

Abstract

The objective of this project is to develop a novel phase-locked loop (PLL) system in cooperation with Xintronix Ltd, who set up the specification. The task is divided into two projects, focusing on the implementation of different parts. The PLL in this part comprises a testbench written in VerilogA, an LC oscillator with a cross-coupled NMOS-PMOS architecture and a quadrature frequency converter consisting of two Gilbert Cell type mixers, sharing their load for up-/down-conversion. The PLL is made of several blocks: a phase-detector, a signal processor with a digital 32-bit output, a $\Delta\Sigma$ -digital-to-analogue-modulator with a low-pass filter and a quadrature frequency converter fed by an LC oscillator and the $\Delta\Sigma$ -modulator's output. The frequency of the LC oscillator was specified to be 5 GHz. A divider, part of the VerilogA testbench, divides that frequency down by two using two D-Flip-Flops. Thereby, two streams with a quadrature relation and 2.5 GHz are created. The quadrature frequency converter is fed by these two streams and quadrature related streams from the $\Delta\Sigma$ -modulator with a frequency in the range of 1 mHz up to 250 kHz. The purpose of the quadrature frequency converter is to compensate for deviations in the LC oscillator and divider from the ideal 2.5 GHz. The output of the PLL is supposed to be exactly 2.5 GHz.

Characteristics of the oscillator and the mixer were measured by means of simulations. A phase noise of 133.3 dBc/Hz at an offset of 1 MHz from the oscillation centre frequency was achieved for the LC oscillator. The mixer has a noise figure of 36 dB at a 1 MHz offset. The conversion gain is 7.6 dB and the IIP3, characterising the linearity, is at -8 dBm.

The achievements made in this project are:

- An ideal testbench was developed in VerilogA for all the blocks of the PLL, which was shown working (Chapter 4.2).
- An LC oscillator was developed on transistor level based on the cross-coupled NMOS-PMOS architecture. Three approaches were made to achieve best results (Chapter 4.3).
- The oscillator has competitive low phase noise and a good stability over PVT variations (Chapter 5.1).
- Two Gilbert Cells were designed sharing one pair of resistive loads to form a quadrature frequency converter (Chapter 4.4).
- The mixer has reasonable characteristics compared to publications (Chapter 5.2).

Acknowledgements

Contents

List of Figures	xi
List of Tables	xiii
Nomenclature	xv
1 Introduction	1
2 Background	3
2.1 LC Oscillators	3
2.2 Gilbert Cell Mixers	8
2.3 Phase-Locked Loops	14
3 State of the Art	19
3.1 LC Oscillators	19
3.2 Gilbert Cell Mixers	22
3.3 Frequency Synthesisers	24
4 System Specification and Design	27
4.1 Specification and Design Introduction	27
4.2 Testbench Design	29
4.3 LC Oscillator Design	33
4.4 Gilbert Cell and Frequency Converter Design	36
5 Results and Evaluation	43
5.1 Oscillator Characteristics	43
5.2 Gilbert Cell Characteristics	51
6 Conclusion and Outlook	57
6.1 Summary	57
6.2 Future work	58
References	59
Appendix A	63
A.1 $\Delta\Sigma$ -digital-to-analogue-modulator	63
A.2 VerilogA Code	65

Contents

List of Figures

1.1	Architectural view of the frequency synthesiser; red components are part of this project and will be described in detail in this paper	1
2.1	LC resonator with parasitic resistances and equivalent circuit	4
2.2	LC resonator with negative compensating resistor	5
2.3	Feedback System	5
2.4	Two different LC oscillator configurations	6
2.5	Output spectrum of an oscillator	7
2.6	Mixer output signal	9
2.7	Mixer linearity parameters	10
2.8	Gilbert Cell Mixer	11
2.9	Quadrature frequency converter with two Gilbert Cells	11
2.10	D-Flip-Flop divider with phase shifting of 90°	12
2.11	Input and output signals of a quadrature frequency converter	13
2.12	Two Gilbert Cells sharing one load for generating the sum or difference signal on the output	14
2.13	Simple Synthesiser architecture	15
2.14	Output locking to the reference over time	15
2.15	Integer-N synthesiser architecture	16
2.16	Dual-loop synthesiser comprising two PLLs and a SSB mixer	17
3.1	LC tank oscillators with more than one active device	20
3.2	Equivalent circuit of an inductor with parasitics	21
3.3	State of the art oscillator designs	22
3.4	Circuit introduced by [15]	23
3.5	Circuit proposed by [19] (from [19])	24
3.6	Frequency Synthesiser proposed by [21] (from [21])	25
4.1	Detailed system view with expected signals	27
4.2	Testbench schematic	29
4.3	Locking of the PLL with high values to show operation	32
4.4	Schematics of the LC oscillator	33
(a)	Original architecture	33
(b)	Modified architecture	33
4.5	Layout view of the inductor from the gpdk180nm library	34
4.6	Oscillator dimensions	35
4.7	Output of oscillator and stabilisation time; blue: real, red: ideal	36

List of Figures

4.8	Schematic of the testbench for the frequency converter	37
4.9	Schematic of one Gilbert Cell	37
4.10	Gilbert Cell dimensions	38
4.11	Gilbert Cell block diagram with voltages	39
4.12	Mixer output; left: overall shape, right: zoomed in; purple: real output, red: ideal output	40
4.13	Frequency converter output; top: overall shape, bottom: zoomed in, purple: real output, red: ideal output	41
5.1	Phase Noise of the original oscillator architecture for $L = 4.06 \text{ nH}$	44
5.2	Frequency over PVT variations for the original architecture with $L = 4.06 \text{ nH}$	45
5.3	Phase noise for $L = 1.13 \text{ nH}$ with the modified architecture	45
5.4	Phase noise for $L = 4.06 \text{ nH}$ with the modified architecture	46
5.5	Process and temperature variation of the frequency for $L = 1.13 \text{ nH}$	47
5.6	Process and temperature variation of the frequency for $L = 4.06 \text{ nH}$	47
5.7	Worst case PVT variations of the frequency for $L = 1.13 \text{ nH}$	48
5.8	Worst case PVT variations of the frequency for $L = 4.06 \text{ nH}$	48
5.9	Voltage spectrum for $L = 1.13 \text{ nH}$	49
5.10	Voltage spectrum for $L = 4.06 \text{ nH}$	49
5.11	Conversion gain for the mixer	51
5.12	Noise Figure depending on the LO input power	52
5.13	Realitive Noise Figure over frequency	53
5.14	Worst case relative Noise Figure with PVT variations	53
5.15	1 dB compression point for the mixer	54
5.16	Input third order intercept point for the mixer	54
5.17	Isolation from the IF port to the RF port depending on the LO input amplitude	55
A.1	1st order Delta Sigma digital to analogue converter	63
A.2	2nd order Delta Sigma digital to analogue converter	63
A.3	Sallen-Key second order filter topology	64

List of Tables

2.1	Truth table for the D-Flip-Flop divider with phase shifting	12
3.1	Comparison of various LC oscillator designs	22
3.2	Comparison of different mixers	23
3.3	Comparison of different frequency synthesisers	25
4.1	General specification	28
4.2	Parameters for the exemplary locking shown in Figure 4.3	32
5.1	Comparison of LC oscillator designs to this design	50
5.2	Comparison of published mixer designs to this design	56

List of Tables

Nomenclature

Variables

Variable	Description	Physical Unit
f	Frequency	Hz
ω	Angular frequency = $2\pi f$	Hz
t	Time	s
V	Voltage	V
$v(t)$	Voltage varying with time	V
$\phi(t)$	Phase noise	dB/Hz
L	Inductance	H
C	Capacitance	F
R	Resistance	Ω
Q	Quality factor of resonant circuit	

Abbreviations

Abbreviation	Description
AC	Alternating Current, periodically varying over time
ADE	Analogue Design Environment (Cadence)
CMOS	Complementary Metal Oxide Semiconductor
DAC	Digital-to-Analogue Converter
DC	Direct Current, not periodically varying with time
$\Delta\Sigma$	Delta-Sigma-Modulator
DFF	D-Flip-Flop
DSB	Double-sideband
FF	fast NMOS, fast PMOS process corner
FPGA	Field Programmable Gate Array
FS	fast NMOS, slow PMOS process corner
HF	High Frequency
IF	Intermediate Frequency
I/Q	Quadrature related streams
LC	Inductor-Capacitor circuit
LF	Low Frequency

Nomenclature

Abbreviation	Description
LO	Local Oscillator
LPF	Low Pass Filter
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
n.a.	not available
NF	Noise Figure
PAC	Periodic AC analysis
PLL	Phase-locked loop
PN	Phase Noise
PNOISE	Perodic Noise analysis
PSS	Periodic Steady State analysis
PVT	Process, Voltage and Temperature variations
QPAC	Quasi-Periodic AC analysis
QPSS	Quasi-Periodic Steady State analysis
QVCO	Quadrature Voltage Controlled Oscillator
RF	Radio Frequency
SNR	Signal to Noise Ratio
SF	slow NMOS, fast PMOS process corner
SS	slow NMOS, slow PMOS process corner
SSB	Single-sideband
VCO	Voltage Controlled Oscillator

Notes

In all circuit schematics where transistors are used and bulk connections are open, the NMOS bulks are connected to ground and the PMOS bulks to the supply voltage V_{dd} . The connections are left open for better readability.

1 Introduction

The project's main objective is to design a mixed-signal phase-locked loop (PLL) with low noise and a high frequency output. This will be done in collaboration with Xintronix Ltd. The realisation is split in two parts both implemented within individual projects.

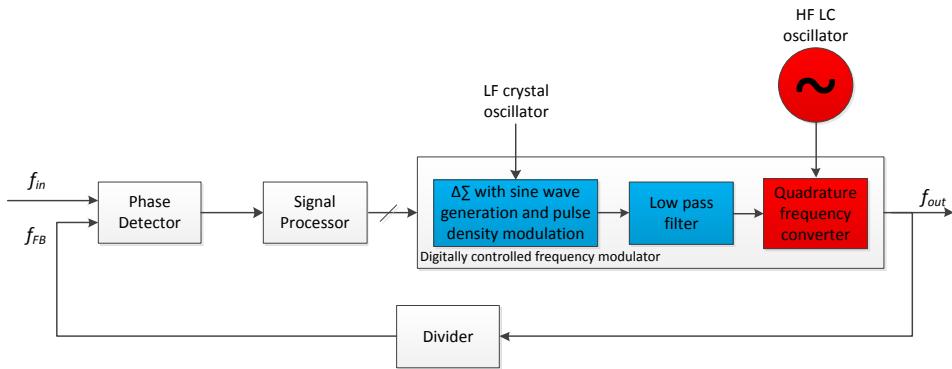


Figure 1.1: Architectural view of the frequency synthesiser; red components are part of this project and will be described in detail in this paper

The PLL developed in this project is illustrated in Figure 1.1. It comprises a digitally controlled frequency modulator and a testbench environment, which is shown in grey in Figure 1.1. The testbench is written in the hardware description language VerilogA and serves as a simulation of the PLL as a whole system. It includes a phase detector as well as a signal processor, which performs an analogue-to-digital conversion and feeds a digital output into the frequency modulator. Additionally the testbench contains a divider for the feedback frequency.

The frequency modulator forms the main part of this work. It consists of a Sigma-Delta-Modulator, which translates the digital input signal via pulse density modulation into an analogue signal and is implemented in VerilogA. To reduce the noise which might have been induced by the input frequency itself, a filter is applied to the output of the Sigma-Delta-Modulator to push the noise out of the spectrum. That part will be implemented in the complementary project and is coloured blue in Figure 1.1.

The components described in the following paragraphs, which are also included in the frequency synthesiser, will be developed as part of this project.

The filtered signal is used as an input for a quadrature frequency converter. The signal consists of two channels with the same frequency but in quadrature. The

1 Introduction

frequency converter contains two mixers for which Gilbert Cells will be used. A Gilbert Cell can work as an analogue multiplier. It is driven by a local oscillator and modulates an input frequency to either the sum of the oscillator and input frequency or the difference of both. The mixers are driven by signals with a frequency of 2.5 GHz, which are generated by an oscillator, the design of which is also part of this project. The oscillator frequency is originally 5 GHz and is then divided by two using two D-flip-flops. The results are two output streams with exactly the same frequencies, but by 90° shifted phases, which therefore makes these streams operate in quadrature. These streams are fed into one mixer each, where they are mixed with the input I/Q and generate the PLL output frequency to be fed back and phase-locked. In high frequency design, parasitic capacitances play a major role and therefore need special attention. By being charged and discharged alternately, they could cause a circuit to oscillate and thus make it unstable. This is of special importance for the design of the Gilbert Cells. The LC oscillator forms an important part of the system, because it sets the limit for the output frequency. Additionally it is noise critical, which has to be taken into account during designing, as any noise in the LC oscillator is multiplied by the Gilbert Cell and adds to the noise on the output. The oscillator also needs to be very accurate and stable over variations in process, supply voltage and temperature (PVT) for the same reasons.

This PLL will be used to clean a clock signal on the input which is impaired by jitter. The output should have very low jitter and therefore the implementation will focus on inducing as little noise as possible. Besides the LC oscillator, the input signal coming into the PLL is a noise critical part. It is firstly filtered by a low-pass filter in the signal processor, to remove the broadband noise on the input and then filtered again by the Sigma-Delta-Modulator. The noise of the LC oscillator has to be compensated by the feedback loop. An important part of the project is to finally determine the actual present noise by means of simulative measurements and if applicable reducing it.

This research review will explain the backgrounds necessary to understand the project's intent and introduce the state of the art of current frequency synthesizers and their components. Chapter 2 explains basics on LC oscillators, Gilbert Cells and quadrature frequency converters, as well as fundamentals of PLLs. Recent publications on those components are evaluated and discussed in Chapter 3. Furthermore, the chosen architectures for the designs are introduced. The implementation is shown in Chapter 4. There, several approaches for the designs are compared. The component geometries are derived from mathematical equations. Chapter 5 shows the simulations of the designs and characteristic figures describing their quality. This review concludes with a summary of the findings and an outlook for future improvement in Chapter 6.

2 Background

This chapter summarises the most important background information and thereby points out the difficulties of this project. Basics of LC oscillators are explained and two fundamental architectures presented. Important terms in the context of oscillators such as phase noise and the quality factor are clarified. Furthermore, mixer behaviour and characteristics are outlined and the circuit of a so called Gilbert Cell is demonstrated in detail. Finally, several approaches for phase-locked loops are illustrated and briefly discussed. After reading this chapter the reader should be able to understand and judge the analysis of the above mentioned components, which will be conducted in the next chapter.

2.1 LC Oscillators

Oscillators provide a periodic output signal with a certain dominant frequency and, generally, harmonically related energy. Several approaches exist, some of which will be described in the following section together with their advantages and disadvantages. The focus is on LC oscillators, which comprise an inductor and a capacitor as their frequency selective components and active feedback to induce oscillation.

The primary metric under control of the designer in an oscillator is the resonant frequency. The metric that can only partially be controlled is phase noise, which is a critical term for oscillators, especially if they are used as a high frequency reference in a more complex system. In that case any inaccuracy is forwarded to the system, which the oscillator is embedded in. Phase noise can influence the timing of data encoding in communication lines and produce errors in transmitted data via jitter. When the oscillator is used for a clocked system, it can cause synchronisation issues. Therefore, techniques for avoiding high phase noise will be introduced and discussed.

The most common types in CMOS high frequency technology are digital ring oscillators and LC oscillators. Both are used for different applications because their advantages and disadvantages satisfy different requirements.

Ring oscillators are easy to implement. They mainly consist of an odd number of cascaded inverters, where the output is connected to the input. That setup forces the system to oscillate. For a ring oscillator no large passive elements such as inductors are necessary, which means they consume only a small area on a chip. When a ring oscillator acts as a voltage-controlled oscillator (VCO) and thus is

2 Background

controllable by a voltage input, the frequency that it can output is variable over a wide range. However, in comparison to LC oscillators ring oscillators consume more power and their noise is typically much higher. [1]

The requirements for the oscillator that will be used as a reference in the PLL of this project are high accuracy, which means low jitter and phase noise, as well as low power consumption. The area consumed on the chip should be as little as possible, but there is no strict limit, which means that use of an inductor is generally possible. Thus, a LC oscillator is the better choice for this implementation.

