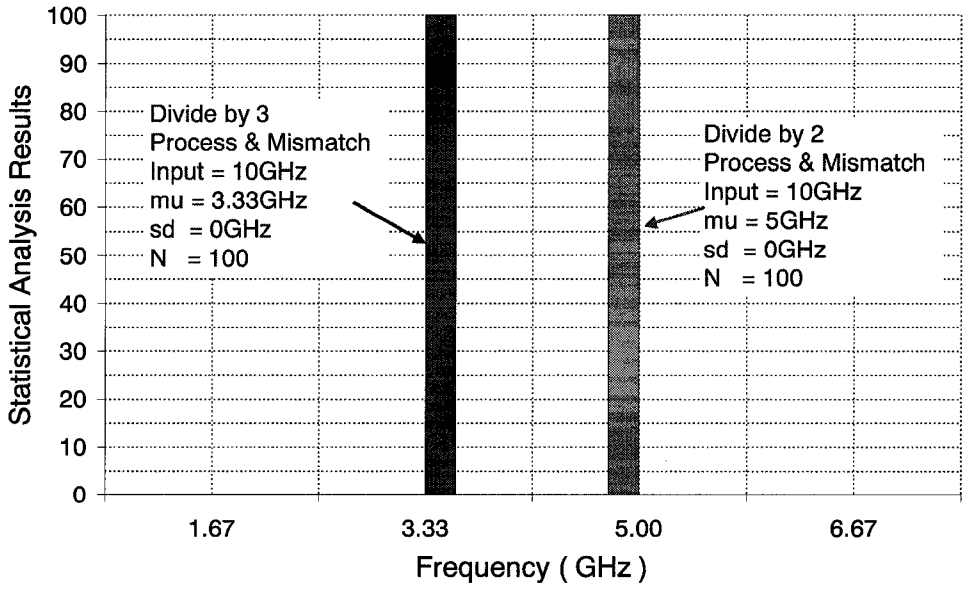


Figure 4.9: Monte Carlo simulations at 10GHz operation: the single 2/3 divider cell was programmed to divide by 2 and by 3, in two separate cases.

Figure 4.9 shows the results of the Monte Carlo simulation when a 10GHz signal was applied to a 2/3 divider cell configured to divide by 2 and to divide by 3 from a 1V supply, respectively. From the total number of one hundred runs, the frequency of the input signal was always correctly divided.

The Monte Carlo simulation was extended by applying a 20MHz signal to the input of the divider cell. The purpose of this simulation was to investigate the frequency range applicable to the divider cell previously optimized to operate at higher frequencies. Figure 4.10 shows that, when the 2/3 divider cell was configured to divide by 2 from a 1V supply, not always the frequency of the input signal was divided by 2. From the total number of one hundred runs, ninety six times the frequency of the output signal form the divider was 10MHz or exactly divided by 2. However in four cases the output frequency from the divider was different from the expected result. As a result, the average output frequency from the divider was 10.4MHz with a standard deviation of 1.96MHz.

Figure 4.11 shows the results of the Monte Carlo simulation when a 20MHz signal was applied to a 2/3 divider cell configured to divide by 3 from a 1V supply. From the total number of one hundred runs, ninety four times the frequency of the output signal was 6.67MHz or exactly divided by 3. However in six cases the output frequency from the divider was different from the expected result. As a result, the average output frequency from the divider was 6.97MHz with a standard deviation of 1.5MHz.

The conclusion of the Monte Carlo simulations is that, for a constant power dissipation, the lower frequency of operation of a single 2/3 divider cell, implementing the resistor bias technique and optimized to operate at high frequencies, is limited.

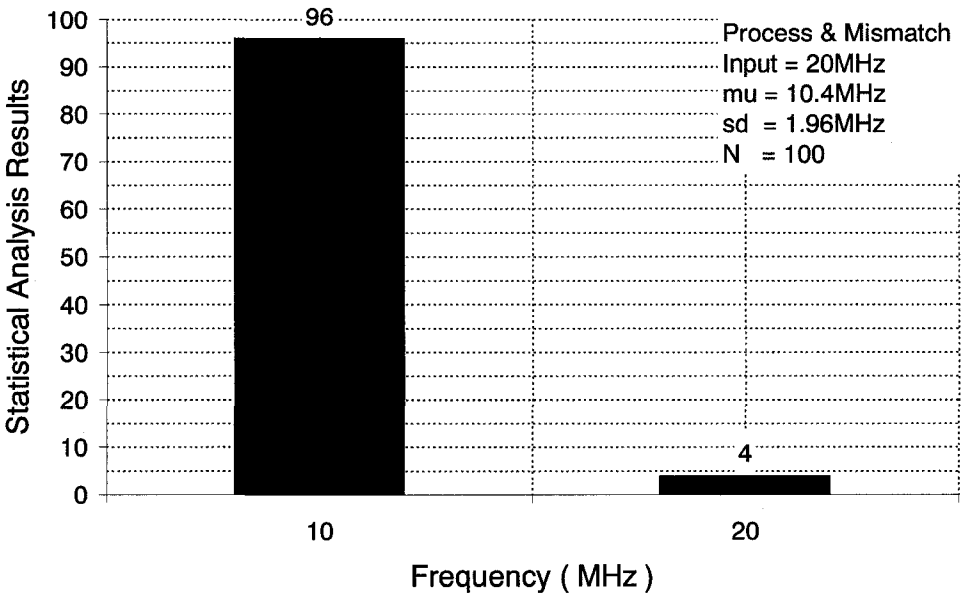


Figure 4.10: Monte Carlo simulation of a single 2/3 divider cell configured to divide by 2 at 20MHz operation.

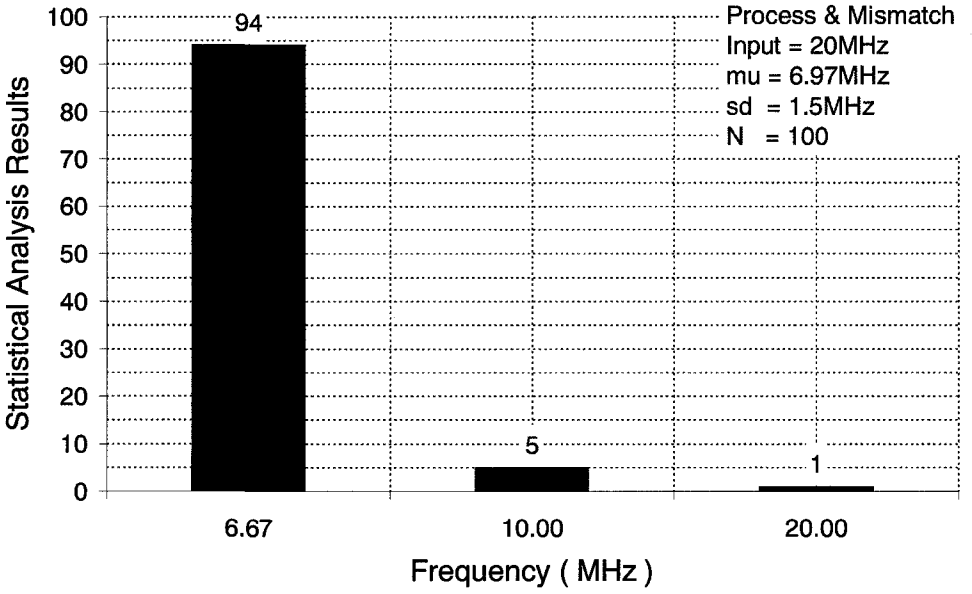


Figure 4.11: Monte Carlo simulation of a single 2/3 divider cell configured to divide by 3 at 20MHz operation.

Therefore, a divider cell optimized for high frequencies would need additional resizings of the custom differential cells in order to use the same divider cell at lower frequencies. Nevertheless, in order to reduce the power dissipation, it is a good practice to optimize the divider cell for the targeted frequency of operation as discussed in [26].

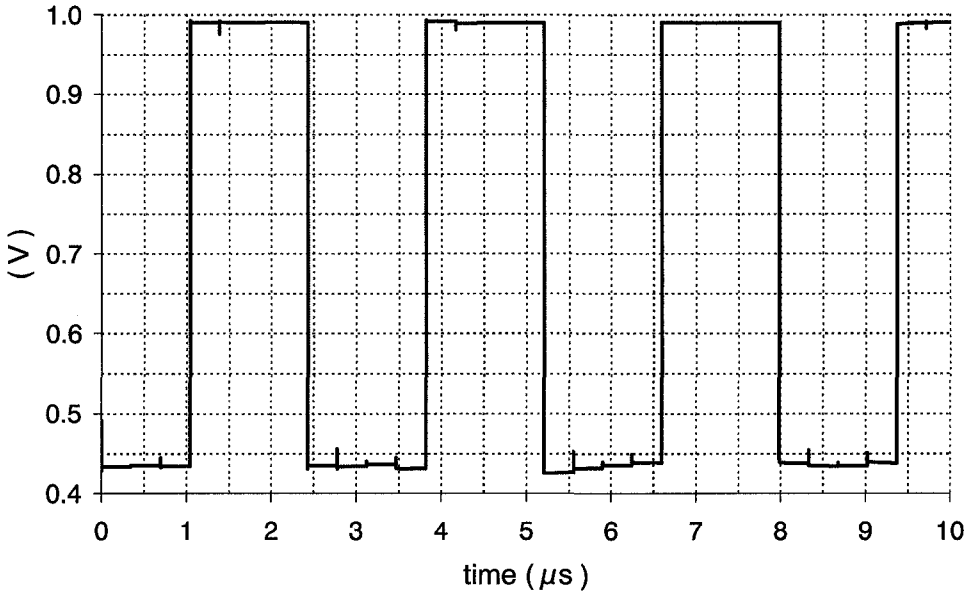


Figure 4.12: Single-ended output waveform from the 14-bit frequency divider at 11.8GHz of operation. The programmable bits of the frequency divider are set to logic one (divide by 32767) and the frequency divider operates from 1V supply.

Finally, Figure 4.12 shows the single-ended waveform of the output signal from the 14-bit frequency divider. The post-layout simulation was used to determine the upper frequency range of the frequency divider operating from 1V power supply. It was found that, with 1V power supply, the frequency of the input signal could be increased as high as 11.8GHz for a correct operation of the 14-bit frequency divider.

4.3 Summary

This chapter depicted the architecture of the frequency divider used with the divide-multiply implementation. In order to design a programmable 10GHz frequency divider to operate from a reduced power supply down to 1V, this thesis proposed modification of the CML gates. The modification included replacement of the tail current source with a passive resistive element. The new differential cells with resistor tail bias have dynamic current supply compared to the static current supply of the classic CML gates. The advantages and the disadvantages of the resistive tail biasing technique was demonstrated through simulation of a single 2/3 divider cell as well as a 14-bit frequency divider.

The major concern of a single 2/3 divider cell is the power supply rejection ratio. A distortion of the output signal from the single divider cell was illustrated for the case when the power supply was accompanied with a signal which frequency was lower compared to the frequency of the output signal from the frequency divider. It was also illustrated that a similar scenario would have a minor effect on the 14-bit frequency divider.

The advantages of the frequency divider implementing the novel differential cells are simplified design, improved phase noise, reduced power supply, and stable GHz operation due to the process and the mismatch variations.

Chapter 5

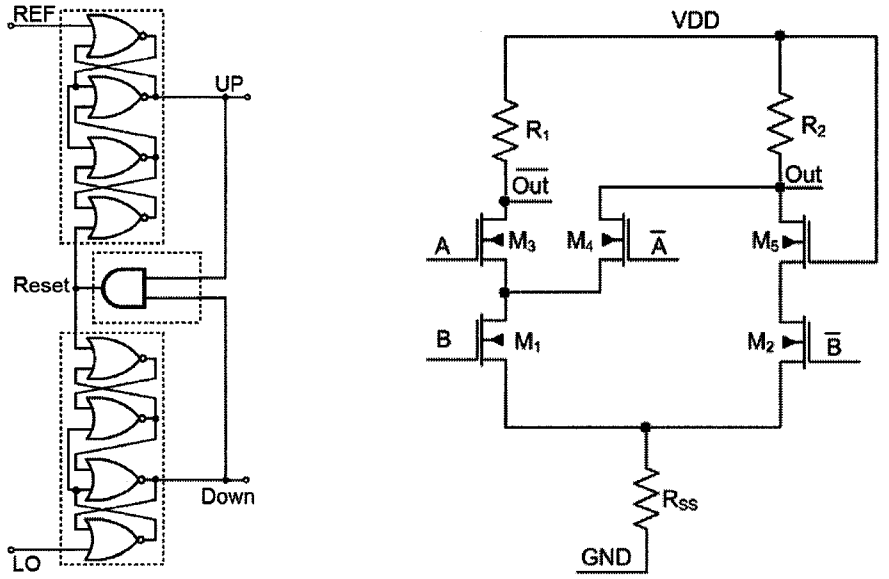
The PFD and the CP Block

The phase frequency detector (PFD) and the charge pump (CP) implemented with the divide-multiply implementation are discussed in this chapter.

5.1 Phase Frequency Detector

The PFD considered with this work is shown in Figure 5.1(a). The PFD consists of two flip-flops (FF) [36] and one AND gate to generate the reset signals. It should be noted that the PFD is a fully differential and for simplification it is drawn as single-ended. There are two building blocks of the PFD: an AND and a NOR gate. The AND gate is depicted in Figure 5.1(b). The tail current source found in a classical implementation of the CML gates is replaced with a common source resistor. This topology was used to reduce the layout area. However, it also helps to reduce the low-frequency content of the noise coming from the circuit [25]. The NOR gate is identical to the AND gate however with swapped inputs.

The function of the PFD is to align the phase and the frequency of the reference signal (Ref) and the signal from the local oscillator (LO). The function is performed by comparing either the rising or falling edges of the Ref and the LO signal. Because



(a) PFD (b) Differential AND gate [25]

Figure 5.1: Phase frequency detector.

the implemented PFD with this work compares the rising edges of the aforementioned signals, the remaining of this document will discuss the function of the PFD accordingly.

If the time when the rising edge of the `Ref` signal arrives is denoted as $t_{\texttt{Ref}}$, and the time when the rising edge of the `LO` signal arrives is denoted as $t_{\texttt{LO}}$, then upon comparing the rising edges of the `Ref` and `LO` signals, the PFD will generate an `UP` signal if the rising edge of the `Ref` signal comes before the rising edge of the $t_{\texttt{LO}}$,

$$\texttt{UP}\,(t) = \begin{cases} \texttt{U}(t) - \texttt{U}(t - \texttt{T}) & t_{\texttt{Ref}} - t_{\texttt{LO}} > 0 \\ 0 & t_{\texttt{Ref}} - t_{\texttt{LO}} \leq 0 \end{cases} \tag{5.1.1}$$

where $\texttt{U(t)}$ is the unit step function, and $\texttt{T} = t_{\texttt{Ref}} - t_{\texttt{LO}}$ is the pulse width of the $\texttt{UP}\,(t)$ signal.

Upon comparing the rising edges of the `Ref` and `LO` signals, the PFD will generate a `Down` signal if the rising edge of the `LO` signal comes before the rising edge of the

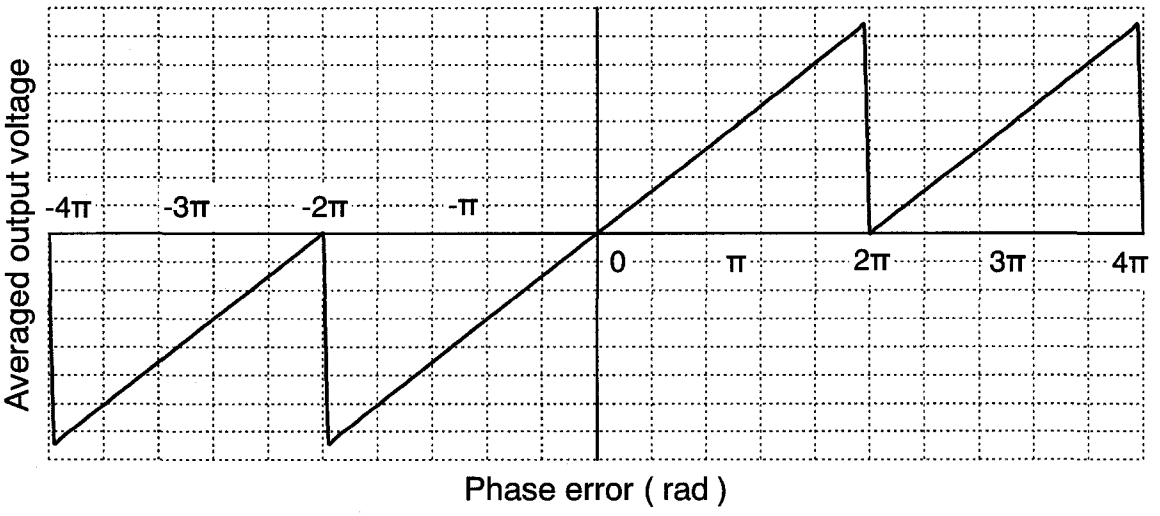


Figure 5.2: Simulated PFD characteristic.

Ref signal,

$$\texttt{Down}\,(t) = \begin{cases} \texttt{U}(t) - \texttt{U}(t - \texttt{T}) & t_{\texttt{LO}} - t_{\texttt{Ref}} > 0 \\ 0 & t_{\texttt{LO}} - t_{\texttt{Ref}} \leq 0 \end{cases} \tag{5.1.2}$$

To obtain the PFD characteristic, two periodic square signals with clock frequency of 20MHz are used to feed the PFD inputs designated as "REF" and "LO". The delay of the REF signal is fixed to 0, while the delay of the LO is used as a variable. A parametric analysis, as a part of the SpectreRF simulator, is used to monitor the average output voltage from the PFD when the delay of the LO signal is varied from -100ns to +100ns. With a clock frequency of 20MHz that implies that the phase of the LO signal is varied from -4π rad to +4π rad. Figure 5.2 shows the simulated PFD characteristic. The plot indicates that the PFD has no "dead" zone[1] and has a linear tracking characteristic over 2π radians of phase error.

[1]A "dead" zone is a region, typically located around zero radians when the phase error is small, where the PFD is loosing it's sensitivity and ability to make correct decisions.

5.2 Charge Pump

Figure 5.3 shows the novel charge pump design used for the research work reported herein. The implemented charge pump has a differential input and a single-ended output. The output signals from the phase frequency detector, designated as "UP" and "Down", are input to the charge pump. The UP differential signals are fed to a p-type differential pair CMOS transistors, while the Down differential signals are fed to an n-type differential pair CMOS transistors.

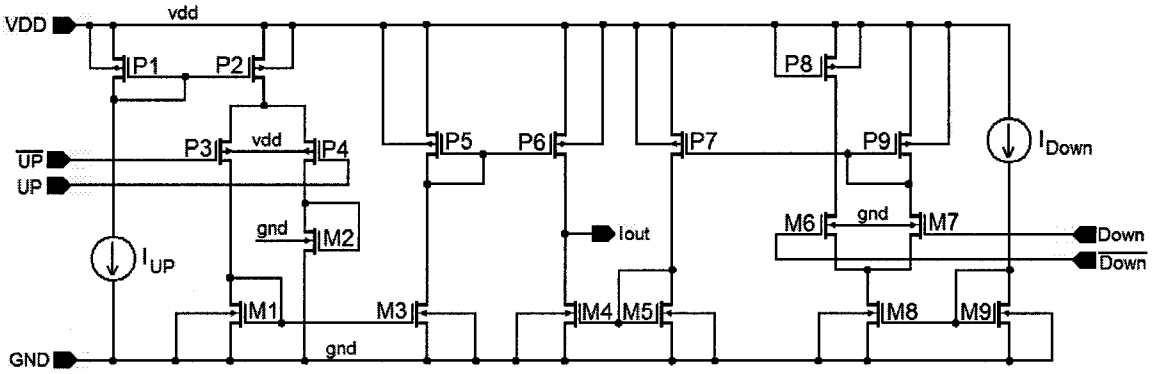


Figure 5.3: Proposed design for a charge pump.

The function of the charge pump is to sink or source a charge to a low pass filter (LPF). For this purpose, the proposed charge pump uses two current sources denoted as "I_{UP}" and "I_{Down}". In this circuit, the input differential pairs (P3-P4 for UP signals and M6-M7 for Down signals) steer current either to a dummy load (M2 for UP signals and P8 for Down signals) or into the two current sources (M1 and P9), respectively. The signal path from the current sources to the charge pump output is equal for the both UP and Down signals (two n-channel and two p-channel transistors).

The proposed charge pump is a modification of the charge pump found in [3] pp. 212 and a bipolar version found in [37]. However, there are several important

differences between the proposed and referenced charge pumps.

The charge pumps in [3, 37] use one current source, while the charge pump presented herein utilizes two current sources. In the charge pumps [3, 37], the UP and Down signals, coming from the PFD, are fed to n-channel differential pairs. Consequently, there is an extra current direction for the Down signals, causing a current matching issue in those designs. The charge pump proposed in this thesis has two complementary differential pairs for the UP and Down signals. Thus, in terms of the number of transistors, the UP and the Down signals are having an equal path from the switching pair to the output of the charge pump. Chapter 8 discusses that the two current sources of the charge pump may be adjusted via a calibration circuit to optimize the charge pump characteristic behavior.

A Monte Carlo simulation was used to analyze the output current from the transistors P6 and M4 due to process and mismatch variation. The output voltage of the charge pump was set so that the currents from P6 and M4 were equal. The number of runs of the Monte Carlo simulation was set to 100. Figures 5.4 and 5.5 show the results for the current from M4 and P6, respectively. The standard deviation of the current from the M4 transistor is 6.16μA, while the standard deviation of the current from the P6 transistor is 4.7μA. The average current from the M4 transistor is 63.4μA, while the average current from the P6 transistor is 63.6μA. Thus, the difference between the two currents is 200nA or deviation of -0.31% due to process and mismatch variations. The work found in [38] practically took the charge pump found in [3, 37] and introduced a regulated cascode circuit to improve the current matching. The charge pump in [38] was implemented in a 0.18μm technology and output 100μA

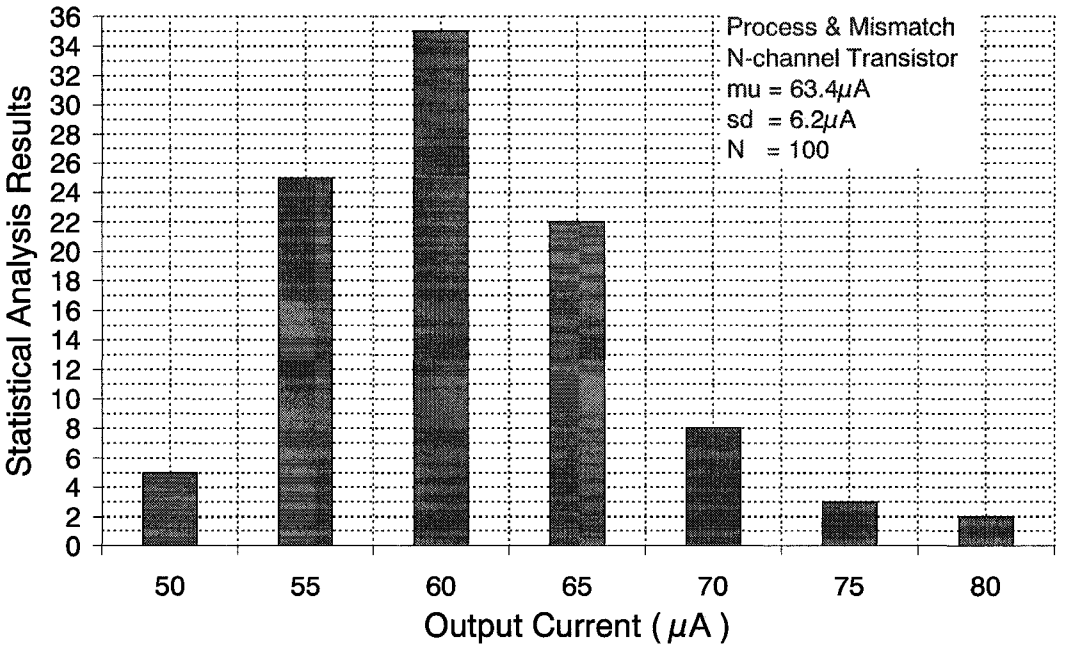


Figure 5.4: Monte Carlo simulation to analyze the output current from the n-channel transistor due to process and mismatch variation.

current. The simulated results found in [38] show that if the charge pump is not optimized, as shown in [3,37], then the mismatch between the UP and Down currents is 24μA. However, if the charge pump is optimized through the use of a regulated cascode circuit, then the deviation between the UP and Down currents is reported to be 1%. Therefore, the current mismatching of the charge pump presented in this thesis work is improved by 3.2 times compared to [38] and it will be shown in Chapter 8 that this improved current matching can be attained for the complete working region of the proposed charge pump.

Figure 5.6 shows the output currents from the n-channel and p-channel transistor of the charge pump as a function of the temperature. The worst deviation of -0.5% is noticed at the upper region when the temperature is set to 100° Celsius.

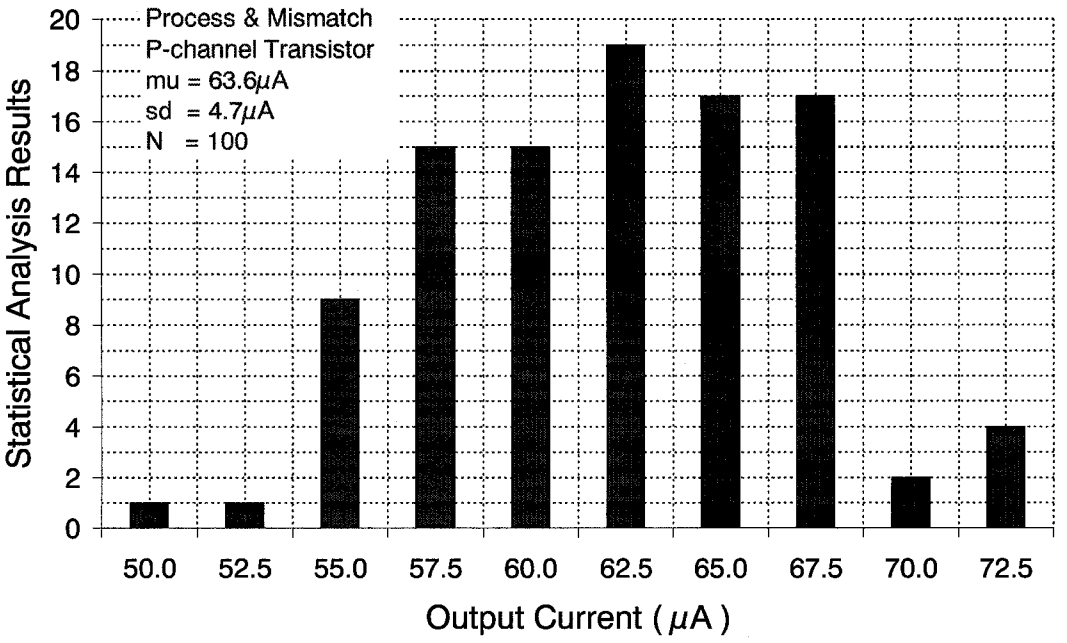


Figure 5.5: Monte Carlo simulation to analyze the output current from the p-channel transistor due to process and mismatch variation.

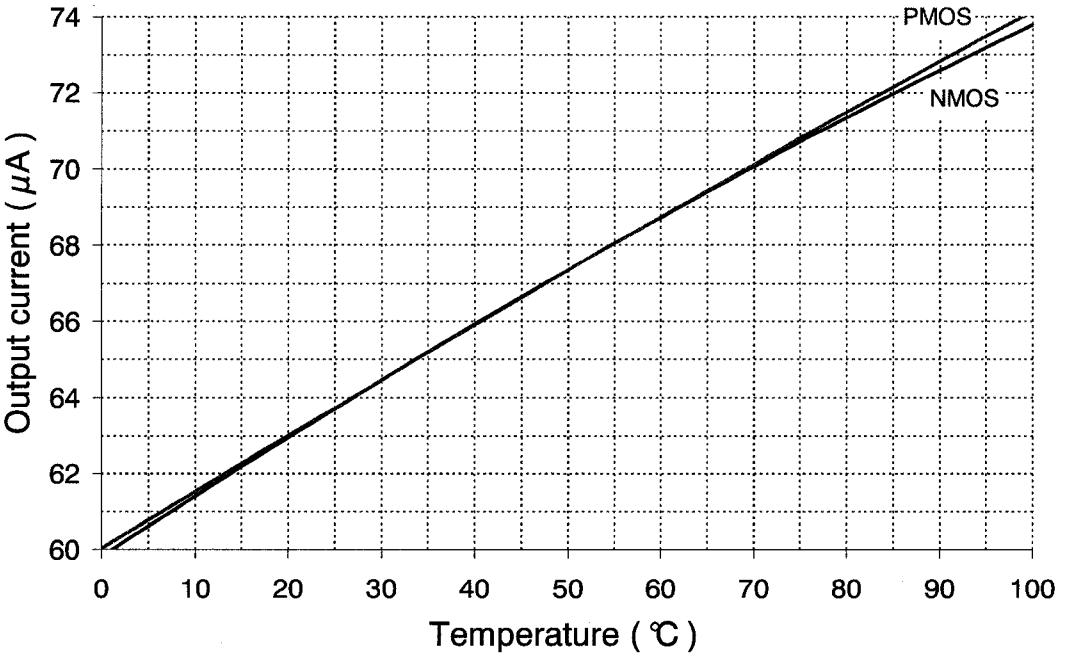


Figure 5.6: Plot of the output current from the n-channel and the p-channel transistor due to temperature variation.

5.3 Phase Noise Contribution

The noise contribution from the phase frequency detector and the charge pump was discussed in the subsection 3.1.5 of the Chapter 3. For completeness, the discussion for the noise contribution from the PFD and the CP in a PLL system is repeated in this part again with more details.

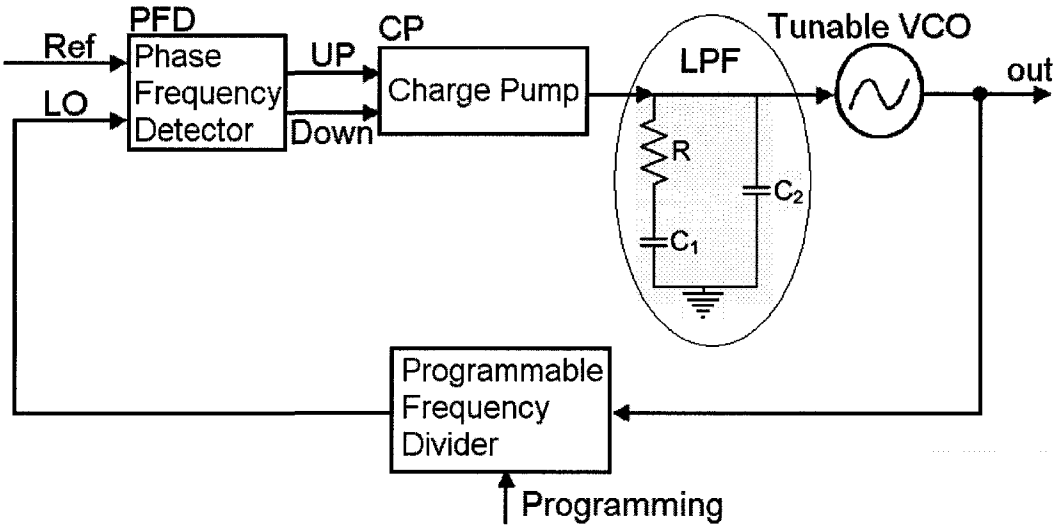


Figure 5.7: Block diagram of a charge pump based phase locked loop.

The open loop noise from the PFD and the CP can be expressed as,

$$\mathtt{Noise}_{\mathtt{PFDCP}} = \frac{i_n}{\mathtt{K}_{\mathtt{phase}}} \tag{5.3.1}$$

where i_n is the output noise current from the CP, while $\mathtt{K}_{\mathtt{phase}}$ is the gain of the PFD.

To estimate the phase noise due to the PFD and the CP in a closed loop system, the transfer function of the feedback system shown in Figure 5.7 can be written as,

$$\frac{\Phi_{\mathtt{out}}(\mathtt{s})}{\Phi_{\mathtt{input}}(\mathtt{s})} = \frac{\mathtt{F}(\mathtt{s})\,\mathtt{K}_{\mathtt{vco}}\mathtt{K}_{\mathtt{phase}}}{\mathtt{s} + \mathtt{F}(\mathtt{s})\,\mathtt{K}_{\mathtt{vco}}\mathtt{K}_{\mathtt{phase}}\frac{1}{\mathtt{N}}} \tag{5.3.2}$$

where $\mathtt{N}$ is the division ratio of the frequency divider.

If the capacitor C_2 is sized to be 10-times smaller than the capacitor C_1, then the transfer function of the LPF can be simplified to,

$$F(s) = R + \frac{1}{sC_1}. \tag{5.3.3}$$

The transfer function of the PFD-CP system is,

$$K_{phase} = \frac{I_{CP}}{2\pi}. \tag{5.3.4}$$

If the expressions (5.3.3) and (5.3.4) are substituted into the expression (5.3.2), and the following substitutions are performed,

$$2\xi\omega_n = \frac{I_{CP}K_{vco}}{2\pi N}R \tag{5.3.5}$$

$$\omega_n^2 = \frac{I_{CP}K_{vco}}{2\pi NC_1} \tag{5.3.6}$$

where ξ and ω_n are the damping constant and the natural frequency of the loop, respectively, then,

$$\left|\frac{\Phi_{out}(\omega)}{\Phi_{input}(\omega)}\right| = \left(N\omega_n^2\right)\sqrt{\frac{1+(RC_1\omega)^2}{\left(\omega_n^2-\omega^2\right)^2+\left(2\xi\omega_n\right)^2\omega^2}}. \tag{5.3.7}$$

The phase noise contribution from the PFD and the CP in the considered loop system expressed in dB is,

$$PN_{PFDCP} = 20\log_{10}\left(\left|\frac{\Phi_{out}(\omega)}{\Phi_{input}(\omega)}\right| \cdot Noise_{PFDCP}\right). \tag{5.3.8}$$

Figure 5.8 shows the phase noise contribution from the PFD and the CP for a damping constant of a 0.707, a division ratio of 500, a reference frequency of 20MHz, and two cases for the natural frequency (100kHz and 1MHz). As expected, a LPF with a lower cut-off frequency will help to reduce the phase noise from the PFD and the CP.

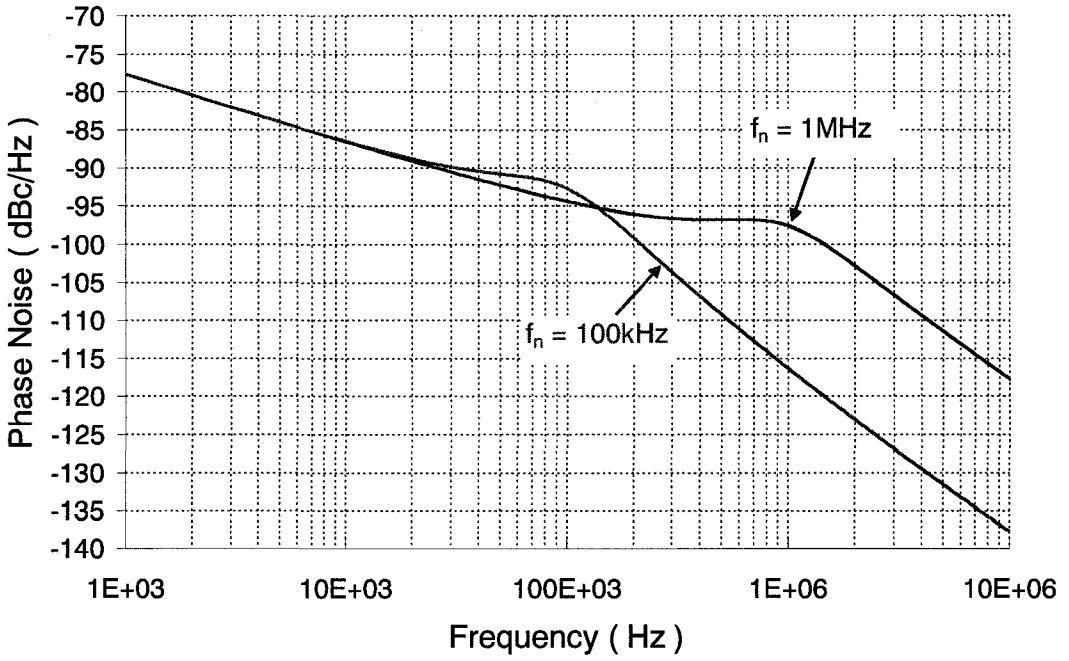


Figure 5.8: Phase noise contribution from the PFD and the CP block for a 10GHz VCO output frequency.

5.4 Summary

The design and performances of the PFD and the CP as building blocks of the divide-multiply implementation were discussed in this chapter. The PFD exhibited a linear tracking characteristic, and implemented the differential gates as discussed in Chapter 4. The charge pump had two complementary current sources, and two complementary differential pairs for the UP and Down signals from the PFD. A Monte Carlo simulation was used to analyze the deviation of the output current from the charge pump due to process and mismatch variation. Compared to the referenced work in the literature, the current mismatching of the proposed charge pump was improved by 3.2 times. Chapter 8 discusses that the improved current matching can be attained for the complete working region of the proposed charge pump.

Chapter 6

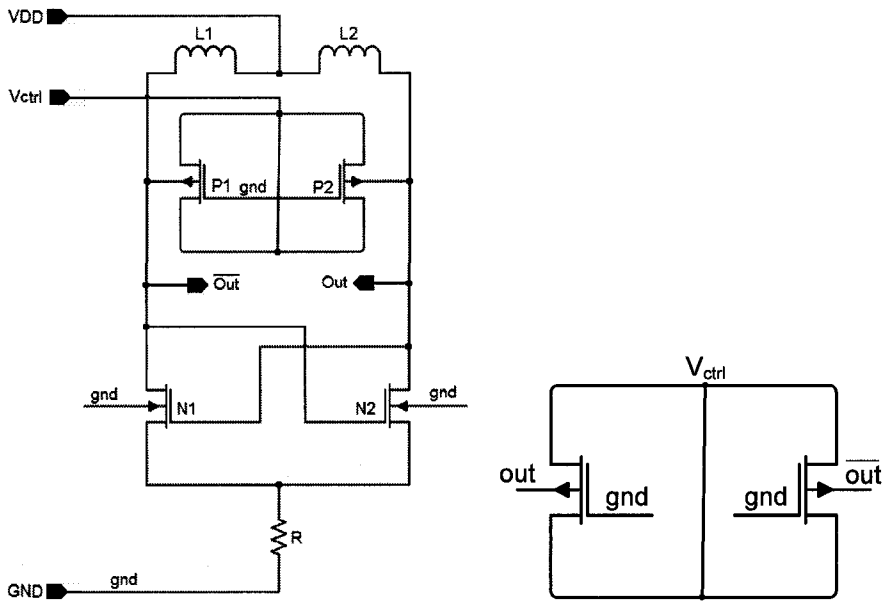
Voltage-Controlled Oscillator

This chapter discusses the implemented voltage-controlled oscillators with the divide-multiply implementation.

6.1 LC VCO

Figure 6.1(a) shows the schematic of the LC VCO implemented with the divide-multiply implementation. As shown, the VCO has a gain stage, inductors and varactors. The gain stage consists of cross coupled n-channel transistors. Because the VCO is sensitive to noise during the zero crossings, the VCO architecture with a resistor instead a current source can give a better phase noise performance [39]. The varactors are implemented from p-channel transistors, denoted as P1 and P2. The voltage signal that tunes the VCO is brought to the source (S) and the drain (D) of the p-channel transistors. The bulk (B) of the transistors is connected to the oscillation node of the VCO. The gates of the p-channel transistors are connected to ground. Figure 6.1(b) depicts a zoom-in to the selected varactor's topology.

The p-channel transistors, as a four-terminal device, can result in various topologies that can be used as a varactor. Figure 6.2 shows the considered configurations in



(a) A -G_m type LC voltage-controlled oscillator.

(b) Selected CMOS varactor for the divide-multiply implementation.

Figure 6.1: LC VCO design and the topology of the selected varactors.

addition to the selected varactor topology. Some of the depicted configurations of the p-channel transistors are less sensitive to the parasitic capacitances due to the layout than others. However, two other factors determined the selection of the topology that at the end was used with the divide-multiply implementation. The first factor was the phase of the impedance of the varactor seen from the oscillation node toward the voltage-controlled signal. The second factor was the monotonicity of the tuning curve as a function of the tuning voltage.

The varactor is used as a variable capacitor in order to tune the frequency of the VCO. The phase of the impedance of an ideal capacitor is -90 degrees. Because the varactors are built from p-channel transistors, in reality the phase of the impedance of the varactor will deviate from -90 degrees. Figure 6.3 shows the phase of the

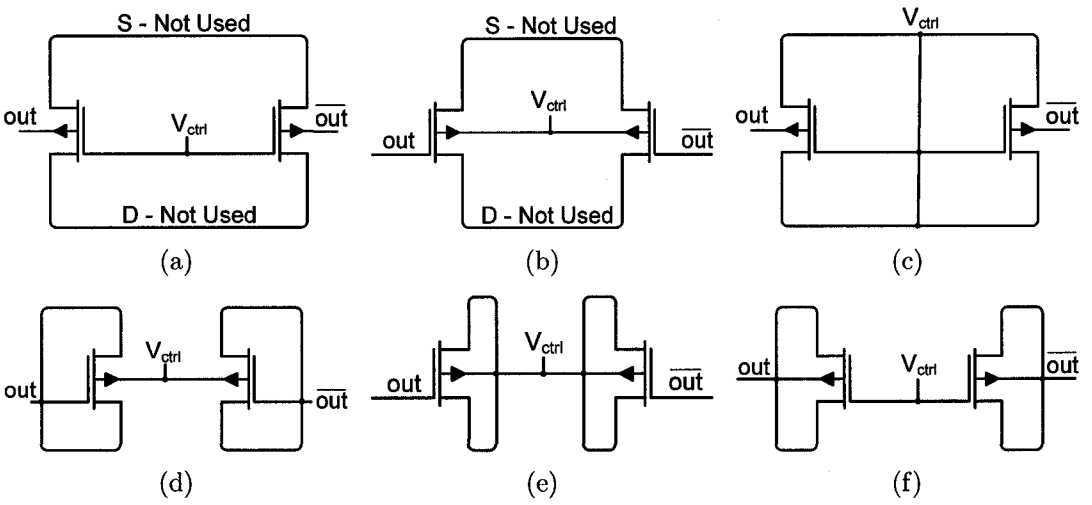


Figure 6.2: Considered CMOS type varactors.

impedance of the varactor topologies shown in Figure 6.2 as a function of the frequency. The p-channel transistors were sized with 144μm in width (36 fingers with unit size of 4μm) and 0.4μm in length. The simulated results show that the deviation from -90 degrees is more pronounced as the frequency is increased for a fixed size of the varactors. In addition, through the simulated results was found that the deviation from -90 degrees is more pronounced as the size of the p-channel transistor is increased in order to increase the tuning range of the VCO for a fixed frequency of operation (ex. 10GHz).

The phase of the impedance of the varactor indicates the quality factor of the varactor. If a simple model for the varactor is used (a series connection of a resistor and a capacitor) then the ideal phase of the impedance of -90 degrees indicates that the associated resistance of the varactor is negligible. As the phase of the impedance of the varactor deviate from the -90 degrees, the associated resistance of the varactor

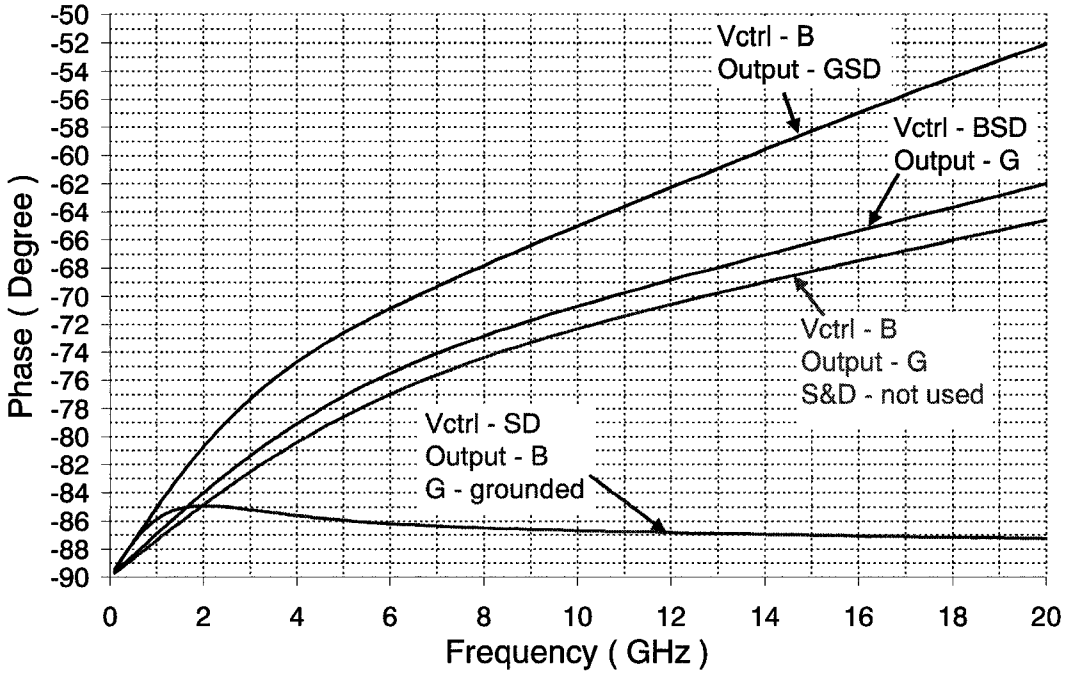


Figure 6.3: Phase of the impedance of the investigated varactor topologies.

becomes more significant causing a poor quality factor of the varactor. As a consequence, the overall phase noise performance of the VCO is affected as well.

If a phase error of -10 degrees is used, then the investigated varactor topologies can result in a good VCO performance (predominantly the phase noise) up to a certain frequency of operation. Beyond that frequency a different varactor topology should be used. For example, if the bulk of the p-channel transistors is one terminal of the varactor, while the gate, the source and the drain (GSD) is the second terminal then, for the aforementioned size of the p-channel transistors, that varactor is good up to 2.1GHz frequency of operation. The frequency of operation could be doubled (4.2GHz) if the bulk and the gate are used as terminals for the varactor, while the

source and the drain of the p-channel transistors are not used. Finally, if the frequency of the application is beyond 4.2GHz (the simulated results are shown up to 20GHz) then the configuration of the p-channel transistors as shown in Figure 6.1(b) could result in a varactor with a good quality factor, and thus improved phase noise performance of the VCO design.

The simulated results show that the selected varactor topology could also result in a VCO design with monotonic tuning curve. As a contrast, if the varactor is configured as shown in Figure 6.2(a) then the tuning curve of the VCO would not be monotonic for all values of the voltage-controlled signal. This type of the varactor is used for the ring oscillator inside the frequency multiplier and its tuning characteristic is depicted further in this chapter.

6.1.1 A Formula to predict the Frequency of Oscillation

The main factor that determines the frequency of operation is the tank circuit. This circuit consists of an inductor in parallel with a varactor capacitance. The varactors are variable CMOS capacitors and, in a parallel combination with the inductors and the parasitic capacitances due to the layout[1], determine the frequency of the VCO.

Figure 6.4 shows an equivalent schematic of the tank circuit for purposes of hand analysis. An inductor L and capacitor C_{var} form a parallel resonant circuit. Their

[1]The schematic drawing of the VCO design is simplified and does not show any parasitic capacitors due to the layout. However, the parasitic capacitance between the varactors and the substrate can be important when the frequency of operation is determined. This parasitic capacitance is due to the reverse bias diode between the p-type substrate and the n-well of the p-channel transistors.

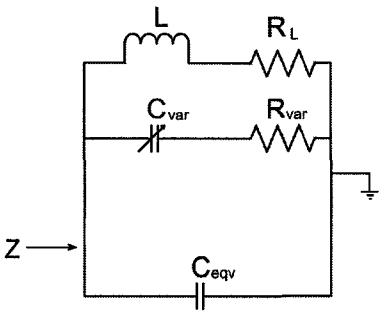


Figure 6.4: A model for the tank circuit.

losses are modeled with resistors, denoted as R_L and R_{var}, respectively. The equivalent capacitance due to the layout is denoted as C_{eqv}. This capacitance includes capacitance due to the reverse biased parasitic diode between the N-well of the varactor and the p-type substrate, denoted as C_{par}, and the parasitic capacitance due to the layout of the gain stage of the VCO, i.e.,

$$C_{\text{eqv}} = C_{\text{par}} + C_{\text{gainstage}} \tag{6.1.1}$$

The expression for the Z of the tank can be found from,

$$Z = (R_L + j\omega L) \,||\, \left(R_{\text{var}} + \frac{1}{j\omega C_{\text{var}}}\right) \,||\, \left(\frac{1}{j\omega C_{\text{eqv}}}\right) \tag{6.1.2}$$

An expanded form for Z is,

$$Z = \frac{A + j\omega B}{D + j\omega E} \tag{6.1.3}$$

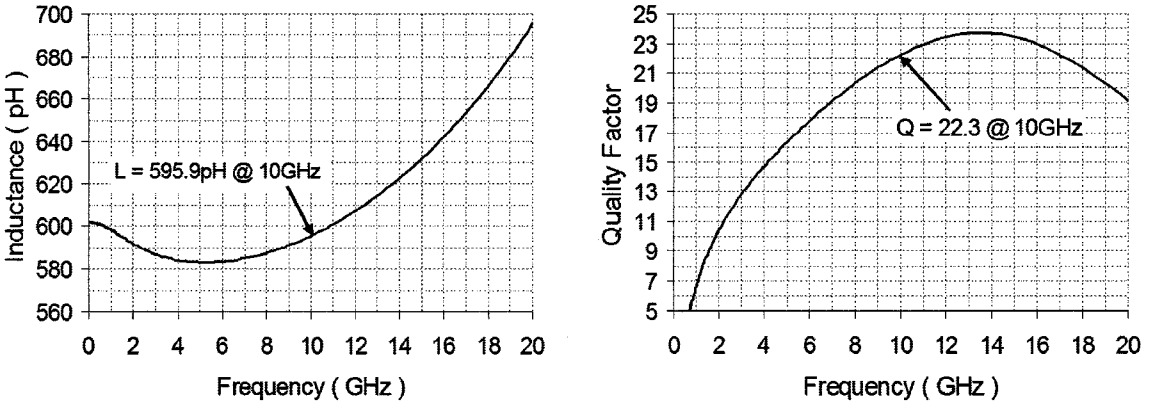
where

$$A = R_L - \omega^2 L R_{\text{var}} C_{\text{var}} \tag{6.1.4}$$

$$B = L + R_L R_{\text{var}} C_{\text{var}} \tag{6.1.5}$$

$$D = 1 - \omega^2 \left[L \left(C_{\text{var}} + C_{\text{eqv}}\right) + R_L R_{\text{var}} C_{\text{var}} C_{\text{eqv}}\right] \tag{6.1.6}$$

$$E = R_L \left(C_{\text{var}} + C_{\text{eqv}}\right) + R_{\text{var}} C_{\text{var}} - \omega R_{\text{var}} L C_{\text{var}} C_{\text{eqv}}. \tag{6.1.7}$$



(a) Inductance of the inductor.

(b) Quality factor of the inductor.

Figure 6.5: Simulated characteristics of the integrated inductor.

It can be shown that the second term in the numerator of the expression (6.1.3) is dominant over the frequencies of interest. Thus, (6.1.3) can be simplified to,

$$Z = \frac{j\omega B}{D + j\omega E}. \tag{6.1.8}$$

When the VCO starts to oscillate, the imaginary component in (6.1.8) would be eliminated. This could happen if,

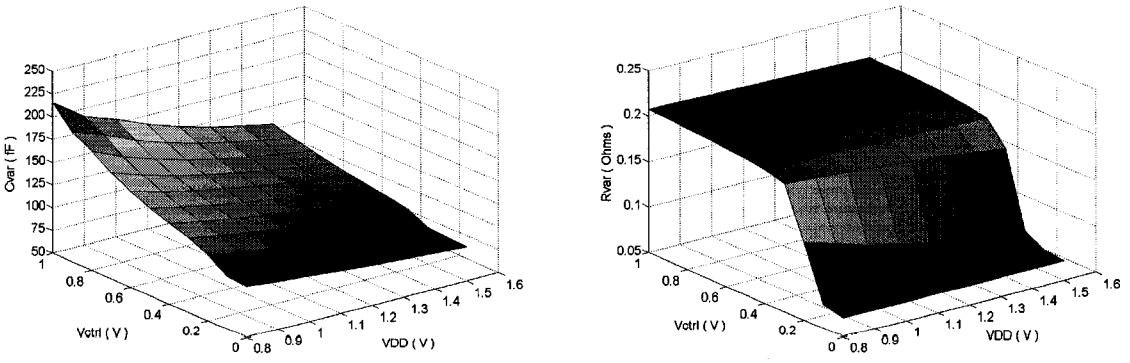
$$D = 0. \tag{6.1.9}$$

Therefore, the approximated formula for predicting the frequency of oscillation for the VCO can be calculated from,

$$F_{\text{osc}} = \frac{1}{2\pi}\sqrt{\frac{1}{L\left(C_{\text{var}} + C_{\text{eqv}}\right) + R_{\text{var}}R_L C_{\text{eqv}}C_{\text{var}}}} \tag{6.1.10}$$

In order to compare the calculated frequency of oscillation with the simulated results, the values of the included parameters in the equation (6.1.10) were obtained by simulating the VCO layout.

Figure 6.5 shows the simulated characteristics of the integrated inductor. The



(a) Series capacitance of the varactor.

(b) Series resistance of the varactor.

Figure 6.6: Simulated characteristics of the integrated varactor.

simulated quality factor of the integrated inductor at 10GHz was found to be Q = 22.3, while the simulated inductance at 10GHz was found to be L = 595.9pH.

The simulated characteristics of the varactor, in terms of series capacitance and resistance, is shown in Figure 6.6. As shown, the values of the equivalent capacitance and resistance of the varactor depend from the voltage controlled signal, $V_{\mathtt{ctrl}}$, and the power supply, VDD. Consequently, the frequency of oscillation of the integrated VCO will be sensitive not only to the voltage controlled signal but also to the power supply.

Finally, the simulated value of the equivalent capacitance was found to be 280fF. The contribution of the reverse biased diode between the N-well of the varactor and the substrate is 242fF. The layout of the gain stage introduces 38fF capacitance. The equivalent capacitance is expected to be a function of the applied voltage. However, to simplify the verification of the calculated frequency of oscillation, this capacitance is set to the aforementioned value. This assumption will affect the deviation between the calculated and the simulated frequency of operation as shown in Figure 6.7.

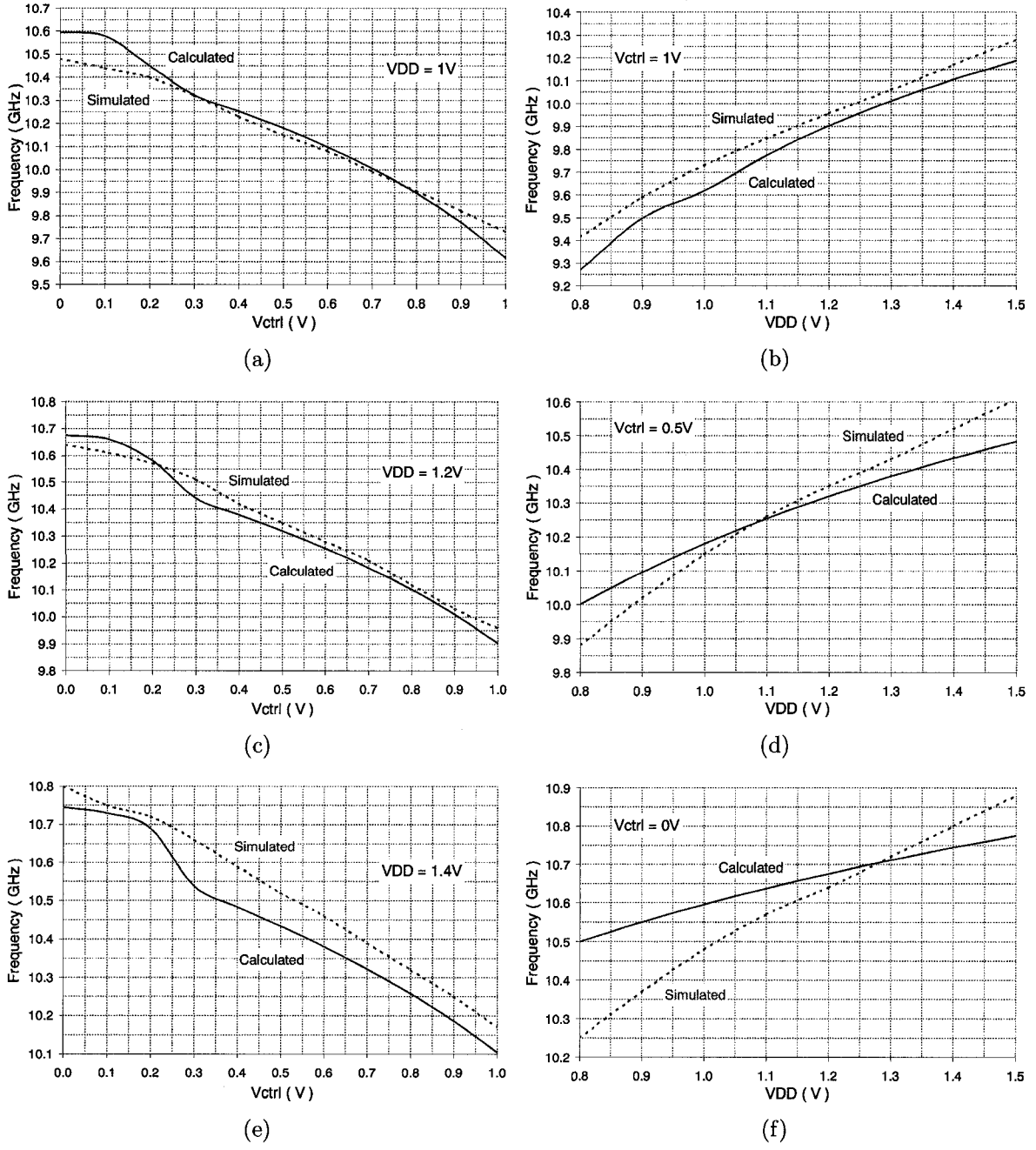


Figure 6.7: Calculated and simulated frequency of oscillation for different power supply and control voltages.

The deviation between the calculated and the simulated frequency of oscillation is defined as,

$$\texttt{Deviation} = \frac{(F_{\texttt{osc}})_{\texttt{calculated}} - (F_{\texttt{osc}})_{\texttt{simulated}}}{(F_{\texttt{osc}})_{\texttt{simulated}}} \cdot 100\,. \tag{6.1.11}$$

Figure 6.7(a) shows the calculated and simulated frequency of oscillation for the case when the supply voltage is set at 1V, and the control signal is swept from 0 to 1V. The worst case error between the calculated and simulated curve is 1.34%.

Figure 6.7(b) shows the calculated and simulated frequency of oscillation for the case when the control signal is set at 1V, and the supply voltage is swept from 0.8 to 1.5V. The worst case error between the calculated and simulated curve is -1.55%.

Figure 6.7(c) shows the calculated and simulated frequency of oscillation for the case when the supply voltage is set at 1.2V, and the control signal is swept from 0 to 1V. The worst case error between the calculated and simulated curve is -0.66%.

Figure 6.7(d) shows the calculated and simulated frequency of oscillation for the case when the control signal is set at 0.5V, and the supply voltage is swept from 0.8 to 1.5V. The worst case error between the calculated and simulated curve is 1.22%.

Figure 6.7(e) shows the calculated and simulated frequency of oscillation for the case when the supply voltage is set at 1.4V, and the control signal is swept from 0 to 1V. The worst case error between the calculated and simulated curve is -1.15%.

Figure 6.7(f) shows the calculated and simulated frequency of oscillation for the case when the control signal is set at 0V, and the supply voltage is swept from 0.8 to 1.5V. The worst case error between the calculated and simulated curve is 2.44%.

Figure 6.8 shows the deviation between the calculated and simulated frequency of

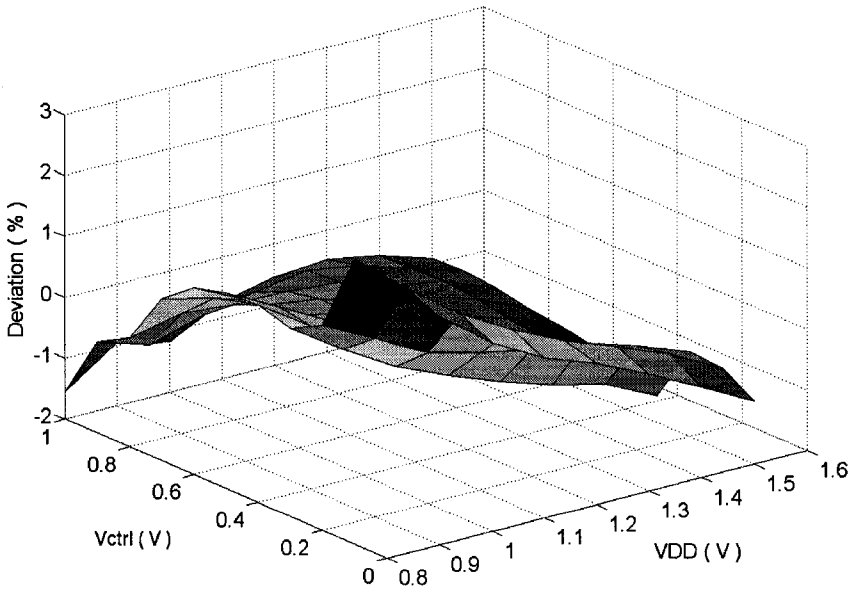


Figure 6.8: Deviation between the calculated and simulated frequency of oscillation.

oscillation for the case when the control signal is swept from 0 to 1V, and the supply voltage is swept from 0.8 to 1.5V. The average deviation error between the calculated and simulated curve is 0.67%. These results indicate that the derived formula has a close agreement with the simulated results.

6.1.2 Additional Simulated Results

Table 6.1 summarizes the simulated performance of the presented VCO. In order to reduce the power dissipation, the VCO supply is set to 1V. The simulated DC current that the VCO draws from the supply is 490μA resulting in only 0.49mW DC power dissipation.

The control voltage denoted as $V_{\mathtt{ctrl}}$ was swept from 0V to 1V resulting in the simulated tuning range of 748MHz. Simulated results show that the VCO can be tuned between 9.72 and 10.47GHz and the result is shown in Figure 6.9.

Table 6.1: Post-layout simulated results of the LC VCO Design

Name	Value	Units	Comments
Supply voltage	1.0	V	
Power Dissipation	0.49	mW	
Tuning Frequencies	9.72 - 10.47	GHz	
K_{VCO}	748	MHz/V	
Phase Noise @ 1MHz offset	-114.4	dBc	$f_0 = 10$GHz
Figure of Merit (FOM)*	-197.6	dB	
V_{pp} @ 27°C	1	V	buffer output
f_{osc} vs. temperature	4.67	MHz/°C	
V_{pp} vs. temperature	3.6	mV/°C	
f_{osc} vs. supply	1.1	GHz/V	
V_{pp} vs. supply	1.82	V/V	
Layout area	373 x 257	μm^2	0.096[mm^2]
Technology	0.13μm CMOS		

$$*FOM = PN(\omega_0, \Delta\omega) - 10\log\left(\frac{[mW]}{P_{VCO}}\left(\frac{\omega_0}{\Delta\omega}\right)^2\right)$$

where $PN(\omega_0, \Delta\omega)$ is the single-side-band noise at the offset frequency $\Delta\omega$ from the carrier frequency ω_0. P_{VCO} denotes the power consumption of the VCO in mW.

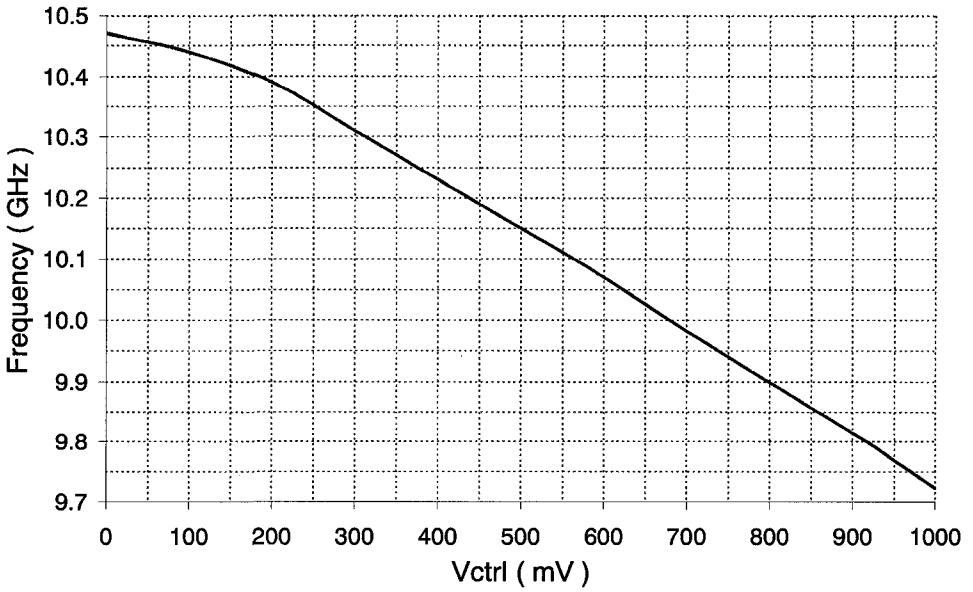


Figure 6.9: Tuning range of the presented LC VCO; the power supply is set to 1V.

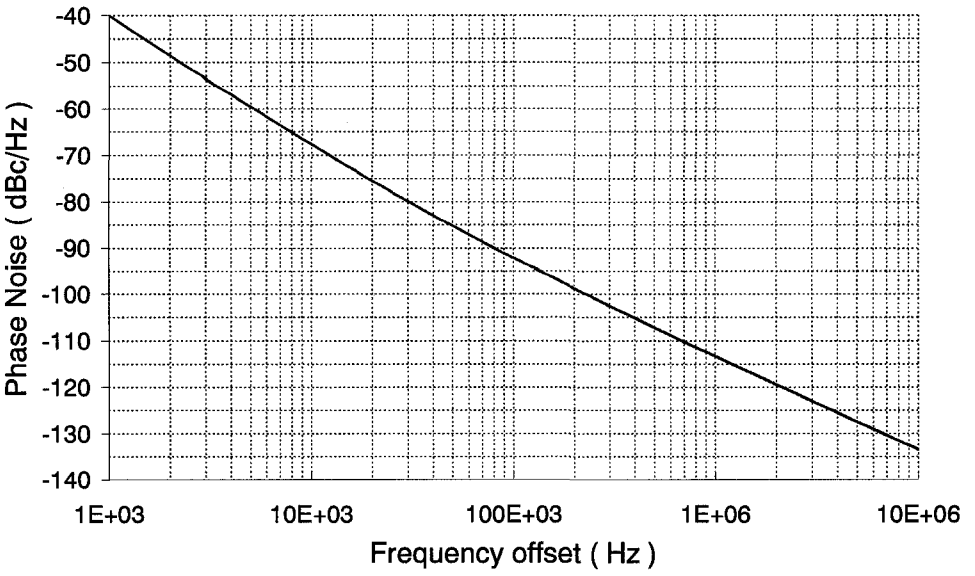


Figure 6.10: Simulated phase noise characteristic of the presented LC VCO.

Figure 6.10 shows the simulated phase noise characteristics of the 10GHz LC voltage-controlled oscillator. The simulated phase noise at 1MHz from the frequency carrier of 10GHz is -114.4dBc/Hz. The power dissipation is 0.49mW and the calculated FOM is better than -197dB.

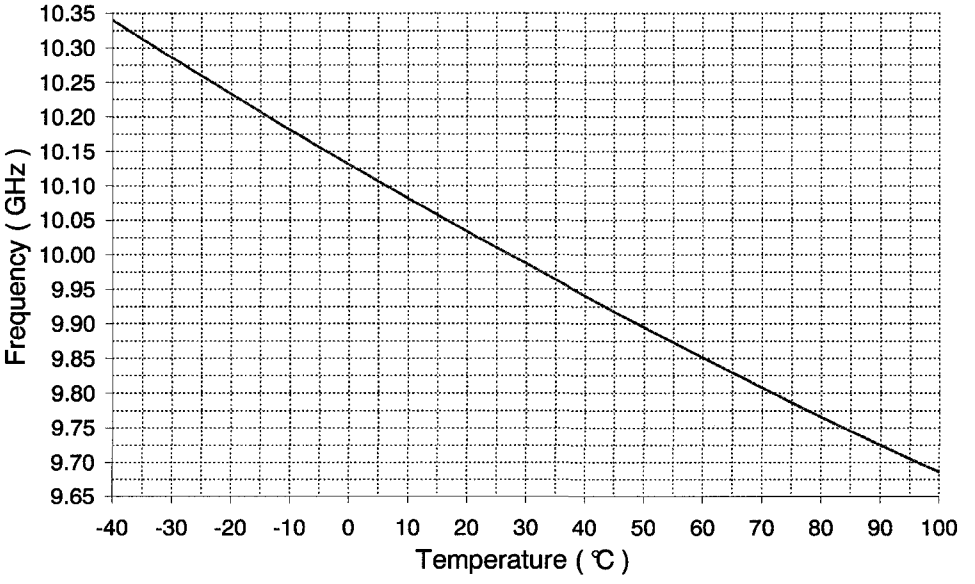
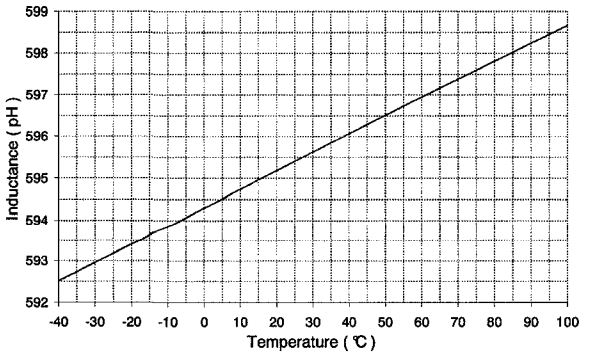
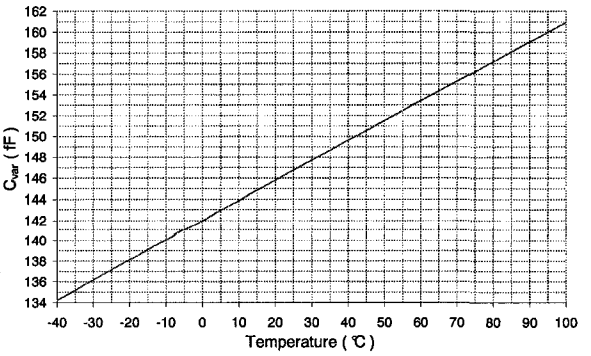


Figure 6.11: Frequency of oscillation v.s. temperature.

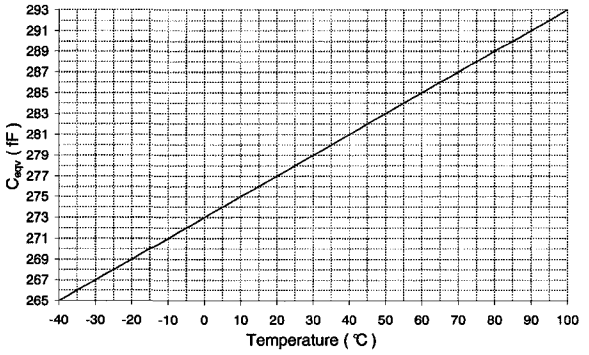
Figure 6.11 shows the dependance of the frequency of oscillation of the VCO on the temperature. The simulated results show that, by sweeping the temperature from -40°C to 100°C, the frequency of the output signal from the VCO in average will increase by 4.67MHz for each 1°C temperature change.



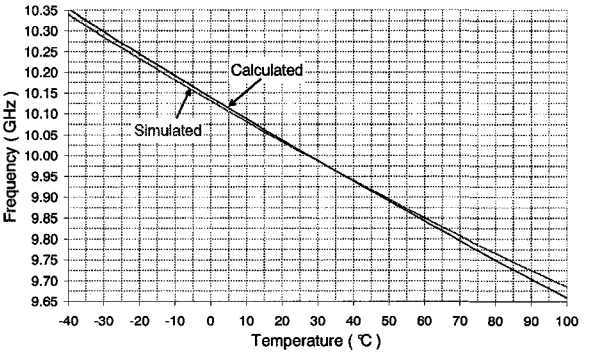
(a) temperature effect on the tank inductance



(b) temperature effect on the varactor capacitance



(c) temperature effect on the equivalent capacitance



(d) calculated and simulated frequency of oscillation as a function of the temperature

Figure 6.12: Investigating the temperature effect on the VCO oscillation frequency.

According to the derived expression (6.1.10), the frequency of oscillation of the VCO is predominantly determined by the tank inductance, the capacitance of the varactor and the parasitic capacitance due to the layout. Figure 6.12(a) shows the simulated tank inductance as a function of the temperature. If the inductance of the tank changes by $\pm 1\%$ then, without changing the other tank parameters, the

frequency of the output signal from the VCO will change by ±0.5%. Figure 6.12(b) shows the capacitance of the varactor as a function of the temperature. It was found that if the capacitance of the varactor changes by ±1%, while all other parameters of the tank circuit are not changed, then the frequency of oscillation will change by ±0.16%. Figure 6.12(c) shows how the parasitic capacitance due to the layout is changing with the temperature. It was also found that if the parasitic capacitance changes by ±1%, while all other parameters of the tank circuit are not changed, the frequency of the VCO signal will change by ±0.34%. Finally, Figure 6.12(d) shows the simulated and the calculated frequency of oscillation as a function of the temperature. The calculated curve was plotted based on the expression (6.1.10) and included the simulated values of the tank inductance, the varactor capacitance, and the parasitic capacitance as a function of the temperature as shown in Figures 6.12(a), 6.12(b), and 6.12(c).

6.1.3 Comparison with other 10GHz VCO designs

The figure of merit of the presented VCO is compared with the FOM of the VCO designs reported in [40–47]. A plot of the FOM as a function of the power dissipation is shown in Figure 6.13. The selection of the VCO designs was based on the frequency of operation (10GHz) and the circuit topology (LC type). The comparison shows that the VCO design from this work has a low power dissipation and an attractive figure of merit relative to the research work prevalent in the literature.