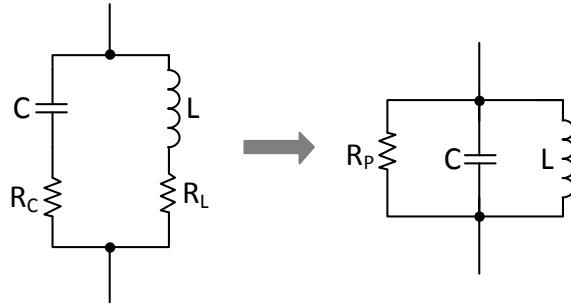


Figure 2.1: LC resonator with parasitic resistances and equivalent circuit

A LC resonator may in its simplest form consist of one inductor and one capacitor. It resonates ideally at a frequency of

$$\omega_0 = \frac{1}{\sqrt{LC}}. \quad (2.1)$$

In reality the devices are not ideal. They have a parasitic resistance, as shown in Figure 2.1. The first circuit shows the devices with parasitic resistance R_C or R_L respectively in series to their corresponding device and the second circuit shows an equivalent, where R_P incorporates both parasitic resistances. Such a resonator circuit is referred to as LC tank. [2]

The problem of the parasitic resistances is, that they attenuate the amplitude of the oscillatory response to an energy input and eventually force it to stop. Figure 2.2 shows a setup for a compensating circuit, where an active device is added which has a resistance of $-R_P$. Ideally, the parasitic resistance and the compensating negative resistance cancel each other completely out. That ensures a steady oscillation with no losses. [1]

An oscillator can be described as a positive feedback system with an amplifier, where the amplified signal is fed back. The block diagram of such a system is illustrated in Figure 2.3. The gain of the amplifier is $A(j\omega)$ and the feedback is $\beta(j\omega)$, both are dependent on the frequency. The product of both is called loop gain. That feedback system can be summarised by

$$\frac{v_{out}}{v_{in}} = \frac{A(j\omega)}{1 - A(j\omega)\beta(j\omega)}. \quad (2.2)$$

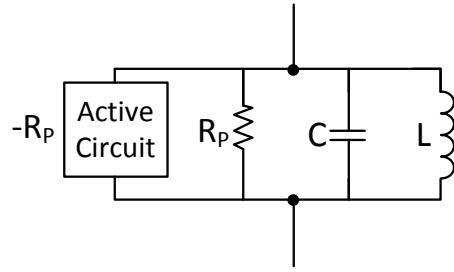


Figure 2.2: LC resonator with negative compensating resistor

Looking at the transfer function of the system in 2.2, one can see that two conditions need to be met to achieve a steady oscillation. If the loop gain is equal to one, the transfer function becomes infinity. The system is unstable and starts oscillating. The phase of the loop gain has to be zero at this point. This is called Barkhausen criterion. [3]

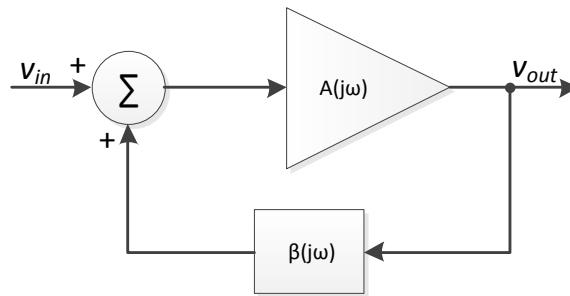


Figure 2.3: Feedback System

There are two commonly used types of feedback LC oscillators, the Colpitts and the Hartley, shown in Figure 2.4a and 2.4b. In both cases the parallel LC circuit is connected between the gate and drain of a MOSFET. In case of the Colpitts oscillator, a capacitive divider is used to feed some of the voltage back to the source of the MOSFET. In case of a Hartley oscillator, an inductive divider is used. The resistor R compensates inductive losses and represents the output resistance of the MOSFET as well as the load of the oscillator. [3]

To minimise the unwanted induced noise, mostly only one transistor is used in the active circuit of a LC oscillator, because transistors are the main sources of noise in a circuit. As the oscillator of this project should provide a high frequency, parasitics play a major role and have to be taken into account for calculating the oscillation frequency. The parasitic capacitors are charged and discharged during oscillation, which might delay or even distort the signal. [2]

Colpitts oscillators are preferred over Hartley oscillators for fixed-voltage oscillators, because they only require one inductor. That also has the advantage of

2 Background

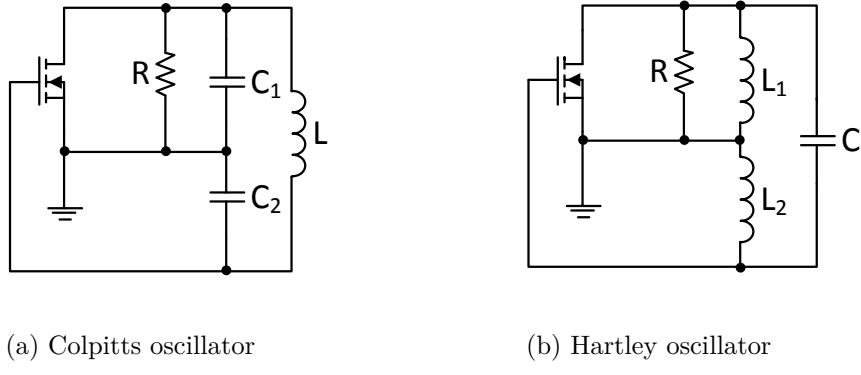


Figure 2.4: Two different LC oscillator configurations

better results in terms of phase noise because the two inductors of the Hartley configuration bear a risk of mutual inductance and therefore noise introduction. [2, 4]

Oscillators are prone to variations in the surrounding temperature of the circuit, the process variations during manufacturing and the supply voltage. This is known as PVT (process, voltage, temperature) variations. For the process variations, there are statistical models available, called corner cases, which were taken from manufactured circuits and can be used for simulations. In the technology used in this project, there are only corner cases available for transistors. This corner cases vary the switching speed of NMOS and PMOS transistors. Four settings are available, SS, which means both NMOS and PMOS are slow, SF, which means NMOS transistors are slow and PMOS types fast, FS, where NMOS are fast and PMOS slow, and finally FF where both types are switching fast. The voltage variation is usually taken into account from $\pm 10\%$ of the desired supply voltage. The temperature range depends on the desired application of the circuit. For commercial products, the range is usually from 0°C to 70°C, for automotive and military applications it is about -55°C to 125 °C.

PVT variations should be taken into account to simulate the circuits realistically and make the specification valid in a realistic environment. Especially oscillators are very sensitive to these variations and the chosen architecture can strongly influence the stability under certain circumstances. This fact is taken into account for choosing the architecture of this project's oscillator.

The deviation of the frequency due to noise is measured as phase noise. A periodic sinusoidal signal oscillating at a frequency of ω_0 can be described by

$$v(t) = V_0 \cdot \cos(\omega_0 t). \quad (2.3)$$

However, taking phase noise into account, the equation changes to

$$v(t) = V_0 \cdot \cos(\omega_0 t + \phi_n(t)), \quad (2.4)$$

where $\phi_n(t)$ represents the deviation from the actual frequency. Figure 2.5 shows the spectrum of an ideal oscillator in (a) and the spectrum of a real oscillator with noise in (b). The ideal oscillator has an impulse response, while the real oscillator gives a spread spectrum. Phase noise is usually measured in dBc/Hz. Therefore the noise power in a 1 Hz wide band at an offset of $\Delta\omega$ from ω_0 is calculated and set in relation to the carrier power. That is where the ‘dBc’ comes from, it states that the signal is measured in relation to the carrier signal. [2]

Jitter is another characteristic parameter for oscillators. It specifies how much in time the oscillator period deviates from its actual desired value and is commonly used for digital square signals.

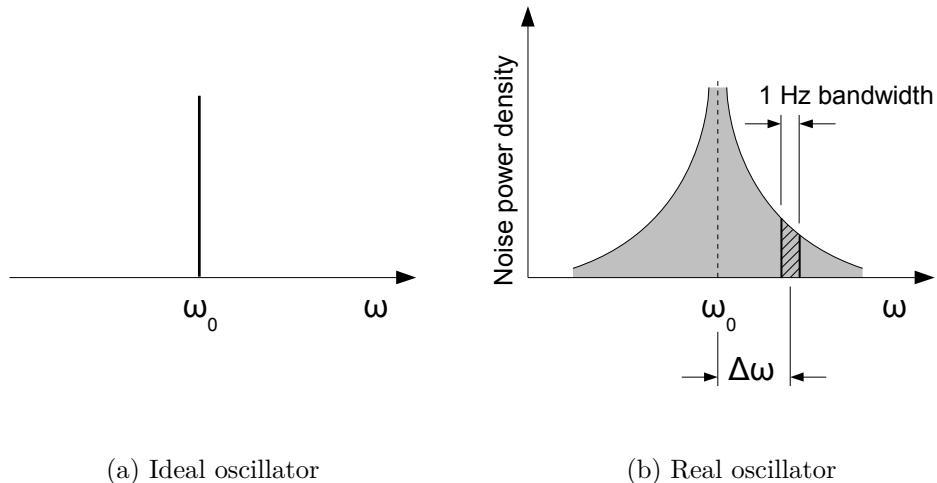


Figure 2.5: Output spectrum of an oscillator

To minimise the phase noise of an oscillator, their Q must be high. Q is a quality parameter of an LC circuit, that can be defined in several ways. One of them is to describe it as $2\pi \cdot \frac{\text{energy stored in LC}}{\text{energy dissipated per cycle}}$. The common way to describe a simple LC tank with a parallel resistance that contains the parasitics is

$$Q = \frac{\omega_0 L}{R_P}. \quad (2.5)$$

Generally, LC oscillators with a Q greater than 20 have a sufficient small phase noise, but that also depends on the supply voltage. The phase noise increases when lowering the supply voltage and therefore the power. [2]

Other types of noise are flicker noise and white noise. Flicker noise is the dominant low-frequency noise in MOSFETs. Although several studies exist, its origin is not fully explored yet. It is also known as $\frac{1}{f}$ -noise, because its spectral density is inversely proportional to frequency. [5]

White noise, which mainly consists of thermal noise and shot noise in MOSFETs, is a random noise that occurs within a certain frequency band. Its power spectrum is flat. [6]

2 Background

Leeson's equation, given in 2.6, includes flicker noise and white noise in its description of phase noise. That shows directly the dependency of the phase noise on the quality factor Q . [1]

$$L(\Delta f) = 10 \log \left[\frac{2FKT}{P_0} \left(1 + \frac{f_0^2}{4Q^2\Delta f^2} \right) \left(1 + \frac{\Delta f_{1/f^3}}{\Delta f} \right) \right] \quad (2.6)$$

where:

F	excess noise factor
K	Boltzmann constant $1.38 \cdot 10^{-23}$ J/K
T	temperature
P_0	power of carrier
f_0	carrier frequency
Q	quality factor of the LC tank
Δf	offset from f_0
$\Delta f_{1/f^3}$	corner frequency of $1/f$ noise

2.2 Gilbert Cell Mixers

A mixer is a non-linear device, which performs a frequency conversion, either up or down, of two input frequencies to one mixed output. They are a class of multipliers because they can multiply their two inputs. One of these inputs is a local oscillator (LO), which usually has a fixed frequency. The other input and output frequency names depend on the way the conversion is performed. For up-conversion the second input is the intermediate frequency (IF) that can vary in a given range. The mixed output signal is then called the radio frequency (RF). For down-conversion the two names are the other way round, as then the second input has a higher frequency than the output, which is also usually in the RF region. [7]

The following section will give an overview of mixer basics and their characteristics and focus on the Gilbert cell architecture [8], which will be used in the project to implement a quadrature frequency converter, the functionality of which will be described as well in this section.

Given the LO and IF inputs of an up-converting multiplier are sinusoidal signals. Then they can be described by

$$v_{LO}(t) = V_{LO} \cdot \cos(\omega_1 t) \quad (2.7)$$

$$v_{IF}(t) = V_{IF} \cdot \cos(\omega_2 t). \quad (2.8)$$

The resulting signal on the RF port is then

$$\begin{aligned} v_{RF}(t) &= V_{LO} \cdot V_{IF} \cdot \cos(\omega_1 t) \cdot \cos(\omega_2 t) \\ &= \frac{V_{LO} \cdot V_{IF}}{2} (\cos((\omega_1 - \omega_2)t) + \cos((\omega_1 + \omega_2)t)). \end{aligned} \quad (2.9)$$

The result term includes a difference frequency and a sum frequency. These terms are responsible for either down-conversion or up-conversion. [7, 9]

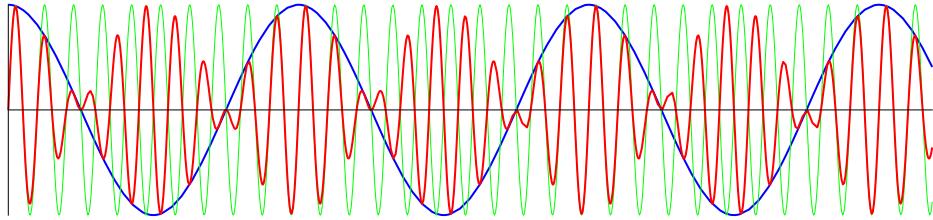


Figure 2.6: Mixer output signal

An example of a typical mixer output plot is shown in Figure 2.6. The blue trace shows the IF input signal and the green trace the LO input frequency. The red trace shows the mixed output signal. It can be seen that the output frequency is a mixture of both input frequencies.

As shown in Equation 2.9 it does not matter if the difference of the input frequencies is positive or negative. Therefore for down-conversion, there always exist two frequencies which can be applied to the input port and lead to the same frequency on the output port, one smaller than the frequency of the local oscillator and one larger. The unwanted frequency that leads to the same result is called image frequency and can be filtered by an image-reject-filter put in front of the mixer. [2]

The quality of a mixer is determined by a number of characteristics, which are conversion gain, noise figure, linearity and isolation.

Conversion gain needs to be separated into voltage conversion gain and power conversion gain. It describes the amplification of the input signal to the output signal. For the voltage conversion gain it is the ratio of the amplitudes of input and output signals. In the above example this is the RF amplitude $\frac{V_{LO} \cdot V_{IF}}{2}$ divided by the IF amplitude V_{IF} and results in $\frac{V_{LO}}{2}$. The power conversion gain is the power delivered to the load on the RF port divided by the IF source power. If the mixer's input and load impedance are equal to the source impedance, voltage and power conversion gain are the same. [2, 7]

Noise Figure (NF) describes the ratio between the input signal to noise ratio (SNR) and the output SNR. There is single-sideband (SSB) noise figure, which describes the case, when the image frequency is filtered out. The double-sideband (DSB) noise figure describes the case where both, image frequency and RF are desired. [7]

Figure 2.7 shows the non-linear behaviour of a mixer. The output power is plotted over the input power on a logarithmic scale. For small input power the mixer behaves linearly, but with increasing power, at some point it starts deviating from its linear extrapolation. The point where the difference between the linear curve and the deviation is 1 dB is called the 1 dB compression point and marks the

2 Background

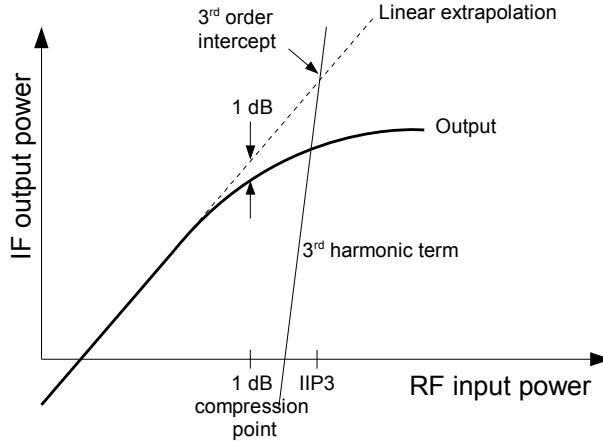


Figure 2.7: Mixer linearity parameters

power at which the mixer is deemed to have entered saturation. The thin trace represents the third harmonic distortion of the non-linear mixer. Its crossing point with the linearly extrapolated first order harmonic is called the third order intercept point. It is often used for comparison of different mixers, because it is a translatable metric. [7]

Isolation explains how much a signal port leaks to a neighbouring port, in other words a linear energy transfer between any two of the mixer ports. [9]

The Gilbert Cell mixer is an active double-sideband, four-quadrant mixer. Its circuit is shown in Figure 2.8. The LO frequency is applied to the v_{in1} input and the IF signal is applied to v_{in2} . The reason therefore is, that the higher frequency signal is more prone to parasitics. In this configuration the high frequency signal does not have to charge and discharge the gate capacitances of the cross coupled transistor pairs. It is loaded by resistors, which have only a very small parasitic capacitance.