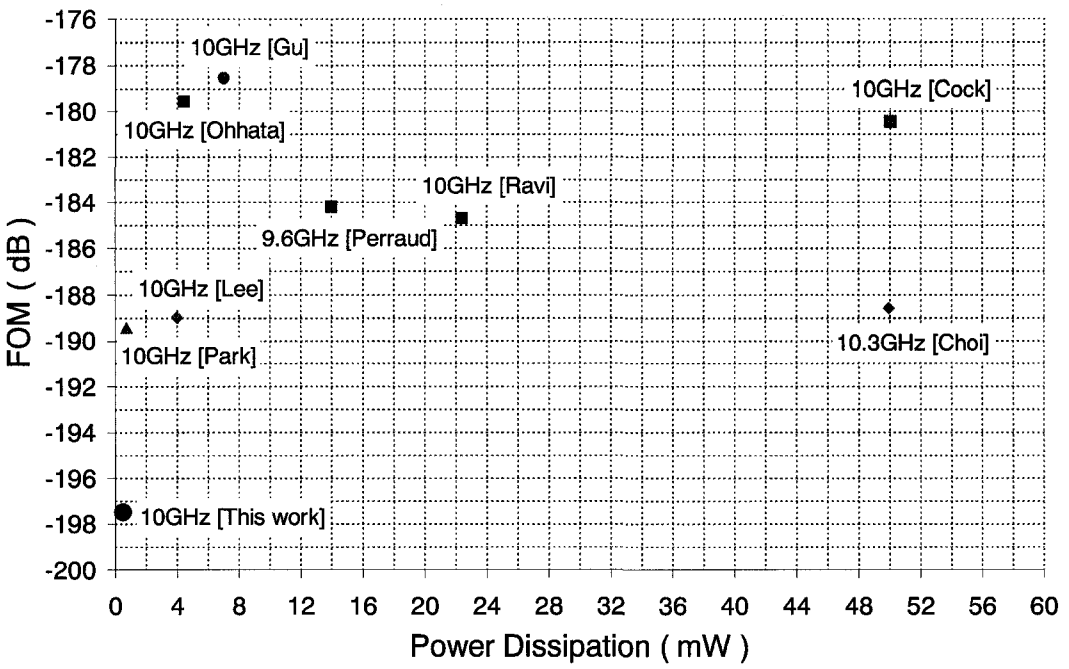


Figure 6.13: Figure of merit as a function of the DC power dissipation for VCO designs with a frequency of operation around 10GHz.

6.2 Ring VCO

The frequency multiplier with the divide-multiply implementation used a ring type of a voltage-controlled oscillator. The VCO consisted with 15 stages. The description of a single delay stage and the analytical analysis can be found in [48].

The tuning characteristic of the VCO with power supply set to 1V is shown in Figure 6.14. The tuning curve is not monotonic over the entire range of the voltage controlled signal due to the characteristic of the implemented varactor. The varactor topology is shown in Figure 6.2(a). A method to improve the linearity of the tuning curve is discussed in [48].

Figure 6.15 shows the phase noise of the ring oscillator. The VCO is tuned to 20MHz and the power supply is set to 1V. The VCO is loaded with a high-impedance buffer. The simulated phase noise at 100kHz offset is -118dBc/Hz.

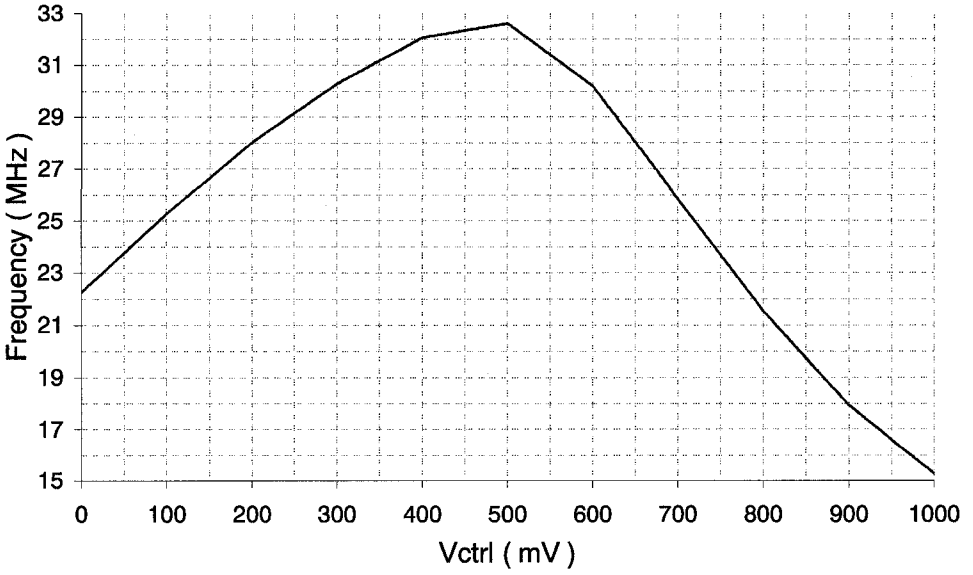


Figure 6.14: Tuning characteristic of the ring oscillator.

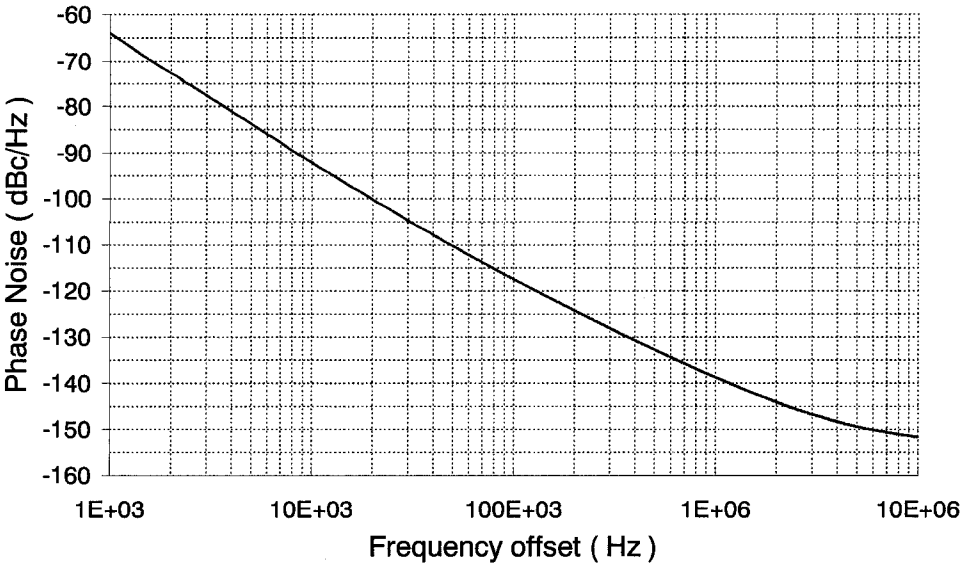


Figure 6.15: Phase noise characteristic of the ring oscillator.

6.3 Summary

The divide-multiply implementation utilized two voltage-controlled oscillator. Because the VCO within the frequency multiplier is completely analyzed in [48], this chapter discussed the LC type VCO within the main system. The possible configurations of p-channel varactors are investigated. The selection of the varactors is performed based on the phase of the impedance of the varactor. As the phase approaches -90 degree, better phase noise performance of the VCO is possible. The simulated results of the VCO were discussed and compared to the state-of-the-art 10GHz VCOs found in the literature. The advantages of the implemented VCO, based on the simulated results, are the phase noise performance and the low power dissipation from a reduced power supply.

Chapter 7

Chip Testing

The divide-multiply implementation of the proposed architectural concept for a frequency synthesizer has been designed and fabricated in a CMOS 0.13μm technology. Figure 7.1 shows the photograph of the fabricated chip. This chapter discusses the test setup, the used equipment to measure the fabricated chip, as well as the measured results.

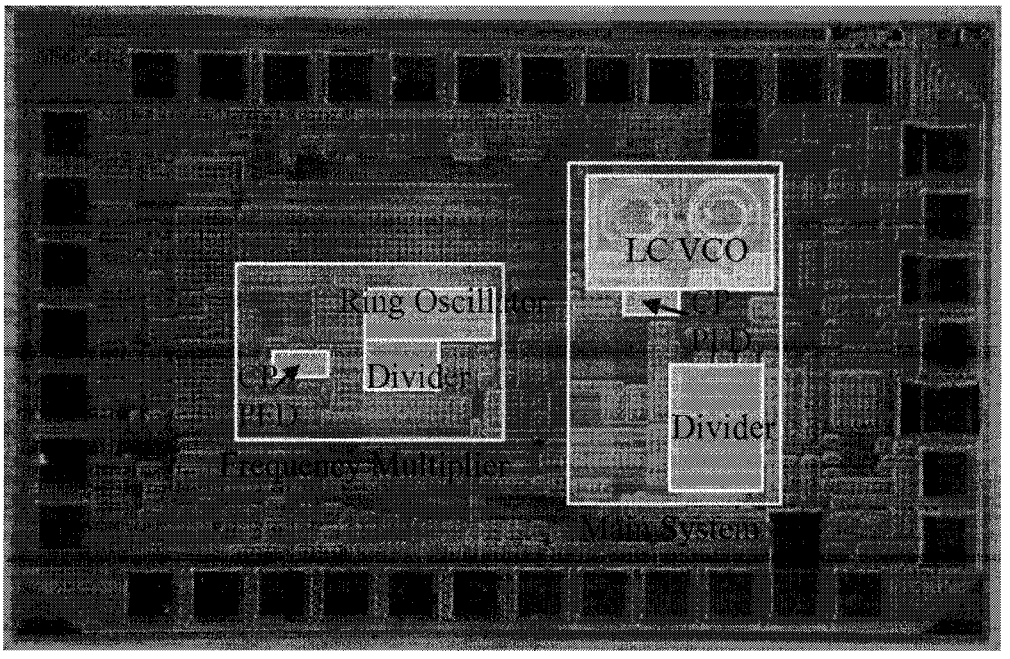


Figure 7.1: Die photograph of the fabricated divide-multiply implementation.

7.1 Test Setup and Used Equipment

Figure 7.2 shows the block diagram of the test setup. The chip expects a differential reference signal, a power supply, programming (binary) bits, and two low pass filters, denoted as LPF 1 and LPF 2, as off-chip components.

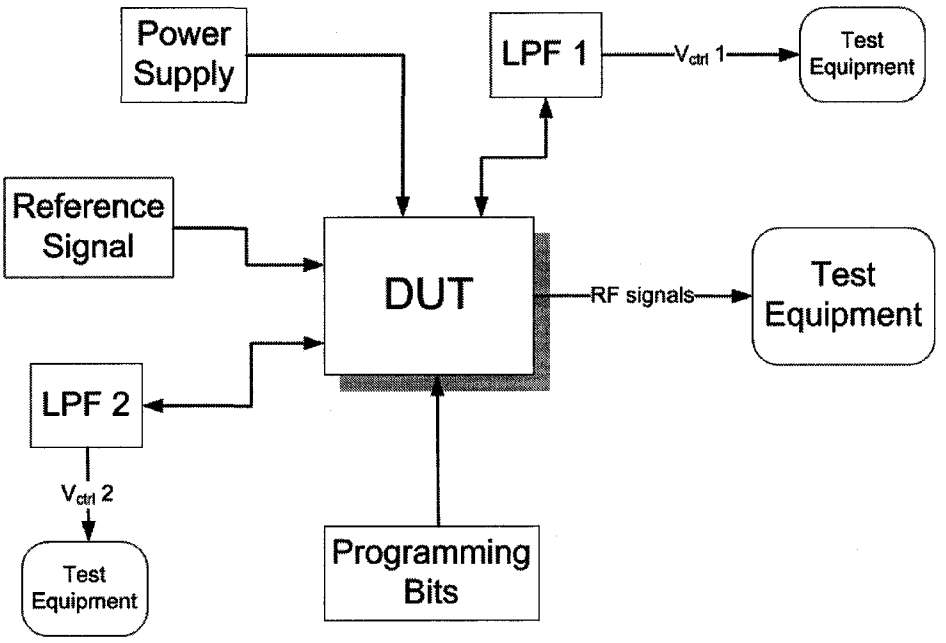


Figure 7.2: Test setup to characterize the divide-multiply implementation.

The following test equipment is used to collect the measured results:

- Agilent 6624A system DC power supply;
- An Agilent 81134A 3.35GHz pulse / pattern generator - to generate the reference signal;
- The following oscilloscopes are used: Tektronix TDS 3032 300MHz 2.5GS/s, Agilent Infiniium 54855A DSO 6GHz, and Agilent Infiniium 54832D MSO 1GHz;

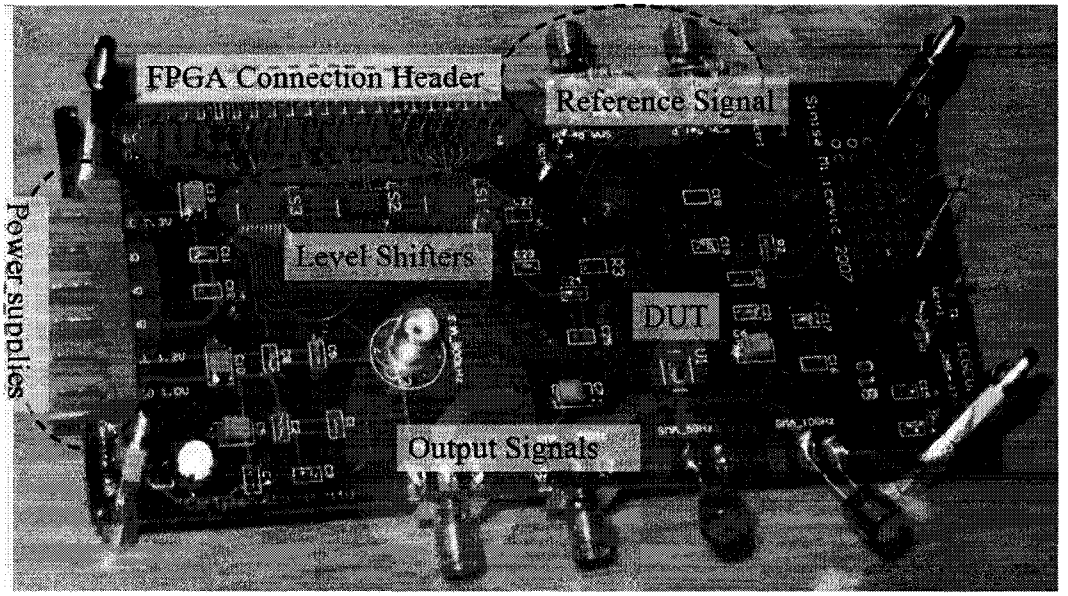


Figure 7.3: Test board to characterize the divide-multiply implementation.

- Agilent E4440A PSA Series Spectrum Analyzer;
- ALTERA Stratix 1S40 FPGA board - to generate the required programming bits.

In order to perform the measurements, the fabricated chip was bonded to a printed circuit board (PCB) as shown in Figure 7.3. The bonded chip is labeled as device under test (DUT). The inputs for the reference signal, the SMA connectors carrying the outputs from the chip and the connector to the FPGA board are highlighted. The MAX3000E (+1.2V to +5.5V, 15kV ESD-Protected, 0.1uA, 230kbps, 8-Channel Level Translators) from MAXIM are used as level shifters to translate the 3.3V signal from the FGPA board to 1.2V.

7.2 Measurements

7.2.1 Measuring the Switching Speed

The switching speed is defined as the time it takes the frequency synthesizer to re-tune the VCO from one frequency to another. Switching speed is measured on an oscilloscope by probing the VCO tuning voltage. The oscilloscope is triggered by the rising or the falling edge of the measured voltage-controlled signal.

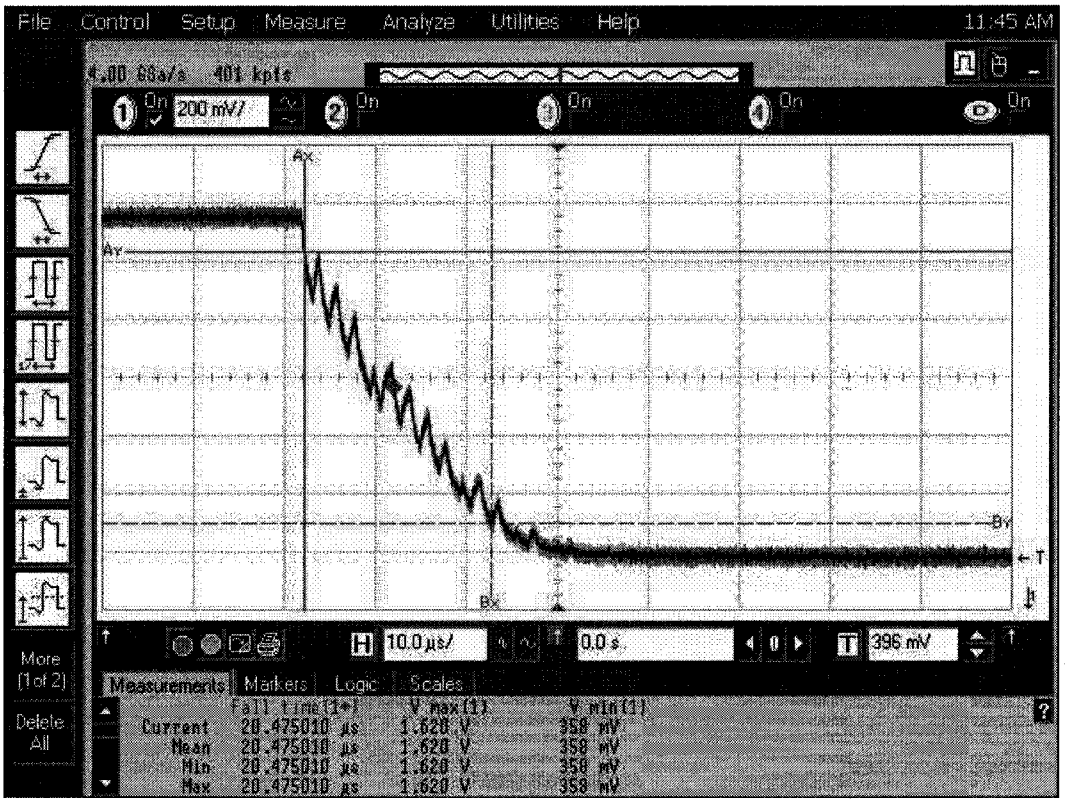


Figure 7.4: Switching time of the frequency multiplier: the loop filter is built with $C_3 = 1nF$, $C_4 = 100pF$ and $R_2 = 8.7k\Omega$. The plot shows the case when the reference signal (20MHz) is enabled.

The divide-multiply implementation utilized a PLL type of a frequency multiplier. Figure 7.4 depicts the switching time of the frequency multiplier once the reference

signal is enabled. The loop bandwidth of the frequency multiplier is designed with $C_3 = 1nF$, $C_4 = 100pF$ and $R_2 = 8.7k\Omega$, and the plot shows that the frequency multiplier can lock for about $30\mu s$. The lock-in time was captured with Agilent Infiniium 54832D MSO 1GHz oscilloscope.

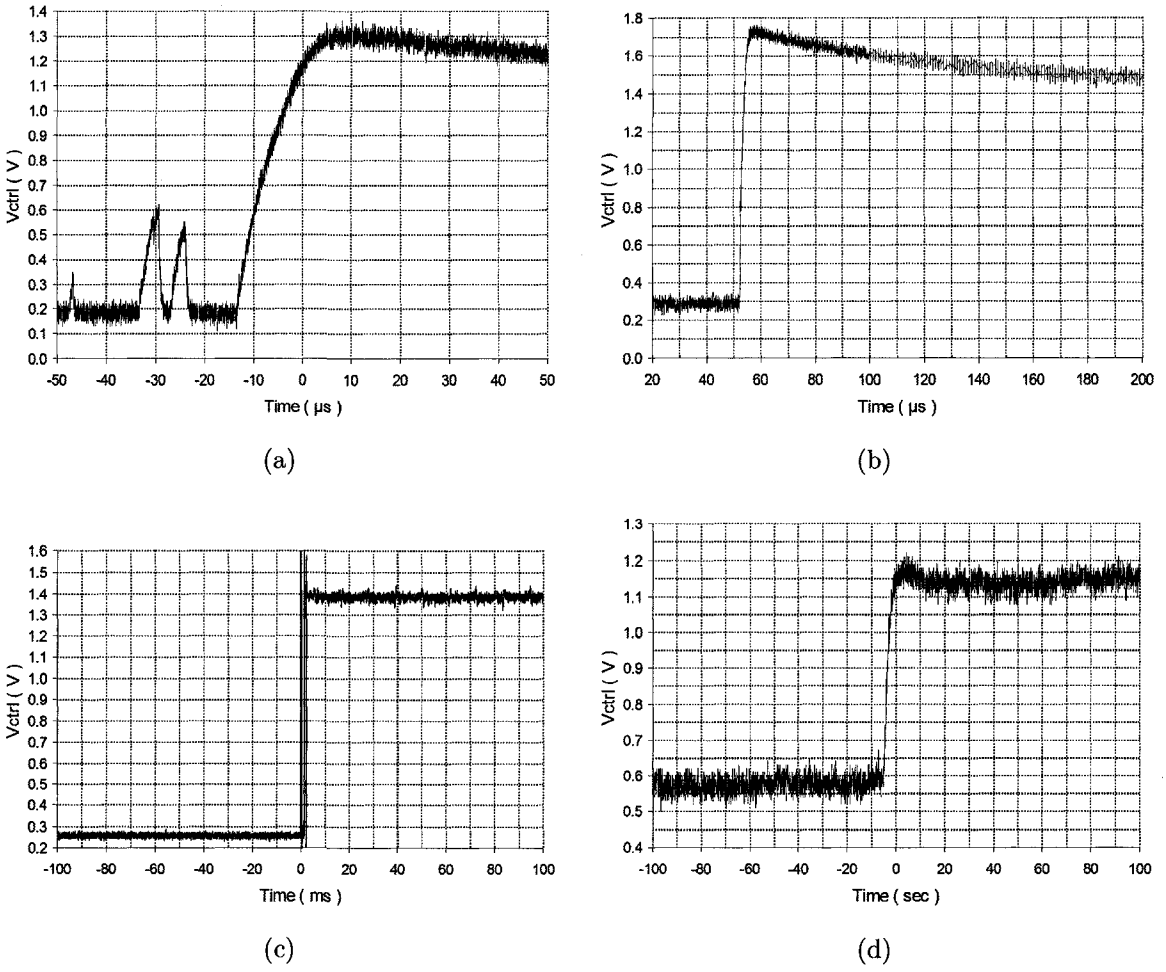


Figure 7.5: Measured lock-in time of the frequency synthesizer.

Figure 7.5 illustrates the lock-in time of the main system for different loop bandwidths. For example, Figure 7.5(a) depicts the measured switching time when the frequency synthesizer locks to a 10.26GHz signal. The power supply was set to 1.64V,

and the main loop was sized with the following parameters: $C_1 = 4.7nF$, $C_2 = 470pF$ and $R_1 = 2.17k\Omega$. A 22kHz loop bandwidth resulted in a switching time of about $65\mu s$. Note that the peaks captured around the time of -30μs are due to the implemented measurement technique. A Tektronix TDS 3032 300MHz 2.5GS/s oscilloscope was used for these measurements.

7.2.2 Measuring the Output Spectrum

Spectrum analyzer was used to monitor the frequency spectrum of the signals. These measurements also monitor appearance of spurious signals. Spurious signals represent any discrete spectral line not related to the signal itself[1] [49].

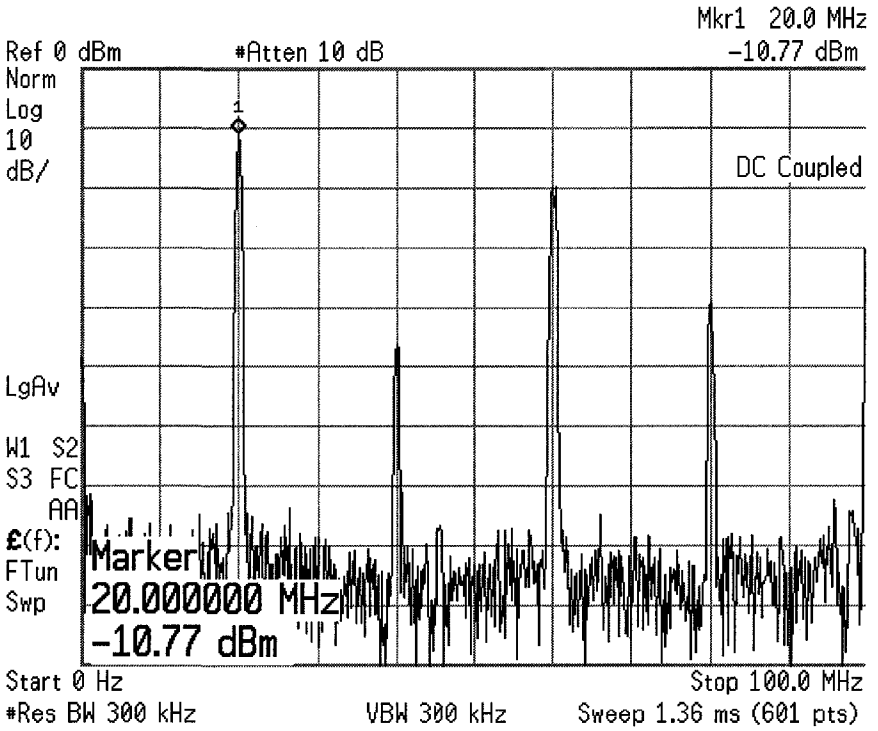


Figure 7.6: Output spectrum of the frequency multiplier. The loop filter of the implemented frequency multiplier is sized with C1 = $510pF$, C2 = $51nF$, and R1 = 12.4kΩ.

[1]Harmonics are usually not considered as spurious signals and are dealt with separately [49].

Figure 7.6 shows the output spectrum of the frequency multiplier. The simulated and the measured results show that the output signal from the frequency multiplier is of a square type. Consequently, in addition to the fundamental signal, strong harmonics are also present in the output spectrum of the frequency multiplier.

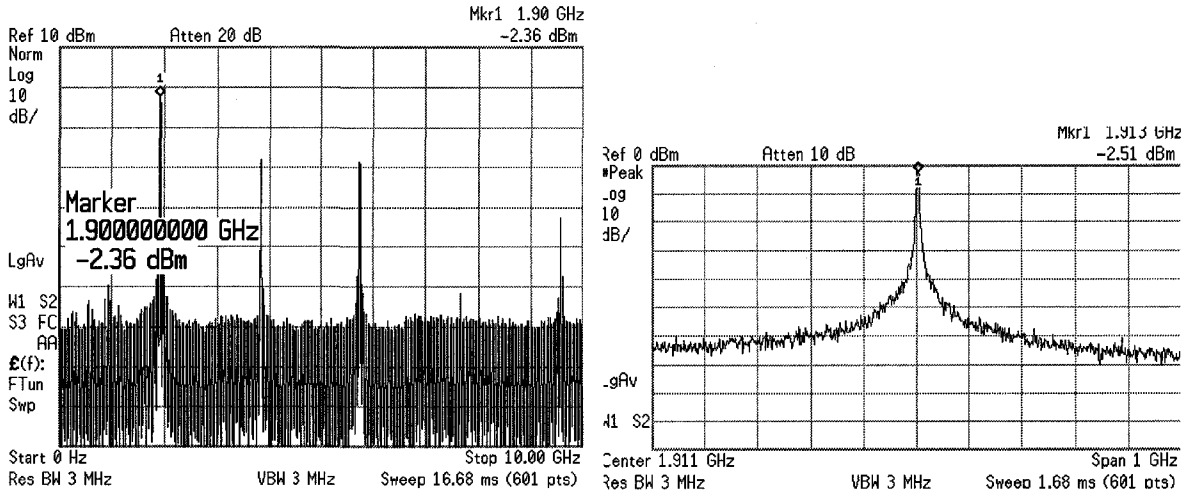


Figure 7.7: Output spectrum of the frequency synthesizer.

Figure 7.7 depicts the output spectrum of a 1.9GHz signal from the frequency synthesizer. The signal from the main VCO was sent through a divide-by-4 frequency divider and a 50Ω output buffer to a spectrum analyzer. Compared to the ΣΔ based frequency synthesizers were the signal power of the spurious tones is strong, as illustrated in Chapter 2, the measured results of the divide-multiply implementation show that the spurious tones of the divide-multiply implementation are reduced without the need of any spur cancelation techniques. This feature of the divide-multiply implementation (as well as any implementation of the proposed architectural concept) is somehow expected because the proposed system is modification of an integer-N

frequency synthesizer. The integer-N frequency synthesizers are sensitive to the non-idealities of the PLL blocks, for example the leakage of the charge pump, which cause spurious tones in the output spectrum. A careful design of the PLL blocks as well as a proper selection of the loop bandwidth would cause reduction of the spurious tones.

7.2.3 Measuring the Phase Noise

A spectrum analyzer is used to measure the phase noise of the output signals. The phase noise is an indicator of the signal quality and is specified in dBc/Hz.

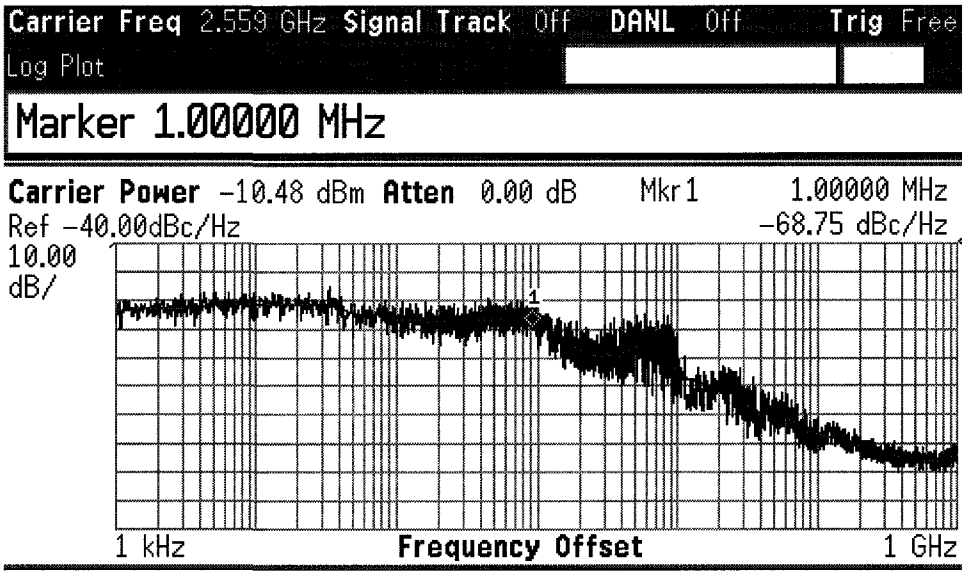


Figure 7.8: Phase noise of the output signal from the frequency synthesizer: the loop filter is built with $C_1 = 4.7nF$, $C_2 = 470pF$ and $R_1 = 2.15k\Omega$.

Figure 7.8 depicts the measured phase noise of 2.559GHz signal generated from the main system. The on-chip LC VCO generates signal with a frequency of 10.24GHz. This signal is sent through an on-chip divide-by-four frequency divider to the Agilent

spectrum analyzer. The loop filter was designed with the following parameters: $C_1 = 4.7nF$, $C_2 = 470pF$ and $R_1 = 2.15k\Omega$.

As discussed in Chapter 2 and particularly recalling the Figure 1.2, the phase noise measurement of the $\Sigma\Delta$ frequency synthesizers shows the appearance of the fractional spurs. These spurs appear close to the fundamental signal and their frequency offset is determined by the frequency resolution of the frequency synthesizer. The channel resolution of the divide-multiply implementation is 500kHz. Thus, one should expect a spur at this frequency offset (500kHz from the carrier). Based on the phase noise measurement shown in Figure 7.8 it can be concluded that the spur at 500kHz offset is reduced such that it is below -70dBc. In a conclusion regarding the spurious tones, Figure 7.8 shows that the spurious tones within the divide-multiply implementation are reduced to the point that are not visible on the phase noise measurement plot as it was the case with the $\Sigma\Delta$ frequency synthesizers.

Regarding the phase noise characteristic, Figure 7.8 shows measured in-band phase noise of 62dBc/Hz at 10kHz offset from the carrier. At 1MHz offset the measured phase noise is 69dBc/Hz. Compared to the simulated results, as discussed in Chapter 3, a deviation between the simulated and the measured phase noise is noticed. The following discussion would search for the cause of the deviation between the simulated and the measured results.

The phase noise analysis of the proposed architecture was performed assuming that the phase noise of the reference signal is negligible. Therefore, the phase noise analysis did not include the contribution of the phase noise from the reference signal. However, in order to measure the fabricated chip, the reference signal was generated

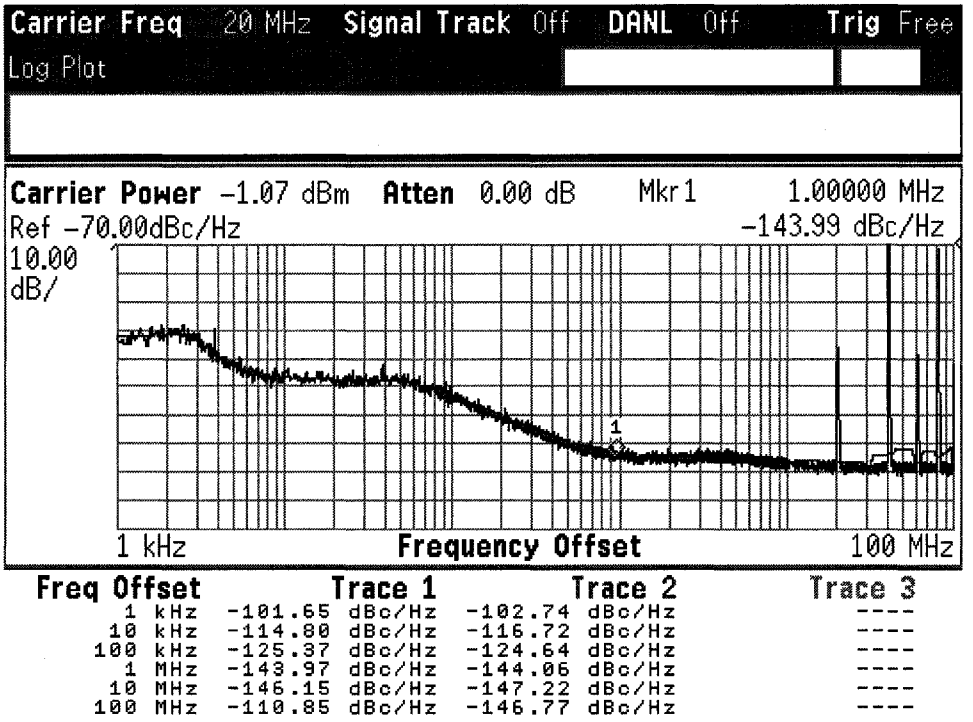
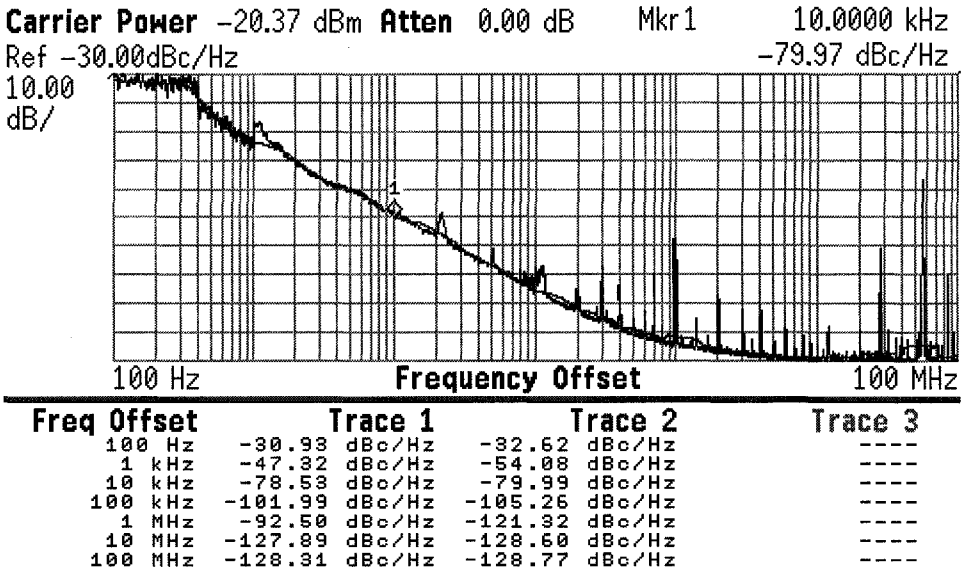


Figure 7.9: Phase noise of the reference signal generated from an Agilent 81134A 3.35GHz pulse / pattern generator.

from an Agilent 81134A 3.35GHz pulse / pattern generator. Figure 7.9 shows the measured phase noise of the reference signal. The phase noise of the reference signal, generated from a pulse generator, should be included into the phase noise analysis because it will affect the measurements of the fabricated frequency synthesizer as it will be shown in this section.

The phase noise analysis of the proposed system, discussed in Chapter 3, indicated that the phase noise from the frequency multiplier is a major contributor to the total phase noise of the output signal from the frequency synthesizer. The fabricated frequency multiplier utilized a ring oscillator, and the same charge pump and the phase frequency detector used within the main system as well. To measure the phase noise



Freq Offset	Trace 1	Trace 2	Trace 3
100 Hz	-30.93 dBc/Hz	-32.62 dBc/Hz	----
1 kHz	-47.32 dBc/Hz	-54.08 dBc/Hz	----
10 kHz	-78.53 dBc/Hz	-79.99 dBc/Hz	----
100 kHz	-101.99 dBc/Hz	-105.26 dBc/Hz	----
1 MHz	-92.50 dBc/Hz	-121.32 dBc/Hz	----
10 MHz	-127.89 dBc/Hz	-128.60 dBc/Hz	----
100 MHz	-128.31 dBc/Hz	-128.77 dBc/Hz	----

Figure 7.10: Measured phase noise of the ring oscillator (frequency multiplier).

of the ring oscillator, the loop bandwidth of the frequency multiplier was reduced such that the measured phase noise of the output signal from the frequency multiplier to be approximately the phase noise of the ring oscillator. This measurement technique resulted in the measured phase noise of the ring oscillator as shown in Figure 7.10.