The Gilbert Cell consists of two differential pairs and two resistive loads. The IF input transistors act as a transconductor and translate the applied voltage into a current. The transconductor can induce undesired noise into the system by imperfect switching of the transistors. Also, when both transistors are conducting at the same time, they produce noise. Additionally, because most mixers of this type have a positive gain, noise generated by the local oscillator is amplified. Therefore it is very important to have a very noise free oscillator as a reference on the LO port. Another problem are the intrinsic capacitances of the transistors, which are charged and discharged by the LC resonator. They also affect linearity of a Gilbert Cell. For designing the mixer, it is important that the amplitudes of the input signals are chosen below the 1 dB compression point, to keep the mixer in the quasi-linear region. Linearity can be improved by adding resistors to the IF input. [7]

The Gilbert Cells will be used in the frequency synthesiser for a quadrature fre-

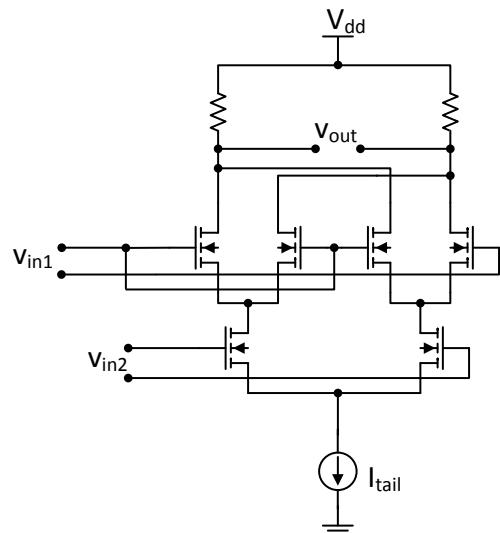


Figure 2.8: Gilbert Cell Mixer

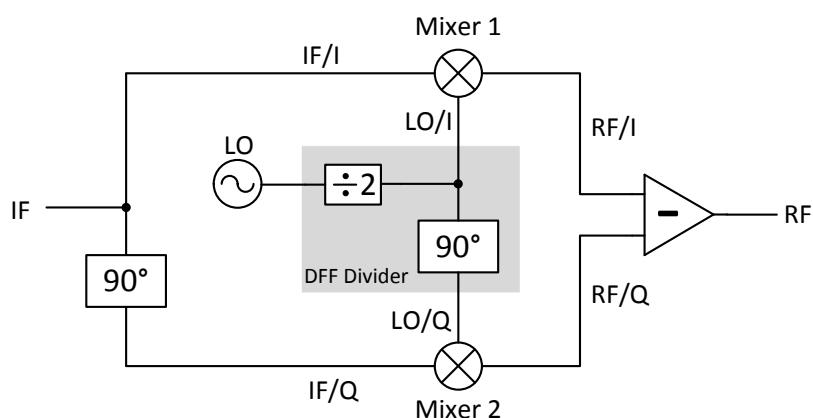


Figure 2.9: Quadrature frequency converter with two Gilbert Cells

2 Background

quency converter, which is illustrated in Figure 2.9. There, two mixers are fed on their IF input by the same signal coming from the sine wave generator, only that for one input the frequency is phase shifted by 90° . To generate these two streams, the LO frequency is divided by two by means of a D-Flip-Flop divider, illustrated in Figure 2.10. The two used D-Flip-Flops are edge sensitive to both, rising and falling edge, to act as a by-2-divider. That can be extracted from the truth-table in Table 2.1. Both streams have half the oscillator frequency and one of the streams is phase shifted by 90° . Each mixer is then fed on their LO input by one of the streams. The phase shifting by 90° is necessary to set the streams in quadrature. By using two mixers operating on the same frequencies but in quadrature, the sum or difference of the two different frequencies can easily be generated. An adder or a subtractor can be used on the generated I and Q streams from the mixers. Adding results in down-converting, subtracting in up-converting, due to equation 2.10.

Oscillator edge	D_1	D_2
1st rising	0	0
1st falling	1	0
2nd rising	1	1
2nd falling	0	1
3rd rising	0	0
.	.	.
.	.	.

Table 2.1: Truth table for the D-Flip-Flop divider with phase shifting

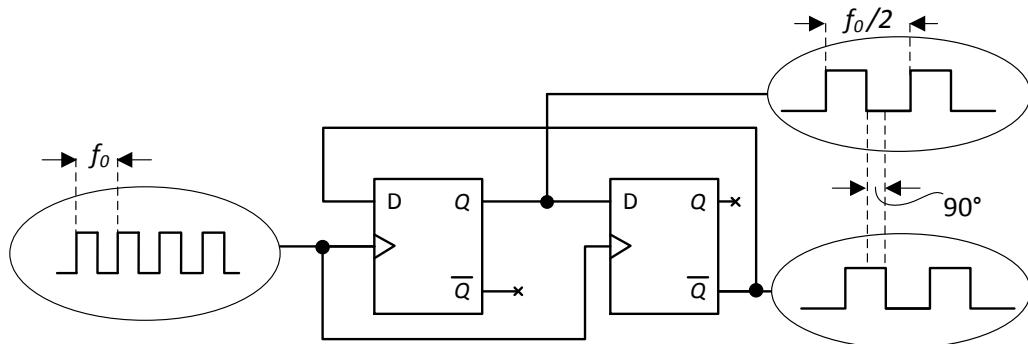


Figure 2.10: D-Flip-Flop divider with phase shifting of 90°

The output signal of the quadrature mixer for up-conversion can be calculated

as follows:

$$\begin{aligned}
 v_{RF} &= V_{IF} \cdot V_{LO} \cdot \left(\underbrace{\cos(\omega_{IF}t) \cdot \cos\left(\frac{1}{2}\omega_{LO}t\right)}_{Mixer1} - \underbrace{\sin(\omega_{IF}t) \cdot \sin\left(\frac{1}{2}\omega_{LO}t\right)}_{Mixer2} \right) \\
 Mixer1 &= \frac{1}{2} \cdot \left(\cos\left(\left(\omega_{IF} - \frac{1}{2}\omega_{LO}\right)t\right) + \cos\left(\left(\omega_{IF} + \frac{1}{2}\omega_{LO}\right)t\right) \right) \\
 Mixer2 &= \frac{1}{2} \cdot \left(\cos\left(\left(\omega_{IF} - \frac{1}{2}\omega_{LO}\right)t\right) - \cos\left(\left(\omega_{IF} + \frac{1}{2}\omega_{LO}\right)t\right) \right) \\
 v_{RF} &= V_{RF} \cdot V_{LO} \cdot \cos\left(\left(\frac{1}{2}\omega_{LO} + \omega_{RF}\right)t\right)
 \end{aligned} \tag{2.10}$$

For down-conversion, the two terms in the first equation need to be subtracted rather than added. The $\frac{1}{2}\omega_{LO}$ is due to the division of the oscillator frequency.

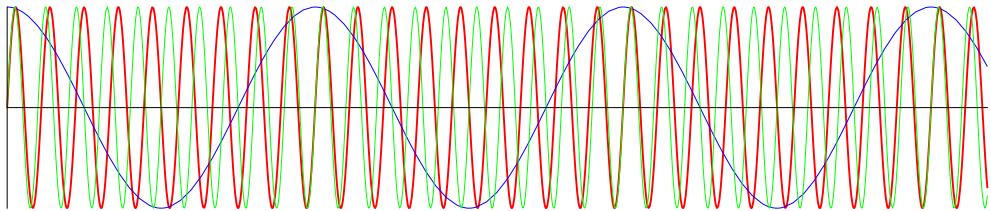


Figure 2.11: Input and output signals of a quadrature frequency converter

Figure 2.11 shows the result graphically. The green line represents the LO input, the blue line represents the IF input and the red trace shows the up-converted RF output. As can be seen, the output signal has a constant amplitude, in contrast to the output of one mixer. This is due to the 90° phase shift, which sets the signals in a relation such that the subsequent addition or subtraction of the two mixer signals results in an amplitude which is the product of the two input signals' amplitudes.

An easy way of implementing the adder or subtractor is by using load sharing, where two mixers are connected to the same load. This is illustrated in Figure 2.12.

By just inverting the signals on the IF port, it can be influenced, if the load sharing acts as up-converting or down-converting. This feature will be used in the implementation to steer the LC oscillator frequency up or down and thus produce a very accurate output.

To minimise the time where both sides of the Gilbert Cell are conducting, the inputs need to be biased. This can be done by using pull up resistors to the biasing voltage.

The LO inputs are high frequency inputs and therefore need coupling capacitors to remove the DC signal which might be present. The IF frequency will be so close to DC in this project, that coupling capacitors would not let the actual signal pass. Therefore the IF inputs don't have coupling capacitors.

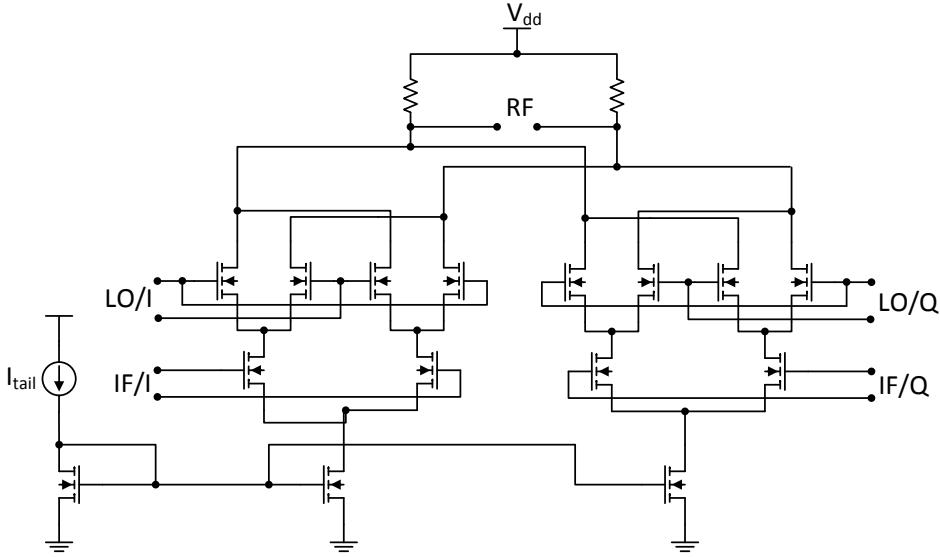


Figure 2.12: Two Gilbert Cells sharing one load for generating the sum or difference signal on the output

2.3 Phase-Locked Loops

The purpose of a phase-locked loop (PLL) is to generate a stable, jitter-free frequency from a reference via a feedback system.

This section will first describe a classical PLL, which consists of a phase detector, a low-pass filter, a voltage controlled oscillator (VCO) and in some cases a divider for the feedback path. Then three popular architecture of frequency synthesisers will be discussed, which are integer-N synthesisers, fractional-N synthesisers and dual-loop synthesisers.

A PLL generates a signal from an input frequency, which is locked to the input frequency by means of feedback. Its phase is proportional to the filtered voltage output v_χ of the phase detector. However, that does not necessarily mean, that their frequencies are the same. By using a divider in the feedback path, the output frequency can be a multiple of the input. A general overview of a PLL system with a divider is illustrated in Figure 2.13.

Figure 2.14 shows how the feedback frequency is locked to the reference over time. It is not a linear process, which is due to the feedback. Because of resolution, the output frequency might slightly oscillate around the actual value, even when locking has occurred.

The voltage controlled oscillator generates the output frequency. Its frequency can be controlled via a voltage applied to the input. In most implementations, increasing the voltage, increases the frequency generated by the VCO and lowering the voltage, decreases the frequency. The voltage is generated by a phase

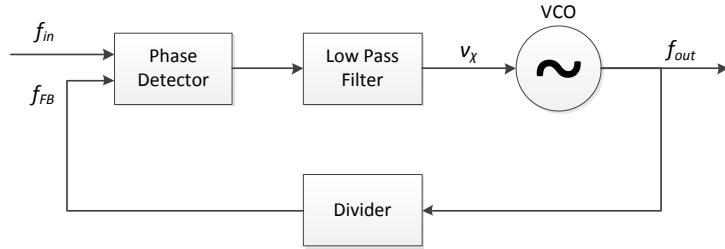


Figure 2.13: Simple Synthesiser architecture

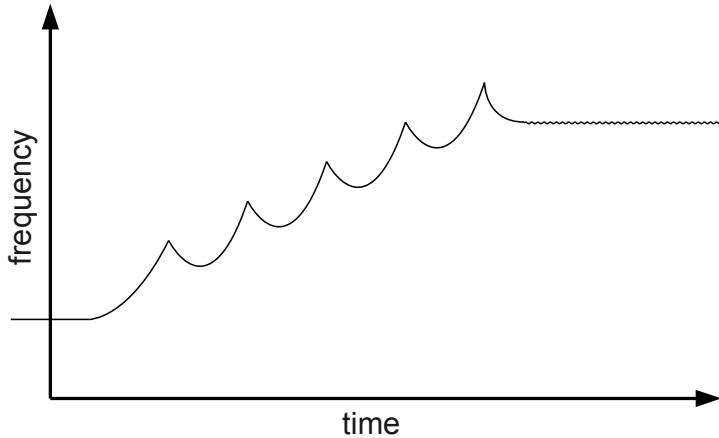


Figure 2.14: Output locking to the reference over time

detector. The phase detector can simply be an XOR-gate, if the system operates on a square wave signal. It compares the phase of the input signal and the fed back output signal, if applicable divided by a certain factor, and generates a voltage accordingly which is output. The voltage is linearly proportional to the difference of the input phases. A low-pass filter removes undesired high frequency components in the signal. [2]

An integer-N synthesiser is shown in Figure 2.15. Other than the basic model above, this synthesiser uses a charge pump to deliver charge to the filter. The phase detector basically controls two switches, which either charge the filter capacitance up to a higher value or discharge it to a lower value. The amount of charge that is delivered to or drawn from the capacitor is proportional to the frequency difference of the input signal and the feedback signal. The notable feature of this architecture is, that the divider in the feedback path is digitally controllable. However, with this implementation the frequency can only be divided by integers. The division ratio has to be fairly large, to lock the VCO to the input, which results in high power consumption of the divider and higher phase noise on the output. [1]

2 Background

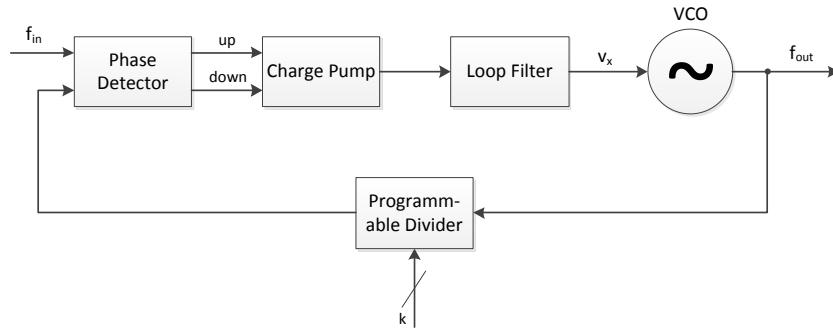


Figure 2.15: Integer-N synthesiser architecture

In contrast, the fractional-N synthesiser overcomes those problems by being able to divide by fractions. It uses an accumulator controlled dual modulus programmable divider, which is incremented each cycle. Every time the accumulator overflows, the divider changes its division ratio. The ratio is still an integer, but when calculating the average division, the result is a fraction. [1]

The fractional-N synthesiser can operate on larger reference frequencies than the integer-N. Therefore the divisor can be smaller and there is less phase noise present in the system. The accumulator consumes however a higher current, because it operates on a higher frequency. To push the noise out of the loop bandwidth to higher frequencies, a $\Delta\Sigma$ -Modulator can be used. The higher its order, the higher the noise suppression but the stability of the system decreases. [1]

The dual-loop synthesiser, as illustrated in Figure 2.16. The reference frequency f_{ref1} is a fixed high frequency source and generates the carrier which is modulated together with the signal generated from the system input frequency f_{ref2} . By varying the divisor M , the frequency which is modulated on the output of the system, is varied as well. Because f_{ref1} is much higher than the channel spacing, the phase noise of the upper PLL's VCO is outside the loop bandwidth. The phase noise of the lower PLL's VCO is small anyway, because f_{ref2} is a low frequency signal.

The difficulties in this implementation are, that the quadrature signal generation inside the quadrature VCOs has to be very accurate. [2]

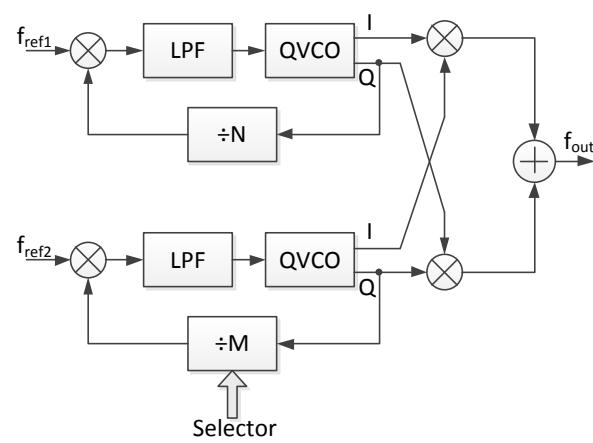


Figure 2.16: Dual-loop synthesiser comprising two PLLs and a SSB mixer

2 Background

3 State of the Art

This chapter's purpose is to demonstrate recent research conducted in the field of this project's interest. Components such as LC oscillators, quadrature frequency mixers and frequency synthesisers published in recent papers are reviewed and their suitability for the design of the frequency synthesiser and its blocks of this project is examined.

3.1 LC Oscillators

This section will introduce state-of-the-art LC oscillators. Common configurations, such as the Colpitts and the Hartley oscillator, were already discussed in chapter 2.1.

The Colpitts oscillator, illustrated in Figure 2.4a, has a fairly simple topology and can be used in circuits, where harmonic distortion is not a critical issue. [10]

The frequency of a Colpitts oscillator is given by $\omega_0 = \frac{1}{\sqrt{LC}}$, where C is determined by the capacitors in series. The result is then

$$\omega_0 = \frac{1}{\sqrt{L \frac{C_1 C_2}{C_1 + C_2}}}. \quad (3.1)$$

To provide sufficient gain, the condition for oscillation is $G_m \geq 4/R$. [10]

Another approach are the cross-coupled configurations. There is a NMOS-only version, which uses two transistors and a NMOS-PMOS version, which uses four transistors.

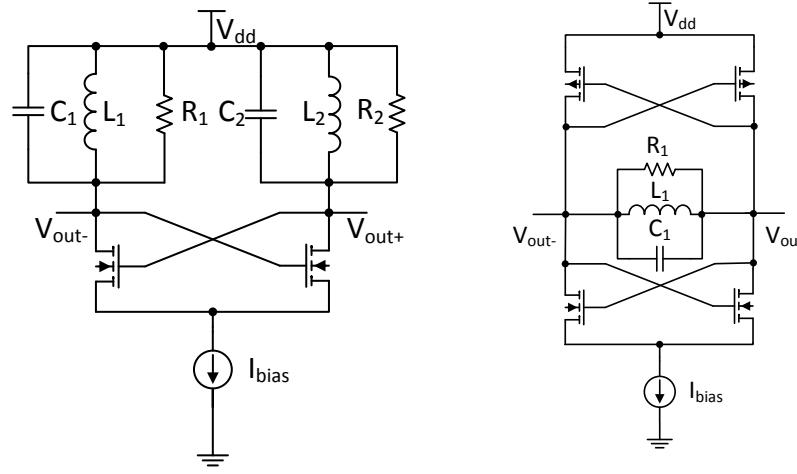
The cross-coupled NMOS oscillator, which is illustrated in Figure 3.1a, needs two inductors. They are usually on-chip and have a low Q. The active circuit is a differential pair fed by a constant current source I_{bias} . The resistors R_1 and R_2 represent the parasitics of the inductors L_1 and L_2 respectively. The capacitors C_1 and C_2 contain the load as well as the parasitic capacitances of the chip. The capacitors could also be replaced by a varactor, to facilitate frequency tuning.[10]

In case the devices of the circuit are symmetrically sized, the equivalent resistance of the two transistors is given by $G_m = -2/g_m$, where g_m is the transconductance of one transistor. To build up the oscillation of the circuit, the parasitic inductor resistance and the resistance of the transistor added together must be negative, which means that $g_m \geq 1/R$, where $R = R_1 = R_2$. [10]

3 State of the Art

Comparison with the G_m of the Colpitts oscillator shows, that its transistor is four times larger than the two transistors of the cross-coupled configuration together. This is a critical issue, if the inductor has a low Q and therefore a low R, because then the g_m of the transistor must be very large. Additionally, a larger area increases the parasitics. In return, the cross-coupled oscillator needs two inductors, which consume a large area. [10]

The NMOS-PMOS cross-coupled oscillator is shown in Figure 3.1b. There, two differential pairs are used, one NMOS and one PMOS. It uses again only one inductor. The advantage over the NMOS-only circuit is, that the output amplitude is twice as high, while using the same current I_{bias} . The NMOS-PMOS circuit has also a higher negative resistance which means, that a lower bias current is needed to compensate for the loss in the tank than for the NMOS-only circuit. [10]



(a) NMOS-only cross-coupled oscillator (b) NMOS-PMOS cross-coupled oscillator

Figure 3.1: LC tank oscillators with more than one active device

The critical design part of the LC oscillator is the inductor. It consumes a larger area on the chip the higher its inductance is. The goal is to keep its size as small as possible, but the higher the inductance is, the better are the oscillator characteristics. Because of its spiral shape, the inductor has many parasitics, that have to be taken into account for the design. The inductance is a function of the number of turns, the dimensions of the metal layer, the average side length of the turns and the spacing between the windings of the inductor. An equivalent circuit of a square spiral inductor, where parasitics are taken into account, is shown in Figure 3.2. Capacitors and resistors model the losses of the inductor. An on-chip inductor can be as high as around 10 nH. This might not sound as a high value, but implemented on chip, an inductor consumes a large area, because of its spiral

shape. [10]

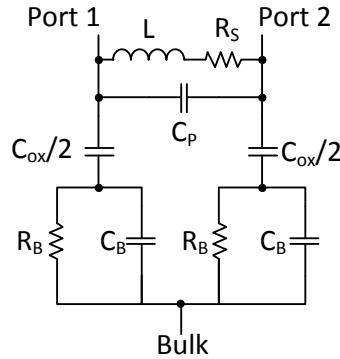


Figure 3.2: Equivalent circuit of an inductor with parasitics

Capacitors are also critical, especially for high frequency applications, such as the one in this project. Capacitances influence the Q of an oscillator, if their resistance is too high. The mostly used capacitors consist of metal-oxide-metal layers and have a fairly high resistance. Integrated-finger capacitors are therefore preferred. [10]

A parasitic compensated circuit is proposed in [11]. The phase noise is improved by 4 dB in comparison to the non-compensated circuit. The schematic is shown in Figure 3.3a. [11]

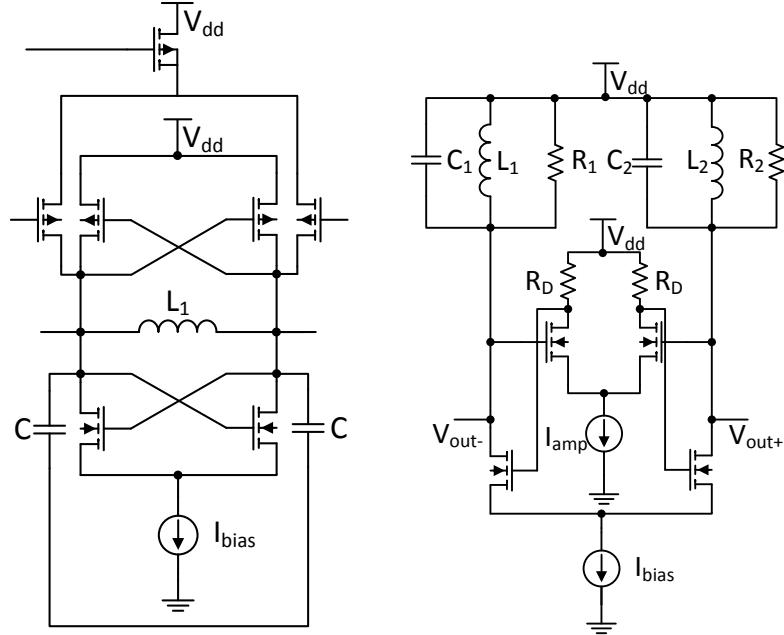
To reduce the power dissipation of an oscillator, [12] proposed a circuit with a differential amplifier, to increase the negative resistance of the active circuit. That technique minimises the current, necessary to start the oscillation. However, the current consumption of the amplifier has to be taken into account and an evaluation of the actual saved current is necessary to verify the profitability. The schematic is illustrated in Figure 3.3b. [12]

In [13], a modified cross-coupled circuit is proposed, where an NMOS transistor is added on each side between the inductor and the NMOS cross-coupled pair, with their gates connected to V_{dd} . These transistors provide the circuit with a sufficient high transconductance to start the oscillation. The power that is consumed by the circuit is thereby minimised. [13]

A differential Hartley oscillator in 180 nm CMOS technology is proposed in [14]. They use a transformer for self-biasing, two more inductors, a varactor and two inverters for their topology. [14]

Table 3.1 shows a comparison of the above mentioned oscillator designs and their characteristics.

The cross-coupled LC oscillator is known to be very stable over PVT variations compared to other common analogue architectures. It also has the advantage that



(a) One half of the parasitic compensated quadrature oscillator proposed in [11]
(b) Oscillator with amplifier for low power proposed in [12]

Figure 3.3: State of the art oscillator designs

it only needs one inductor, which significantly saves area on the chip compared to architectures using more inductors. For these reasons, this architecture is chosen for the oscillator implemented in this project.

3.2 Gilbert Cell Mixers

In this section, several mixers will be compared by means of their performance characteristics.

A 2 GHz quadrature down-conversion mixer is introduced by [15]. Its schematic is shown in Figure 3.4a. The circuit has a good LO-IF port isolation of -15.65 dB

Ref.	f_0	Phase Noise	Δf	Power
[11]	2.63 GHz	-112.3 dBc/Hz	600 kHz	7.5 mW
[12]	1.00 GHz	-40.0 dBc/Hz	1 kHz	0.8 mW
[13]	12.50 GHz	-106.0 dBc/Hz	400 kHz	0.1 mW
[14]	5.60 GHz	-123.6 dBc/Hz	1 MHz	6.5 mW

Table 3.1: Comparison of various LC oscillator designs

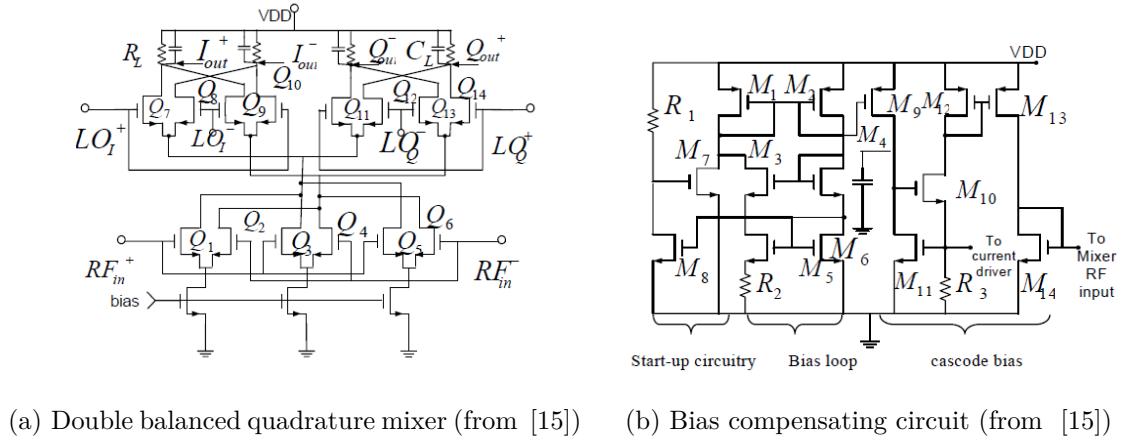


Figure 3.4: Circuit introduced by [15]

and low substrate noise. Q1 to Q6 of Figure 3.4a form the driver stage and amplify the input signal to reduce switching noise. The input differential pairs have a linear transconductance over a small range of input voltages. Therefore the overall transconductance is constant over a certain input range. A bias circuit, illustrated in Figure 3.4b, is attached to the quadrature mixer, to compensate for temperature and supply voltage variations and to make the amplification of the mixer stable. [15]

Table 3.2 shows the characteristics of several mixers in comparison.

Ref.	f _{RF}	f _{LO}	NF	Δf	Conv. Gain	IIP3	Power
[15]	2.120 GHz	2.119 GHz	8.96 dB	1 MHz	20.55 dB	-0.54 dBm	6.0 mW
[16]	2.450 GHz	2.449 GHz	16 dB	1 MHz	13.36 dB	-1 dBm	14.5 mW
[17]	900 MHz	910 MHz	n.a.	n.a.	-9.1 dB	10 dBm	3 mW
[18]	2.4 GHz	2.3 GHz	n.a.	n.a.	6.7 dB	-7.5 dBm	11.9 mW

Table 3.2: Comparison of different mixers

Gilbert Cells, introduced in recent papers, are mostly used as down-converters, where the LO and RF input signals are in a similar order of magnitude. For this project, the Gilbert Cell will be used for both up- and down-conversion to compensate for inaccuracies in the reference on the input of the frequency synthesiser and in the LC oscillator. That way, a very clean and noise free signal is generated on the output. As the inaccuracies are expected to be not more than a couple of hundreds of kHz, the IF input range is very small. It also means, that the RF output frequency is very close to the LO frequency, which means it is very high.

3.3 Frequency Synthesisers

This section will give some brief examples of state-of-the-art frequency synthesisers. The given examples will at a later stage be used to evaluate the frequency synthesiser developed in this project. Typical characteristics will be compared to show the added value of the new development.

[19] introduces an architecture which has low noise and consumes a small area on chip. It is configured for a low frequency reference, however can be extended to operations in the RF area.

The analogue PLL uses an active loop filter and can therefore omit the charge pump. Two large resistors are used to emulate a small current normally provided by a charge-pump. The disadvantage is, that the high resistance is prone to thermal noise and thus effects the phase noise of the whole circuit. Therefore two switches are added in series with the high resistances (see Figure 3.5), which close only short before a pulse of the phase detector arrives and open short after that.

If that switching window is too small, the loop becomes unstable because of the parasitic capacitances of the resistors, where charge is stored and flows into the op-amp, which results in ringing. If the window is long enough, the charge can flow back to the phase detector.

The windowing technique achieves a significant noise reduction of 15 dB compared to the conventional architecture. [19]

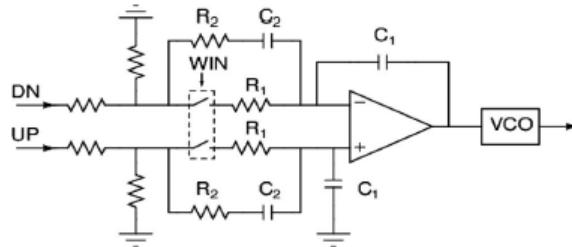


Figure 3.5: Circuit proposed by [19] (from [19])

In the paper [20], a dual-loop frequency synthesiser is proposed which uses a multi-phase VCO instead of the conventional VCO. The overall circuit is similar to the one shown previously in Figure 2.16. The multi-phase VCO provides signals of four different phases and thus has four outputs, which are fed into a multi-phase divider. The circuit has a very good frequency resolution in comparison to conventional dual-loop synthesisers. Compared to a conventional $\Delta\Sigma$ -modulated fractional-N synthesiser, the phase noise is also much better.

The design was implemented on a FPGA and measurements were carried out and compared to results of a conventional dual-loop frequency synthesiser. By using the multi-phase VCO, either the loop bandwidth could be four times in-

creased or a high frequency resolution could be achieved, according to the desired application. [20]

In Yang et al. a mixed-signal frequency synthesiser is proposed, which operates at around 9 GHz. Their goal was to design a low noise and low power system with a high frequency output. To achieve this, a quadrature VCO, consisting of two LC oscillators, with frequency doubling is used. For the frequency doubler, they use a transformer. The block diagram is illustrated in Figure 3.6. [21]

The measurements gave a frequency output range of 8.8 GHz to 9.2 GHz with a reference clock in the range of tenth of MHz. The measured phase noise at 1 MHz offset is -104.5 dBc/Hz for a 9.1 GHz output frequency. The synthesiser consumes 12 mW of power. [21]

Table 3.3 shows a comparison of several state of the art frequency synthesisers.