The signal from the ring oscillator was sent through an on-chip output buffer to the on-chip bond pad. The chip was bonded onto a double-side printed circuit board (PCB). The phase noise was measured with a spectrum analyzer which added a 50Ω load to the test board. The simulated results were performed on a free-running ring oscillator loaded with a high impedance buffer. The simulated results represent the phase noise characteristic of the on-chip signal. The effects of the output buffer, the bond pads, the bondwires, the PCB, and the load of the spectrum analyzer were not included. Consequently, the measured phase noise result shows a deviation compared to the simulated results. In order to have a better comparison between the simulated

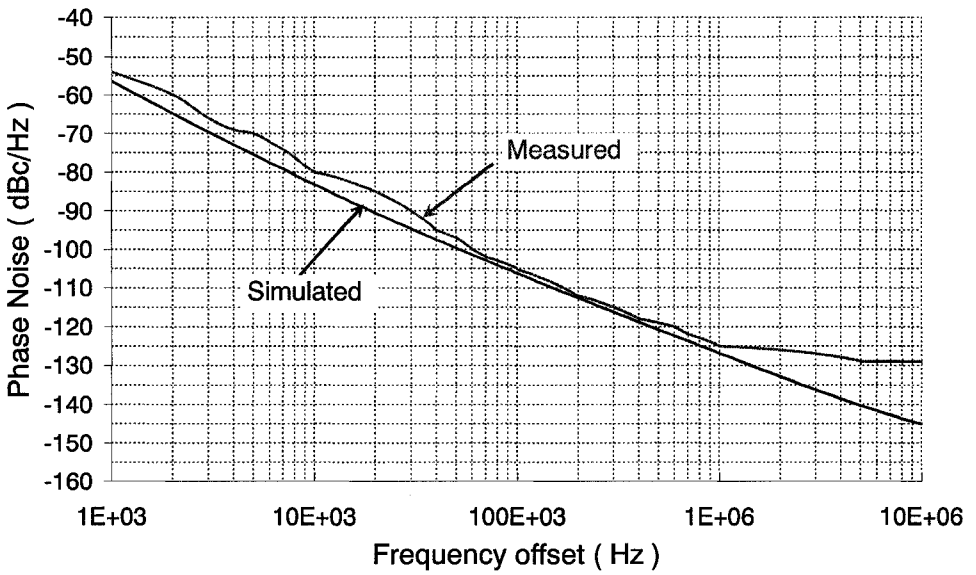


Figure 7.11: Simulated and measured phase noise of the ring oscillator within the frequency multiplier.

and the measured phase noise of the ring oscillator, the phase noise of the free-running ring oscillator was re-simulated by including the aforementioned effects. The effect of the PCB was simulated by adding a small resistor with enabled noise on the signal path (both DC and AC). In addition to the noise signal path resistance, a 1pF capacitor between the measured signal line and the ground was added during the simulation.

Figure 7.11 shows the comparison of the new simulated and the measured results. The plot depicts that the simulated and the measured results are relatively close within the frequency offsets up to 1MHz. The closed loop of the frequency multiplier (not a free-running VCO), and the noise from the external sources (power supply, pulse generator) are believe to increased the noise floor of the measured signal. Consequently, there is a deviation between the simulated and the measured results

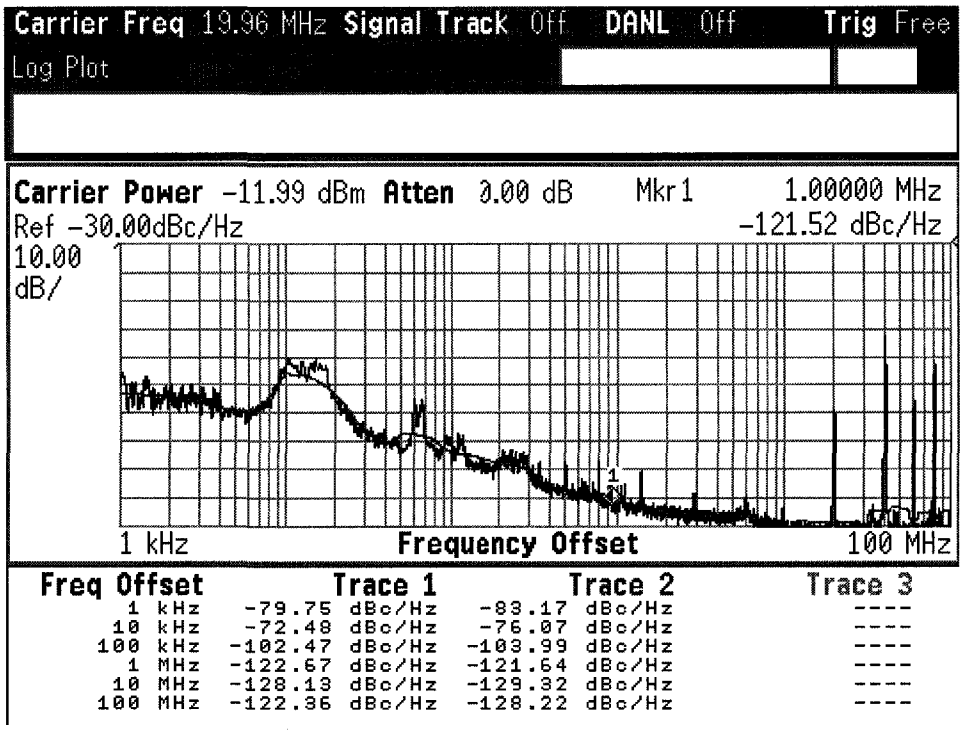


Figure 7.12: Phase noise of the output signal from the frequency multiplier

beyond the 1MHz frequency offset. Nevertheless, comparing the simulated phase noise of the free-running ring oscillator, as shown in Chapter 6 - Figure 6.15, and the simulated/measured phase noise of the ring oscillator shown in Figure 7.11, it is clear that the test board and the measurement equipment affected the measured results.

In order to determine the phase noise due to the phase frequency detector and the charge pump, the loop bandwidth of the frequency multiplier was increased to 15kHz (compared to the previous measurement) and the measured phase noise is shown in Figure 7.12. Once the measured data for the phase noise of the output signal from the frequency multiplier and the phase noise of the ring oscillator is obtained, the phase noise analysis depicts the phase noise contribution due to the phase frequency detector and the charge pump as shown in Figure 7.13. It is interesting to note that

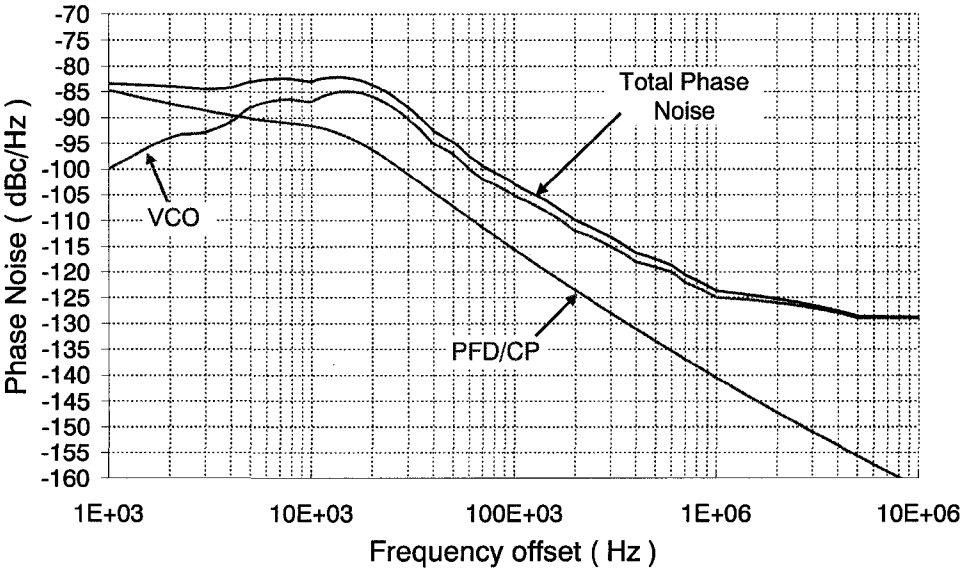


Figure 7.13: Contribution to the total phase noise of the individual blocks of the frequency multiplier based on the measured results.

neither the phase noise of the reference signal nor the phase noise of the main VCO of the divide-multiply implementation would have an effect, at lower frequency offsets, on the phase noise of the output signal from the frequency multiplier. Regarding the divide-multiply implementation, the output signal from the frequency divider is a reference signal for the frequency multiplier. The phase noise of that signal is equal to the phase noise of the main VCO signal reduced by the 20log of the division ratio of the frequency divider. Thus, the phase noise of the output signal from the frequency multiplier is determined by the phase noise characteristic of the implemented VCO (within the frequency multiplier), and the phase noise characteristic of the phase frequency detector and the charge pump. The noise floor, however, would be determined by the noise floor of the system.

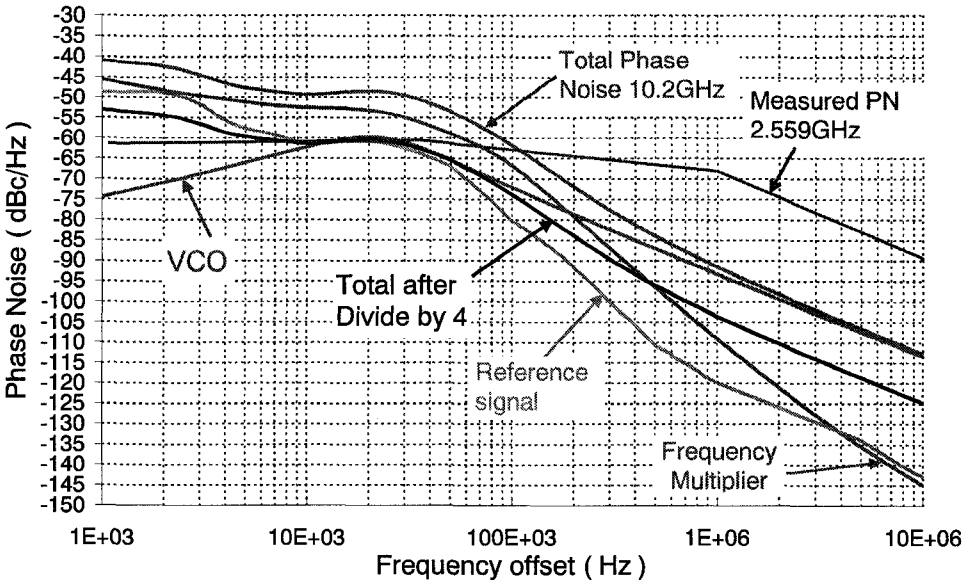


Figure 7.14: Calculated and measured phase noise of the output signal from the frequency synthesizer.

Finally, the measured phase noise of the output signal from the frequency synthesizer, shown in Figure 7.8, is compared to the calculated phase noise of the frequency synthesizer. Figure 7.14 shows the measured phase noise and the output referred phase noise of the reference signal, the frequency multiplier, and the VCO as major contributors to the total phase noise of the frequency synthesizer[2]. In addition to the phase noise due to the PFD, the CP, the VCO, and the frequency multiplier, the calculated phase noise includes the effect of the measured phase noise of the reference signal as well. Figure 7.14 depicts that, after the total phase noise is lowered by 12dB (because the signal from the VCO is sent through divide by 4 frequency divider to the spectrum analyzer) a close matching between the measured and the calculated

[2]The phase noise due to the PFD and the CP is included with the calculations, however this curve is not shown since its contribution to the total phase noise is smaller compared to the phase noise due to the reference signal and the frequency multiplier. Thus, to simplify the plot, the curve representing the phase noise due to the PFD and the CP is not shown in Figure 7.14.

phase noise is visible only within the frequency offset up to 30kHz[3]. The deviation between the calculated and the measured curve increases as the offset frequency increases. Assuming that the off-chip loop filter had the appropriate attenuation after the cut-off frequency, the reason for having higher out-of-band measured phase noise is due to the increased phase noise on the output node of the VCO. The test board accommodates signals with frequencies of 20MHz (the reference signal and the output of the frequency multiplier) and three output signals from the frequency synthesizer with frequencies around 2.5GHz, 5GHz, and 10GHz. Due to the attenuation of the PCB (FR4 material) the measurements were performed on the 2.5GHz signal from the frequency synthesizer (the VCO signal sent through divide by 4 frequency divider). All of these signals interfere with the measured signal and cause increase of the phase noise. Moreover, any noise coupled onto the power supply line of the frequency synthesizer directly affects the phase noise of the VCO. Because the feedback system acts as a high pass loop filter for the phase noise due to the VCO, this phase noise will cause increase of the out-of-band phase noise of the measured signal from the frequency synthesizer.

7.2.4 Measuring the Signal Waveforms

To get the on-chip signal waveforms, an Agilent Infiniium 54832D MSO 1GHz oscilloscope is used. The measurements are performed as single-ended and a high impedance input of the oscilloscope is used. The signals are taken from the output of the 14-bit frequency divider.

[3]The loop bandwidth of the frequency synthesizer is set to 22kHz.

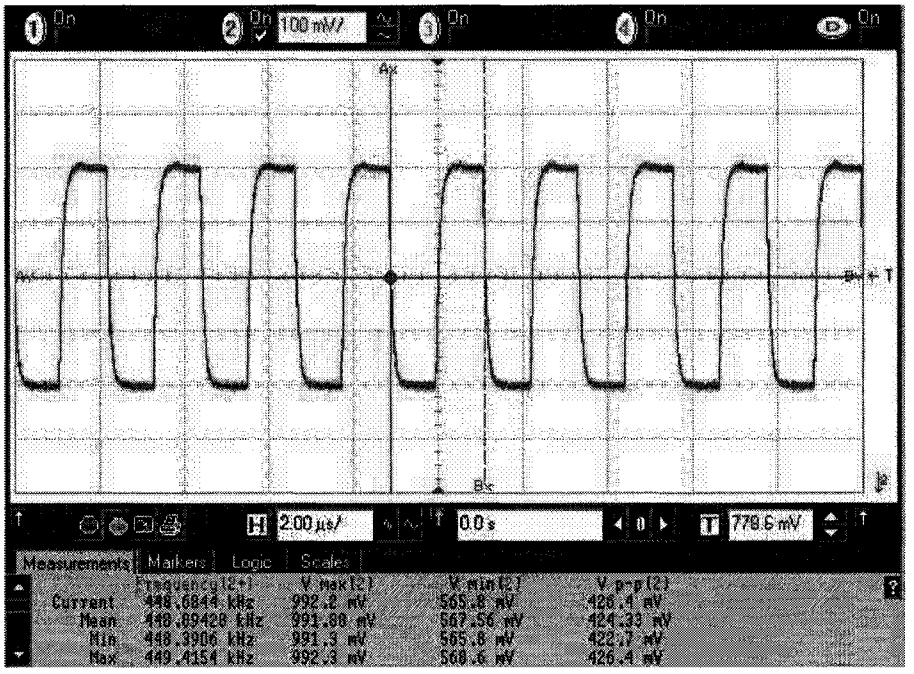


Figure 7.15: Output of the 14-bit frequency divider. The control bits are set to logic zero, and the power supply is set to 1V.

Figure 7.15 shows the measured waveform of the output signal from the 14-bit frequency divider. With 1V power supply and control bits set to logic zero, the frequency of the output signal from the frequency divider is 448.7kHz with peak-to-peak voltage of 426mV. The signal waveform is similar to the simulated results as shown in Figure 4.12 (Chapter 4).

Figure 7.16 shows a similar case with only difference that the control bits of the frequency divider were set to logical one. The measured frequency of the output signal from the frequency divider is 224.7kHz with 436.5mV peak-to-peak voltage swing.

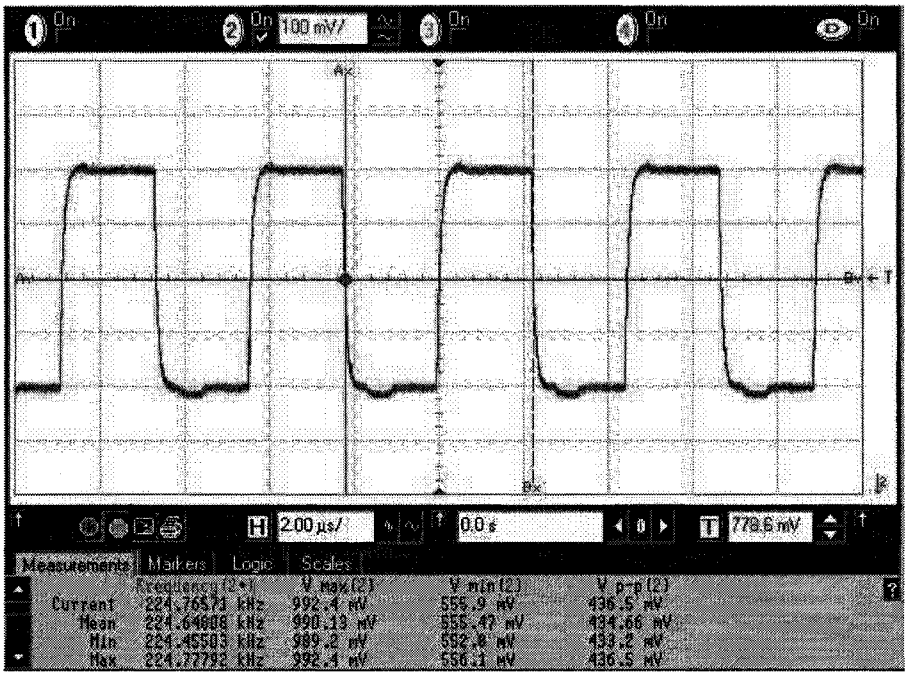


Figure 7.16: Output of the 14-bit frequency divider. The control bits are set to logic one, and the power supply is set to 1V.

7.3 Comparison to State-of-the-Art

7.3.1 Integer-N Frequency Synthesizers

The proposed architectural concept for a frequency synthesizer is modification of an integer frequency synthesizer. Thus, the simulated and the measured results of the divide-multiply implementation are compared to the results of the integer-N frequency synthesizers found in [50–54]. The cited work was selected based on the frequency of operation (around 10GHz). The main disadvantage of the integer-N frequency synthesizers is the channel resolution. The resolution of the integer-N frequency synthesizers is equal to the frequency of the reference signal. The divide-multiply implementation of the proposed frequency synthesizer used a 20MHz reference signal and 500kHz illustrative channel resolution was selected.

Table 7.1: Comparison of the experimental results with the integer-N frequency synthesizers prevalent in the literature.

Reference	[50]	[51]	[52]	[53]	[54]	This work	
						simulated	measured
Technology	CMOS 0.18μm	CMOS 0.18μm	CMOS 0.18μm	CMOS 0.18μm	CMOS 0.18μm	CMOS 0.13μm	CMOS 0.13μm
Supply	1.8V	1.8V	1.6V	2V	1.8V	1V	1-1.8V
Current Consumption	38.9mA Core	42.8mA Core	12.5mA Core	35mA Core	32.3mA Core	150.9mA Core	220mA Chip
Frequency	8.67-10.12 GHz	10-11 GHz	9-10.6 GHz	14.8-16.9 GHz	6.3-9 GHz	9.72-10.47 GHz	7.6-12 GHz
Reference	20MHz	10MHz	140MHz	23.5MHz	528MHz	20MHz	20MHz
In-band Phase noise @ Offset	NG	-90 dBc/Hz 10kHz	-105 dBc/Hz 10kHz	-56 dBc/Hz 10kHz	-97 dBc/Hz 10kHz	-55 dBc/Hz 10kHz	-62.1 dBc/Hz 10kHz
Out-of-band Phase noise @ Offset	-102 dBc/Hz 1MHz	-130 dBc/Hz 10MHz	-120 dBc/Hz 20MHz	-104.5 dBc/Hz 1MHz	-109.6 dBc/Hz 1MHz	-103 dBc/Hz 1MHz	-99.2 dBc/Hz 10MHz
Resolution	20MHz	10MHz	140MHz	23.5MHz	528MHz	500kHz	500kHz
Settling time	3μs	NG	0.5μs	4μs	150ns	35μs	65μs
Bandwidth	670kHz	500kHz	7MHz	NG	13.5MHz	30kHz	22kHz
Spurs @ Offset	-41dBc 20MHz	-48dBc 10MHz	-58 140MHz	-50dBc 23.5MHz	-52dBc 528MHz	-94.1dBc 36MHz	below noise floor
Layout area	1.35mm^2 Chip	0.43mm^2 Core	NG NG	0.96mm^2 Chip	0.77mm^2 Core	0.35mm^2 Core	2.5mm^2 Chip

The divide-multiply implementation presented in this thesis was not optimized with respect to the current consumption, in order to increase the probability of first-silicon operation. However, it was found that the optimized design would dissipate below 50mW power from a 1V supply, when operating with a 10GHz VCO. The optimized power dissipation of the divide-multiply implementation is comparable to the cited work.

Because the channel resolution of the integer-N frequency synthesizers is equal to

the reference frequency, the frequency divider within the feedback loop of the frequency synthesizer would have smaller division ratio compared to the divide-multiply implementation where the division ratio is in the range of 20000. Consequently, the effect that the frequency divider would have on the phase noise performance of the divide-multiply implementation is significantly higher compared to the cited work. For example, the in-band phase noise at 10kHz offset of the divide-multiply implementation is -55dBc/Hz. This number is only comparable to the reported in-band phase noise found in [53]. Thus, if the 500kHz channel resolution of the illustrated divide-multiply implementation is increased then the in-band phase noise would be improved as well. Similarly, the division ratio would affect the out-of-band phase noise performance as well. However, due to the narrow loop bandwidth of the divide-multiply implementation, the out-of-band phase noise is comparable to the cited work.

The frequency resolution of the cited integer-N frequency synthesizers allows implementation of high loop bandwidths. Consequently, the cited work reported fast switching speed. The loop bandwidth is disadvantage for the divide-multiply implementation. The 30kHz loop bandwidth resulted in a simulated lock-in time of 35μs, and the 22kHz loop bandwidth resulted in a measured lock-in time of 65μs.

The spurious performance is advantage for the divide-multiply implementation. The simulated results indicated a spur at 36MHz offset from 10GHz carrier. However, the spur was found to be 94dB below the carrier. The spurs of the measured frequency synthesizer were below the noise floor.

7.3.2 Fractional-N Frequency Synthesizers

The following part will compare the simulated and the measured results of the divide-multiply implementation to the results of $\Delta\Sigma$ based fractional frequency synthesizer found in [5, 55, 56].

Table 7.2: Comparison of the experimental results with the $\Delta\Sigma$ fractional frequency synthesizers prevalent in the literature.

Reference	[5]	[55]	[56]	This work	
				simulated	measured
Technology	CMOS 0.13μm	CMOS 0.09μm	CMOS 0.13μm	CMOS 0.13μm	
Supply	1.5V	1.4V	1.2V	1V	1-1.8V
Current Consumption	63mA Core	3mA Core	33.3mA Core	150.9mA Core	220mA Chip
Frequency	10.24-12.55 GHz	7.6-8.4 GHz	4.6-5.4 GHz	9.72-10.47 GHz	7.6-12 GHz
Reference	64MHz	NG	40MHz	20MHz	20MHz
In-band Phase Noise @ Offset	-80 dBc/Hz 10kHz	-100 dBc/Hz 10kHz	-70 dBc/Hz 10kHz	-55 dBc/Hz 10kHz	-62.1 dBc/Hz 10kHz
Out-of-band Phase Noise @ Offset	-104 dBc/Hz 10MHz	-117 dBc/Hz 10MHz	-134 dBc/Hz 10MHz	-130 dBc/Hz 10MHz	-99.2 dBc/Hz 10MHz
Resolution	18kHz	10kHz	125kHz	500kHz	500kHz
Settling time	8μs	20μs	10μs	35μs	65μs
Bandwidth	1MHz	500kHz	800kHz	30kHz	22kHz
Frac-spurs @ Offset	-44dBc 18kHz	-60dBc 10kHz	NG	-94.1dBc 36MHz	below noise floor
Ref-spurs @ Offset	-52dBc NG	below noise floor	NG	-98.8dBc 40MHz	below noise floor
Layout area	0.64mm^2 Core	NG	NG	0.35mm^2 Core	2.5mm^2 Chip

The comparison indicates that disadvantage of the divide-multiply implementation, illustrated with a PLL based frequency multiplier, is the in-band phase noise

performance. Nevertheless, 20-30dB improvements of the phase noise due to the frequency multiplier is practically hard to accomplish even with a different type of a frequency multiplier. Thus, regarding the phase noise performance, the divide-multiply implementation is not recommended (practical) for 10GHz applications.

The measured lock-in time of the divide-multiply implementation is 65μs with a loop bandwidth of 22kHz. The lock-in speed attained in this design iteration already exceeds that required for many wireless standards, for example, Bluetooth [7] and WiMAX [57]. Nevertheless, the narrow loop bandwidth is another disadvantage of the divide-multiply implementation. The cited work reported better switching speed practically due to the implementation of a higher loop bandwidth.

The reduction of the fractional and the reference spurs is the primary advantage of the divide-multiply implementation compared to the $\Delta\Sigma$ fractional frequency synthesizers prevalent in the literature. The fractional and the reference spurs were not visible in the output spectrum of the measured signals from the frequency synthesizer. The reason is that the fractional and the reference spurious tones were below the measured noise floor of the signals. Based on the measured results it can be concluded that the fractional and the reference spurs are below -70dBc. The simulated results indicated fractional spurs of -94.1dBc and reference spurs of -98.8dBc. However, the fractional spurs appeared at 36MHz offset, while the reference signal appeared at 40MHz offset from the carrier. Additional filtering of the output signal can further reduce the spurious tones. Therefore, the frequency synthesizer that implements the divide-multiply implementation can be denoted as a spur-free fractional frequency synthesizer.

Finally, the core of the divide-multiply implementation required only $0.35mm^2$ chip area, assuming off-chip loop filters.

7.4 Summary

The divided-multiply implementation of the proposed architectural concept for a frequency synthesizer was fabricated in a $0.13\mu m$ CMOS technology. This chapter discussed the test setup, the equipment and the performed measurements of the fabricated chip. The chip was bonded onto a double-side PCB in order to facilitate the measuring process. The measured phase noise was discussed and compared to the simulated results in order to find the cause of discrepancy between the measured and the simulated results. It was found that the test board, the bond wires and the used equipment affected the measured results. When the aforementioned effects were included with the simulated results, the agreement between the simulated and the measured results was significantly improved. The signal waveform and the output spectrum were also discussed in the measurement section. The simulated and the measured results of the divide-multiply implementation were compared to the integer-N and the fractional-N frequency synthesizers. The advantage of the divide-multiply implementation is the spurious performance. The disadvantages of the divide-multiply implementation are the phase noise performance, and the requirements for a narrow loop bandwidth.

Chapter 8

Conclusion and Future Work

A frequency synthesizer with reduced spurious tones was the primary contribution with this thesis work. In order to reduce the spurious tones, this thesis proposed a novel system architecture for a frequency synthesizer along with the improvements of the custom PLL building blocks.

Chapter 3 presented a novel frequency synthesizer. The new system was derived from an integer-N frequency synthesizer by replacing the frequency divider, within the feedback loop, with subsystem of frequency dividers and frequency multipliers. Out of many possible implementation of the novel system architecture, this chapter illustrated three possible implementations and they were discussed in this thesis. The simulated results of the first implementation, denoted as divide-multiply implementation, were discussed in this chapter. In addition, the transient and the phase noise analysis of the novel system were included in this chapter.

Chapter 4 discussed the architecture of the frequency divider used with the divide-multiply implementation. The high frequency of operation (10GHz) and the reduced power supply resulted in new differential gates with resistor tail bias. The potential disadvantage of the frequency divider which implemented the differential cells with

resistor tail bias is the PSRR. This chapter discussed the PSRR of a single divider 2/3 cell as well as a 14-bit frequency divider based on the simulated results. The conclusion of the simulated results, regarding the power supply rejection ratio, was that if the frequency of the output signal from the frequency divider is higher compared to the frequency of the coupled signal on the power supply (for example 5GHz compared to 900MHz) then the amplitude modulation of the output signal is a problem. However, if the frequency of the output signal from the frequency divider is lower compared to the frequency of the coupled signal (for example 305.2kHz compared to 900MHz), then the coupled signal would not modify the transition edges of the output signal resulting in a correct functionality of the phase frequency detector.

The advantages of using the proposed differential cells for a frequency divider included simplified design, operation from a reduced power supply and improvements of the phase noise of the frequency divider.

Moreover, through the Monte Carlo simulation, this chapter discussed the effect of the process variations and the circuit mismatch on the operation of a single 2/3 divider cell. The conclusion of the Monte Carlo simulations was that, the functionality of the divider cell (determined through a correct division of the input frequency) designed and optimized to operate at 10GHz was not affected by the process variations and the circuit mismatch. However, for a constant power dissipation, the lower frequency of operation of a single 2/3 divider cell, implementing the resistor bias technique and optimized to operate at high frequencies, is limited. Therefore, a divider cell optimized for high frequencies would need additional re-sizings of the custom differential cells in order to use the same divider cell at lower frequencies.

Chapter 5 discussed the design and the phase frequency detector and the novel charge pump of the divide-multiply implementation. The PFD had a linear tracking characteristic, and implemented the differential gates as discussed in Chapter 4. The charge pump had two complementary current sources, and two complementary differential pairs for the UP and Down signals improving the current matching of the charge pump. Through the comparison of the simulated results of the proposed charge pump and the charge pump that was used as a reference, the current mismatching of the charge pump presented in this thesis work was improved by 3.2 times. A possible calibration of the proposed charge pump, which will allow the improved current matching to be attained for the complete working region of the proposed charge pump, is discussed further in this chapter.

Chapter 6 discussed the LC-type voltage controlled oscillator implemented with the divide-multiply implementation. In order to accomplish a low power and a low phase noise performance, the VCO used a resistor tail bias technique. Moreover, this chapter discussed the selection of a varactor topology based on the simulated phase of the varactor impedance seen from the VCO oscillation node.

The theoretical discussion in this chapter include the derivation of a formula to predict the frequency of oscillation of a fully integrated 10GHz LC VCO in a 0.13μm CMOS technology. This formula was used to investigate the temperature effect on the VCO frequency.

The simulated results of the VCO (FOM and power dissipation) were compared with other 10GHz VCO designs prevalent in the literature. The low power dissipation and the low phase noise resulted in an attractive FOM of the presented VCO.

Chapter 7 discussed the test setup and the used equipment to measure the divide-multiply implementation. The test setup included a double-side PCB fabricated with FR4 material and finished with Emerson gold. Compared to the $\Sigma\Delta$ based frequency synthesizers, the measured results of the divide-multiply implementation show that the spurious tones were reduced without the need of any spur cancelation techniques. This feature of the divide-multiply implementation (as well as any implementation of the proposed architectural concept) is somehow expected because the proposed system is modification of an integer-N frequency synthesizer. The integer-N frequency synthesizers are sensitive to the non-idealities of the PLL blocks, for example the leakage of the charge pump, which cause ripples on the loop voltage and thus spurious tones in the output spectrum. A careful design of the PLL blocks as well as a proper selection of the loop bandwidth would cause reduction of the spurious tones. The cause for the discrepancy between the simulated and the measured phase noise of the output signal from the frequency synthesizer was investigated. It was found that the test board, the bond wires and the used equipment affected the measured results.

The simulated and the measured results of the divide-multiply implementation were compared to the integer-N and the fractional-N frequency synthesizers. The main advantage of the divide-multiply implementation was found to be the spurious performance. The disadvantages of the divide-multiply implementation are the phase noise performance, and the narrow loop bandwidth.

Three appendixes are following this chapter. While Appendix C discusses the transition processes in the loop filter while the PLL acquires a stable state, the promising

performances of the proposed system (phase noise, loop bandwidth, switching speed, etc) are further discussed in Appendixes A and B through analyzing two additional implementation of the proposed system denoted as multiply-divide implementation and divide-multiply-divide implementation.

The multiply-divide implementation of the proposed frequency synthesizer was discussed in Appendix A. This implementation was illustrated for 1GHz frequency of operation with 500kHz frequency resolution. Based on the comparison between the simulated results of the multiply-divide implementation and the experimental results of the $\Sigma\Delta$ based frequency synthesizers found in the literature as well as the numerical specification for the phase noise of the GSM standard, the advantages of the multiply-divide implementation are the phase noise performance (both the in-band and the out-of-band phase noise), the size of the loop filter, the switching speed, and the spurious performance. The main disadvantage of the multiply-divide implementation is the maximum frequency of operation for a given channel spacing.

The divide-multiply-divide implementation of the proposed frequency synthesizer was illustrated for two frequencies (1GHz and 10GHz) with frequency resolution of 500kHz. The divide-multiply-divide implementation solves the disadvantage of the multiply-divide implementation regarding the highest application frequency. Meanwhile, the other advantages of the divide-multiply-divide implementation (the phase noise, the size of the loop filter, the switching speed, and the spurious performance) are similar to the advantages of the multiply-divide implementation. The main disadvantage of the divide-multiply-divide implementation is the programming of the frequency dividers and the frequency multiplier for a given channel spacing.

The conclusion based on the simulated (measured) results of the proposed frequency synthesizer is that the implementation of the proposed system can affect the performance of the frequency synthesizer. For example, the divide-multiply implementation is practically not useful for 10GHz application due to the poor phase noise performance, limited loop bandwidth and thus the switching speed. The multiply-divide implementation, however, is illustrated to have good performances but limited frequency of operation making it not practical for applications in GHz range. On the other hand, the divide-multiply-divide implementation is illustrated to have attractive performances compared to $\Sigma\Delta$ frequency synthesizers found in the literature. Thus, the divide-multiply-divide implementation can be the potential implementation of the proposed frequency synthesizer in a practical application. Nevertheless, it should be pointed that the $\Sigma\Delta$ frequency synthesizers have superior advantage regarding the frequency resolution compared to any implementation of the proposed frequency synthesizer. Although the proposed architecture can allow fine channel resolution, it is practically not useful if the channel resolution of the frequency synthesizer is required to be around 1Hz or less. For those applications, the $\Sigma\Delta$ frequency synthesizers are the frequency generators of choice.

Finally, the proposed frequency synthesizer and the implemented circuit designs were illustrated in 0.13μm CMOS technology. The system design itself does not prevent the implementation of the proposed frequency synthesizer in any new modern technologies. For example, the proposed frequency synthesizer was demonstrated at 10GHz of operation. As the CMOS technology scales down, the unity gain (f_t) of the device increases. In the modern processes, the actual performance of the circuit

design including the maximum frequency is highly layout dependent [58]. For example, [59] reported that the maximum measured frequency for the 0.13μm CMOS technology is 135GHz. In the advanced technologies (for example 90nm, 65nm, etc) this frequency is further increased. Therefore, the maximum frequency of the application that eventually will use the proposed system architecture for a frequency synthesizer is only limited by the implemented technology.

Regarding the phase noise performance, it was illustrated, with the discussion of the three possible implementations of the new frequency synthesizer, that the in-band phase noise is determined by the phase noise of the PFD and the CP (multiply-divide and divide-multiply-divide implementation) or by the phase noise due to the frequency multiplier (divide-multiply implementation). However, the in-band phase noise is predominantly determined by the division ratio between the VCO frequency and the reference frequency but not the utilized CMOS technology. Regarding the out-of-band phase noise, if a narrow-band loop bandwidth is used then this phase noise would be determined by the phase noise of the VCO. The phase noise of the VCO is process dependent due to the following reasons.

Assuming that an LC type of a VCO would be used, then the phase noise of the VCO will depend from the quality factor of the tank circuit (i.e. the quality of the passive devices). In CMOS scaling, both active devices and lowest interconnection line scale down with technology [60]. However, the top metal thickness as well as total dielectric insulator thickness becomes thicker. The thicker top metal, the lower dielectric constant and farther top-level to substrate distance, cause reduction of the

substrate loss and parasitics. As a result, the quality-factor of the inductor is improved [61]. Moreover, the implementation of better interconnect technology (for example Cu compared to Al) will cause the performance of on-chip inductors further to improve[1].

The CMOS scaling has an effect on the quality factor of the CMOS varactor as well. The quality factor of the varactor depends from the series equivalent resistance. The dominant contribution of the series equivalent resistance is the channel resistance [60]. Because the channel resistance scales down as the channel length scales, the quality factor will increase significantly [60].

Due to the improvements of the quality factor of the tank circuit, it would be expected that the phase noise of the VCO will improve as well. However, as CMOS technology scales down, the gate-oxide thickness also scales down. The thin gate oxide results in lower breakdown voltage of the device. The reduction in the device breakdown voltage reduces the power supply and the RF voltage swing, which lowers the RF output power, and degrades the phase noise [62].

In conclusion regarding the phase noise of the VCO, although the modern processes is expected to accommodate passive devices with improved quality factor, the reduced RF output power from the VCO causes degradation of the VCO phase noise. Nevertheless, a high loop bandwidth of the frequency synthesizer suppresses the phase noise of the VCO. Thus, for the implementation of the proposed synthesizer which allows a high loop bandwidth the phase noise of the VCO is not a significant problem.

[1]However, inductance will not scale as transistor. In other words, silicon areas being occupied by on-chip inductors will not scale down even though CMOS technology advances [60].

8.1 Future Work

8.1.1 Charge Pump Calibration

Chapter 5 briefly pointed that an additional feature that should be noted for the proposed charge pump is that the complementary current sources may be adjusted via a calibration circuit to optimize the charge pump characteristic behavior. The calibration circuit was not included with the divide-multiply implementation. The advantage of a calibration circuit can be illustrated through the following discussions.