Ref.	Technology	Frequency	Power	Phase Noise @ 1 MHz
[21]	130 nm	8.8 – 9.2 GHz	12 mW	-104 dBc/Hz
[22]	180 nm	1.06 – 1.4 GHz	4.9 mW	-121 dBc/Hz
[23]	180 nm	1.9 – 2.1 GHz	4.5 mW	-120 dBc/Hz
[24]	90 nm	2.4 – 2.6 GHz	5.5 mW	-113 dBc/Hz

Table 3.3: Comparison of different frequency synthesisers

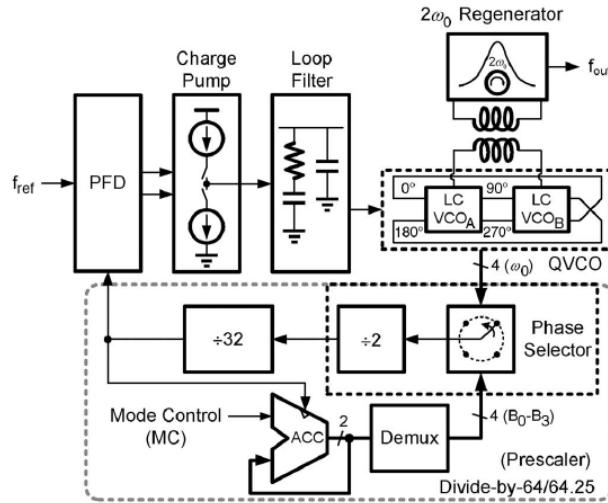


Figure 3.6: Frequency Synthesiser proposed by [21] (from [21])

The architectures introduced in this chapter and Chapter 2.3 have proven useful for many applications, but do not satisfy the criteria for the desired application of

3 State of the Art

this project's design. Usually, PLLs produce a frequency on their output, which is variable in a fairly large range, i.e. several hundreds of MHz. The frequency synthesiser of this project will be used for cleaning clock signals impaired by jitter and therefore has to produce a very stable and noise-free output of 2.5 GHz. The mixer mainly compensates for inaccuracies in the LC oscillator, which might have a magnitude of only several hundreds of Hz. For that reason, existing PLL architectures are not suitable for the design.

To implement such a synthesiser, established components, such as the Gilbert Cell and the cross-coupled LC oscillator, can be used to form a novel system. This system and its specification will be further described in the next chapter.

4 System Specification and Design

This chapter will introduce the specification given by Xintronix Ltd and describe the design of all the components. The VerilogA code of the testbench blocks will be explained as well as the transistor-level design of the LC oscillator and the Gilbert Cell. The schematics will be shown and mathematically calculated geometries will be compared to geometries, which lead to the desired function of the circuit. Where difficulties occurred in the design, reasons and possible solutions will be analysed. It will furthermore show, in which order the designing was conducted and how the project progressed over time.

4.1 Specification and Design Introduction

The frequency synthesiser system was already shown in the introduction. Figure 4.1 illustrates again a detailed view with the signals which are expected on the interfaces of the blocks.

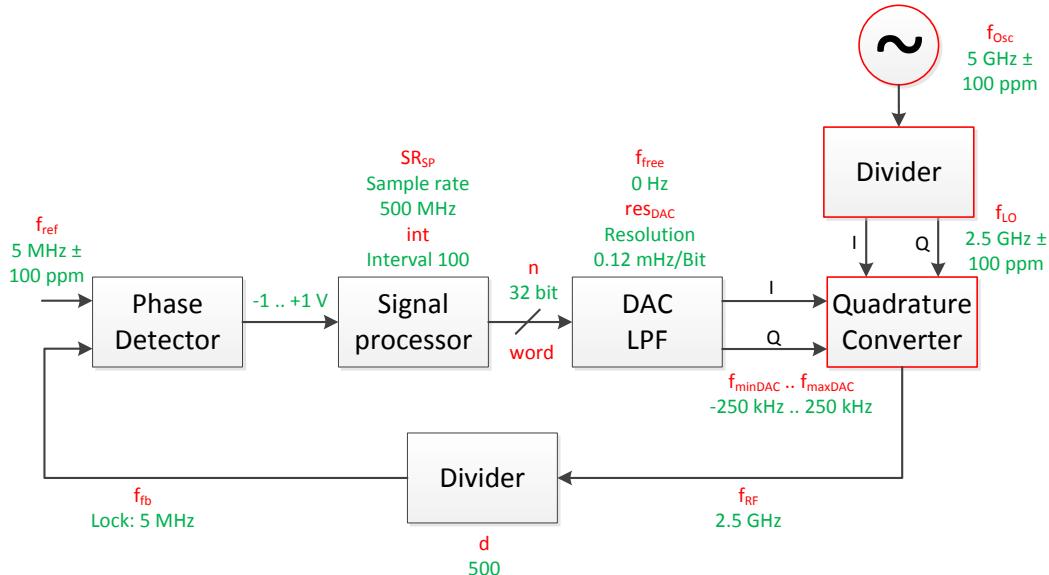


Figure 4.1: Detailed system view with expected signals

The input frequency for testing was chosen to be 5 MHz. The output of the frequency synthesiser should be 2.5 GHz. A 5 GHz oscillator, which in reality will be given by another PLL with an accuracy of ± 100 ppm ($= 0.01\%$), provides the

4 System Specification and Design

basic output frequency. Therefore it is divided by two and mixed with a correction frequency which ranges from -250 kHz to 250 kHz with a 1 mHz resolution, to balance the 100 ppm deviation, which corresponds to 250 kHz for the 2.5 GHz signal.

From Xintronix's specification the lock time of the system, after which the output matches the reference, can be very high, up to 1 minute.

The system will operate with a maximum supply voltage of $1.8 \text{ V} \pm 10\%$. The current available on the chip is $10 \mu\text{A} \pm \sqrt{2}$, which gives a range of $8.49 \mu\text{A} \dots 11.51 \mu\text{A}$. The temperature range in which all measurements were taken is from -40 °C to 125 °C.

The specification is again summarised in Table 4.1.

	min	typ	max	
V_{dd}	1.62	1.8	1.98	V
I_{ss}	8.49	10	11.51	μA
f_{LO}	2.499925	2.500000	2.500075	GHz
f_{IF}	$1 \cdot 10^{-3}$		$250 \cdot 10^3$	Hz
f_{RF}		2.5		GHz
T	-40	27	125	°C
t_{lock}			60	s

Table 4.1: General specification

The specification mainly describes the overall parameters and interfaces between blocks. It leaves room for deciding on some values inside each block. These will be further described and their chosen definition justified in the following sections.

At the beginning of the design phase, a testbench was created, which contained simple models for all the necessary components written in VerilogA. The testbench was modified until a lock time was reached within a reasonable simulation time. That required using parameters which don't follow the specification and would not make sense in a real design but show that the system is working.

The next step was to make the models more realistic by including noise to the most noise critical parts. That is on the one hand the input reference and on the other hand the 5 GHz oscillator.

The above explained parts were done in cooperation with the complementary project done by Shruti Gudi. Experiences were shared and the behaviour of certain parts was discussed. However, the implementation was done separately and the final designs differed in several parts, which will be covered more deeply in the next section, when alternative implementations are described.

The subsequent stages were undertaken completely independent of one another's projects.

For this project, the oscillator was designed on transistor level. The architecture had been chosen by Xintronix to be a cross-coupled LC design, as shown in Chapter 3.1. The geometries were calculated and fine tuned until the best results in terms of PVT variations and phase noise were achieved.

Then, the Gilbert Cell mixer was designed. This was the most time consuming part, as many factors influence the output and the mixer finally defines the output signal and thus noise. Not only the geometries had to be chosen correctly, but also the exact input signals had to be defined to achieve the desired output.

Two Gilbert Cells were later connected together, to form the quadrature frequency converter. That required redesigning the transistor geometries, load resistors and input levels, to optimise the output.

Finally, the divider for the LC oscillator was designed in VerilogA and the frequency converter could be connected to the oscillator. This system again needed redesigning of single parts, mainly of the mixer, for optimisation purposes.

The environment for the development of both, VerilogA models and transistor level schematics, is Cadence Virtuoso. It is used for designing and simulations of several kinds. For components such as resistors, capacitors and voltage or current sources, models from the ideal analogue library of Cadence were chosen. The transistors and the inductance for the LC oscillator were taken from the gpdk180nm library. This is a generic library, which models realistic behaviour but without having underlying parameters extracted from real measurements of fabricated circuits.

4.2 Testbench Design

The testbench includes VerilogA models of the components. Firstly, all parts were modelled and later, specific parts were replaced by real circuits.

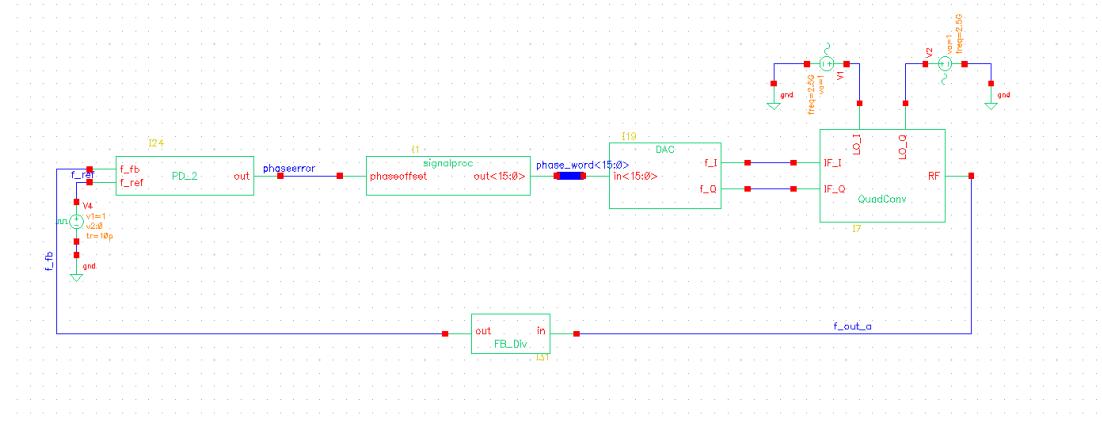


Figure 4.2: Testbench schematic

All oscillators used in the testbench, including the input reference and the LC oscillator, are in the beginning replaced by ideal voltage sources from the analogue

4 System Specification and Design

library of Cadence. The remaining components, which are modelled in VerilogA are the phase frequency detector, the signal processor, the digital-to-analogue converter – later a $\Delta\Sigma$ modulator with a low pass filter – furthermore the quadrature frequency converter and the feedback divider. The code for these models can be found in the Appendix.

The phase detector is implemented in a way, that the output is positive if the reference phase is ahead of the feedback and negative if the reference phase is behind the feedback. It compares the occurrence of rising edges of both signals and infers from that the phase relation of the two frequencies. The change of the pulse width over time gives information on the frequency relation of the two signals. The output has three states, positive, neutral and negative.

Alternatively, the phase detector could be implemented as a subtractor, which forms the difference of the reference and the feedback signal and has also three output levels. Or it could be an XOR, which outputs a high level if the signals have different levels and a low level, if the signals have the same level. Another approach would be an analogue multiplier with again three output levels. These implementations would also require a different signal processor design.

The signal processor samples the phase detector's output signal at certain time steps, the frequency of which was chosen to be 500 MHz. During the sampling the average over the input samples is calculated over a defined range, which was chosen to be 100. That average value is then compared to the previous average, which was stored internally, and according to whether the difference is positive or negative, the output 32-bit digital value is increased or decreased. The digital value is equivalent to the control voltage of a VCO. Its absolute value controls the frequency that is produced by the $\Delta\Sigma$ -DAC.

To translate the signals from a subtractor-, multiplier- or XOR-phase detector, the signal processor has to be designed in a slightly different way. It would again sample its input and take the average over a certain time interval. The average should ideally be $\frac{\text{highlevel} - \text{lowlevel}}{2}$. Depending on the deviation from that ideal average, the signal processor increases or decreases the digital word. In lock, the reference and the feedback frequency would therefore be in quadrature, as that relation forces the output error of the phase detector to have equally distributed high and low times and therefore the ideal average.

The DAC model was not implemented as a $\Delta\Sigma$ -modulator because that task forms part of the complementary project. In the simplified model, the DAC samples its input, which is the digital word coming from the signal processor, at a frequency of 5 MHz. That gives the signal processor enough time for producing an updated value at its output, as it runs at 500 MHz but in intervals of 100, which leaves an update rate of 5 MHz, according to Equation 4.8. The DAC has a free running frequency which can be arbitrarily chosen. It forms the initial value for the IF mixer input and therefore determines, what the initial feedback frequency

on system start-up is. In the testbench, it is chosen to be 0 Hz, as ideally the LO frequency is exactly 2.5 GHz. The word is set to be 0 on system start-up.

The quadrature converter performs the ideal mathematical function behind a mixer architecture by multiplying the voltages of the two input frequencies at each iteration. The equation behind the model is:

$$V_{out} = (V(f_{LO}) \cdot V(f_{IF}))_I - (V(f_{LO}) \cdot V(f_{IF}))_Q \quad (4.1)$$

The divider counts the rising edges on its input and depending on the divisor, which can be set as a parameter, outputs a rising or falling edge for the divided signal.

The following equations show, how the parameters of the system shown in Figure 4.1 depend on each other:

$$goal : f_{fb} = f_{ref} \quad (4.2)$$

$$f_{fb} = \frac{f_{RF}}{d} \quad (4.3)$$

$$f_{RF} = f_{LO} + f_{IF} \quad (4.4)$$

$$f_{IF} = f_{free} \pm res_{DAC} \cdot word \quad (4.5)$$

$$res_{DAC} = \frac{f_{maxDAC} - f_{minDAC}}{\frac{2^n}{2}} \quad (4.6)$$

$$word = \left[\frac{avg}{int} \cdot 2^n \right]_{current} - \left[\frac{avg}{int} \cdot 2^n \right]_{previous} \quad (4.7)$$

$$SR_{DAC} = \frac{SR_{sp}}{int} \quad (4.8)$$

$$\Rightarrow f_{fb}(word) = \frac{1}{d} \cdot \left(f_{LO} + f_{free} \pm \frac{f_{maxDAC} - f_{minDAC}}{\frac{2^n}{2}} \cdot word \right) \quad (4.9)$$

where f_{ref} is the frequency of the input signal of the PLL, f_{fb} is the output frequency of the PLL after division in the feedback divider. f_{RF} is the frequency of the output signal of the quadrature frequency converter and therefore the output signal of the PLL. d is the divisor of the feedback divider and f_{LO} is the frequency coming from the LC oscillator and being divided by two. f_{IF} is the frequency of the signal generated by the $\Delta\Sigma$ digital to analogue converter which also has a free-running frequency f_{free} . res_{DAC} represents the resolution of the DAC in Hz/bit. The parameter $word$ stands for the digital word generated by the signal processor, which controls the output frequency of the DAC. n is the number of bits of that word. avg is the average input over an interval int sampled at a rate of SR_{sp} in the signal processor. SR_{DAC} represents the sample rate at which the DAC samples its input.

4 System Specification and Design

Simulating the system with the actual values is not possible on the university machines, because the locking takes fairly long. The simulation of a period of 25 μ s takes approximately 40 minutes in real time and produces amounts of data that can't be handled by the PCs. However, to show that the system is still working, the above explained specification was modified so that locking can be reached within around 3 μ s simulation time. The results are shown in Figure 4.3. Figure a) shows the overall locking procedure and Figure b) shows a detailed view of the locking which happens within the first 3 μ s. After that time, the feedback frequency oscillates around the actual value. The oscillation shown in the graph is very high. That is due to the chosen resolution of the DAC, which is set very low in this example, to show the circuit is working. The detailed view shows, how the desired frequency is stepwise approached. Comparing the graph to Figure 2.14 makes clear that the result is similar to the expected outcome.

The values which were taken for the example in the graph are summarised in Table 4.2

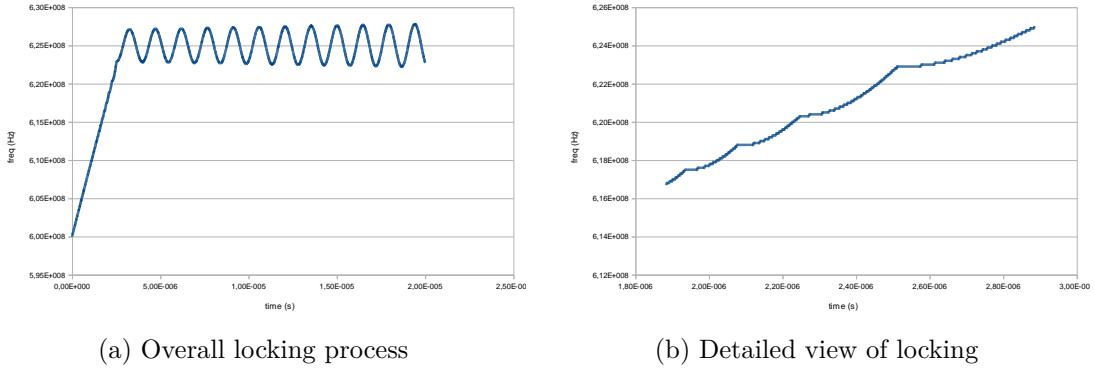


Figure 4.3: Locking of the PLL with high values to show operation

f_{ref}	625	MHz
SR_{sp}	10	GHz
int	16	
n	16	bit
SR_{DAC}	625	MHz
f_{free}	80	MHz
f_{minDAC}	140	MHz
f_{maxDAC}	20	MHz
res_{DAC}	1831	Hz/bit
f_{LO}	2.4	GHz
d	4	

Table 4.2: Parameters for the exemplary locking shown in Figure 4.3

4.3 LC Oscillator Design

This section will discuss the architecture of the chosen LC oscillator in detail and explain the geometries of the components mathematically as well as by means of simulations. Two approaches were made in terms of the architecture which will be compared.

As mentioned in previous sections, the oscillator was specified to be of a LC type, which means that the oscillation is caused by a resonator consisting of an inductor (L) and a capacitor (C). The commonly used architecture for that type is a cross-coupled NMOS-PMOS architecture. The original version shown in Figure 4.4a was designed and compared to the design in Figure 4.4b, which has two additional transistors. The differences in the results will be discussed in Chapter 5.

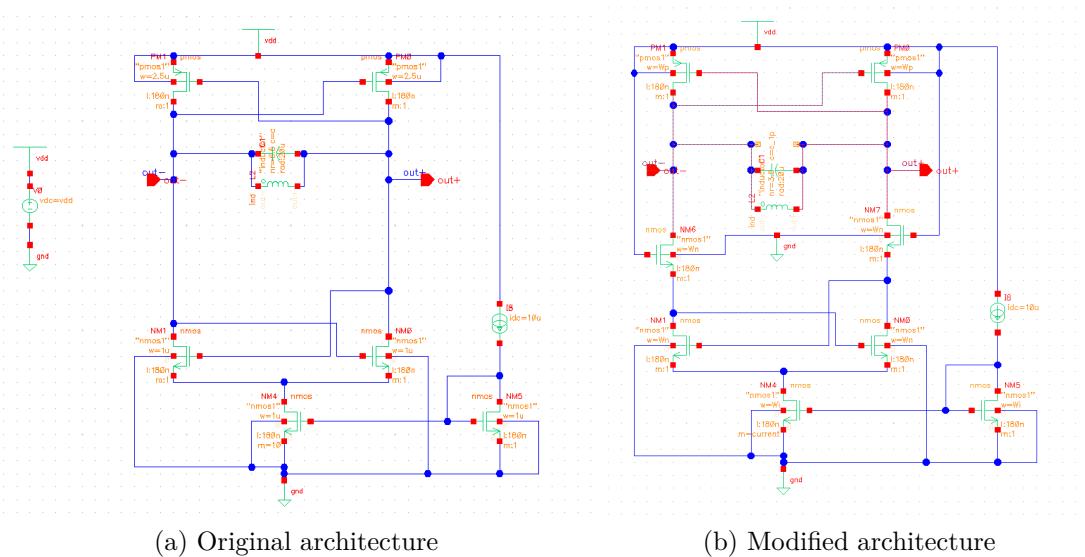


Figure 4.4: Schematics of the LC oscillator

The transistors used for the design as well as all further designs were taken from the gpdk180nm library provided by Cadence as a generic library. Process variation corner cases are available in that library for the MOSFETs, which were used in the PVT variation analysis and will be further explained in the next chapter. The inductor was also taken from that library, but process corner cases are not available. It is a spiral inductor with a square shape, which is shown in Figure 4.5. The inductance is determined by setting the inner radius R, the inductor width W, the space S and the number of turns N and can be calculated as in Equation 4.10.

$$L \approx \frac{1}{25.4} \cdot \frac{N^2 \cdot \left(\frac{2R+N(W+S)}{2}\right)^2}{30 \cdot \frac{2R+N(W+S)}{2} - 22R} \quad (4.10)$$

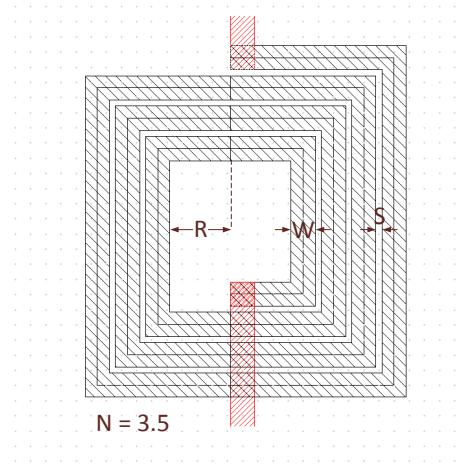


Figure 4.5: Layout view of the inductor from the gpdk180nm library

Two different inductor values were taken to show how they influence phase noise and PVT variations. For both values, the inner radius is 20 μm , the width of the inductor 8 μm and the spacing 2 μm . With 3.5 turns, the inductance is 1.13 nH and with 6.5 turns it is 4.06 nH according to Equation 4.10.

The capacitor is an ideal model from the analogLib standard Cadence library.

The output frequency of the oscillator is determined by the geometries of the inductor and the capacitor as follows:

$$f_0 = \frac{1}{2\pi\sqrt{LC}} \quad (4.11)$$

The desired frequency of the oscillator is 5 GHz. That leaves a capacitance of 897 fF for an inductance of 1.13 nH and a capacitance of 250 fF for an inductance of 4.06 nH according to Equation 4.11. Due to parasitics, these values need to be adjusted by means of simulations. The original version (Figure 4.4a) needs a capacitance of 241.2 fF for an inductance of 4.06 nH and outputs a frequency of 5.0002 GHz. For the version with the extra transistors (Figure 4.4b) and $L = 1.13$ nH a capacitance of 889.7 fF produces an output frequency of 4.99999 GHz, for $L = 4.06$ nH a capacitance of 240.7 fF produces 5.0001 GHz.

Figure 4.6 summarises the dimensions for all used components of the extended architecture. The original architecture uses the same dimensions for its components except for the capacitance, which is chosen as mentioned above.

The output of the oscillator with the extra transistors is shown in Figure 4.7. There, the blue trace shows the actual output and the red trace shows the ideal output given by

$$V_{out} = \hat{V} \cdot \sin(2\pi \cdot 5 \text{ GHz} \cdot t). \quad (4.12)$$

The real and ideal trace have the same frequency but are phase shifted, which is not important, as it does not affect the functionality.

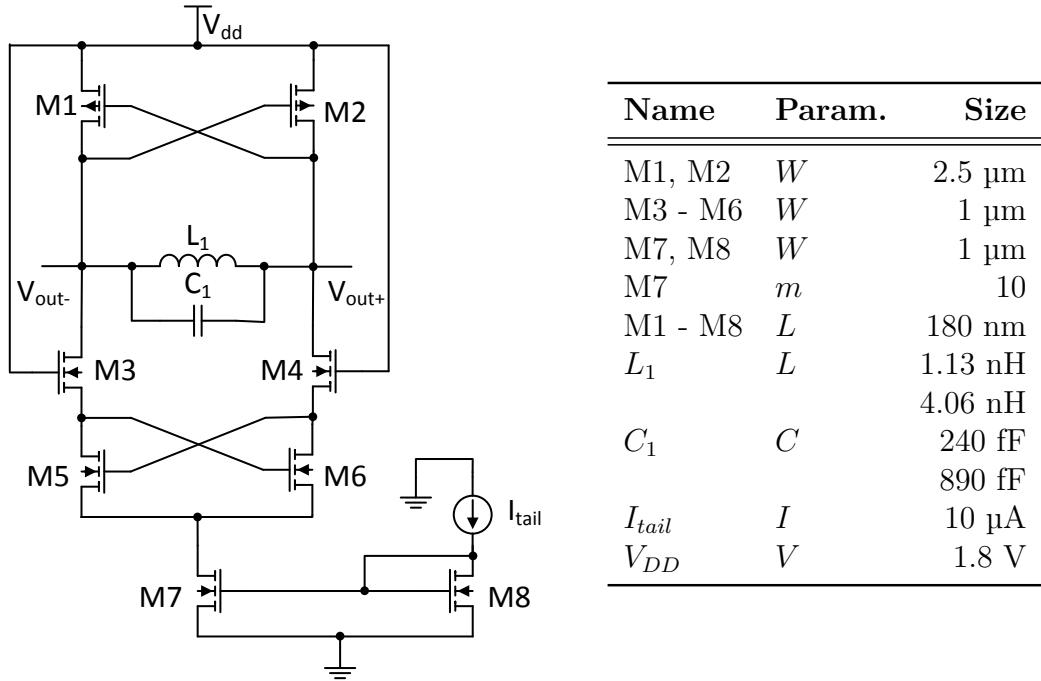


Figure 4.6: Oscillator dimensions

The oscillator needs an impulse at the beginning on one node of the differential output to induce the energy which will oscillate from one node to the other. Without that impulse, there is no oscillation. For the shown simulations, a 1 V impulse was used in the beginning. This is done by a special feature called "initial condition" in the analogue design environment (ADE) in Cadence used for simulations.

The stabilisation time for the LC oscillator with $L = 4.06 \text{ nH}$ where the full amplitude is reached is approximately 60 ns, as shown in the graph. For $L = 1.13 \text{ nH}$ the time to stabilisation is approximately 300 ns. That is due to the higher capacitance used in the LC tank.

For the original architecture, the time is around 20 ns. The reason for that being shorter than the stabilisation time of the extended architecture is, that using less transistors means having less parasitic capacitances and therefore a faster stabilisation.

The W/L ratio of the transistors can be determined approximately, by using the equation for current flowing through a transistor. All transistors should be in saturation for correct operation so that Equation 4.13 is valid.

$$I_D = \frac{k' W}{2 L} V_{ov}^2 \quad (4.13)$$

Usually, a V_{ov} of 0.2 V is desired to still be in the saturation region of the transistor.

4 System Specification and Design

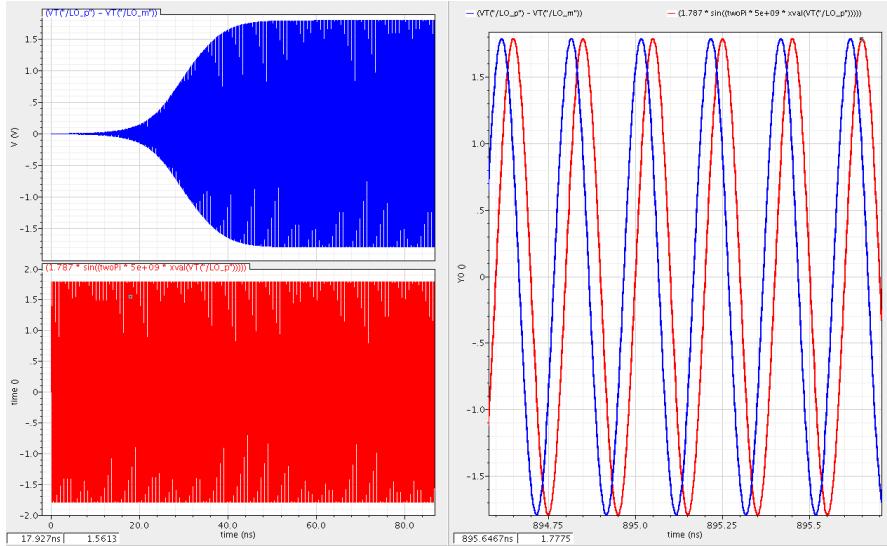


Figure 4.7: Output of oscillator and stabilisation time; blue: real, red: ideal

k' for the NMOS is here approximated by $100 \mu\text{A}/\text{V}^2$. The drain current of the transistors is $100 \mu\text{A}$. That leaves a W/L ratio of 50 for NMOS and a width of $9 \mu\text{m}$ for a length of 180 nm. To lower the parasitic capacitance which slows down the switching speed of the transistors and extends the time needed for stabilisation of the oscillator, the width was chosen to be only $1 \mu\text{m}$ for the NMOS transistors and 2.5 the size for the PMOS. A smaller W/L ratio has also a positive effect on PVT variations, which will be further explained in Chapter 5.

4.4 Gilbert Cell and Frequency Converter Design

In this section the design of the Gilbert Cell mixer and the frequency converter built from that will be explained. The ideal mathematical calculation of the geometries is illustrated and deviations from that in the actual design will be explained. The output waveforms will be shown and discussed and the necessary inputs for the mixers to produce a reasonable output are mentioned.

The netlist driven schematic for the quadrature frequency converter in a test-bench, which was created in Cadence Virtuoso, is shown in Figure 4.8. The schematic of the upper mixer "GC" is shown in Figure 4.9. The lower mixer "GC2" contains a similar circuit with the same dimensions but without the resistors to the supply voltage. The reason for that is, that load-sharing is used for subtraction of the I and Q signals in the frequency converter as explained in Chapter 2.2 and previously shown in Figure 2.12. With that design the outputs of the mixer cells "GC" and "GC2" can be directly connected and form the desired overall circuit.

The dimensions are listed in Figure 4.10. All transistors have the minimum

4.4 Gilbert Cell and Frequency Converter Design

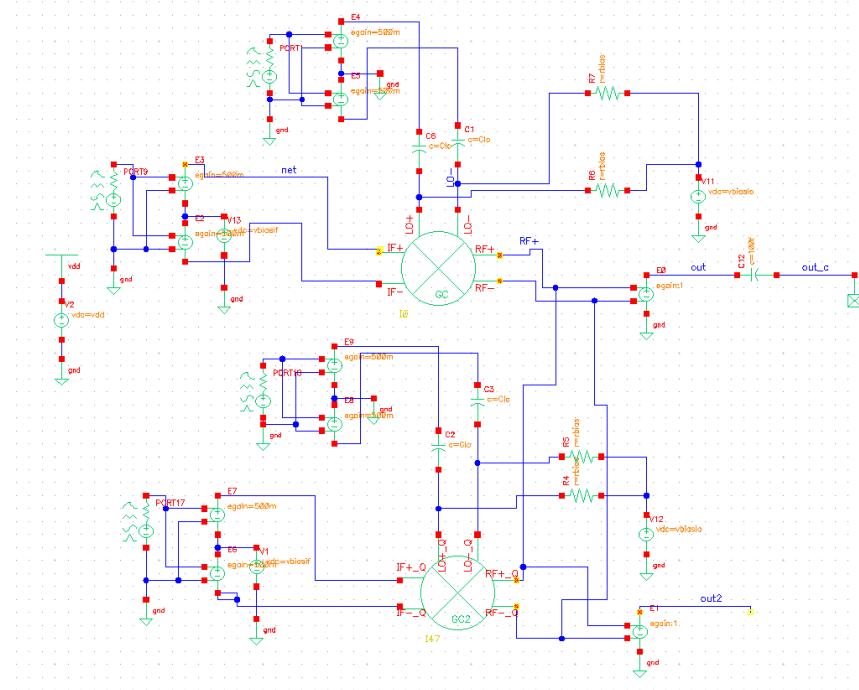


Figure 4.8: Schematic of the testbench for the frequency converter

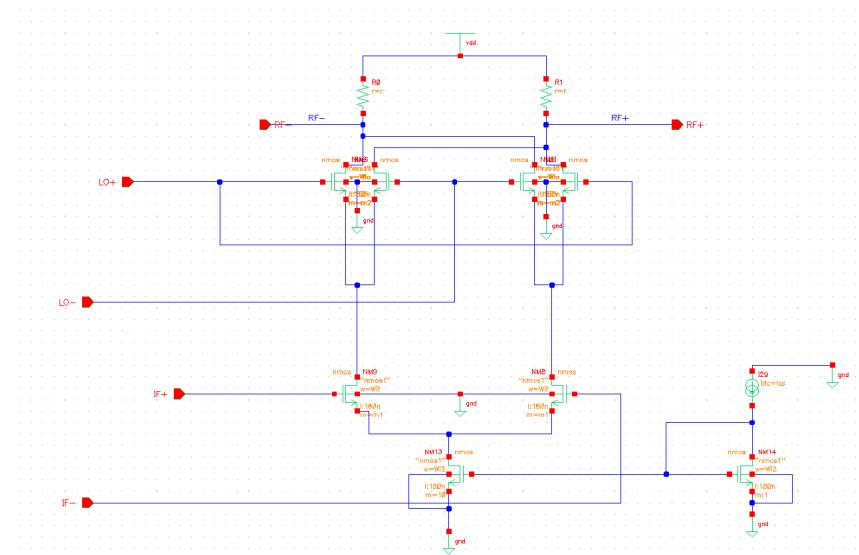


Figure 4.9: Schematic of one Gilbert Cell

Name	Param.	Size
M1, M2	W	$20 \mu\text{m}$
M3 - M6	W	500 nm
M7, M8	W	$9 \mu\text{m}$
M1 - M8	L	180 nm
M7	m	10
R	R	$8 \text{ k}\Omega$
R_P	R	$5 \text{ k}\Omega$
C_c	C	6.15 fF
I_{tail}	I	$10 \mu\text{A}$
V_{DD}	V	1.8 V
V_{bias}	V	1 V
v_{in1}	f	2.5 GHz
v_{in1}	\hat{V}	450 mV
v_{in2}	f	250 kHz
v_{in2}	\hat{V}	30 mV
v_{in2}	$offset$	700 mV

Figure 4.10: Gilbert Cell dimensions

length and a width based on Equation 4.13 with adjustments to improve the design. Figure 4.11 shows a block diagram of one Gilbert Cell and the voltages over each block. For the transistors of the tail current mirror and the differential transconductance pair with the low frequency applied, the desired overdrive voltage is again about 200 mV. The upper switching transistors need to be sized according to their on-resistance. They influence the linearity of the mixer. The resistance of the load resistors defines the amplitude range of the output signal, the higher it is, the higher is the voltage swing on the output due to the increased gain. The following calculations show the ideal parameters, where $I_{tail} = 10 \mu\text{A}$ and with a multiplier of 10 for M7 $I_D = 100 \mu\text{A}$. k' is again assumed to be around $100 \mu\text{A}/\text{V}^2$ for the NMOS transistors.