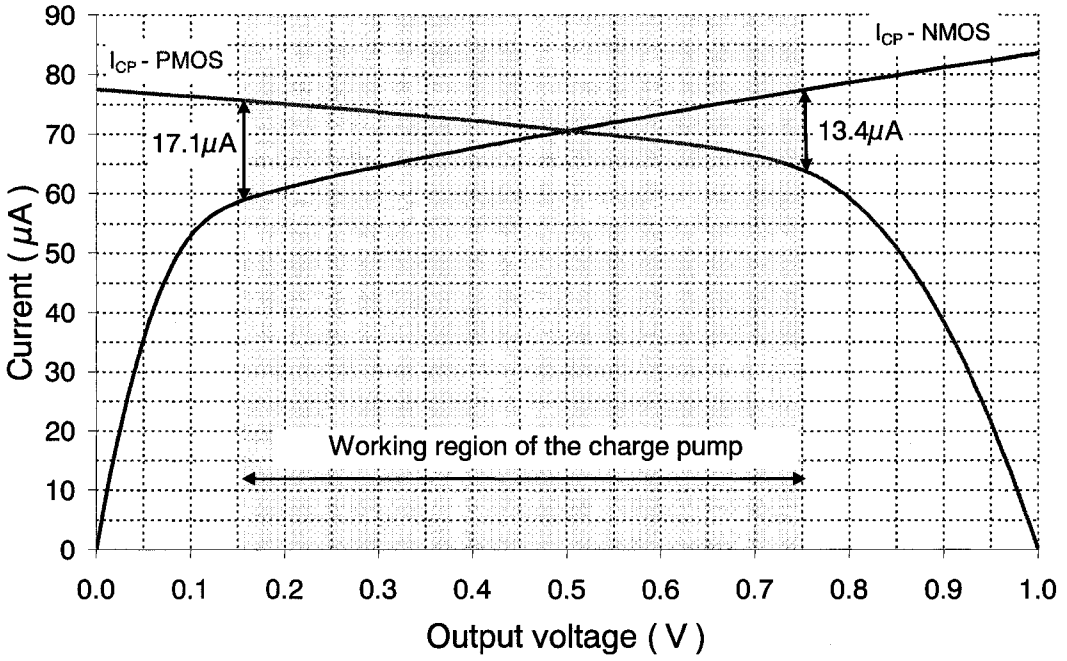


Figure 8.1: Charge pump current as a function of the output voltage when the complementary current sources are not calibrated.

Figure 8.1 shows the simulated charge pump currents (both UP and Down signals) as a function of the loop voltage. As the loop voltage varies, the mismatch between the UP and the Down charge pump current is evident. If a Monte Carlo simulation is performed (process and mismatch, 100 runs) when the loop voltage is set at 0.3V

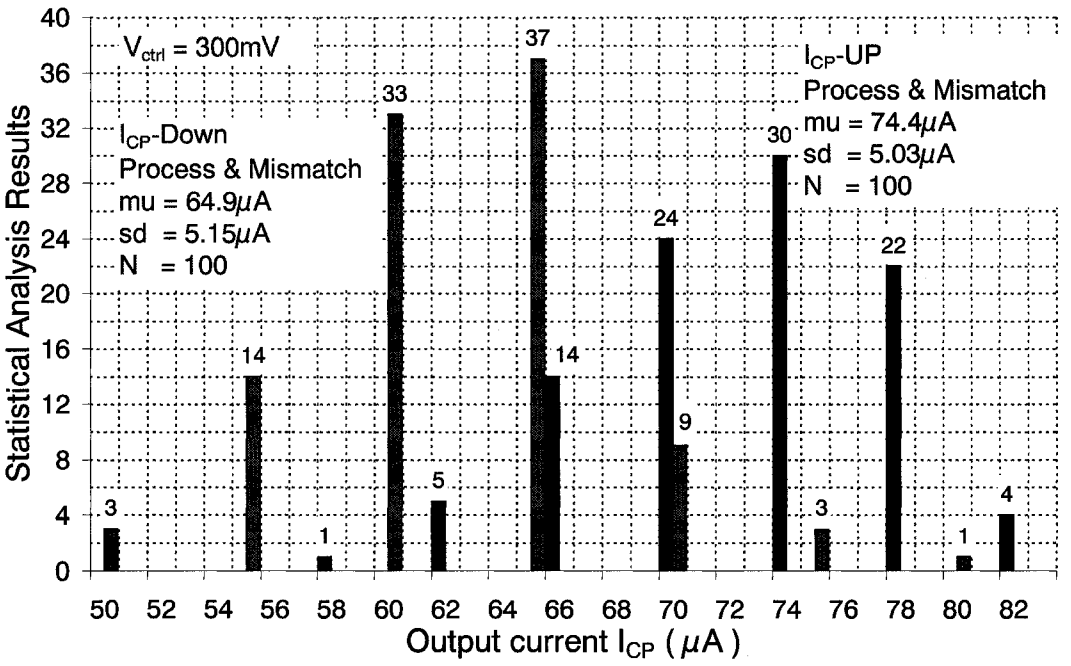


Figure 8.2: Monte Carlo simulation for a non-calibrated charge pump current sources.

then Figure 8.2 shows that the averaged difference between the UP and the Down charge pump current would be 9.5μA. Thus, the deviation between the UP and the Down currents of the non-calibrated charge pump is 14.64%.

Figure 8.3 shows the case when the complementary current sources are tracking the loop voltage and are calibrated appropriately. The calibrated charge pump current generates equal UP and Down currents for the entire working region of the charge pump. If a Monte Carlo simulation is performed (process and mismatch, 100 runs) when the loop voltage is set at 0.3V then Figure 8.4 shows that the averaged difference between the UP and the Down charge pump current would be 0.3μA. Thus, the deviation between the UP and the Down currents of the calibrated charge pump is only 0.43%. The smaller deviation between the UP and the Down charge pump currents would reduce the ripples that can appear on the loop voltage (control signal).

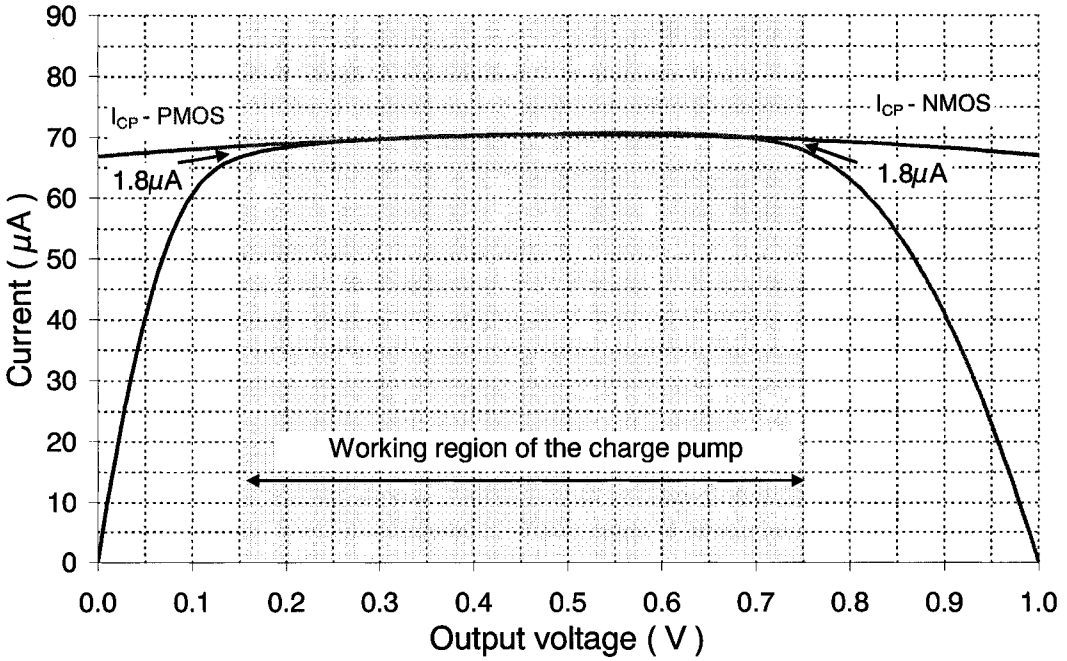


Figure 8.3: Charge pump current as a function of the output voltage when the complementary current sources are calibrated.

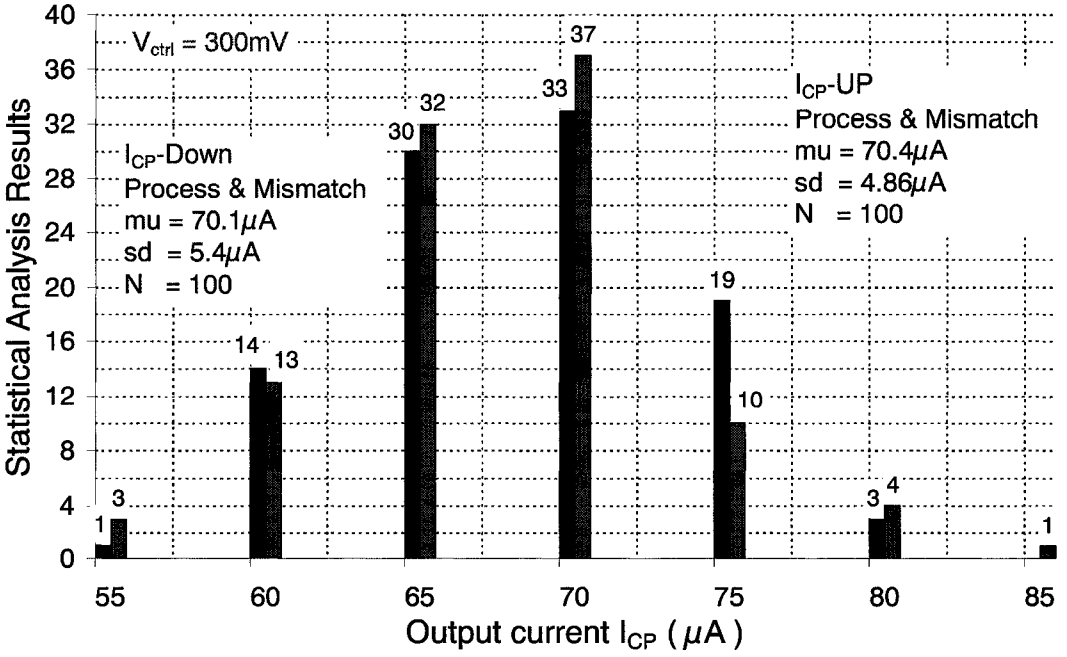


Figure 8.4: Monte Carlo simulation for a calibrated charge pump current sources.

The reduced ripples would also reduce the fractional and the reference spurs.

8.1.2 Optimization and Targeting a Specific Application

A future work would be an optimized implementation of the proposed frequency synthesizer regarding the phase noise performance, the power dissipation, the size of the loop bandwidth, the switching speed, and the spurious performance. In order to know the numerical specifications regarding the aforementioned performances, the future work would be to implement the proposed system for a frequency synthesizer for a specific application. The channel spacing, the phase noise and the application frequency would determine which implementation to be used (i.e. the divide-multiply implementation, the multiply-divide implementation, or the implementation that would utilize multiple number of frequency dividers and multiple number of frequency multipliers). Regarding the phase noise, it was illustrated that the divide-multiply implementation is not recommended particularly when the division ratio of the frequency divider is high as illustrated in the thesis. Thus, if the phase noise is the criteria of merit of the targeted application then a different implementation of the proposed system architecture should be considered.

Appendix A

Multiply-Divide Implementation

As discussed in Chapter 3, there are multiple implementations of the proposed architectural concept for a frequency synthesizer. As an alternative to the divide-multiple implementation that was discussed in the previous chapters is the multiply-divide implementation.

A.1 An Example Case

For completeness of this chapter, the block diagram of the multiply-divide implementation is repeated here and shown in Figure A.1.

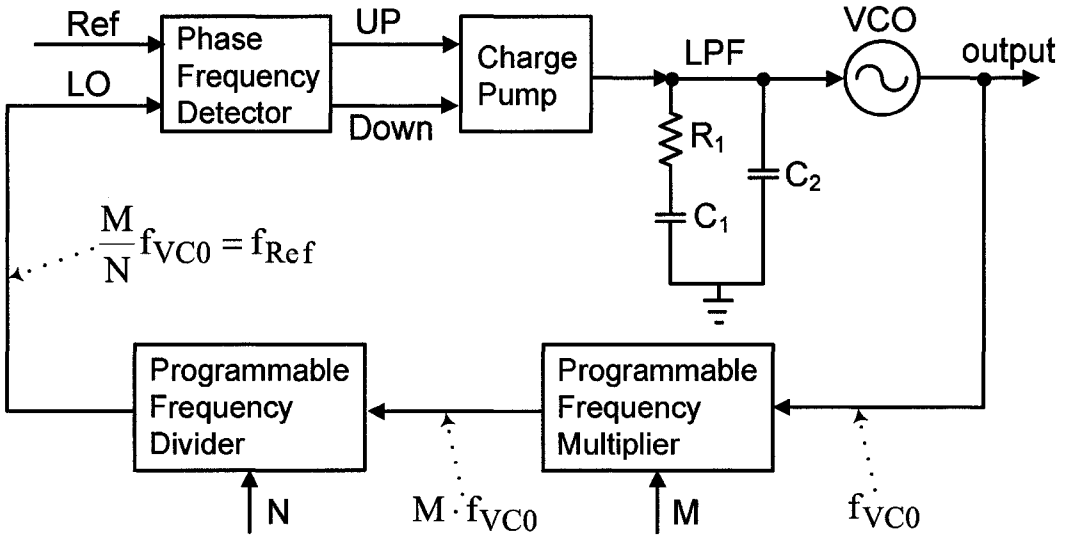


Figure A.1: A block diagram of the multiply-divide implementation.

To illustrate this implementation, the proposed fractional frequency synthesizer

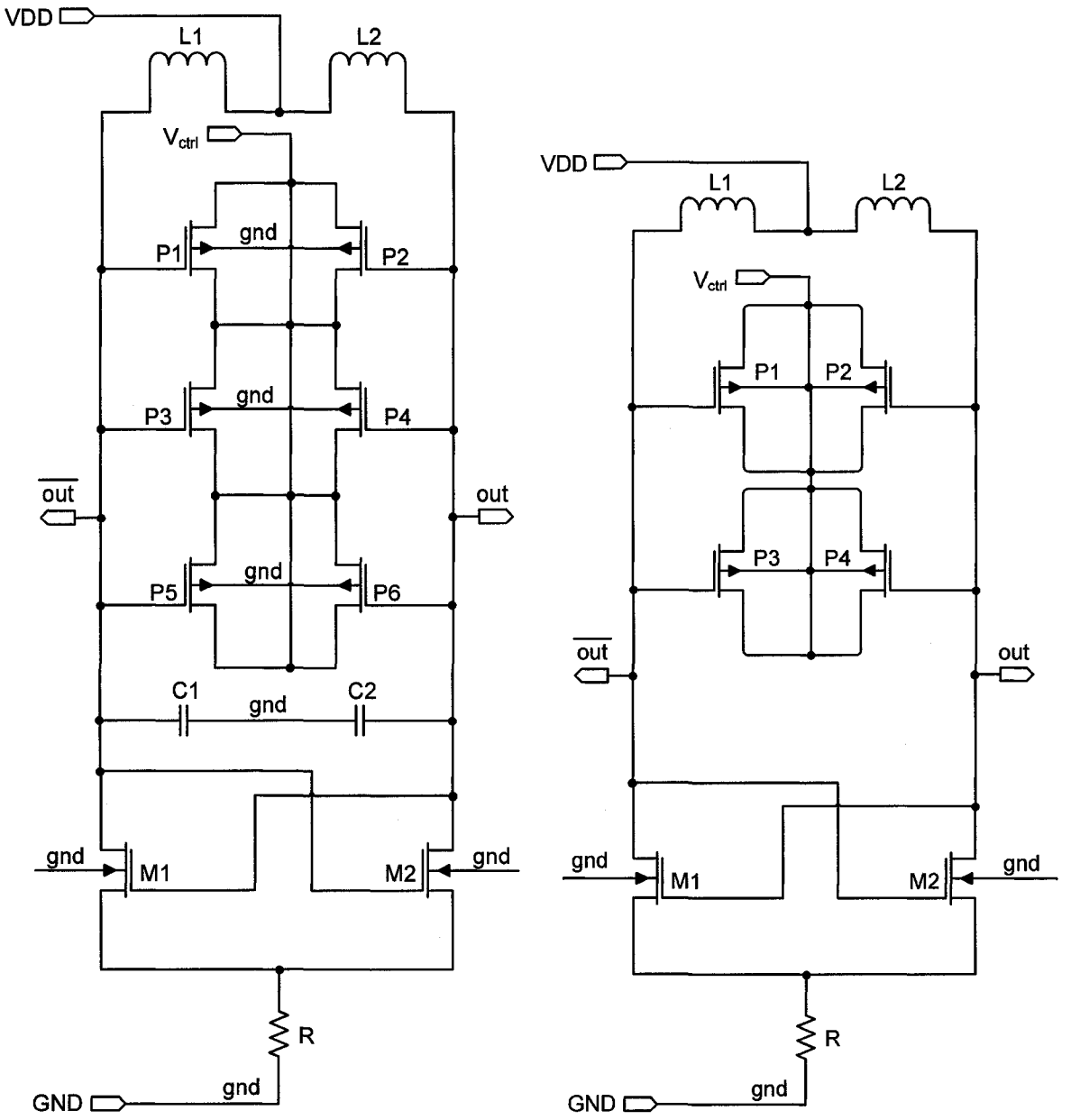
was implemented in a 0.13μm CMOS technology. The frequency of the reference signal was chosen as 20MHz again, and the charge pump, the phase frequency detector, and the frequency divider used with the divide-multiply implementation were reused.

The very first disadvantage of the multiply-divide implementation is the frequency of operation. For example, if a 10GHz operation with 500kHz spacing is targeted then the frequency multiplier should operate around 400GHz. This frequency is not visible for the 0.13μm CMOS technology. Therefore, to illustrate the multiply-divide implementation, 1GHz of operation with 500kHz channel spacing was chosen.

Similarly to the divide-multiply implementation, a PLL type of the frequency multiplier was used for illustration purposes only. Consequently, the multiply-divide implementation requires two voltage-controlled oscillators. The VCO within the main system operates around 1GHz, while the VCO within the frequency multiplier operates around 40GHz.

The 1GHz oscillator, within the main system, implemented the VCO topology depicted in the Chapter 6 (Figure 6.1) and is shown in Figure A.2(a). Figure A.3 shows the simulated results of this oscillator. The 1GHz VCO utilized two inductors sized with outer dimension of 350μm, metal width of 10μm[1], spacing between the metal paths of 5μm (default), and number of turns set to 7.25. Figure A.3(a) shows the simulated inductance of the aforementioned inductor for the frequencies between 100MHz and 100GHz. The simulated inductance at 1GHz is 14.4nH. The simulated quality factor at 1GHz is 13.3 as shown in Figure A.3(b). SpectreRF (the pss and the pnoise analysis) utilizing the bsim 4 models was used to simulate the phase noise of the VCO. Figure A.3(c) shows the phase noise when the VCO operates at 1GHz from 1V supply. The phase noise at 10kHz offset is -81dBc/Hz, at 100kHz offset is -107dBc/Hz, and at 1MHz offset is -129dBc/Hz. With a power dissipation of 788.3μW

[1]The inductors throughout this thesis were sized such that the peak of the quality factor to be close to the oscillation frequency. The metal width of 10μm was found, through the simulations, to be the optimum sizing regarding the phase noise of the VCO.



(a) VCO within the main system

(b) VCO within the frequency multiplier

Figure A.2: Schematic of the 1GHz VCO and the 40GHz VCO within the multiply-divide implementation.

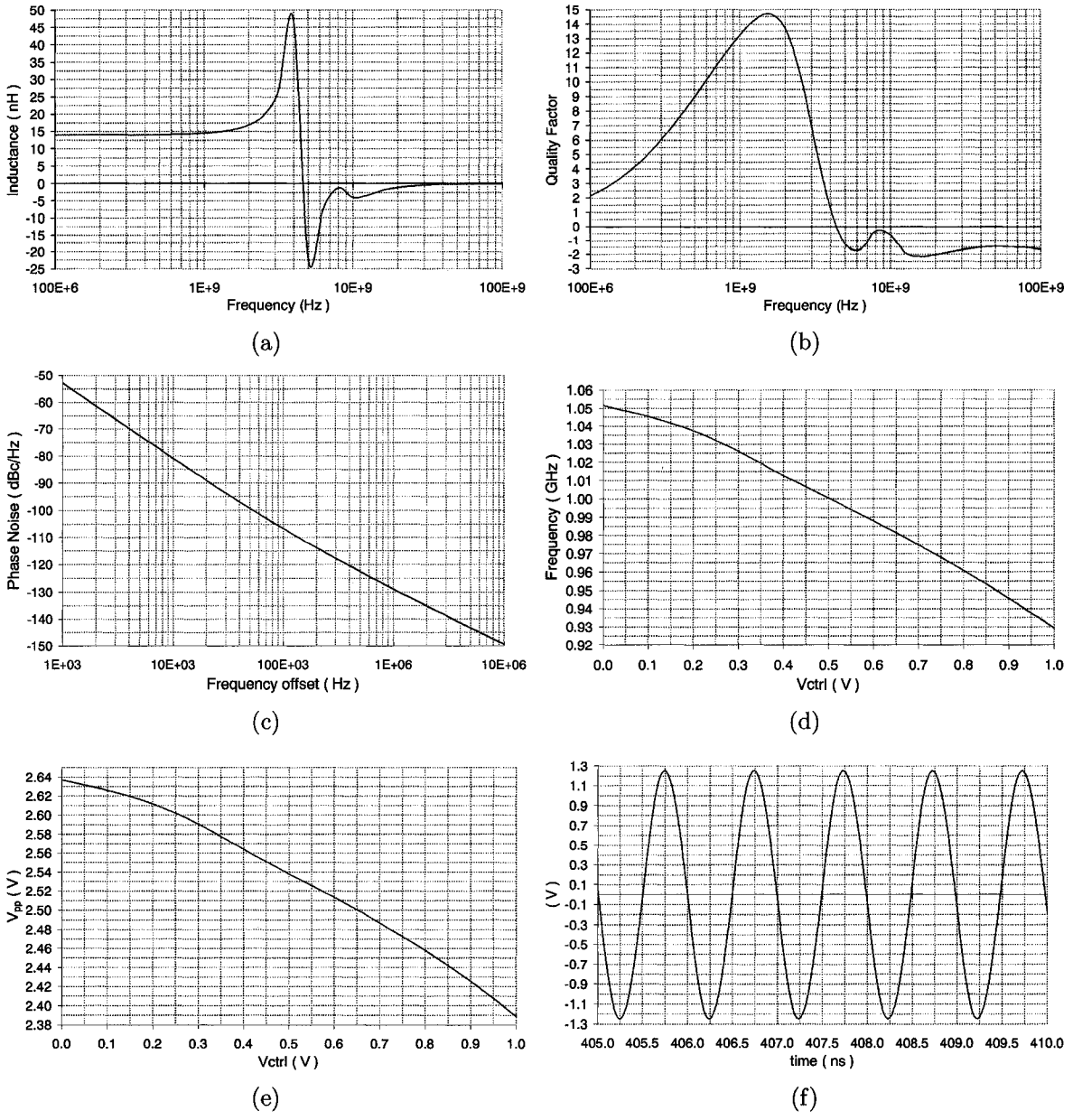


Figure A.3: Simulated performances of the 1GHz VCO within the main system.

from 1V supply the figure of merit (FOM) of this VCO at 1MHz offset is -190dB. The VCO can be tuned between 930MHz and 1.05GHz as shown in Figure A.3(d). The VCO generated signals with sinusoidal waveform. The peak-to-peak voltages (differential) vary between 2.39V and 2.64V as depicted in Figure A.3(e). Figure A.3(f) shows the sinusoidal waveform of the output signal from the VCO.

The 40GHz VCO, within the frequency multiplier, implemented a similar LC topology (with a resistor tail bias) as well and is shown in Figure A.2(b). The inductors of this VCO were sized with outer dimension of 100μm, metal width of 10μm, spacing between the metal paths of 5μm (default), and number of turns set to 1. Figure A.4(a) shows the simulated inductance of one inductor for the frequencies between 100MHz and 100GHz. The simulated inductance at 40GHz is 150pH. The simulated quality factor at 40GHz is 30 as depicted in Figure A.4(b). Figure A.4(c) shows the phase noise when the VCO operates at 40GHz from 1V supply. The phase noise at 10kHz offset is -32dBc/Hz, at 100kHz offset is -60dBc/Hz, and at 1MHz offset is -87dBc/Hz. With a power dissipation of 1mW from 1V supply the FOM of this VCO at 1MHz offset is -179dB. The VCO can be tuned between 31.2GHz and 42.7GHz as shown in Figure A.4(d). The VCO generated signals with peak-to-peak voltages (differential) between 0.78V and 1.5V as depicted in Figure A.4(e). The steady-state differential waveform of the output signal from the VCO is illustrated in Figure A.4(f).

To illustrate the values for the switching speed of the multiply-divide implementation, the loop bandwidth of the frequency multiplier was sized with a damping factor of 0.707 and a natural frequency of 5MHz. The loop bandwidth of the main system (Figure 3.2) was sized with a damping factor of 0.707 and a natural frequency of 1MHz. The aforementioned values were selected in order to accomplish a fast switching speed. Figure A.5 shows the acquisition time for the aforementioned scenario. The proposed system was first locked at 1GHz. The control signal for the VCO was

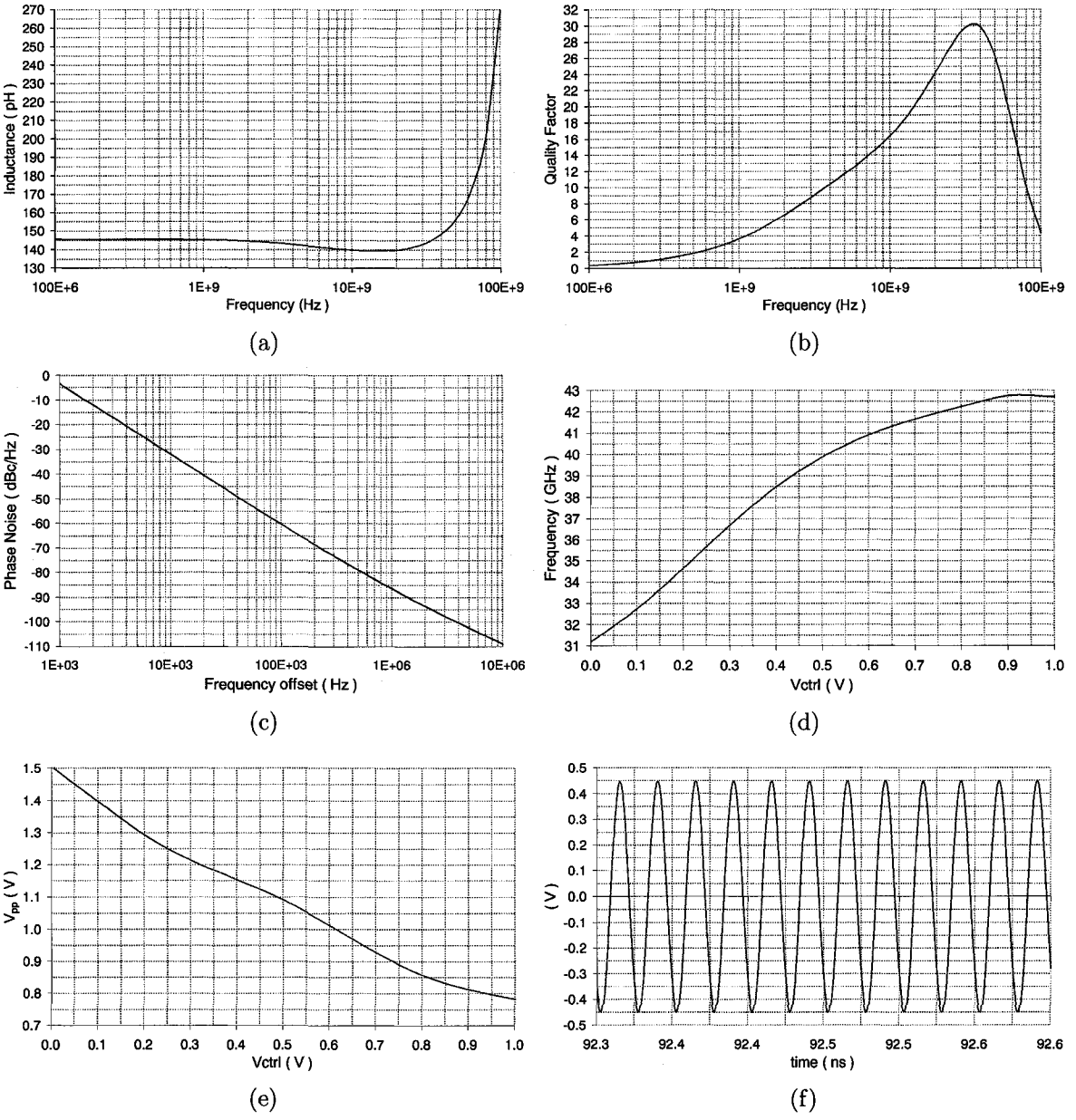


Figure A.4: Simulated performances of the 40GHz VCO within the PLL type frequency multiplier.

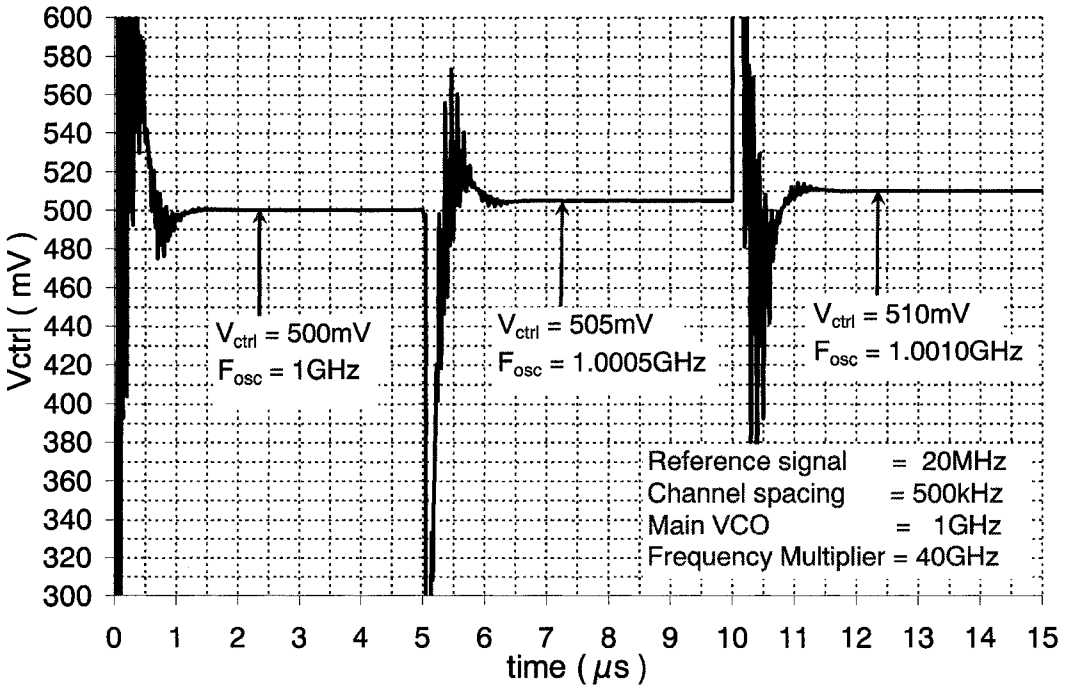


Figure A.5: Acquisition time for the multiply-divide implementation.

settled at 500mV. At 5μs the system was forced to lock at the neighboring channel separated by 500kHz. After approximately 1.5μs ($\pm$0.06% of the final control voltage value) the control signal for the VCO settled at its new value and the fractional frequency synthesizer generated a signal with a frequency of 1.0005GHz. At 10μs the system was forced to lock at the next neighboring channel. After the acquisition time, the control signal for the VCO settled at its new value and the fractional frequency synthesizer generated a signal with a frequency of 1.0010GHz.

The frequency multiplier, in the multiply-divide implementation, acquires lock to the output signal of the main VCO. As a result of the non-idealities of the PLL blocks of the frequency multiplier, ripples, with period equal to the main VCO frequency (around 1GHz), can appear on the loop voltage within the frequency multiplier. These ripples will cause spurious tones to appear on the output signal from the frequency multiplier. However, the loop bandwidth of the frequency multiplier is small enough

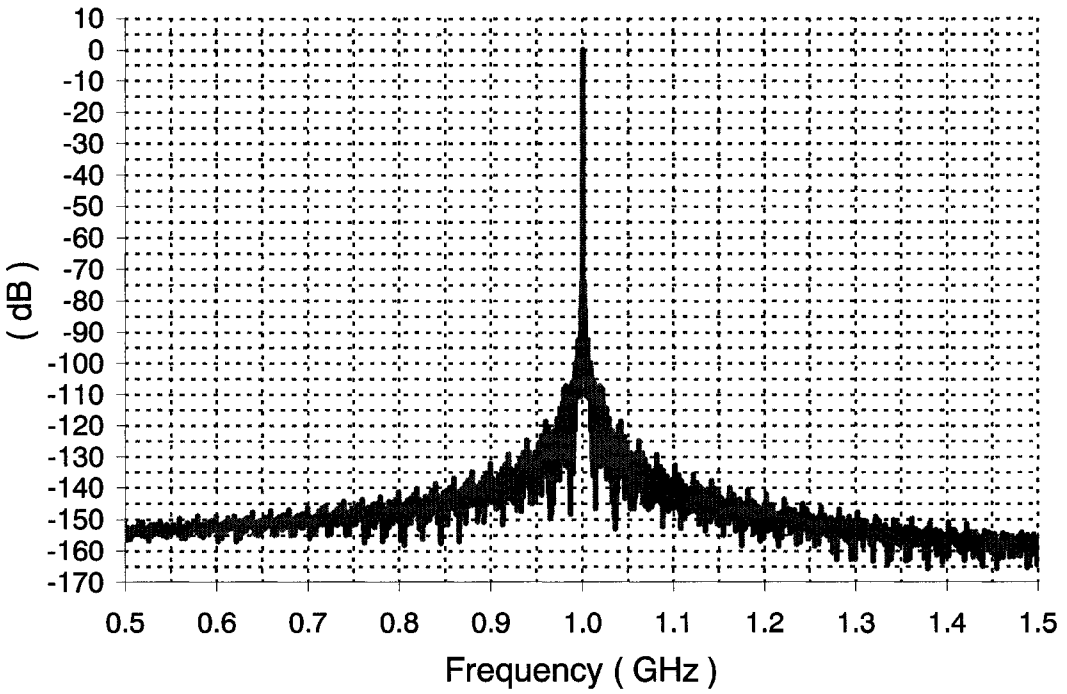


Figure A.6: Multiply-divide implementation: DFT of the output signal from the frequency synthesizer.

to reduce these spurs. Therefore, similar to the divide-multiply implementation, the fractional spurs theoretically can appear in the output spectrum of the proposed system. Practically, however, these spurs are reduced through the loop filtering such that the proposed loop system does not need any known fractional spurs cancellation techniques implemented with the $\Delta\Sigma$ frequency synthesizers, for example.

Figure A.6 shows the DFT of the generated signal with a frequency of 1.0005GHz, within the time period where the system is locked. The noise floor of the output spectrum is around -150dB. The noise floor is due to the performance of the main VCO (operating at 1GHz in the particular example).

Figure A.7 shows the estimated total phase noise of the generated signal from the proposed multiply-divide implementation that implements a PLL type of a frequency multiplier. The curves are drawn based on the theory discussed in section 3.1.5. The phase noise due to the frequency multiplier is lowered through the frequency divider

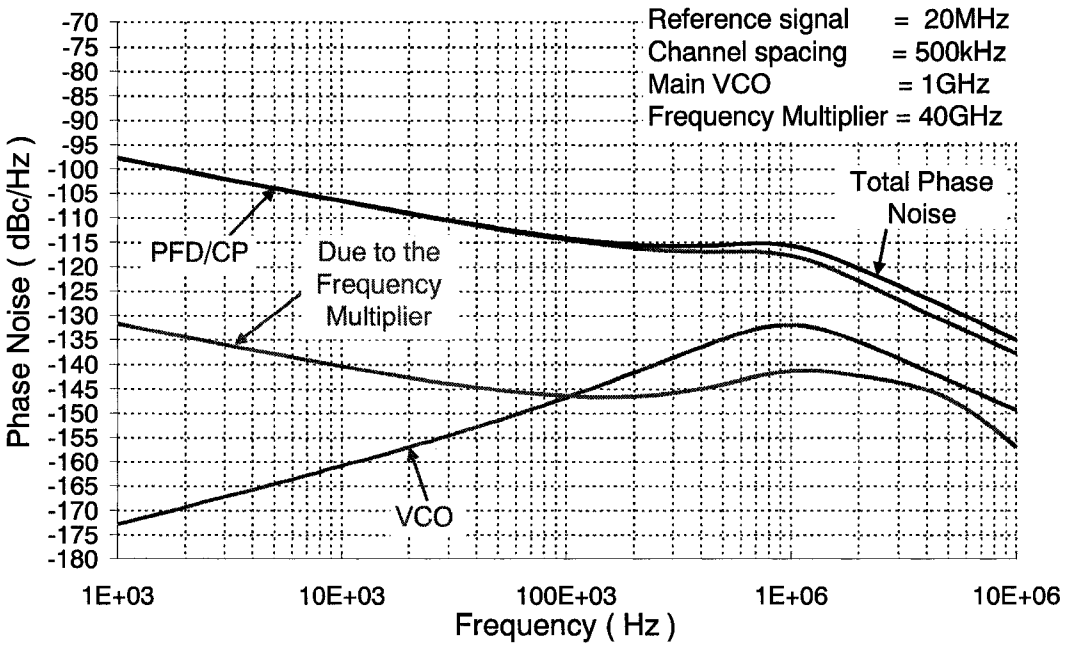


Figure A.7: A particular example of the phase noise of the multiply-divide implementation.

by 20logN. Due to the high loop bandwidth, the in-band phase noise of the VCO is suppressed resulting the phase noise due to the PFD-CP in the main system (Figure 3.2) to be a major contributor to the total phase noise. Although the results are based on the design choices while implementing the proposed concept, it should be evident that the phase noise from the phase frequency detector and the charge pump, within the main system, determines the overall phase noise performance of the proposed system. Thus, in order to improve the phase noise of the proposed multiply-divide implementation, an emphasis should be put on the the minimization of the phase noise due to the PFD-CP within the main system.

A.2 Comparison to Divide-Multiply Implementation

The following part will discuss the advantages and the disadvantages of the divide-multiply and the multiply-divide implementation based on the simulated results.

Table A.1: Summary of the simulated results of divide-multiply and the multiply-divide implementation.

Concept	divide-multiply	multiply-divide
Technology	CMOS 0.13μm	CMOS 0.13μm
Supply voltage	1V	1V
Current consumption	150.9mA	180mA
Output frequency	9.72-10.47GHz	0.93-1.05GHz
Reference frequency	20MHz	20MHz
In-band phase noise @ Offset	-55dBc/Hz 20kHz	-109dBc/Hz 20kHz
Out-of-band phase noise @ Offset	-130dBc/Hz 10MHz	-135dBc/Hz 10MHz
Resolution	500kHz	500kHz
Settling time	35μs	1.5μs
Loop bandwidth	30kHz	1MHz
Fractional spurs @ Offset	-94.1dBc 36MHz	-77.1dBc 500kHz
Reference spurs @ Offset	-98.8dBc 40MHz	-117.8dBc 20MHz

Table A.1 summarizes the simulated results of the divide-multiply and the multiply-divide implementation. Implemented in a 0.13μm CMOS technology, the both implementation operated from 1V supply and were illustrated for two different frequencies. The main reason is the disadvantage of the multiply-divide implementation regarding the frequency of operation for a specific frequency resolution. The divide-multiply implementation, regarding the upper frequency of operation on the other hand, would be limited only by the corner frequency of the implemented process.

The current consumed by the core frequency synthesizer is 150.9mA and 180mA

for the divide-multiply and the multiply-divide implementation from 1V supply, respectively. The most power dissipation of the proposed concepts is due to the frequency dividers. The frequency dividers were designed to operate at the highest frequency from a 1V supply and the power dissipation was not optimized. It is estimated that over 70% improvements could be achieved if the currents of the 2/3 divider cells of the implemented frequency dividers [25] is scaled with the frequency as discussed in [26]. Therefore, it is expected that an optimized (regarding the current consumption) divide-multiply implementation would draw below 50mA of current from 1V supply at 10GHz of operation. If, however, a different type of a frequency divider is used, not necessarily a multi-modulus divider, then further optimization of the power dissipation is possible.

The simulated in-band phase noise of the divide-multiply implementation, for the particular example discussed in this paper, is -55dBc/Hz at 20kHz offset from 10GHz carrier, while the simulated in-band phase noise of the multiply-divide implementation is -109dBc/Hz at 20kHz offset from 1GHz carrier. Because the phase noise of the multiply-divide implementation is determined by the phase noise characteristic of the phase frequency detector and the charge pump within the main system, then it would be relatively easy to estimate that, if the multiply-divide implementation is designed for 10GHz operation then the in-band phase noise simulated at 20kHz would be -90dBc/Hz. Consequently, regarding the in-band phase noise the multiply-divide implementation performs better compared to the divide-multiply implementation.

The simulated out-of-band phase noise of the divide-multiply implementation is -130dBc/Hz at 10MHz offset from 10GHz carrier and loop bandwidth of 30kHz, while the simulated out-of-band phase noise of the multiply-divide implementation is -135dBc/Hz at 10MHz offset from 1GHz carrier and 1MHz loop bandwidth. If the loop bandwidth of the multiply-divide implementation is reduced to 30kHz then the phase noise at 10MHz offset would be -149dBc/Hz. It is clear that an additional advantage

of the multiply-divide implementation over the divide-multiply implementation is the out-of-band phase noise performance.

The divide-multiply implementation accomplished a simulated switching speed of $35\mu s$. The simulated switching speed of the multiply-divide implementation is $1.5\mu s$. In order to prevent the stability concern, the main disadvantage of the divide-multiply implementation is the small loop bandwidth. The multiply-divide implementation, on the other hand, allows implementation of a high loop bandwidth resulting in 20-times faster switching speed. The loop bandwidth is additional advantage of the multiply-divide implementation over the divide-multiply implementation.

The divide-multiply implementation reduced the fractional and the reference spurs such that they are noticed far a way from the carrier at 36MHz and 40MHz offset, respectively. The multiply-divide implementation sees a fractional spur at 500kHz offset practically due to the high loop bandwidth. Thus, in order to improve the fractional spurs, the loop bandwidth of the multiply-divide implementation should be decreased. The spurious performance is assumed an advantage for the both implementations.

In conclusion regarding the comparison between the divide-multiply and the multiply implementation, the limiting factor when selecting the multiply-divide implementation is the frequency of operation for a specific frequency resolution. If a CMOS $0.13\mu m$ is considered, then the multiply-divide implementation is practical for applications only up to 1GHz if the channel spacing is 500kHz. If the channel spacing is required to be smaller then the multiply-divide implementation will be practical for (low) MHz range applications. The finer channel spacing will cause the multiply-divide implementation not to be practical (not even possible) for GHz range. The limitations of the divide-multiply implementation, on the other hand, are the small loop bandwidth (thus the switching speed), and the poor phase noise performance. The divide-multiply implementation does not limit the application frequency. However, regarding the phase noise performance, the divide-multiply implementation is

not practical for 10GHz applications. In addition, the limited loop bandwidth limits the practicality of the divide-multiply implementation as well.

A.3 Comparison to State-of-the-Art

The simulated results of the multiply-divide implementation are compared to the results of $\Delta\Sigma$ based fractional frequency synthesizer found in the literature [63–68] and the results are summarized in Table A.2.

Table A.2: Comparison of the simulated results (multiply-divide implementation) with the $\Delta\Sigma$ fractional frequency synthesizers prevalent in the literature.

Reference	[63]	[64]	[65]	[66]	[67]	[68]	This work simulated
Technology	CMOS 0.18μm	CMOS 0.25μm	CMOS 0.5μm	CMOS 0.5μm	CMOS 0.13μm	CMOS 0.18μm	CMOS 0.13μm
Supply	2.8V	2.5V	2.5V	3.3V	1.5V	3.3V	1V
Current	10mA Core	NG	10.8mA Core	6.7mA Core	28mA Core	NG	180mA Core
Frequency	0.9-1.73 GHz	0.9-0.95 GHz	0.9-1.13 GHz	0.86-0.95 GHz	1.44-1.94 GHz	1-1.1 GHz	0.93-1.05 GHz
Reference	13MHz	50MHz	6.4MHz	27.8MHz	26MHz	13MHz	20MHz
In-band PN @ Offset	-84 dBc/Hz 10kHz	-78 dBc/Hz 100kHz	-90 dBc/Hz 10kHz	-110 dBc/Hz 100kHz	-90 dBc/Hz 10kHz	-106 dBc/Hz 100kHz	-107 dBc/Hz 10kHz
Out-of-band PN @ Offset	-136 dBc/Hz 1.25MHz	NG	-135 dBc/Hz 10MHz	NG	-145 dBc/Hz 10MHz	-135 dBc/Hz 1MHz	-135 dBc/Hz 10MHz
Resolution	6MHz	10kHz	200kHz	200kHz	0.39Hz	0.72Hz	500kHz
Lock-in	300μs	800μs	NG	500μs	43μs	160μs	1.5μs
Bandwidth	16kHz	NG	15kHz	NG	30kHz	14kHz	1MHz
Frac-spurs @ Offset	NG	-52dBc 160kHz	-85dBc 200kHz	NG	NG	NG	-77.1dBc 500kHz
Ref-spurs @ Offset	-77dBc 13MHz	NG	-95dBc 7.994MHz	NG	-93.9dBc 26MHz	NG	-117.8dBc 20MHz
Chip area	1.43mm^2 Core	NG	11mm^2 Chip	2.25mm^2 Chip	2.1mm^2 Chip	NG	2.5mm^2 Chip

The multiply-divide implementation is not optimized regarding the power dissipation. Nevertheless, even the best optimization to be used, the multiply-divide implementation would be current starving compared to the other work. The reason is that the frequency multiplier operates at 40GHz, frequency that is much higher than the operation frequency of the frequency synthesizer. The 40GHz operation requires more current compared to the 1GHz operation.

The in-band and the out-of-band phase noise performance of the multiply-divide implementation shows superior characteristic compared to the cited work. The in-band phase noise at 10kHz offset from 1GHz carrier is -107dBc/Hz. This is at least 17dB improvements of the in-band phase noise of this work compared to cited work. Regarding the out-of-band phase noise, the very first loop indicates that the [67] promises the best performance. However, the loop bandwidth of the [67] is 30kHz. If the loop bandwidth of the multiply-divide implementation is decreased from 1MHz down to 30kHz then the out-of-band phase noise characteristic at 10MHz offset would be -150dBc/Hz. This is 5dB improvements compared to [67].

The fine resolution is a strong advantage of the $\Sigma\Delta$ based frequency synthesizers. Their frequency resolution is determined by the number of programming bits of the $\Sigma\Delta$ modulator and the reference frequency. For example, [67] uses a 12-bit $\Sigma\Delta$ modulator and 26MHz reference signal. Equivalently that means that [67] can accomplish a frequency resolution of 0.39Hz.

The switching speed of 1.5μs and the loop bandwidth of 1MHz[2] are strong advantage of the multiply-divide implementation compared to the cited work.

Another advantage of the multiply-divide implementation is the spurious performance. With 1MHz loop bandwidth, a spur is noticed at 500kHz offset from the 1GHz carrier. The -77dBc power of that spur could be significantly reduced if the

[2]The simulated results determined that the loop bandwidth of the multiply-divide implementation could be increased as high as 2MHz when 20MHz reference signal is used. It is clear that a higher reference signal would also allow a higher loop bandwidth.

loop bandwidth is decreased. For example, [65] reports a fractional spur at 200kHz offset with -85dBc power and loop bandwidth of 15kHz. If the loop bandwidth of the multiply-divide implementation is reduced to 15kHz then the power of the spur at 500kHz offset would be reduced below -100dBc.

In conclusion, the advantages of the multiply-divide implementation are the phase noise performance, the switching speed, the high loop bandwidth, and the spurious performance. GSM standard requires that all spurious be 80dB below the carrier [69]. It was illustrated that 1MHz loop bandwidth of the multiply-divide implementation resulted in a spur at 500kHz offset that was 77.1dB below the carrier. The 3dB margin to 80dB could be accomplished by lowering the loop bandwidth. For illustration purposes only, the phase noise of the multiply-divide implementation can be compared to the numerical specification of the phase noise for the GSM standard. Table A.3 shows an attractive phase noise characteristic of the illustrated multiply-divide implementations with a 30kHz loop bandwidth.

Table A.3: Numerical specification of phase noise for GSM Standard.

Offset frequency	GSM specification	multiply-divide
100kHz	-52.3dBc/Hz	-105.4dBc/Hz
200kHz	-82.8dBc/Hz	-112.5dBc/Hz
250kHz	-85.8dBc/Hz	-115dBc/Hz
400kHz	-112.8dBc/Hz	-119dBc/Hz
600kHz	-112.8dBc/Hz	-123dBc/Hz
1.8MHz	-121dBc/Hz	-133dBc/Hz
3MHz	-123dBc/Hz	-138dBc/Hz
6MHz	-129dBc/Hz	-144dBc/Hz
10MHz	-150dBc/Hz	-150dBc/Hz

The disadvantage of the multiply-divide implementation is the frequency of operation for a given frequency resolution, the power dissipation, and the frequency resolution compared to the work from the literature.

A.4 Summary

The multiply-divide implementation of the proposed frequency synthesizer was discussed in this chapter. This implementation was illustrated for 1GHz frequency of operation with 500kHz frequency resolution. Utilizing two voltage controlled oscillators, the simulated results of the 1GHz and the 40GHz VCO were shown and discussed. Some simulated results of the frequency synthesizer, such as the switching speed, the DFT, and the phase noise, were discussed as well. The simulated results of the multiply-divide implementation were compared to the simulated results of the divide-multiply implementation and the experimental results of the $\Sigma\Delta$ based frequency synthesizers found in the literature. In addition, the phase noise characteristic of the multiply-divide implementation was compared to the numerical specification for the phase noise of the GSM standard. The advantages of the multiply-divide implementation are the phase noise performance (both the in-band and the out-of-band phase noise), the size of the loop filter, the switching speed, and the spurious performance. The main disadvantage of the multiply-divide implementation is the maximum frequency of operation for a given channel spacing.

Appendix B

Divide-Multiply-Divide Implementation

When the simulated results of the divide-multiply and the multiply implementation were compared, the conclusion was that the limiting factor for the multiply-divide implementation is the maximum frequency of operation for a specific frequency resolution. The limitations of the divide-multiply implementation, on the other hand, were the small loop bandwidth, and the poor phase noise performance. Consequently, if a low phase noise frequency synthesizer operating at high (GHz) frequency is targeted with a fine frequency resolution, then a different implementation of the proposed architectural concept should be considered. This chapter will illustrate the divide-multiply-divide implementation of the proposed system for a frequency synthesizer.

B.1 An Example Case for 10GHz of operation

Figure B.1 shows the block diagram of the divide-multiply-divide implementation. The feedback loop of the frequency synthesizer consists of a cascade connection of two frequency dividers and a frequency multiplier. The output signal from the VCO first is brought to a frequency divider. The output signal from the frequency divider is brought to the frequency multiplier, which output signal goes to another frequency divider before is brought to the input of the phase frequency detector.

To illustrate this implementation, the proposed fractional frequency synthesizer

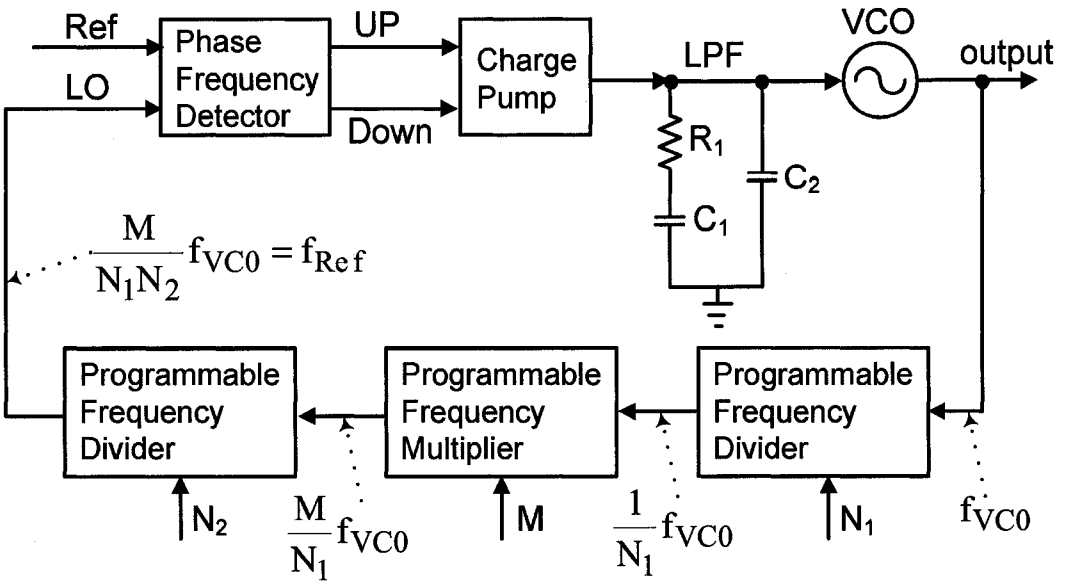


Figure B.1: A block diagram of the divide-multiply-divide implementation.

Table B.1: Selecting the programming numbers for the frequency dividers and the frequency multiplier.

f_{VCO}	N_1	$\frac{1}{N_1}f_{VCO}$	M	$\frac{M}{N_1}f_{VCO}$	N_2	$\frac{M}{N_1N_2}f_{VCO}$	f_{Ref}	$\frac{M}{N_1N_2}$
10.0000GHz	200	50.0MHz	40	2.00GHz	100	20.00MHz	20MHz	500.000
10.0005GHz	177	56.5MHz	40	2.26GHz	113	20.00MHz	20MHz	500.025
10.0010GHz	146	68.5MHz	40	2.74GHz	137	20.00MHz	20MHz	500.050
10.0015GHz	241	41.5MHz	40	1.66GHz	83	20.00MHz	20MHz	500.075
10.0020GHz	1667	6.00MHz	400	2.40GHz	120	20.00MHz	20MHz	500.100

was implemented in a 0.13μm CMOS technology. The frequency of the reference signal was chosen as 20MHz again, and the charge pump, the phase frequency detector, and the frequency divider used with the divide-multiply implementation were reused. Similarly to the divide-multiply implementation, a PLL type of the frequency multiplier was used for illustration purposes only. Consequently, the divide-multiply-divide implementation requires two voltage-controlled oscillators. Moreover, the 10GHz VCO of the divide-multiply implementation was reused as well. The channel spacing was also selected to be equal to the divide-multiply implementation (500kHz). The divide-multiply-divide implementation was designed to operate from 1V power supply.

Compared to the divide-multiply and the multiply-divide implementation, the channel selection with the divide-multiply-divide implementation is more complicated and can be seen as a potential disadvantage of this implementation. To illustrate this issue, Table B.1 summarizes five examples of a possible programming ratios of the frequency dividers and the frequency multiplier. Assuming that the frequency synthesizer is in a lock condition, the numbers N_1, M, and N_2 are selected in order to illustrate their impact on the tuning rage of the VCO within the frequency multiplier, the type of the frequency dividers to be used, as well as the sizing of the loop bandwidth within the frequency multiplier.

For example, in order to generate a 10GHz signal and for illustration purposes only, Table B.1 illustrates that the first divider would be programmed to divide by 200, the frequency multiplier would be programmed to multiply by 40, and the second divider would be programmed to divide by 100. In this particular example, the frequency multiplier would generate a signal with a frequency of 2GHz.

The second example illustrates a possible programming in order to generate a signal spaced 500kHz away from the 10GHz (i.e. to generate the signal 10.0005GHz). To do so, the first divider would divide by 177, the frequency multiplier would multiply by 40, and the second divider would divide by 113. In this second example, the VCO within the frequency multiplier would generate a signal with a frequency of 2.26GHz.

The third example illustrates that a 10.0010GHz signal can be generated by programming the frequency divider to divide by 146, the frequency multiplier to multiply by 40, and the second frequency divider to divide by 137. In this particular example, the VCO within the frequency multiplier would generate a signal with a frequency of 2.74GHz.

The forth example illustrates the generation of the 10.0015GHz signal. According to the Table B.1, one possible solution is if the first frequency divider is programmed to divide by 241, and the second frequency divider is programmed to divide by 83.

With the constant multiplication ratio of 40, the frequency multiplier will generate a signal with a frequency of 1.66GHz.

Finally, the last example illustrates a possible programming of the frequency dividers and the frequency multiplier in order to generate the 10.0020GHz signal. If the multiplication ratio M is kept 40, then the one of the frequency dividers should be programmed to divide by 12 and the second frequency divider to divide by 1667. The case the first divider to divide by 12 is excluded because the frequency multiplier in that case should generate a 33.34GHz signal. Therefore, the first divider is programmed to divide by 1667, and the second divider is programmed to divide by 12. With a multiplication ratio of 40, in this particular case, the frequency multiplier will generate a signal with frequency of 240MHz.

From the aforementioned five examples it can be concluded that the VCO within the frequency multiplier should be able to cover the frequency range between 240MHz and 2.74GHz. A ring type of an oscillator, as explained in [48], can be a possible solution for a VCO design. However, that type of a VCO will have a poor phase noise performance compared to an LC type of a VCO. Therefore, in order to reduce the tuning range of the VCO within the frequency multiplier, is was decided to set the multiplication ratio of the frequency multiplier to 400, and to program the second frequency divider to divide by 120. In this case, the frequency multiplier will generate a 2.4GHz signal. Thus, the required tuning range of the VCO within the frequency multiplier will be reduced between 1.66GHz and 2.74GHz.

Based on the Table B.1 it can be concluded that multi-modulus frequency dividers probably are not solution of choice for the divide-multiply-divide implementation. For example, the first frequency divider, according to the illustrated five examples, should be able to cover the range between 146 and 1667. A 7-bit frequency divider, based on a divide by 2/3 divider cell, can cover the range between 128 and 255. This would be enough for the first four examples but not for the fifth example. To cover the fifth

example, a 10-bit frequency divider would be necessary. A possible solution would be to use a switching circuit that would enable or disable additional stages of a multi-modulus frequency divider. The direct trade-off would be the increase of the chip area and the power dissipation.

Similar discussions are for the second frequency divider, which should cover the range between 83 and 137, and for the frequency multiplier, which should cover the range between 40 and 400.

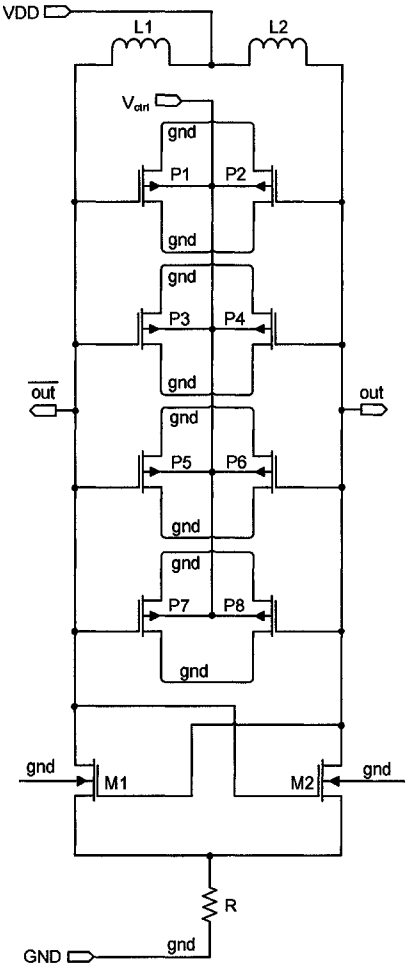


Figure B.2: The schematic of the VCO within the frequency multiplier.

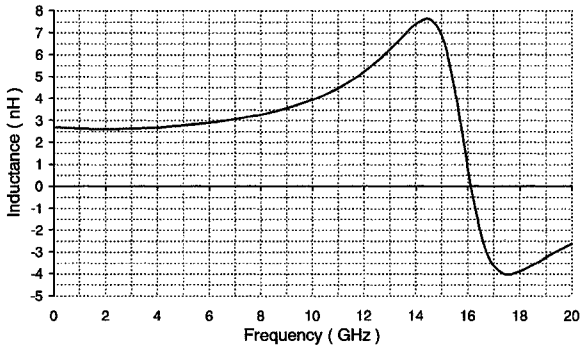
Finally, another conclusion that can be made based on the Table B.1 is the sizing of the loop bandwidth of the frequency multiplier. It can be seen that in four out of five cases the input frequency of the frequency multiplier is higher than the

reference frequency. However, the last example shows that a 6MHz signal goes to the frequency multiplier. A rule of thumb is saying that the maximum loop bandwidth of the frequency multiplier should be 600kHz. To reduce the ripples on the controlled voltage, the loop bandwidth should be reduced further. The simulated results presented further in the text are performed with 500kHz loop bandwidth.

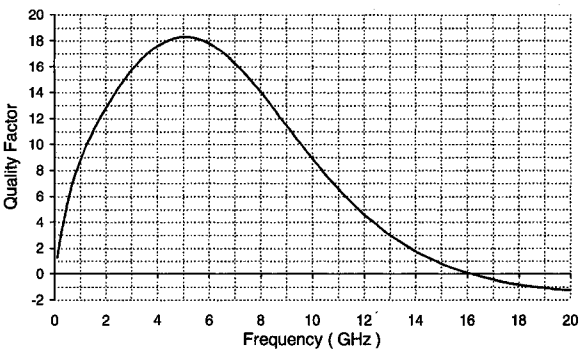
Figure B.2 shows the schematic of the VCO within the frequency multiplier. The VCO within the main system is identical to the VCO described in Chapter 6.

The VCO, within the frequency multiplier, utilized two inductors sized with outer dimension of 200μm, metal width of 10μm, spacing between the metal paths of 5μm (default), and number of turns set to 4. Figure B.3(a) shows the simulated inductance of the aforementioned inductor for the frequencies between 100MHz and 20GHz. The simulated inductance at 2GHz is 2.61nH. The simulated quality factor at 2GHz is 12.8 as shown in Figure B.3(b).

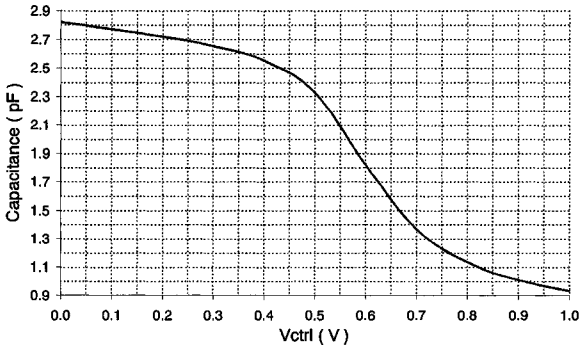
In order to cover the frequency range between 1.66GHz and 2.74GHz, eight p-channel transistors were used to create variable capacitance. Each p-channel transistor was sized with 400nm in length and 250μm in width (25 x 10μm). The gates of four transistors were connected to one oscillation node of the VCO. The gates of the other four transistors were connected to the second oscillation node of the VCO. The bulk of all transistors was shortened and connected to the voltage controlled signal (V_{ctrl}). The source and the drain of all transistors were shortened to ground such that the p-channel transistors were biased to operate in depletion region. The simulated capacitance of the varactor, seen from the oscillation node of the VCO, as a function of the V_{ctrl} signal at 2GHz of operation is shown in Figure B.3(c). The capacitance of the varactor can be tuned between 0.93pF and 2.82pF. The 3:1 tuning capacitance of the varactor resulted in the tuning range of the VCO shown in Figure B.3(d). The VCO, within the frequency multiplier, can be tuned between 1.62GHz and 2.88GHz. Figure B.3(e) shows the peak-to-peak voltage of the output signal of the VCO over



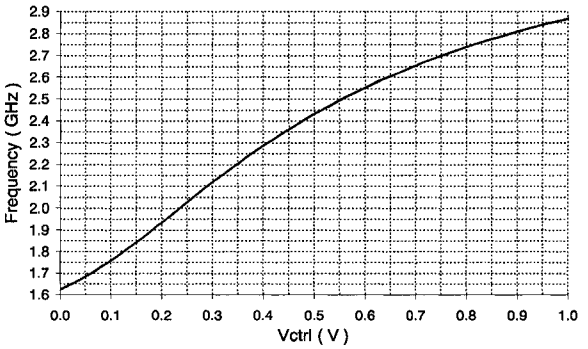
(a) Inductance of the inductor



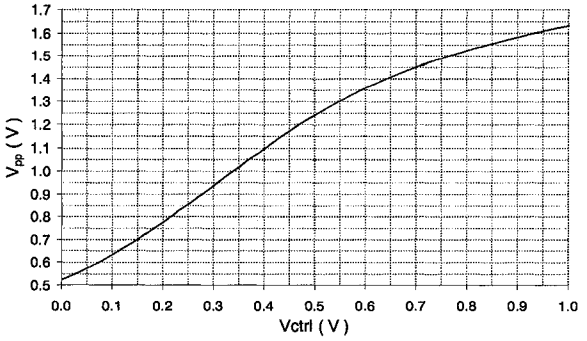
(b) Quality factor of the inductor



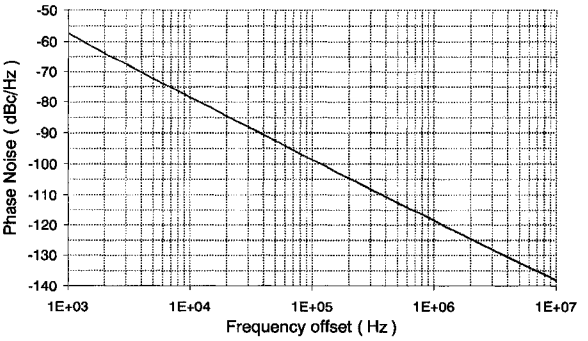
(c) Capacitance of the varactor



(d) Tuning range of the VCO



(e) Peak-to-peak voltage



(f) Phase noise of the 2GHz signal

Figure B.3: Simulated performances of the VCO within the frequency multiplier.

the tuning range of the VCO.

The SpectreRF simulator (the pss and the pnoise analysis) implementing the bsim 4 models was used to simulate the phase noise of the VCO. The simulated phase noise of the VCO operating at 2GHz is shown in Figure B.3(f). The simulated phase noise at 100kHz offset is -98.7dBc/Hz, at 1MHz offset is -118.8dBc/Hz, and at 10MHz offset is -138.2dBc/Hz. Operating from 1V supply, the VCO dissipated 712μW resulting in FOM of -186.3dB at 1MHz offset.

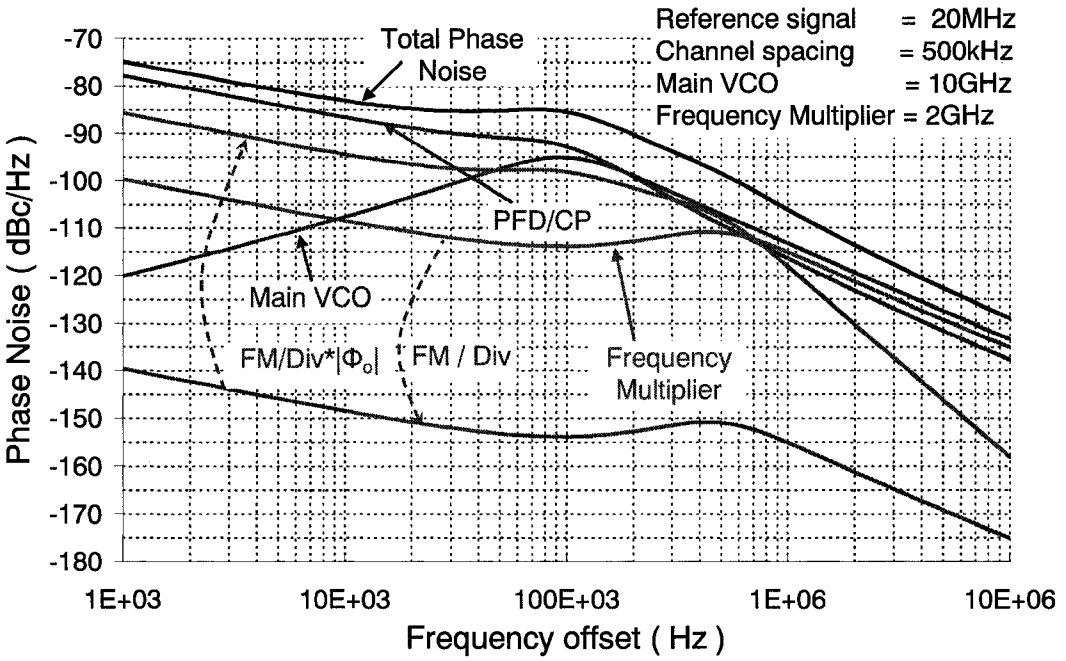


Figure B.4: Phase noise characteristic of the divide-multiply-divide implementation at 10GHz.

Figure B.4 shows the estimated total phase noise of the 10GHz signal from the divide-multiply-divide implementation that implements a PLL type of a frequency multiplier. The phase noise due to the frequency multiplier is first lowered through the frequency divider by 20log100 and then multiplied by the appropriate transfer function of the loop system. The loop filter within the frequency multiplier was sized with a damping factor of 0.707 and a natural frequency of 500kHz. The loop filter of

the main system was sized with a damping factor of 0.707 and a natural frequency of 100kHz. It can be concluded from the Figure B.4 that a 100kHz natural frequency of the main loop filter is the optimum sizing for the illustrated divide-multiply-divide implementation regarding the phase noise. If the natural frequency is increased then the phase noise due to the VCO would be suppressed, however, more phase noise would be allowed due to the phase frequency detector and the charge pump. On the other hand, if the natural frequency is reduced, then the phase noise due to the phase frequency detector and the charge pump would be suppressed, however, the phase noise due to the VCO would become dominant.

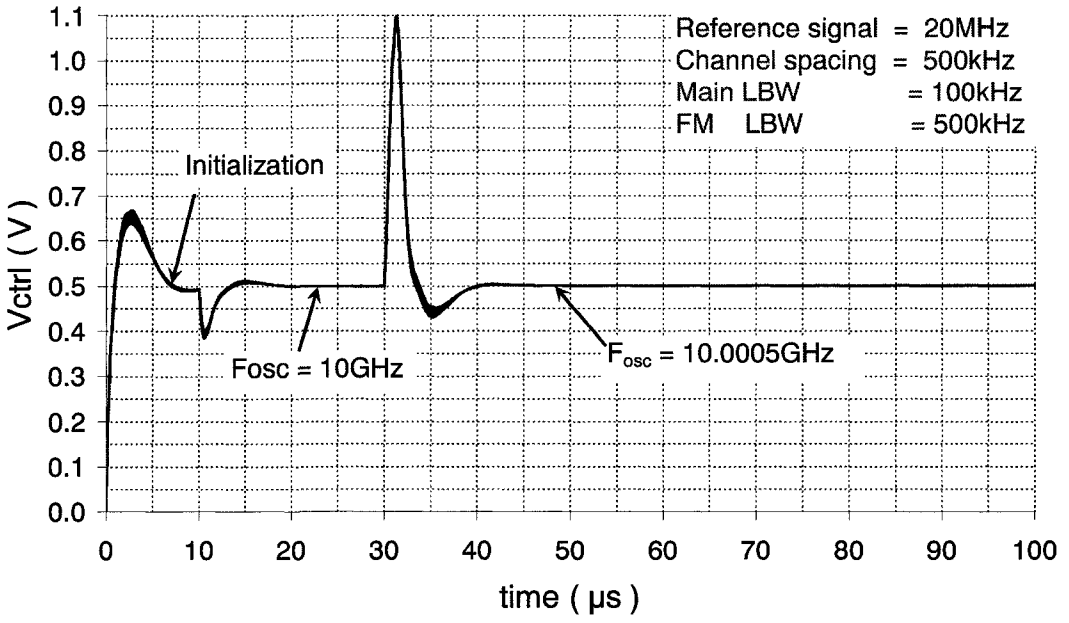


Figure B.5: Switching time for the divide-multiply-divide implementation.

Figure B.5 illustrates the switching speed of the divide-multiply-divide implementation. The loop bandwidth of the frequency multiplier was sized with a damping factor of 0.707 and a natural frequency of 500kHz. The loop bandwidth of the main system was sized with a damping factor of 0.707 and a natural frequency of 100kHz. The proposed system was first locked at 10GHz. The control signal for the VCO was

settled at 500mV. At 30μs the system was forced to lock at the neighboring channel separated by 500kHz. After approximately 17μs ($\pm$0.006% of the final control voltage value) the control signal for the VCO settled at its new value and the fractional frequency synthesizer generated a signal with a frequency of 10.0005GHz. Through the simulated results was found that if the natural frequency of the frequency multiplier is increased to 1MHz and the natural frequency of the main system is increased to 500kHz, while keeping the damping constant of 0.707, the switching time of the frequency synthesizer will be about 8μs.

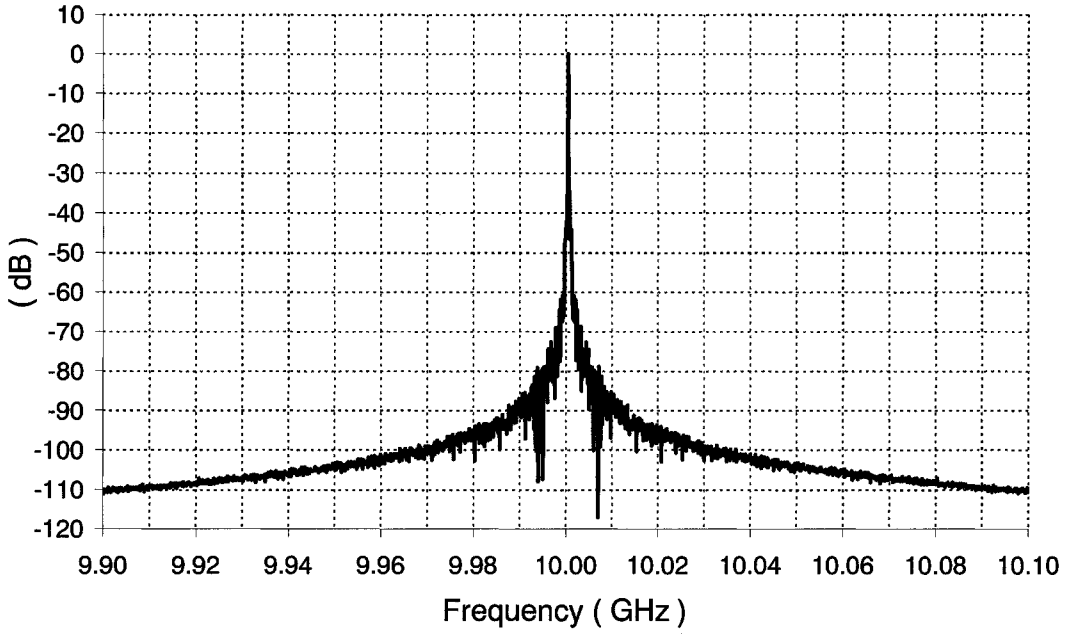


Figure B.6: DFT of the output signal from the divide-multiply-divide implementation.

Figure B.6 shows the DFT of the generated 10.0005GHz signal from the frequency synthesizer. A close look of the plot indicated a spur at 500kHz offset and 62dB bellow the carrier.

B.1.1 Comparison to Divide-Multiply Implementation

The following part will compare the divide-multiply and the divide-multiply-divide implementation based on the simulated results.

Table B.2: Summary of the simulated results of the divide-multiply and the divide-multiply-divide implementation.

Concept	divide-multiply	divide-multiply-divide
Technology	CMOS 0.13μm	CMOS 0.13μm
Supply voltage	1V	1V
Current consumption	150.9mA	130mA
Output frequency	9.72-10.47GHz	9.72-10.47GHz
Reference frequency	20MHz	20MHz
In-band phase noise @ Offset	-55dBc/Hz 20kHz	-85dBc/Hz 20kHz
Out-of-band phase noise @ Offset	-130dBc/Hz 10MHz	-130dBc/Hz 10MHz
Resolution	500kHz	500kHz
Settling time	35μs	17μs
Loop bandwidth	30kHz	100kHz
Fractional spurs @ Offset	-94.1dBc 36MHz	-62dBc 500kHz
Reference spurs @ Offset	-98.8dBc 40MHz	-101.2dBc 40MHz

Table B.2 summarizes the simulated results of the divide-multiply and the divide-multiply-divide implementation. In order to do a better comparison, the both implementations are illustrated at the same frequency of operation with a same channel spacing. In addition, the building blocks of the divide-multiply implementation (CP, PFD, frequency divider, and 10GHz VCO) were reused by the divide-multiply-divide implementation.