Tail current mirror:

$$\frac{W}{L} = \frac{2I_D}{k'V_{ov}^2} = 50 \Rightarrow W = 50 \cdot L = 50 \cdot 180 \text{ nm} = 9 \mu\text{m} \quad (4.14)$$

Differential Pair (IF):

$$\frac{W}{L} = \frac{2I_D}{k'V_{ov}^2} = \frac{2 \cdot 100 \mu\text{A}}{100 \mu\text{A}/\text{V}^2 \cdot (0.2 \text{ V})^2} = 50 \Rightarrow W = 9 \mu\text{m} \quad (4.15)$$

Load resistors / Switching transistors (LO):

$$R = R_L + R_{switch} = \frac{V_R}{I_D} = \frac{1.4 \text{ V}}{100 \mu\text{A}} = 14 \text{ k}\Omega \quad (4.16)$$

$$R_{switch} = \frac{1}{k' \frac{W}{L} V_{ov}} \quad (4.17)$$

for $R_L = 8 \text{ k}\Omega \Rightarrow R_{switch} = 6 \text{ k}\Omega$.

$$\Rightarrow \frac{W}{L} = \frac{1}{k' R_{switch} V_{ov}} = \frac{1}{100 \mu\text{A}/\text{V}^2 \cdot 6 \text{ k}\Omega \cdot 0.2 \text{ V}} = 8.33 \quad (4.18)$$

$$\Rightarrow W = 1.5 \mu\text{m} \quad (4.19)$$

$$(4.20)$$

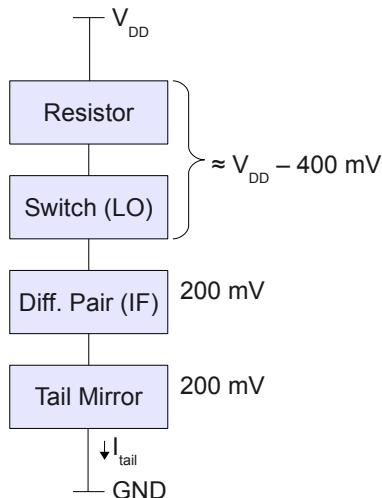


Figure 4.11: Gilbert Cell block diagram with voltages

The width of the switching transistors M3 to M6 was chosen to be smaller than the calculated value, to make the parasitic capacitance as small as possible due to the high frequency applied to their gates. This allows a cleaner and faster switching and a better shape on the output.

The width of the differential pair transistors, however, was chosen to be larger than the calculated value, to increase the voltage swing on the output and improve the accuracy of the output signal.

The correct functionality of the mixer strongly depends on the input signals and their levels and amplitudes. It is important that the transistors M1 and M2 are only conducting at the same time for a very short period, ideally not at all. Therefore, the bias level needs to be around the threshold voltage of the

4 System Specification and Design

transistors. A high amplitude of the input IF signal on v_{in2} results in a cut off of the output signal, because there is not enough headroom. That is, the amplitude needs to be very small. It is also limited by the linearity of the mixer. The amplitude needs to be in the linear region, which is further explained in Chapter 5.

The bias voltage of v_{in2} is 700 mV and the amplitude 30 mV for all simulations, as these values have proven to give best results.

The switching transistors M3 to M6 can have a higher amplitude which means they can have a higher bias voltage as well. For them it is again important that ideally two transistors on the same branch of the mixer are not conducting at the same time. Thus, the bias voltage, which is created by pulling up the decoupled nets through resistors of $5\text{ k}\Omega$, is chosen to be 1 V. The amplitude is 450 mV.

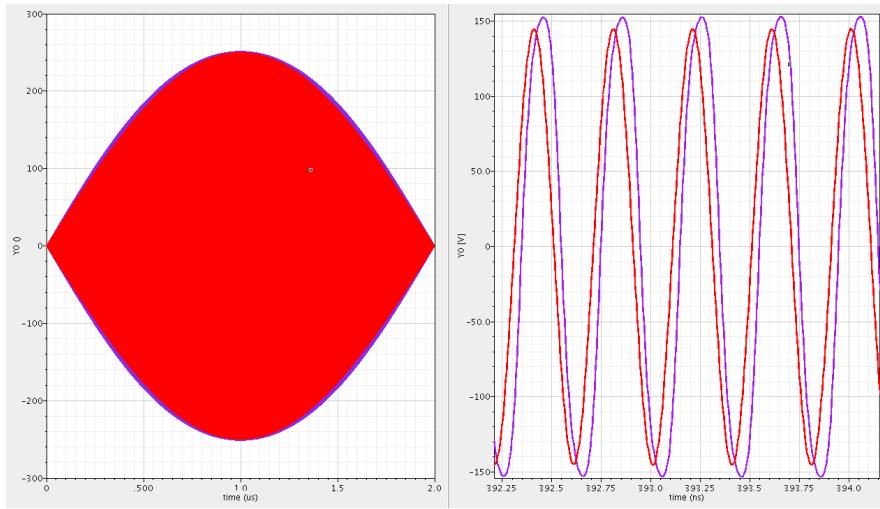


Figure 4.12: Mixer output; left: overall shape, right: zoomed in; purple: real output, red: ideal output

The output signal of one mixer is shown in Figure 4.12. The purple trace shows the output of the circuit and the red trace shows the ideal output according to Equation 2.9.

The overall shape differs slightly from the ideal trace, which is due to transistors in both branches of the mixer circuit being in the "on"-state at the same time and also due to non-ideal port isolation. That is, the port's signals leak through to their neighbouring ports and interfere with their signals.

Figure 4.13 shows the output of the whole frequency converter, where two Gilbert Cells share a resistive load. The upper graph shows the overall signal and details the amplitude, the lower graph shows the frequency in detail.

For the same reasons the mixer output differs from the ideal, the amplitude of the overall converter output is not steady. The IF frequency is leaking through and causes a variation in the amplitude. This will be explained and shown in further detail again in Chapter 5.

4.4 Gilbert Cell and Frequency Converter Design

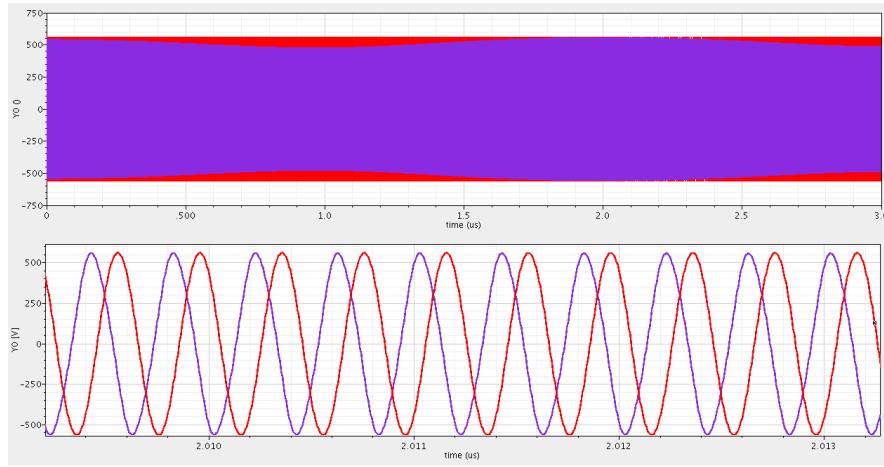


Figure 4.13: Frequency converter output; top: overall shape, bottom: zoomed in, purple: real output, red: ideal output

This chapter has described the design of the main components of this project. The geometries of the components were chosen based on mathematical analysis and adjustments were made based on the simulation results. The characteristics, which are directly related to the designs, will be explained and discussed in the next chapter.

4 System Specification and Design

5 Results and Evaluation

The design process goes hand in hand with simulations of several characteristics which have to be taken into account and need to be constantly observed. The key characteristics for all circuits, which have been already explained in Chapter 2, were simulated, measured and plotted with spectreRF in Cadence Virtuoso. This chapter will show the phase noise for the LC oscillator and discuss PVT variations. For the mixer and the quadrature frequency converter noise, linearity, isolation and gain are shown. At the end of each section, the design will be compared to recent publications and thereby its quality will be evaluated.

5.1 Oscillator Characteristics

The analogue LC oscillator designed as described in Chapter 4.3 is evaluated in this section. Phase noise and PVT variations are used to characterise the quality of the design. The accuracy of the output frequency is the major concern. All measurements were carried out for the original architecture and the modified one with the two extra transistors for comparison. For the extended architecture, two different values for the inductor and therefore the capacitor as well were used to show how the performance differs. The settings in spectreRF for the simulations are described based on the Cadence Manual [25, 26].

Figure 5.1 shows the relative phase noise of the original cross-coupled LC oscillator for an inductor with $L = 4.06 \text{ nH}$. The red trace shows the statistical average of the phase noise at 27°C and 1.8 V supply voltage. The worst case is shown in green and is achieved with a temperature of 125°C at a supply voltage of 1.98 V ($= 1.8 \text{ V} + 10\%$). The process variations were set to fast for both NMOS and PMOS transistors (FF).

The best case is shown in orange, where the temperature is -40°C , the voltage is 1.68 V ($= 1.8 \text{ V} - 10\%$) and the corner cases are set to FS, so fast for NMOS and slow for PMOS transistors.

At 1 kHz , the average phase noise is -33.3 dBc/Hz , at 1 MHz it is -118.8 dBc/Hz , the best and worst case differ by up to $\pm 5 \text{ dBc/Hz}$ from that value.

The simulation was done using a pss (= periodic steady state) and a pnoise (periodic noise) analysis in spectreRF of the Cadence analogue design environment. The pss analysis has to be performed in order to run a pnoise analysis prior to that. It is the equivalent to a DC analysis but with periodic signals.

5 Results and Evaluation

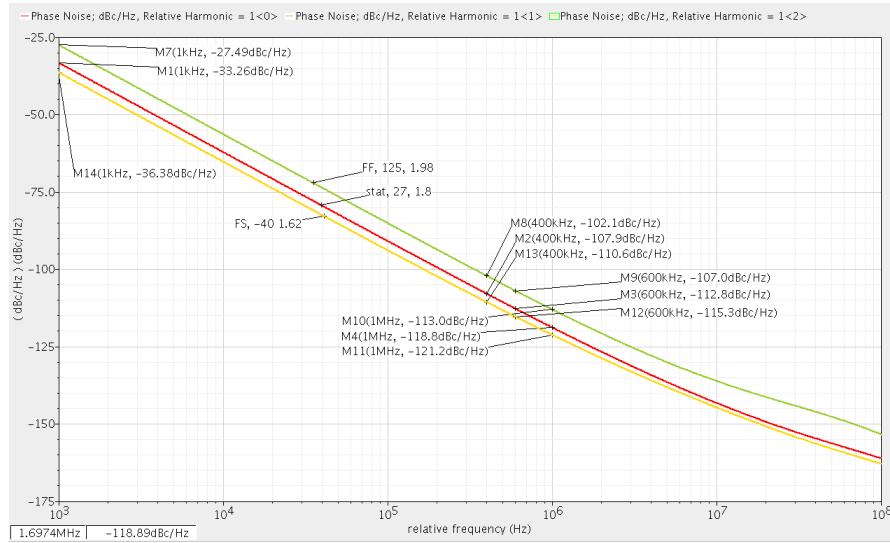


Figure 5.1: Phase Noise of the original oscillator architecture for $L = 4.06 \text{ nH}$

As there are no periodic input signals to the oscillator, no fundamental frequencies are displayed in the setup window for the pss analysis. The beat frequency specifies an approximate of the expected output frequency and is set to 5 GHz in this case. The number of output harmonics was chosen to be 6. The higher this value is, the more accurate is the phase noise later, because more tones are taken into account for calculation. The stabilisation time is set to 1 μs to allow time for the oscillator to reach its stable output frequency. For the oscillator there is a special feature in the pss analysis. To activate it the box called "oscillator" needs to be ticked off. This allows to specify the output nodes out+ and out- as shown in Figure 4.4a.

The pnoise analysis sweeps the frequency over a specified range and plots the noise in a way that can be chosen from several settings. In this case, the sweep type is relative, so that the phase noise is plotted as an offset from the oscillation frequency of 5 GHz. The sweep range is chosen to be from 1 kHz to 100 MHz on a logarithmic scale with 501 points per decade to get an accurate graph. The maximum number of sidebands is set to 6 to match the number of output harmonics of the oscillator and take these into account for the noise analysis. The output is set to "voltage" and the differential nodes of the oscillator out+ and out- are chosen in the schematic. The input is set to "none" as there is no input and the noise type is set to "sources" to achieve the output shown in Figure 5.1. By setting the noise type to jitter, the jitter is automatically calculated and output as an absolute value with a unit of fs ($= 10^{-15} \text{ s}$) or even as ($= 10^{-18} \text{ s}$).

Figure 5.2 shows how the frequency varies with PVT variations. The light blue trace shows the average at 27 °C, statistically distributed corner cases and a supply voltage of 1.8 V. The frequency there is 5.0002 GHz. The dark blue trace shows the minimum case with 4.99513 GHz for 125 °C, 1.98 V supply voltage and corner

5.1 Oscillator Characteristics

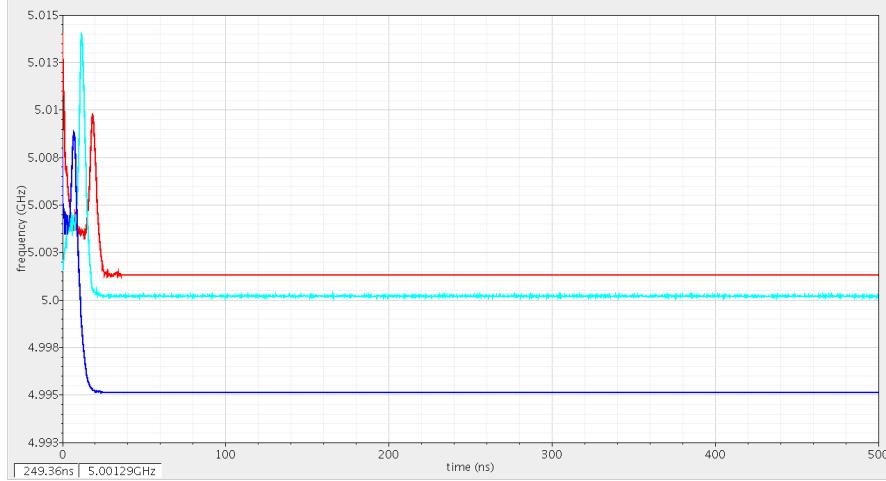


Figure 5.2: Frequency over PVT variations for the original architecture with $L = 4.06 \text{ nH}$

cases set to FF. The red trace shows the maximum frequency with 5.00129 GHz at -40°C , 1.68 V supply voltage and for SF as process variation. This means that the fundamental frequency only varies $+0.02\%$ and -0.10% . That is very low for an LC oscillator and in reality can't be achieved. The reason these values are so low is, that there is no process variation model for the inductor in the design kit used for this project. The inductor, however, varies a lot over process variations. A realistic accuracy would be several percent starting from $\pm 5\%$.