The main contributors to the power dissipation of the divide-multiply implementation were the frequency dividers. The frequency dividers were based on a multi-modulus 2/3 divider cell optimized to operate at 10GHz. Assuming that the

divide-multiply-divide implementation is using a similar architecture of the frequency dividers, then in order to switch between 10GHz and 10.0005GHz the divide-multiply-divide implementation will need three frequency dividers: a 7-bit (the first frequency divider), a 5-bit (within the frequency multiplier) and a 6-bit (the second frequency divider) frequency divider. In this particular case, the divide-multiply-divide implementation will use about 20mA less current compared to the divide-multiply implementation. However, in order to switch to 10.0020GHz, the first divider would use 10-bits divider, the frequency multiplier would incorporate an 8-bit frequency divider, and the second frequency divider would use 6-bits divider. In that case, the DC current consumption of the divide-multiply-divide implementation would increase to 170mA (or 20mA more current). In addition, more current would by used assuming that the divide-multiply-divide implementation would incorporate additional switching circuit to turn on or off additional stages of the frequency dividers. In conclusion regarding the power dissipation, it is expected that the divide-multiply-divide implementation will need about 10-20% more current compared to the divide-multiply implementation.

Comparing the in-band phase noise of the divide-multiply and the divide-multiply-divide implementation, the divide-multiply-divide implementation improved the in-band phase noise performance of the frequency synthesizer by 30dB. The improved phase noise makes the divide-multiply-divide implementation of the proposed system attractive for some practical applications. This will be further discussed in the following section.

Additional advantages of the divide-multiply-divide implementation are the size of the loop bandwidth and thus the switching speed. While the 30kHz is probably the maximum loop bandwidth of the illustrated divide-multiply implementation, the loop bandwidth of the divide-multiply-divide implementation in most of the cases is

determined by the frequency of the reference signal. It was found through the simulated results that the loop bandwidth of the divide-multiply-divide implementation could be increased up to 1MHz for the cases when the input signal of the frequency multiplier operates at higher frequency compared to the reference signal. Nevertheless, tt was already discussed that 100kHz is an optimum loop bandwidth regarding the phase noise performance of the divide-multiply-divide implementation. This loop bandwidth resulted in a switching time of about 17μs. That is twice faster compared to the divide-multiply implementation.

Finally, regarding the fractional and the reference spurs the both implementations gives attractive results. The narrow loop bandwidth of the divide-multiply implementation moved the fractional spurs far away from the carrier. The 100kHz loop bandwidth of the divide-multiply-divide implementation resulted in a spur at 500kHz offset. However, this spur appears 62dB bellow the carrier. If the loop bandwidth is reduced to 30kHz then this spur would be reduced as well. Regarding the reference spurs, and taking into the account the size of the loop bandwidth, the divide-multiply-divide implementation has small advantages over the divide-multiply implementation.

B.1.2 Comparison to Fractional-N Frequency Synthesizers

The following part will compare the simulated results of the divide-multiply-divide implementation to the results of $\Delta\Sigma$ based fractional frequency synthesizer found in [5, 55, 56].

Implementing the same blocks (PFD, CP, 10GHz VCO, and divider cells) as the divide-multiply implementation, Table B.3 shows that the phase noise performance of the divide-multiply-divide implementation is attractive for practical applications. Additional advantages of the divide-multiply-divide implementation are the loop bandwidth and the spurious performance. Depending of the reference frequency as well

Table B.3: Comparison of the experimental results with the $\Delta\Sigma$ fractional frequency synthesizers prevalent in the literature.

Reference	[5]	[55]	[56]	divide-multiply-divide
Technology	CMOS 0.13μm	CMOS 0.09μm	CMOS 0.13μm	CMOS 0.13μm
Supply	1.5V	1.4V	1.2V	1V
Current Consumption	63mA Core	3mA Core	33.3mA Core	130mA Core
Frequency	10.24-12.55 GHz	7.6-8.4 GHz	4.6-5.4 GHz	9.72-10.47 GHz
Reference	64MHz	NG	40MHz	20MHz
In-band Phase Noise @ Offset	-80 dBc/Hz 10kHz	-100 dBc/Hz 10kHz	-70 dBc/Hz 10kHz	-83 dBc/Hz 10kHz
Out-of-band Phase Noise @ Offset	-104 dBc/Hz 10MHz	-117 dBc/Hz 10MHz	-134 dBc/Hz 10MHz	-130 dBc/Hz 10MHz
Resolution	18kHz	10kHz	125kHz	500kHz
Settling time	8μs	20μs	10μs	17μs
Bandwidth	1MHz	500kHz	800kHz	100kHz
Frac-spurs @ Offset	-44dBc 18kHz	-60dBc 10kHz	NG	-62dBc 500kHz
Ref-spurs @ Offset	-52dBc NG	below noise floor	NG	-101.2dBc 40MHz

as the channel selection, the divide-multiply-divide implementation can allow loop bandwidth comparable to the cited work in Table B.3. The simulated switching time of the divide-multiply-divide implementation with 100kHz is 17μs. However, if the loop bandwidth is increased to 500kHz, then the switching speed will be 8μs and will be comparable to [5]. Nevertheless, while [5] reported strong appearance of in-band fractional spurs, the simulated results of the divide-multiply-divide implementation indicated attractive spurious performance.

B.2 An Example Case for 1GHz of operation

The upper frequency of operation for a given channel spacing was found to be disadvantage of the multiply-divide implementation. The multiply-divide implementation was illustrated in a 0.13μm CMOS technology for 1GHz of operation with 500kHz channel spacing. Due to the placement of the frequency multiplier, the VCO within the frequency multiplier generated frequencies around 40GHz. If the channel spacing is reduced, for example to 200kHz, the multiply-divide implementation would not be practically possible in 0.13μm CMOS technology.

In order to overcome the disadvantage of the multiply-divide implementation, the proposed system for a frequency synthesizer could use the divide-multiply-divide implementation.

Table B.4: Selecting the programming numbers for the frequency dividers and the frequency multiplier.

f_{VCO}	N_1	$\frac{1}{N_1} f_{VCO}$	M	$\frac{M}{N_1} f_{VCO}$	N_2	$\frac{M}{N_1 N_2} f_{VCO}$	f_{Ref}	$\frac{M}{N_1 N_2}$
1.0000GHz	25	40.0MHz	40	1.60GHz	80	20.00MHz	20MHz	50.000
1.0005GHz	29	34.5MHz	40	1.38GHz	69	20.00MHz	20MHz	50.025
1.0010GHz	26	38.5MHz	40	1.54GHz	77	20.00MHz	20MHz	50.050
1.0015GHz	1	1.0015GHz	40	40.06GHz	2003	20.00MHz	20MHz	50.075
1.0020GHz	12	83.5MHz	40	3.34GHz	167	20.00MHz	20MHz	50.100

Table B.4 illustrates a possible programming of the frequency dividers and the frequency multiplier of the divide-multiply-divide implementation. In order not to limit the loop bandwidth of the frequency multiplier, and thus the overall system, the smaller number between N_1 and N_2 is allocated to the first frequency divider.

The disadvantage related with the programming of the divide-multiply-divide implementation, as it was discussed in Section B.1, is visible for the 1GHz operation with 500kHz spacing as well. The five examples listed in Table B.4 show that the VCO within the frequency multiplier should be able to tune between 1.38GHz and

40.06GHz in order to cover all five illustrated examples. The cause of a wide frequency range, and thus the difficulty of programming the two frequency dividers, is more pronounced if the product of N_1 and N_2 is a prime number. For example, in order to generate the 1.0015GHz signal, the product of N_1 and N_2 should be 2003, which is a prime number. So, the first divider (N_1) of the divide-multiply-divide implementation should be programmed to "divide" by 1, while the second divider to divide by 2003. Similarly, 167 is a prime number causing the required frequency of the VCO within the frequency multiplier to be twice as higher as the required frequency of the first three examples.

Table B.5: Selecting the programming numbers for the frequency dividers and the frequency multiplier of the divide-multiply-divide implementation.

$fvco$	N_1	$\frac{1}{N_1}fvco$	M	$\frac{M}{N_1}fvco$	N_2	$\frac{M}{N_1N_2}fvco$	f_{Ref}	$\frac{M}{N_1N_2}$
1.0000GHz	25	40.0MHz	40	1.60GHz	80	20.00MHz	20MHz	50.000
1.0005GHz	29	34.5MHz	40	1.38GHz	69	20.00MHz	20MHz	50.025
1.0010GHz	26	38.5MHz	40	1.54GHz	77	20.00MHz	20MHz	50.050
1.0015385GHz	31	32.3MHz	39	1.26GHz	63	20.00MHz	20MHz	50.07692
1.0019512GHz	26	38.5MHz	41	1.58GHz	79	20.00MHz	20MHz	50.09756

A possible solution for the aforementioned disadvantage is to allow a flexible channel resolution. For example, Table B.5 shows that, instead of generating the signal 1.0015GHz, the frequency synthesizer to generate the signal 1.0015385GHz. With other words, the new signal will have a frequency which is 38.46kHz away from the accurate 1.0015GHz. Similarly, the fifth example shows how to generate the signal 1.0019512GHz when the required frequency is 1.002GHz (or -48.78kHz difference). The flexible channel resolution (500kHz $\pm$ 50kHz) will result in a tuning range between 1.26GHz and 1.6GHz (for the VCO within the frequency multiplier). In addition, the input signal of the frequency multiplier in all five examples has a frequency which is higher compared to the reference frequency. Consequently, a high

loop bandwidth can be used for the frequency multiplier as well as the overall system. Moreover, the flexible channel resolution resolve the problem of the need of multiple frequency dividers. A 4-bit frequency divider will be enough to cover the range of the first frequency divider of the divide-multiple-divide implementation (N_1 = 25 - 31). Similarly, a 5-bit frequency divider will be needed within the frequency multiplier, and a 6-bit frequency divider will be needed for the second frequency divider.

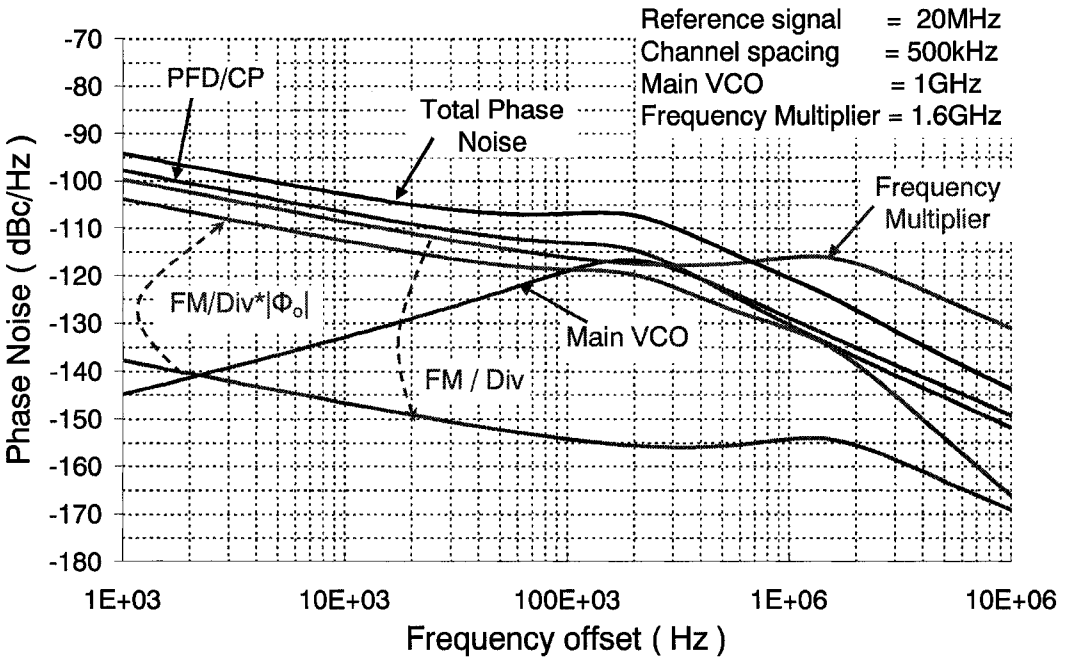


Figure B.7: Phase noise characteristic of the 1GHz divide-multiply-divide implementation.

Figure B.7 shows the estimated total phase noise of the 1GHz signal from the divide-multiply-divide implementation. The phase noise due to the frequency multiplier is first lowered through the frequency divider by 20log80 and then multiplied by the appropriate transfer function of the loop system. The loop filter within the frequency multiplier was sized with a damping factor of 0.707 and a natural frequency of 1.5MHz. The loop filter of the main system was sized with a damping factor of 0.707 and a natural frequency of 200kHz.

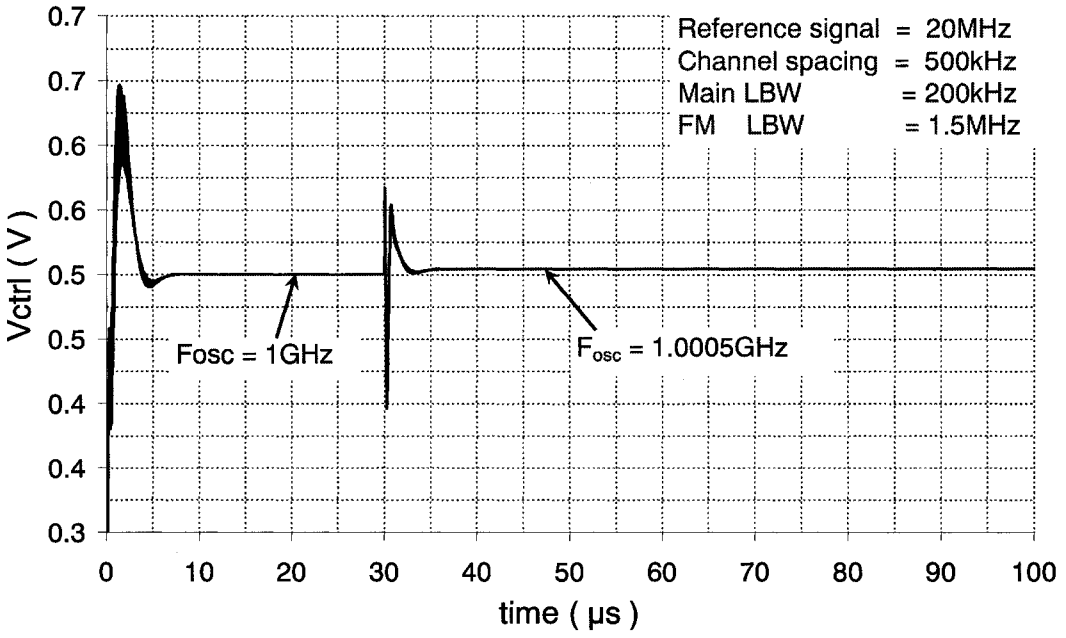


Figure B.8: Switching time for the divide-multiply-divide implementation.

The aforementioned loop bandwidth for the frequency multiplier and the main system was used to illustrate the switching speed of the 1GHz divide-multiply-divide implementation as shown in Figure B.8. The proposed system was first locked at 1GHz. The control signal for the VCO was settled at 500mV. At 30μs the system was forced to lock at the neighboring channel separated by 500kHz. After approximately 6μs the control signal for the VCO settled at its new value and the fractional frequency synthesizer generated a signal with a frequency of 1.0005GHz.

Figure B.9 shows the lock-in time of the divide-multiply-divide implementation when the natural frequency of the frequency multiplier was increased to 2MHz, and the natural frequency of the main system was increased to 1MHz. The damping constant for the frequency multiplier and the main system was kept at 0.707. The high loop bandwidth resulted in approximately 3.1μs switching time between the two neighboring channels 1GHz and 1.0005GHz.

Figure B.10 shows the DFT of the 1.0005GHz signal when the loop bandwidth of

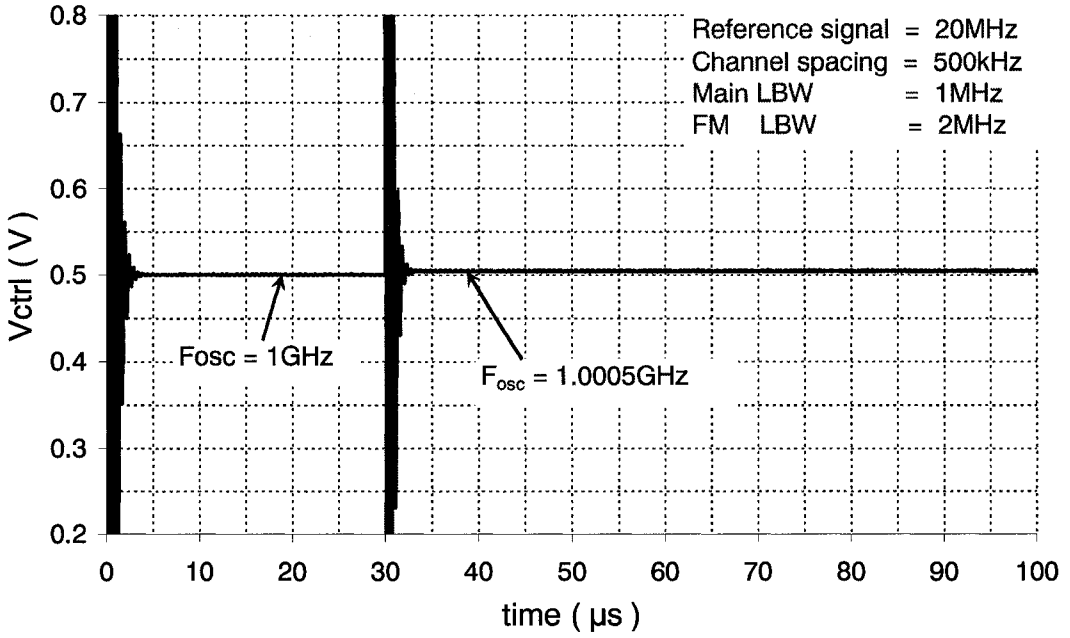


Figure B.9: Switching time for the divide-multiply-divide implementation for high loop bandwidth.

the frequency multiplier was size with 1.5MHz natural frequency and 0.707 damping constant, while the loop bandwidth of the main system was sized with 200kHz natural frequency and 0.707 damping constant. The spurs caused by the reference signal were found to be below the noise floor of the signal. However, a spur, 84.3dB below the carrier, was noticed at 3.5MHz offset. Through the simulated results was found that, if the natural frequencies of the frequency multiplier and the main system are increased to 2MHz and 1MHz, respectively, then the spurious tones would become a bigger problem. The high loop bandwidth will cause a spur, 55.3dB below the carrier, to appear at 2.5MHz offset from the carrier. In addition, a reference spur, 90dB below the carrier, will be noticed at 20MHz offset. Thus, a 200kHz natural frequency of the main system is an optimal size of the main loop bandwidth regarding the spurious tones, switching speed and the phase noise of the divide-multiply-divide implementation.

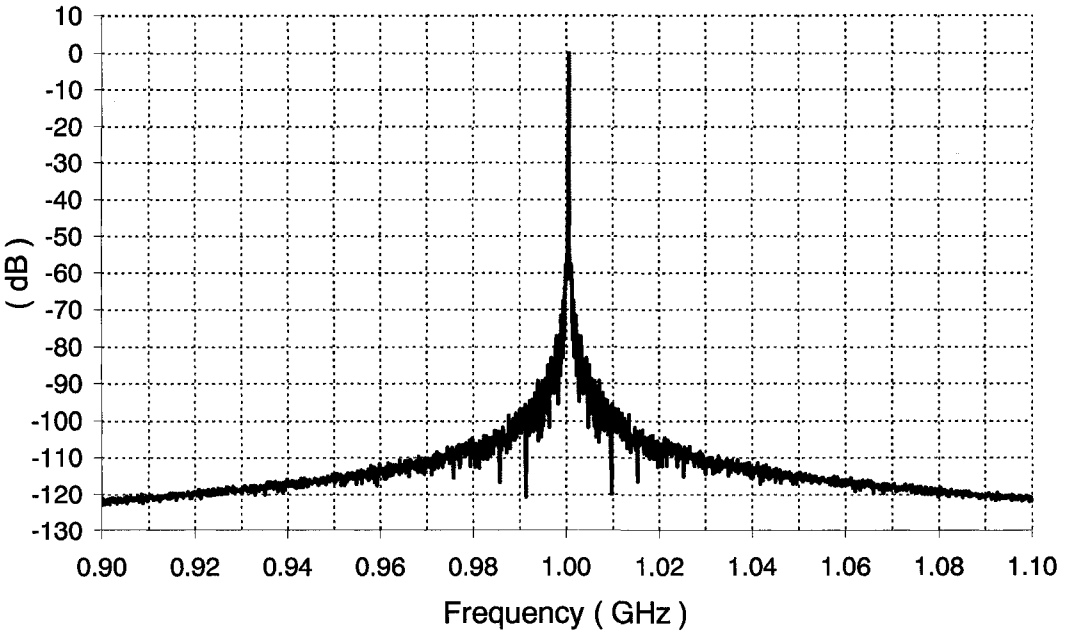


Figure B.10: DFT of the output signal from the divide-multiply-divide implementation.

B.2.1 Comparison to Multiply-Divide Implementation

The following part will compare the divide-multiply and the divide-multiply-divide implementation based on the simulated results.

Table B.6 shows the summary of the simulated results of the multiply-divide and the divide-multiply-divide implementation.

Although the both implementations are not optimized regarding the power dissipation, the estimated power dissipation of the divide-multiply-divide implementation is 70mW less compared to the multiply-divide implementation. This is expected since the frequency multiplier within the multiply-divide implementation operates at 40GHz, compared to the frequency multiplier of the divide-multiply-divide implementation which operates between 1.2-1.6GHz. In addition, a 40GHz signal is going to the frequency divider that follows the frequency multiplier within the multiply-divide implementation. It is understandable that the 40GHz operation requires more power

Table B.6: Summary of the simulated results of the two considered implementations.

Concept	multiply-divide	divide-multiply-divide	
Technology	CMOS 0.13μm	CMOS 0.13μm	
Supply voltage	1V	1V	
Current consumption	180mA	110mA	
Output frequency	0.93-1.05GHz	0.93-1.05GHz	
Reference frequency	20MHz	20MHz	
In-band phase noise @ Offset	-109dBc/Hz 20kHz	-105.7dBc/Hz 20kHz	-105.1dBc/Hz 20kHz
Out-of-band phase noise @ Offset	-135dBc/Hz 10MHz	-134.4dBc/Hz 10MHz	-143.8dBc/Hz 10MHz
Resolution	500kHz	500kHz$\pm$50kHz	
Settling time	1.5μs	3.1μs	6μs
Loop bandwidth	1MHz	1MHz	200kHz
Fractional spurs @ Offset	-77.1dBc 500kHz	-55.3dBc 2.5MHz	-84.3dBc 3.5MHz
Reference spurs @ Offset	-117.8dBc 20MHz	-90dBc 20MHz	below noise floor

compared to 1GHz operation.

The in-band phase noise performance of the divide-multiply-divide implementation is 3.3dB higher compared to the in-band phase noise of the multiply-divide implementation illustrated with 1MHz loop bandwidth. The increase of the phase noise of the divide-multiply-divide implementation is due to the contribution of the phase noise of the frequency multiplier. The contribution of the phase noise of the frequency multiplier to the total in-band phase noise of the multiply-divide implementation was negligible due to the division ratio of the frequency divider placed between the frequency multiplier and the phase frequency detector.

The out-of-band phase noise of the divide-multiply-divide implementation sized with 1MHz loop bandwidth, estimated at 10MHz offset, is only 0.6dB more compared to the same phase noise of the multiply-divide implementation. As expected, the lower loop bandwidth (200kHz) will improve the out-of-band phase noise as shown in Table

B.6.

The switching speed of the divide-multiply-divide implementation sized with 1MHz loop bandwidth is about twice slower compared to the multiply-divide implementation. The frequency multiplier within the multiply-divide implementation was illustrated with 5MHz loop bandwidth. The frequency multiplier within the divide-multiply-divide implementation was illustrated with 2MHz loop bandwidth. The faster lock-in time of the frequency multiplier within the multiply-divide implementation is believed to resulted in faster switching time of the main system as well.

Finally, the comparison between the multiply-divide and the divide-multiply-divide implementation regarding the spurious tones concludes that smaller loop bandwidth of the divide-multiply-divide implementation is recommended for attractive spurious performance.

B.3 Summary

The divide-multiply-divide implementation of the proposed frequency synthesizer was discussed in this chapter. This implementation was illustrated for two frequencies (1GHz and 10GHz) with frequency resolution of 500kHz. The simulated results of the divide-multiply-divide implementation were compared to the simulated results of the divide-multiply implementation (10GHz) and the multiply-divide implementation (1GHz). The divide-multiply-divide implementation solves the disadvantage of the multiply-divide implementation regarding the highest application frequency. Meanwhile, the other advantages of the divide-multiply-divide implementation (the phase noise, the size of the loop filter, the switching speed, and the spurious performance) are similar to the advantages of the multiply-divide implementation. The main disadvantage of the divide-multiply-divide implementation is the programming of the frequency dividers and the frequency multiplier for a given channel spacing.

Appendix C

Loop Filter

This chapter discusses the transition processes in the loop filter while the PLL acquires a stable state. The theoretical analysis is compared to the simulated and the measured results. Additional readings regarding the theoretical models which will explain and predict the functionality and behavior of the PLL can be found in [70–74].

C.1 Formulation of the Voltage-Controlled Signal

Figure C.1 shows the loop filter under consideration in this work. A current source used to model the charge pump is also shown.

A time domain analysis of the loop filter output yields an expression for $\mathtt{V_{ctrl}}(t)$ given by,

$$\mathtt{V_{ctrl}}(t) = \mathtt{V_{ctrl}}(\infty) - \left(\mathtt{V_{ctrl}}(\infty) - \mathtt{V_{ctrl}}\left(0^{+}\right)\right) e^{-\frac{t}{\tau_F}}. \tag{C.1.1}$$

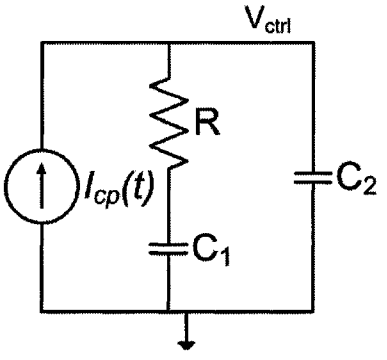


Figure C.1: Model of the charge pump and the implemented loop filter.

where $V_{ctrl}(\infty)$ is the voltage level that the voltage-controlled signal asymptotically approaches with time constant $\tau_F = R_F C_F$, and $V_{ctrl}(0^+)$ is the initial value of the $V_{ctrl}(t)$ signal. The values for the resistor R_F and capacitor C_F can be shown to be,

$$R_F = \frac{1+\omega^2 C_1^2 R^2}{\omega^2 C_1^2 R} \tag{C.1.2}$$

$$C_F = \frac{C_1}{1+\omega^2 C_1^2 R^2} + C_2. \tag{C.1.3}$$

C.1.1 Investigation of PFD UP Signals:

The function of the PFD is to compare the rising or falling edges of the reference signal (Ref) and the signal from the local oscillator (LO). If the Ref signal leads compared to the LO signal then the PFD generates an UP signal. If however, the LO signal leads then the PFD generates a Down signal. Figure C.2 depicts the case when the UP signal is generated by the PFD.

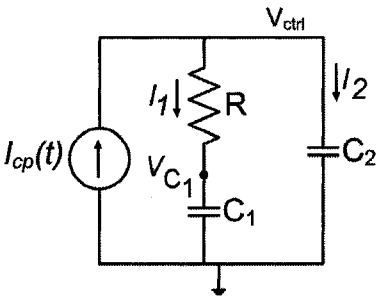


Figure C.2: A model of the charge pump together with the loop filter for incoming Up signals from the PFD.

If the time when the rising edge of the Ref signal arrives is denoted as t_{Ref}, and the time when the rising edge of the LO signal arrives is denoted as t_{LO}, then upon comparing the rising edges of the Ref and LO signals, the PFD will generate an UP signal if the rising edge of the Ref signal comes before the rising edge of the t_{LO},

$$UP(t) = \begin{cases} U(t) - U(t-T) & t_{Ref} - t_{LO} > 0 \\ 0 & t_{Ref} - t_{LO} \leq 0 \end{cases} \tag{C.1.4}$$

where U(t) is the unit step function, and $T = t_{Ref} - t_{LO}$ is the pulse width of the $UP(t)$ signal.

Each time the $\mathtt{UP}(t)$ signal is high, the charge pump generates a current pulse with an amplitude of $\mathtt{I_{CP}}$ and a pulse width equal to the pulse width of the UP signal,

$$\mathtt{I_{cp}}(t) = \mathtt{I_{CP}} \cdot \mathtt{UP}(t). \tag{C.1.5}$$

As shown in Figure C.2, the current $\mathtt{I_1}(t)$ flows through the resistor R and the capacitor $\mathtt{C_1}$. The current $\mathtt{I_1}(t)$ can be expressed as,

$$\mathtt{I_1}(t) = \mathtt{C_1}\frac{dv_{\mathtt{C_1}}}{dt}. \tag{C.1.6}$$

Thus, an expression for the voltage $v_{C1}(t)$ can be derived,

$$v_{C1}(t) = \int_{t_{\mathtt{start}}}^{t_{\mathtt{end}}} \frac{\mathtt{I_1}}{\mathtt{C_1}}\mathtt{dt} + \mathtt{V}_{C1(\mathtt{initial})} \tag{C.1.7}$$

where, t indicates the time the charge pump sources a charge to the loop filter starting at the time $t_{\mathtt{start}}$ i.e. arriving of the UP signal, and finishing at the time $t_{\mathtt{end}}$ when the charge pump is disconnected from the loop filter. The $\mathtt{V}_{C1(\mathtt{initial})}$ indicates the initial voltage stored on the capacitor $\mathtt{C_1}$. Equation (C.1.7) indicates that the charging of the capacitor $\mathtt{C_1}$ is a linear function of the time.

When the charge pump sources a current to the loop filter, the $\mathtt{V_{ctrl}}(t)$ signal rises, subject to the upper limit of its voltage range. Between two current pulses, the loop filter starts to discharge. The voltage values that the $\mathtt{V_{ctrl}}(t)$ signal will reach during the charging and discharging time depend on the timing constant of the loop filter and the pulse width of the current pulses. For the present work, the timing constant of the loop filter is much larger compared to the pulse width of the current pulses that are coming from the charge pump. Figure C.3 indicates the voltage levels reached at each point of the time during the observed acquisition.

Figure C.4 illustrates the charging process of the loop filter during the time interval between t_1 and t_2. At the time $t = t_1$ the charge pump sends the first current pulse. The loop resistor R will accommodate a voltage drop of,

$$\Delta\mathtt{V_R} = \mathtt{I_{CP}R}. \tag{C.1.8}$$

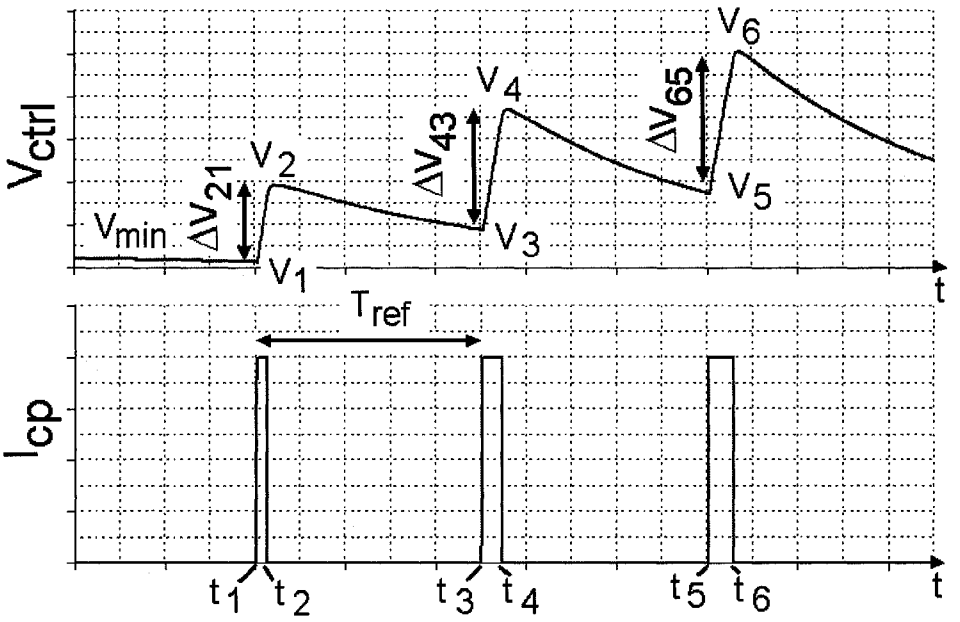


Figure C.3: The waveforms of the $\mathtt{V_{ctrl}}$ and $\mathtt{I_{cp}}$ for the case when the UP signals are generated.

From Figure C.2, the $\mathtt{V_{ctrl}}(t)$ signal can be expressed as,

$$\mathtt{V_{ctrl}}(t) = \Delta \mathtt{V_R} + \mathtt{V}_{C1}. \tag{C.1.9}$$

If the charge stored on the capacitor $\mathtt{C_1}$ before the time t_1 is ignored, then from (C.1.9) the $\mathtt{V_{ctrl}}(t)$ signal will approach the value given by (C.1.8) within the time $\tau_{\mathtt{F}}$. Thus, using the general expression (C.1.1) one can calculate the voltage level denoted as $\mathtt{V_2}$,

$$\mathtt{V_2} = \mathtt{I_{CP}R} - (\mathtt{I_{CP}R} - \mathtt{V_1})\, e^{-\frac{t_2 - t_1}{\tau_{\mathtt{F}}}}. \tag{C.1.10}$$

where,

$$\mathtt{V_1} = \mathtt{V_{min}}. \tag{C.1.11}$$

At the end of the aforementioned time interval the voltage across the capacitor $\mathtt{C_1}$ will be,

$$\mathtt{V_A} = \mathtt{V_1} + \frac{\mathtt{I_{CP}}}{\mathtt{C_1}}(t_2 - t_1). \tag{C.1.12}$$

During the time interval $t_2 < t < t_3$ the charge pump is disconnected from the loop filter. The capacitor $\mathtt{C_1}$ will source charge to the loop filter. The voltage-controlled

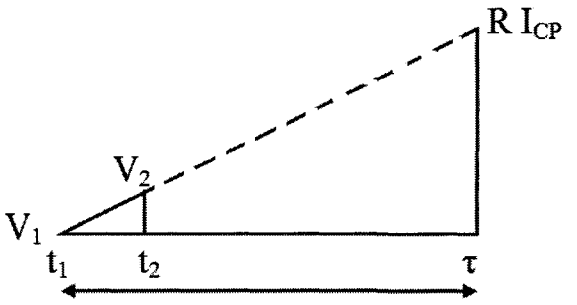


Figure C.4: Illustrated targeted and reached voltage levels during the charging interval.

signal will try to reach the voltage value V_A with a time constant τ_F. However, at the time $t = t_3$, due to $\tau_F > t_3 - t_2$ the voltage-controlled signal will reach the voltage value V_3 that can be expressed as,

$$V_3 = V_A - (V_A - V_2)\, e^{-\frac{t_3 - t_2}{\tau_F}}. \tag{C.1.13}$$

Following the same analogy, the voltage levels V_4, V_5, and V_6 can be expressed as,

$$V_4 = I_{CP}R - (I_{CP}R - V_3)\, e^{-\frac{t_4 - t_3}{\tau_F}} \tag{C.1.14}$$

$$V_5 = V_B - (V_B - V_4)\, e^{-\frac{t_5 - t_4}{\tau_F}} \tag{C.1.15}$$

$$V_6 = I_{CP}R - (I_{CP}R - V_5)\, e^{-\frac{t_6 - t_5}{\tau_F}}. \tag{C.1.16}$$

where,

$$V_B = V_A + \frac{I_{CP}}{C_1}(t_4 - t_3). \tag{C.1.17}$$

The $V_{ctrl}(t)$ signal now can be written for each time interval as follows,

$$V_{ctrl}(t) = \begin{cases} I_{CP}R - (I_{CP}R - V_1)\, e^{-\frac{t}{\tau_F}} & t_1 < t \le t_2 \\ V_A - (V_A - V_2)\, e^{-\frac{t}{\tau_F}} & t_2 < t \le t_3 \\ I_{CP}R - (I_{CP}R - V_3)\, e^{-\frac{t}{\tau_F}} & t_3 < t \le t_4 \\ V_B - (V_B - V_4)\, e^{-\frac{t}{\tau_F}} & t_4 < t \le t_5 \\ I_{CP}R - (I_{CP}R - V_5)\, e^{-\frac{t}{\tau_F}} & t_5 < t \le t_6 \\ \ldots \end{cases} \tag{C.1.18}$$

The theoretical analysis was compared to results obtained from PLL simulations. A frequency synthesizer of a type shown in Figure 5.7 was simulated. A transient

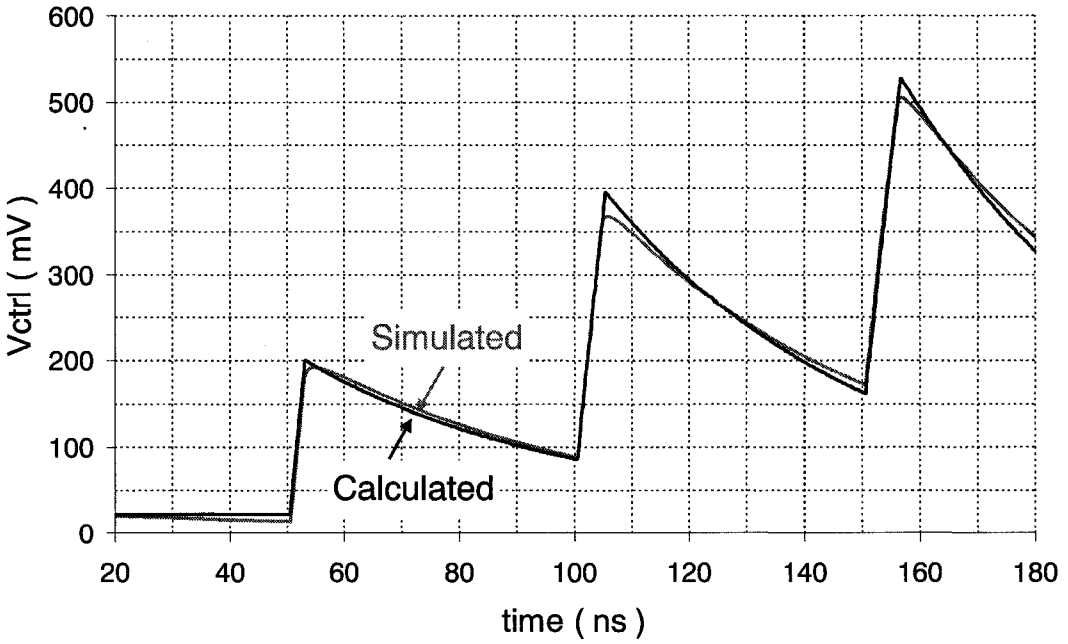


Figure C.5: Plot showing the calculated and simulated voltage controlled signal for the case of train of UP signals.

analysis (SpectreRF simulator) was used to get the waveform of the voltage-controlled signal in two cases.

In the first case, to simplify the simulation, the circuit design is kept for the PFD and the CP block while the VCO and the divider were implemented using behavior models. Figure C.5 shows that the proposed formulation of the voltage-controlled signal has a close agreement with the simulated results (within 6% over the interval shown).

In the second case, the full circuit design was used for all building blocks of the frequency synthesizer. In addition, a post-layout simulation was performed. Figure C.6 shows the calculated and the simulated results. The deviation between the two curves is within 7%.

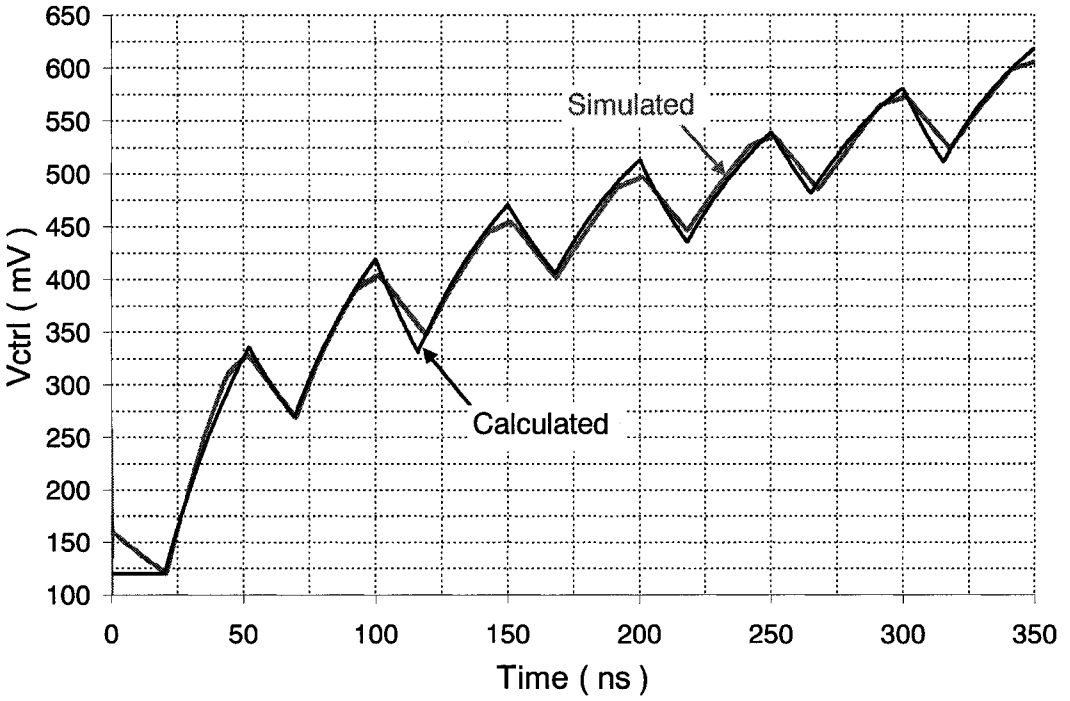


Figure C.6: The calculated and the post-layout simulated voltage controlled signal for the case of UP signals.

C.1.2 Investigation of PFD Down Signals:

Figure C.7 shows a model of the charge pump for the case when the PFD generates a Down signal.

Upon comparing the rising edges of the Ref and LO signals, the PFD will generate a Down signal if the rising edge of the LO signal comes before the rising edge of the Ref signal,

$$\mathtt{Down}\,(t) = \begin{cases} \mathtt{U}(t) - \mathtt{U}(t - \mathtt{T}) & t_{\mathtt{LO}} - t_{\mathtt{Ref}} > 0 \\ 0 & t_{\mathtt{LO}} - t_{\mathtt{Ref}} \leq 0 \end{cases} \tag{C.1.19}$$

where U(t) is the unit step function, and $\mathtt{T} = t_{\mathtt{LO}} - t_{\mathtt{Ref}}$ is the pulse width of the $\mathtt{Down}\,(t)$ signal.

Each time the $\mathtt{Down}\,(t)$ signal is high, the charge pump sinks a current pulse with an amplitude of $\mathtt{I_{CP}}$ and a pulse width equal to the pulse width of the Down signal,

$$\mathtt{I_{cp}}\,(t) = -\mathtt{I_{CP}} \cdot \mathtt{Down}\,(t)\,. \tag{C.1.20}$$

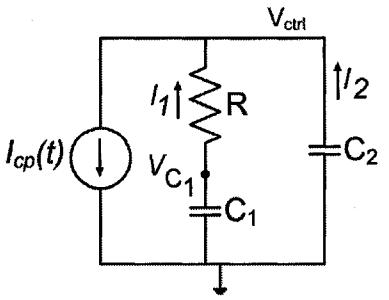


Figure C.7: A model of the charge pump together with the loop filter in a case of incoming Down signals.

The amplitude of the CP current in the case of the Down signals is not necessarily equal to the amplitude of the CP in the case of the UP signals. However, in order to simplify the writing, the notation of the CP current as well as discussed voltage points of the voltage-controlled signal is kept the same for both cases.

Figure C.8 illustrates the waveform of the voltage-controlled signal when the charge pump sinks a current from the loop filter. At the time $t = t_1$ the voltage-controlled signal has an initial value denoted as $\mathtt{V}_1$. The rising edge of the $\mathtt{Down}\,(t)$ signal generated by the PFD at the time $t = t_1$ causes the charge pump to be connected to the loop filter. The current sunk by the charge pump will cause a voltage drop over the loop resistor. Looking from the $\mathtt{V_{ctrl}}$ node, the voltage drop can be expressed as,

$$\Delta \mathtt{V}_\mathtt{R} = -\mathtt{I}_\mathtt{CP}\mathtt{R}. \tag{C.1.21}$$

At the same time, the charge pump creates a discharge path for the charge stored by the loop capacitor $\mathtt{C}_2$. As illustrated in Figure C.8, the $\mathtt{V_{ctrl}}\,(t)$ signal will drop and with a time constant $\tau_\mathtt{F}$ will target the value given with the expression (C.1.21). However, at the time $t = t_2$ the $\mathtt{V_{ctrl}}\,(t)$ signal will reach the value denoted as $\mathtt{V}_2$,

$$\mathtt{V}_2 = -\mathtt{I}_\mathtt{CP}\mathtt{R} + (\mathtt{I}_\mathtt{CP}\mathtt{R} + \mathtt{V}_1)\, e^{-\frac{t_2 - t_1}{\tau_\mathtt{F}}}, \tag{C.1.22}$$

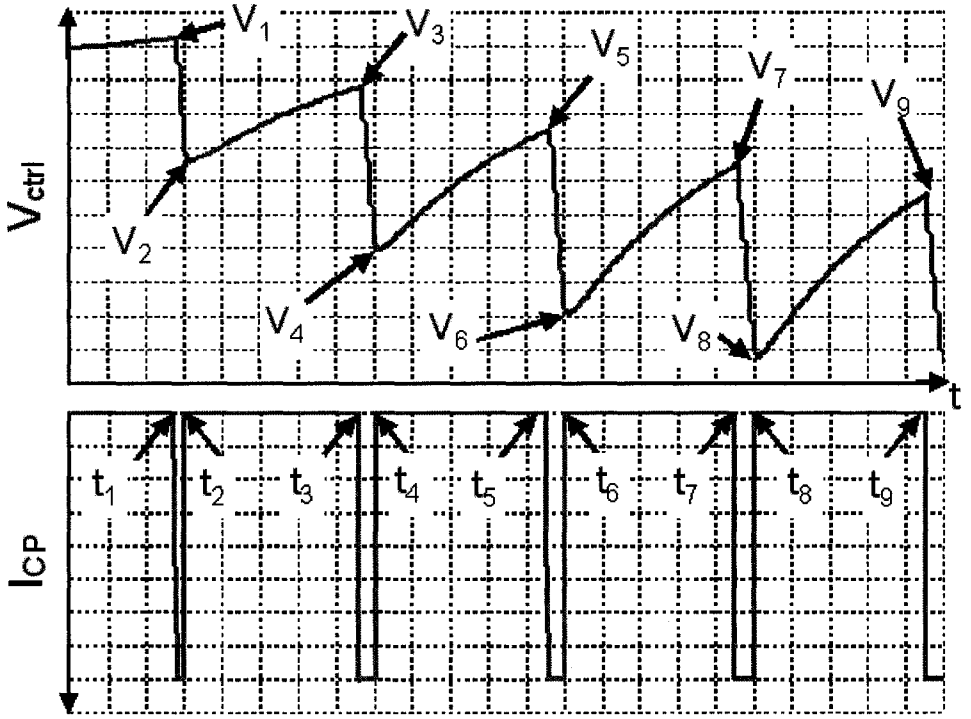


Figure C.8: The waveforms of the V_{ctrl} and I_{cp} for the case when the Down signals are generated from the PFD.

and the voltage across the capacitor C_1 will be reduced to,

$$V_A = V_1 - \frac{I_{CP}}{C_1}(t_2 - t_1). \tag{C.1.23}$$

During the time interval $t_2 < t < t_3$ the charge pump is disconnected from the loop filter. The capacitor C_1 will source charge to the loop filter. The voltage-controlled signal will try to reach the voltage value V_A with a time constant τ_F. However, at the time $t = t_3$, due to $\tau_F > t_3 - t_2$ the V_{ctrl} signal will reach the voltage value V_3 that can be expressed as,

$$V_3 = V_A - (V_A - V_2)\, e^{-\frac{t_3 - t_2}{\tau_F}}. \tag{C.1.24}$$

Following the same analogy, the voltage levels V_4, V_5, V_6, and V_7 can be expressed as,

$$V_4 = -I_{CP}R - (-I_{CP}R - V_3)\, e^{-\frac{t_4 - t_3}{\tau_F}} \tag{C.1.25}$$

$$V_5 = V_B - (V_B - V_4)\, e^{-\frac{t_5 - t_4}{\tau_F}} \tag{C.1.26}$$

$$V_6 = -I_{CP}R - (-I_{CP}R - V_5)\, e^{-\frac{t_6 - t_5}{\tau_F}} \tag{C.1.27}$$

$$V_7 = V_C - (V_C - V_6)\, e^{-\frac{t_7 - t_6}{\tau_F}} \tag{C.1.28}$$

where,

$$V_B = V_A - \frac{I_{CP}}{C_1}(t_4 - t_3) \tag{C.1.29}$$

$$V_C = V_B - \frac{I_{CP}}{C_1}(t_6 - t_5)\,. \tag{C.1.30}$$

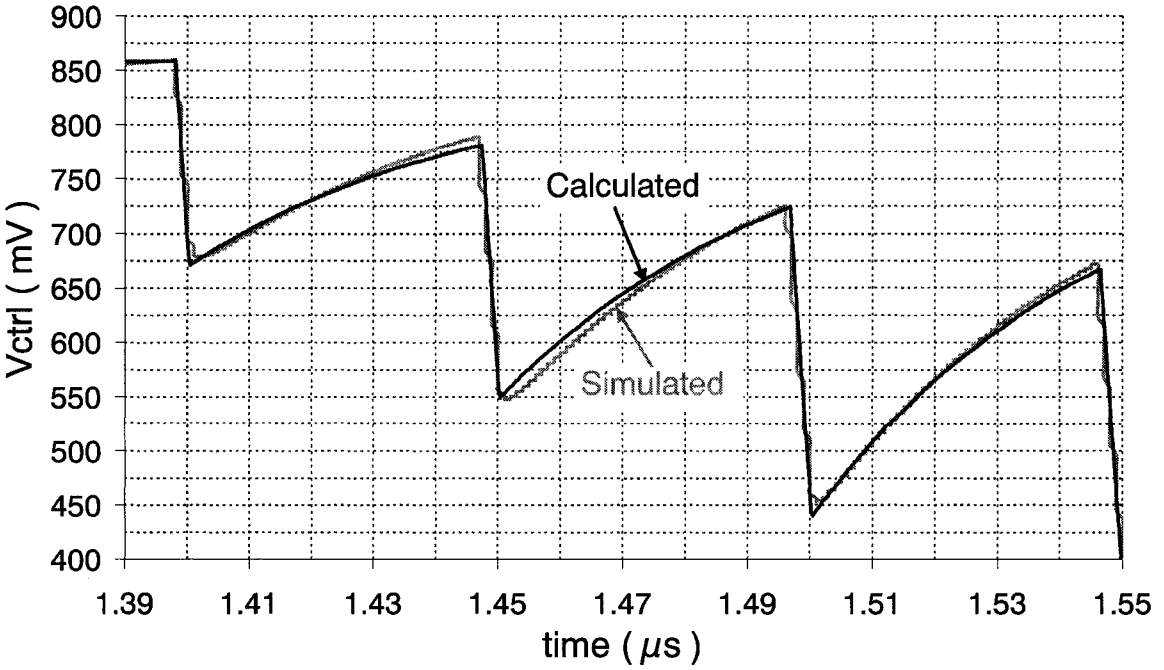


Figure C.9: The calculated and the simulated voltage controlled signal for the case of Down signals.

The $V_{ctrl}(t)$ signal now can be written for each time intervals as follows,

$$V_{ctrl}(t) = \begin{cases} -I_{CP}R + (I_{CP}R + V_1)\, e^{-\frac{t}{\tau_F}} & t_1 < t \leq t_2 \\ V_A - (V_A - V_2)\, e^{-\frac{t}{\tau_F}} & t_2 < t \leq t_3 \\ -I_{CP}R + (I_{CP}R + V_3)\, e^{-\frac{t}{\tau_F}} & t_3 < t \leq t_4 \\ V_B - (V_B - V_4)\, e^{-\frac{t}{\tau_F}} & t_4 < t \leq t_5 \\ -I_{CP}R + (I_{CP}R + V_5)\, e^{-\frac{t}{\tau_F}} & t_5 < t \leq t_6 \\ V_C - (V_C - V_6)\, e^{-\frac{t}{\tau_F}} & t_6 < t \leq t_7 \\ -I_{CP}R + (I_{CP}R + V_7)\, e^{-\frac{t}{\tau_F}} & t_7 < t \leq t_8 \\ \ldots \end{cases} \tag{C.1.31}$$

The simulated results show that the presented formulation of the voltage-controlled signal has a close agreement with the simulated results as shown in Figure C.9.

C.1.3 Measured Results

Figure C.10 shows the comparison between the measured and calculated waveform of the $V_{ctrl}(t)$ signal. As it was a case with the simulated results, the measurement results show a close agreement with the presented formulation of the $V_{ctrl}(t)$ signal. The reference signal was generated from Agilent 81134A 3.35GHz Pulse/Pattern Generator and the acquisition of the V_{ctrl} signal was monitored by the Tektronix TDS 3032 300MHz 2.5GS/s two channel oscilloscope.

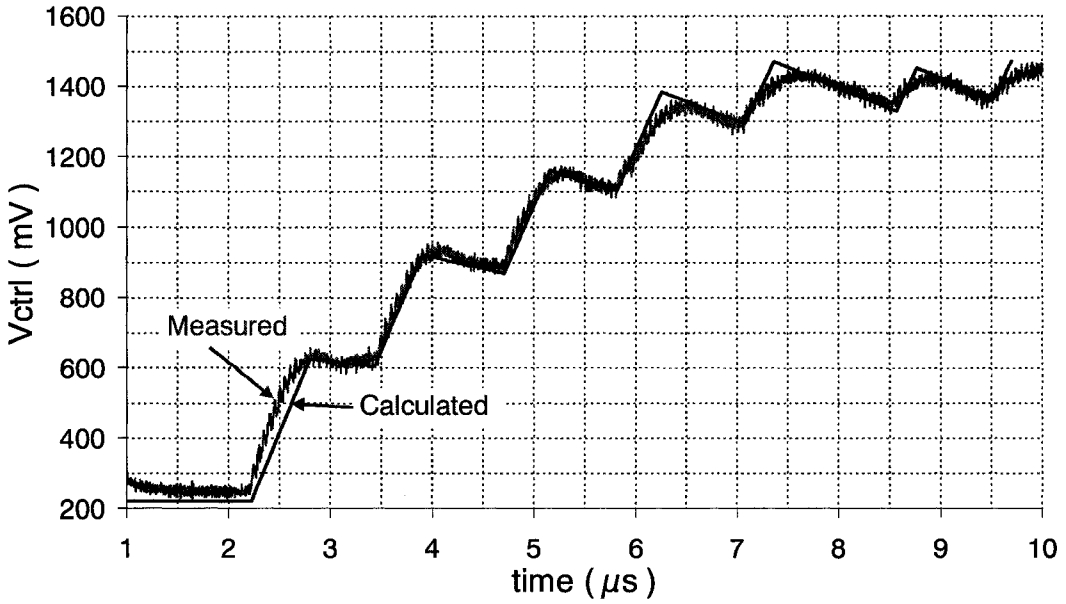


Figure C.10: Plot showing the calculated and measured voltage controlled signal for the case of UP signals.

C.2 Summary

A mathematical formulation of the transient portion of the signal used to control a tunable oscillator has been presented. A frequency synthesizer fabricated in a 0.13μm CMOS technology was used to compare the theoretically derived expressions for the voltage-control signal to the simulated and measured results. The comparison indicated that the presented formulation is in a close agreement (deviation less than a 7%) with the simulated and the measured results.

Appendix D

Literature Search for Modified CML Logic with Resistor Tail Bias

In order to investigate the existence of differential (CML based) cells with resistor tail bias in the literature, other than those produced with this thesis, the following resources were considered:

- IEEE Xplore (Institute of Electrical and Electronics Engineers - www.ieee.org):
  - IEEE Periodicals,
  - IET Periodicals,
  - IEEE Conference Proceedings,
  - IET Conference Proceedings,
  - IEEE Standards.
  - IEEE Books,
  - Educational Courses.
- google (www.google.com)
  - public educational courses and publications,
- Carleton University library:
  - books,

  - online web resources,
  - master thesis,
  - doctorate thesis,
  - scholars portal search,
  - Ei Engineering Village.

- Springer - Academic journals, books, and online media (www.springer.com);
- United States and most foreign patents (Patent Hunter).

Bibliography

[1] A. Blanchard, "*Phase-Locked Loops: Application to Coherent Receiver Design*," Wiley 1976.

[2] F. M. Gardner, "*Phaselock Techniques*," Hoboken, NJ : John Wiley, 3rd edition, 2005.

[3] J. Rogers, C. Plett, and F. Dai, "*Integrated Circuit Design for High-Speed Frequency Synthesis*," Artech House 2006.

[4] A. Marques, M. Steyaert, and W. Sansen, "Theory of PLL fractional-N frequency synthesizers," *Wireless Networks 4*, pp. 79–85, 1998.

[5] V. Solomko and P. Weger, "11 GHz CMOS $\Sigma\Delta$ frequency synthesiser," *Electronics Letters*, vol. 42, no. 21, pp. 1199 – 1200, October 2006.

[6] B. Muer and M. Steyaert, "A CMOS Monolithic $\Delta\Sigma$-Controlled Fractional-N Frequency Synthesizer for DCS-1800," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 7, pp. 835–844, July 2002.

[7] M. Marletta, P. Aliberti, M. Pulvirenti, A. Cavallaro, S. Terryn, P. Filoramo, R. Iardino, V. Spalma, and S. Cosentino, "Fully Integrated Fractional PLL for Bluetooth Application," *IEEE Radio Frequency Integrated Circuits Symposium*, pp. 557–560, June 2005.

[8] Z. Shu, K. L. Lee, and B. H. Leung, "A 2.4-GHz Ring-Oscillator-Based CMOS Frequency Synthesizer With a Fractional Divider Dual-PLL Architecture," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 3, pp. 452 – 462, March 2004.

[9] J. Yang, S. Kim, S. Kim, and B. Jeon, "Fast Switching Frequency Synthesizer Using Direct Analog Techniques for Phased-Array Radar," *Radar 97*, vol. 449, pp. 386–390, October 1997.

[10] A. Rokita, "Direct Analog Synthesis Modules for an X-Band Frequency Source," *12th International Conference on Microwaves and Radar, MIKON '98*, vol. 1, pp. 63–68, May 1998.

[11] J. Vankka, M. Waltari, M. Kosunen, and K. Halonen, "A Direct Digital Synthesizer with an On-Chip D/A-Converter," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 2, pp. 218–227, February 1998.

[12] F. Dai, W. Ni, S. Yin, and R. Jaeger, "A Direct Digital Frequency Synthesizer With Fourth-Order Phase Domain $\Delta\Sigma$ Noise Shaper and 12-bit Current-Steering DAC," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 4, pp. 839–850, April 2006.

[13] S. Turner and D. Kotecki, "Direct Digital Synthesizer With Sine-Weighted DAC at 32-GHz Clock Frequency in InP DHBT Technology," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 10, pp. 2284–2290, October 2006.

[14] C. Lam and B. Razavi, "A 2.6-GHz/5.2-GHz Frequency Synthesizer in 0.4μm CMOS Technology," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 5, pp. 788–794, May 2000.

[15] M. Margarit, D. Shih, P. Sullivan, and F. Ortega, "A 5-GHz BiCMOS RFIC Front-End for IEEE 802.11a/HiperLANWireless LAN," *IEEE Journal of Solid-State Circuits*, vol. 38, no. 7, pp. 1284–1287, July 2003.

[16] G. Leung and H. Luong, "A 1-V 5.2-GHz CMOS Synthesizer for WLAN Applications," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 11, pp. 1873–1882, November 2004.

[17] H. Lee, J. Cho, K. Lee, I. Hwang, T. Ahn, K. Nah, and B. Park, "A Σ-Δ Fractional-N Frequency Synthesizer Using a Wide-Band Integrated VCO and a Fast AFC Technique for GSM/GPRS/WCDMA Applications," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 7, pp. 1164–1169, July 2004.

[18] C. Heng and B. Song, "A 1.8-GHz CMOS Fractional-N Frequency Synthesizer With Randomized Multiphase VCO," *IEEE Journal of Solid-State Circuits*, vol. 38, no. 6, pp. 848 – 854, June 2003.

[19] C. Wang, Y. Tseng, H. She, and R. Hu, "A 1.2 GHz Programmable DLL-Based Frequency Multiplier for Wireless Applications," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 12, pp. 1377 – 1381, December 2004.

[20] Q. Du, J. Zhuang, and T. Kwasniewski, "A Low-Phase Noise, Anti-Harmonic Programmable DLL Frequency Multiplier With Period Error Compensation for Spur Reduction," *IEEE Transactions on Circuits and Systems - II: Express briefs*, vol. 53, no. 11, pp. 1205 – 1209, November 2006.

[21] P. Maulik and D. Mercer, "A DLL-Based Programmable Clock Multiplier in 0.18μm CMOS With -70 dBc Reference Spur," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 8, pp. 1642 – 1648, August 2007.

[22] S. Mahmud, S. Ganesan, A. Rusek, and M. Hillis, "A Programmable Self-Adaptive Digital Frequency Multiplier," *IEEE Transactions on Instrumentation and Measurement*, vol. 37, no. 2, pp. 227 – 230, June 1988.

[23] S. Milicevic and L. MacEachern, "Frequency of Oscillation of a Cross-Coupled CMOS VCO with Resistor Tail Biasing," *The 50th IEEE International Midwest Symposium on Circuits and Systems, MWSCAS 2007*, pp. 490 – 493, August 2007.

[24] S. Milicevic and L. MacEachern, "A Phase-Frequency Detector and a Charge Pump Design for PLL Applications," *International Symposium on Circuits and Systems, ISCAS 2008*, pp. 1532 – 1535, May 2008.

[25] S. Milicevic and L. MacEachern, "Frequency Dividers Implementing Custom Cells with Resistor Tail Bias," *The International Symposium on Signals, Systems and Electronics ISSSE 2007*, pp. 493 – 496, August 2007.

[26] S. Vaucher, I. Ferencic, M. Locher, S. Sedvallson, U. Voegeli, and Z. Wang, "A Family of Low Power Truly Modular Programmable Dividers in Standard 0.35μm CMOS Technology," *IEEE Journal of Solid State Circuits*, vol. 35, no. 7, pp. 1039 – 1045, July 2000.

[27] M. Ali and E. Hegazi, "A Multigigahertz Multimodulus Frequency Divider in 90-nm CMOS," *IEEE Transactions on Circuits and SystemsII:Express Briefs*, vol. 53, no. 12, pp. 1333 – 1337, December 2006.

[28] S. Cheng, H. Tong, J. Silva-Martinez, and A. I. Karscilayan, "A Fully Differential Low-Power Divide-by-8 Injection-Locked Frequency Divider Up to 18 GHz," *IEEE Journal of Solid State Circuits*, vol. 42, no. 3, pp. 583 – 591, March 2007.

[29] C. Kromer, G. Bueren, G. Sialm, T. Morf, F. Ellinger, and H. Jaeckel, "A 40GHz Static Frequency Divider With Quadrature Outputs in 80nm CMOS," *IEEE Microwave and Wireless Components Letters*, vol. 16, no. 10, pp. 564 – 566, October 2006.

[30] N. Sheng, R. Pierson, K. Wang, R. Nubling, P. Asbeck, M. Chang, W. Edwards, and D. Phillips, "A high-speed multimodulus HBT prescaler for frequency synthesizer applications," *IEEE Journal of Solid State Circuits*, vol. 26, pp. 1362 – 1367, October 1991.

[31] U. Singh and M. Green, "High-frequency CML clock dividers in 0.13 μm CMOS operating up to 38 GHz," *IEEE Journal of Solid State Circuits*, vol. 40, no. 8, pp. 1658 – 1661, August 2005.

[32] C. Vaucher and Z. Wang, "A low-power truly modular 1.8 GHz programmable divider in standard CMOS technology," *in Proc. 25th Eur. Solid-State Circuits Conference*, pp. 406 – 409, September 1999.

[33] P. Heydari and R. Mohavavelu, "Design of Ultra High-Speed CMOS CML buffers and Latches," *Proceedings of the 2003 International Symposium on Circuits and Systems*, vol. 2, pp. II–208 – II–211, May 2003.

[34] A. Tanabe, M. Umetani, I. Fujiwara, T. Ogura, K. Kataoka, M. Okihara, H. Sakuraba, T. Endoh, and F. Masuoka, "0.18μCMOS 10-Gb/s Multiplexer/Demultiplexer ICs using current mode logic with tolerance to threshold voltage fluctuation," *IEEE Journal of Solid State Circuits*, vol. 36, no. 6, pp. 988 – 996, June 2001.

[35] M. Mizuno, M. Yamashina, K. Furuta, H. Igura, H. Abiko, K. Okabe, A. Ono, and H. Yamada, "A GHz MOS adaptive pipeline technique using MOS current-mode logic," *IEEE Journal of Solid State Circuits*, vol. 31, no. 6, pp. 784 – 791, June 1996.

[36] I. Shahriary, G. Brisay, S. Avery, and P. Gibson, "GaAs Monolithic Digital Phase/Frequency Discriminator," *GaAs IC Symposium Technical Digest*, pp. 183 – 186, 1985.

[37] J. Rogers, F. Dai, M. Cavin, and D. Rahn, "A Multiband $\Delta\Sigma$ Fractional-N Frequency Synthesizer for a MIMO WLAN Transeiver RFIC," *IEEE Journal of Solid State Circuits*, vol. 40, no. 3, pp. 678–689, March 2005.

[38] J. H. Noh and H. G. Jeong, "Charge-Pump with a Regulated Cascode Circuit for Reducing Current Mismatch in PLLs," *Proceedings of World Academy of Sciency, Engineering and Technology*, vol. 25, no. 2007 ISSN 1307-6884, pp. 185 – 187, November 2007.

[39] J. Rogers and C. Plett, "*Radio Frequency Integrated Circuit Design*," Artech House 2003.

[40] W. Cock and M. Steyaert, "A CMOS 10GHz Voltage Controlled LC-Oscillator with Integrated high-Q Inductor," *Proceedings of the 27th European Solid-State Circuits Conference, ESSCIRC 2001*, pp. 498–501, September 2001.

[41] Y. Lee and H. Yu, "A Transformer-based Low Phase Noise and Widely Tuned CMOS Quadrature VCO," *IEEE International Symposium on Circuits and Systems, ISCAS 2006. Proceedings*, pp. 4041–4017, May 2006.

[42] A. Ravi, K. Soumyanath, L. Carley, and R. Bishop, "An Integrated 10/5GHz Injection-locked Quadrature LC VCO in a 0.18m digital CMOS process," *Proceedings of the 28th European Solid-State Circuits Conference, ESSCIRC 2002*, pp. 543–546, 2002.

[43] D. Park and B. Jung, "Low Power LC-VCO Design Using Direct Cross-coupled Cell Biasing," *IEEE International Symposium on Circuits and Systems, ISCAS 2006. Proceedings*, pp. 4018–4021, May 2006.

[44] L. Perraud, J. Bonnot, N. Sornin, and C. Pinatel, "Fully-Integrated 10 GHz CMOS VCO for multi-band WLAN applications," *Proceedings of the 29th European Solid-State Circuits Conference, ESSCIRC 2003*, pp. 353–356, 2003.

[45] T. Choi, H. Lee, L. Katehi, and S. Mohammadi, "A Low Phase Noise 10GHz VCO in 0.18μm CMOS Process," *34th European Microwave Conference*, pp. 273–276, October 2005.

[46] K. Ohhata, K. Harasawa, M. Honda, and K. Yamashita, "Design of Low-Noise, Low-Power 10-GHz VCO Using 0.18μm CMOS Technology," *IEICE Transactions on Electronics 2006*, vol. E89C, no. 2, pp. 203–205, February 2006.

[47] Z. Gu and A. Thiede, "10 GHz Low-Noise Low-Power Monolithic Integrated VCOs in Digital CMOS Technology," *IEICE Transactions on Electronics 2006*, vol. E89C, no. 1, pp. 88–93, January 2006.

[48] S. Milicevic, "*A Multi-Band Voltage-Controlled Ring Oscillator*," Carleton University, M.A.Sc. thesis, May 2005.

[49] C. Barrett, "Fractional/Integer-N PLL Basics," *Texas Instruments, Wireless Communication Business Unit*, pp. 1–55, August 1999.

[50] T.-H. Lin and Y.-J. Lai, "An Agile VCO Frequency Calibration Technique for a 10-GHz CMOS PLL," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 2, pp. 340 – 349, Febraury 2007.

[51] N. Pavlovic, J. Gosselin, K. Mistry, and D. Leenaerts, "A 10 GHz frequency synthesiser for 802.11a in 0.18μm CMOS," *Proceeding of the 30th European Solid-State Circuits Conference, ESSCIRC 2004*, pp. 367 – 370, September 2004.

[52] A. Ravi, G. Banerjee, R. E. Bishop, B. A. Bloechel, L. R. Carley, and K. Soumyanath, "10 GHz, 20 mW, fast locking, adaptive gain PLLs with on-chip frequency calibration for agile frequency synthesis in a 0.18μm digital CMOS process," *Symposium on VLSI Circuits, 2003. Digest of Technical Papers*, pp. 181 – 184, June 2003.

[53] Y.-H. Peng and L.-H. Lu, "A Ku-Band Frequency Synthesizer in 0.18μm CMOS Technology," *IEEE Microwave and Wireless Component Letters*, vol. 17, no. 4, pp. 256 – 258, April 2007.

[54] G.-Y. Tak, S.-B. Hyun, T. Y. Kang, B. G. Choi, and S. S. Park, "A 6.39-GHz CMOS Fast Settling PLL for MB-OFDM UWB Applications," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 8, pp. 1671 – 1679, August 2005.

[55] A. Ravi, R. E. Bishop, L. R. Carley, and K. Soumymath, "8 GHz, 20mW, fast locking, fractional-N frequency synthesizer with optimized 3rd order, 3/5-bit IIR and 3rd order 3-bit FIR noise shapers in 90nm CMOS," *IEEE Custom Integrated Circuits Conference*, vol. 28, no. 6, pp. 625 – 628, 2004.

[56] M.-T. Tsai and C.-Y. Yang, " A Fast-Locking Agile Frequency Synthesizer for MIMO Dual-mode WiFi / WiMAX Applications," *14th IEEE International Conference on Electronics, Circuits and Systems*, pp. 1384 – 1387, December 2007.

[57] B. Bisla, R. Eline, and L. Franca-Neto, "RF System and Circuit Challenges for WiMAX," *Intel Technology Journal*, vol. 8, no. 3, August 2004.

[58] B. Heydari, M. Bohsali, E. Adabi, and A. M. Niknejad, "Low-Power mm-Wave Components up to 104GHz in 90nm CMOS," *IEEE International Solid-State Conference*, pp. 200 – 597, February 2007.

[59] C. Doan, S. Emami, A. M. Niknejad, and R. W. Broaderson, "Millimeter - Wave CMOS Design," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 1, pp. 144 – 155, January 2005.

[60] K. Lee, I. Nam, I. Kwon, J. Gil, K. Han, S. Park, and B.-I. Seo, "The Impact of Semiconductor Technology Scaling on CMOS RF and Digital Circuits for Wireless Application," *IEEE Transactions of Electron Devices*, vol. 52, no. 7, pp. 1415 – 1422, July 2005.

[61] J. Gil and H. Shin, "A simple wide-band on-chip inductor model for silicon-based RF ICs," *IEEE Transactions on Microwave Theory and Techniques*, vol. 51, no. 9, pp. 2023 – 2028, September 2003.

[62] U. L. Rohde, A. K. Poddar, and G. Boeck, "*Modern Microwave Oscillators for Wireless Applications: Theory and Optimization*," John Wiley and Sons Inc., 2005.

[63] Y. jin Kim, J. wan Kim, V. N. Parkhomenko, D. Baek, J. heon Lee, E. yung Sung, I. Nam, and B. ha Park, "A Wideband 0.18-μm CMOS $\Sigma\Delta$ Fractional-N Frequency Synthesizer with a single VCO for DVB-T," *Asian Solid-State Circuits Conference*, pp. 193 – 196, November 2005.

[64] S. Kim and Y. Kim, "A Fractional-N PLL Frequency Synthesizer Design," *IEEE SoutheastCon, Proceedings*, pp. 84 – 87, April 2005.

[65] W. Rhee, A. Ali, and B.-S. Song, "A 1.1GHz CMOS Fractional-N Frequency Synthesizer with a 3b 3rd-Order ΣΔ Modulator," *IEEE Solid-State Circuits Conference*, pp. 198 – 199, February 2000.

[66] K. Waheed, K. Desai, and F. M. Salem, "A low power frequency synthesizer with an integrated negative transconductance LC-tuned VCO," *IEEE International Conference on Robotics, Intelligent Systems and Signal Processing*, vol. 1, pp. 582 – 587, October 2003.

[67] S. T. Lee, S. J. Fang, D. J. Allstot, A. Bellaouar, A. R. Fridi, and P. A. Fontaine, "A quad-band GSM-GPRS transmitter with digital auto-calibration," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 12, pp. 2200 – 2214, December 2004.

[68] B. Chi, X. Zhu, S. Huang, and Z. Wang, "1GHz monolithic high spectrum purity fractional-N frequency synthesizer with a 3-b third-order delta-sigma modulator," *7th International Conference on Solid-State and Integrated Circuits Technology*, vol. 2, pp. 1504 – 1507, October 2004.

[69] H. Mnif, M. Fakhfakh, I. Krout, M. Barhoumi, and M. Loulou, "A ΣΔ fractional- N synthesizer for GSM standard specifications," *13th IEEE International Conference on Electronics, Circuits and Systems*, pp. 1121 – 1124, December 2006.

[70] F. M. Gardner, "Charge-Pump Phase-Lock Loops," *IEEE Transactions on communicatins*, vol. COM-28, no. 11, pp. 1849–1858, November 1980.

[71] M. V. Paemel, "Analysis of a Charge-Pump PLL: A New Model," *IEEE Transactions on communicatins*, vol. 42, no. 7, pp. 2490–2498, July 1994.

[72] P. Hanumolu, M. Brownlee, K. Mayaram, and U. Moon, "Analysis of charge-pump phase-locked loops," *IEEE Transactions on circuits and systems - I: regular papers*, vol. 51, no. 9, pp. 1665–1674, September 2004.

[73] Z. Wang, "An Analysis of Charge-Pump Phase-Locked Loops," *IEEE Transactions on circuits and systems - I: regular papers*, vol. 52, no. 10, pp. 2128 – 2138, October 2005.

[74] S. Sancho, A. Suarez, and J. Chuan, "General Envelope-Transient Formulation of Phase-Locked Loops Using Three Time Scales," *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, no. 4, pp. 1310 – 1320, April 2004.