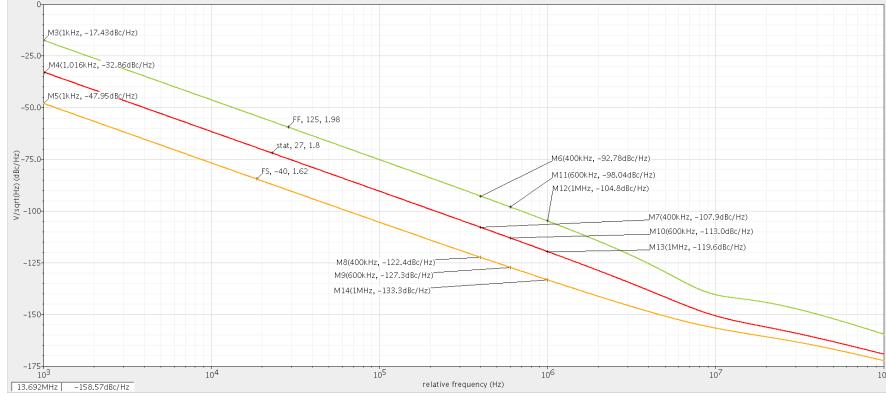


Figure 5.3: Phase noise for $L = 1.13 \text{ nH}$ with the modified architecture

For the modified cross-coupled architecture of the oscillator the phase noise average, best and worst case results are plotted in Figure 5.3 and 5.4 for two different inductances.

The average relative phase noise is represented by the red trace and at 1 kHz for the circuit with an inductance of $L = 1.13 \text{ nH}$ it is -32.9 dBc/Hz and at an offset of 1 MHz it is -119.6 dBc/Hz . The best case at -40°C with a supply voltage of 1.68 V and process variations set to FS is shown in orange and the worst case

5 Results and Evaluation

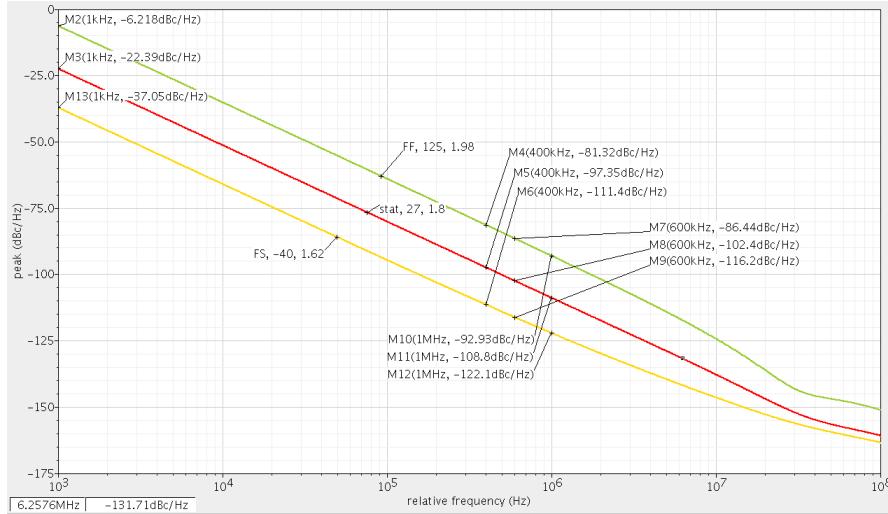


Figure 5.4: Phase noise for $L = 4.06$ nH with the modified architecture

at 125 °C and 1.98 V supply voltage with FF set as corner case is shown in green. Comparing this graph to the previous graph of the original architecture, it can be seen, that the average phase noise is slightly better than before measured at higher offsets from the oscillation frequency. At 1 kHz it is very similar.

The deviation of the best and worst case according to PVT variations is however greater in the new architecture with approximately ± 15 dBc/Hz. This can be explained by the number of transistors. Because the extended architecture uses two more transistors, the whole circuit is more prone to PVT variations than the circuit with less transistors.

The oscillator with an inductance of $L = 4.06$ nH has an average phase noise of -22.4 dBc/Hz at 1 kHz and -108.0 dBc/Hz at 1 MHz, again shown as a red trace in Figure 5.4. The orange trace shows the best case and the green trace the worst case both under the same circumstances as in the previous graph with the lower inductance. The variation of these from the average is again around ± 15 dBc/Hz.

The circuit with the lower inductance shows a better performance in terms of phase noise. The PVT variations are similar, but again under the condition that there is no inductor model for process variations so that the inductor's size is not taken into account and therefore leads to similar results.

The variation of the output frequency of the extended oscillator with process and temperature variations is illustrated in Figure 5.5 and 5.6 for the two different inductances.

All measurements were taken with a supply voltage of 1.8 V. The red traces show the average process corners at the three different temperatures, the orange traces show the corner cases for fast NMOS and slow PMOS transistors. The green traces show the frequencies for both in fast mode. The pink traces represent the SF setting and the blue traces the SS process variations.

5.1 Oscillator Characteristics

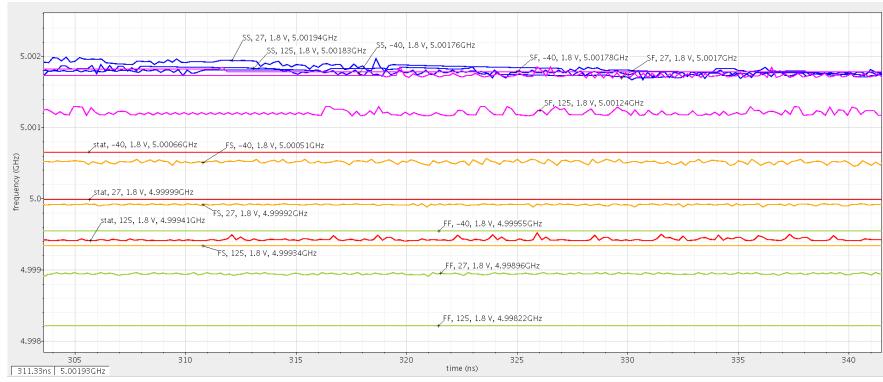


Figure 5.5: Process and temperature variation of the frequency for $L = 1.13 \text{ nH}$

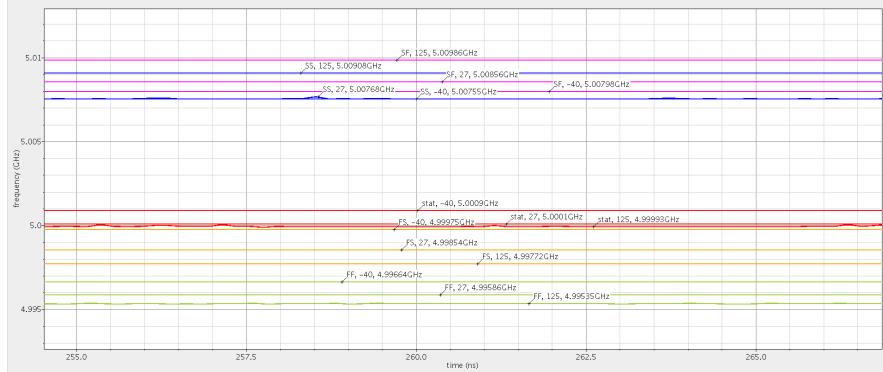


Figure 5.6: Process and temperature variation of the frequency for $L = 4.06 \text{ nH}$

5 Results and Evaluation

It can be generally seen, that for the lower inductance, the frequency is less stable. However for the larger inductor, the deviation from the desired frequency with the process and temperature variation is larger.

Figure 5.7 and 5.8 show only the best and worst case, also taking a 10 % deviation of the supply voltage into account. The centre frequency for the circuit with an inductance of 1.13 nH is 4.99999 GHz. The worst case deviation is + 0.09 % and - 0.04 %. For the oscillator with the 4.06 nH inductance, the centre frequency for the specified design is 5.0001 GHz with a deviation of + 0.40 % and - 0.16 % in the worst cases.

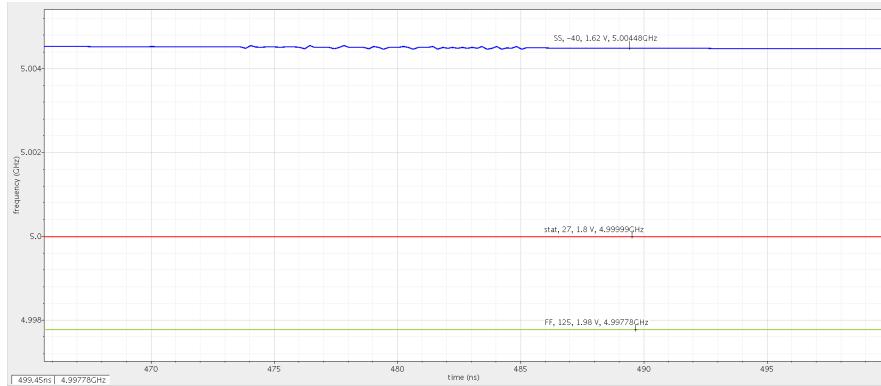


Figure 5.7: Worst case PVT variations of the frequency for $L = 1.13 \text{ nH}$

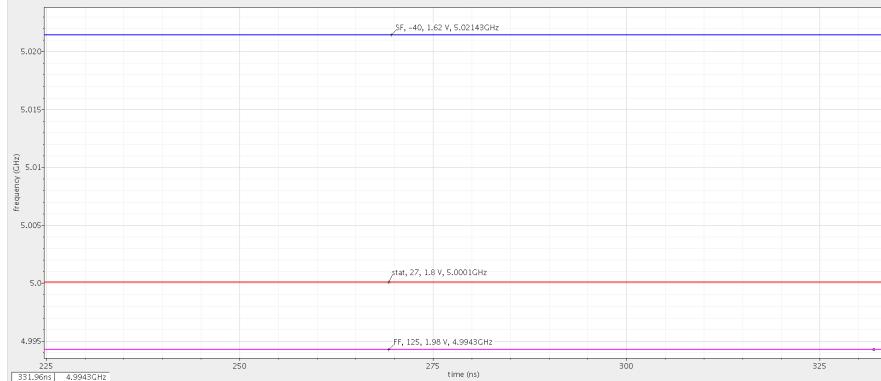


Figure 5.8: Worst case PVT variations of the frequency for $L = 4.06 \text{ nH}$

The pss analysis which was run to gain the phase noise graphs, also gives a voltage spectrum of the magnitude of the harmonic frequencies. These are plotted in Figure 5.9 and 5.10 for the extended architecture. The peak is at 5 GHz with approximately 5 dB for both inductances.

Table 5.1 summarises the characteristics of the three in this project proposed designs and compares them to recent publications.

5.1 Oscillator Characteristics

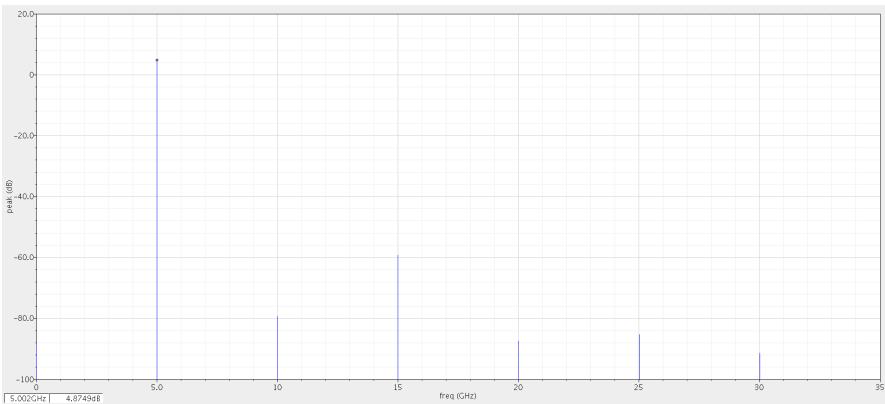


Figure 5.9: Voltage spectrum for $L = 1.13 \text{ nH}$

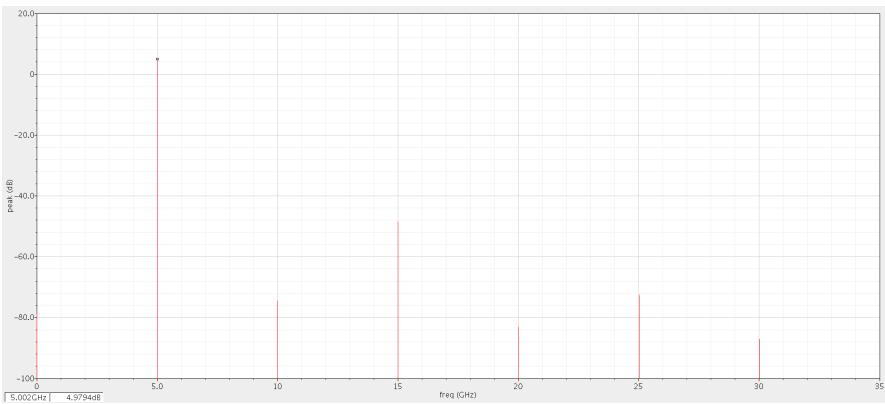


Figure 5.10: Voltage spectrum for $L = 4.06 \text{ nH}$

5 Results and Evaluation

Taking only the best case into account, the extended architecture with an inductance of 1.13 nH shows the best results. Considering the worst case, the simple cross-coupled oscillator has a better phase noise performance. The best case results beat all the mentioned publications by up to 15 dBc/Hz. Overall, all three designs in this project show reasonable phase noise results which can compete against other designs.

The publications do not mention any variations or worst case values for their phase noise, therefore their data was assumed to be the best case result. Generally, there are only a few papers considering PVT variations, most designs omit these, because the results can be very poor, if the design is not matched properly. Further it is to be mentioned, that [12–14] took their measurements from real circuits and not only the simulations as in [11] and in this project’s design.

Design	f_0	PN (dBc/Hz)			Δf
		worst	average	best	
[12]	1.00 GHz			-40.0	1 kHz
[13]	12.50 GHz			-106.0	400 kHz
[11]	2.63 GHz			-112.3	600 kHz
[14]	5.60 GHz			-123.6	1 MHz
Original L \approx 4 nH	5.00 GHz	-27.5	-33.3	-36.4	1 kHz
		-102.1	-107.9	-110.6	400 kHz
		-107.0	-112.8	-115.3	600 kHz
		-113.0	-118.8	-121.2	1 MHz
Extended L \approx 1 nH	5.00 GHz	-17.4	-32.9	-47.9	1 kHz
		-92.8	-107.9	-122.4	400 kHz
		-104.8	-113.0	-127.3	600 kHz
		-104.8	-119.6	-133.3	1 MHz
Extended L \approx 4 nH	5.00 GHz	-6.22	-22.4	-37.1	1 kHz
		-81.3	-97.4	-111.4	400 kHz
		-86.4	-102.4	-116.2	600 kHz
		-92.9	-108.8	-122.1	1 MHz

Table 5.1: Comparison of LC oscillator designs to this design

5.2 Gilbert Cell Characteristics

This section will explain the results of the Gilbert Cell simulations based on the design specifications in the previous chapter. Thereby, characteristics such as conversion gain, noise figure, linearity and isolation are analysed. Where possible adequate mathematical equations are used for comparison to the simulative results.

The conversion gain dependent on the input power of the LO port is plotted in Figure 5.11. It shows the gain from the input power to the output power. At the chosen amplitude on the LO input, which is 450 mV, the conversion gain has a magnitude of 7.6 dB. The maximum value is around 13 dB, but to reach that value the amplitude of the LO signal needs to be higher than 1 V.

The conversion gain was also tested for PVT variations, but these were very small, there was nearly no deviation from the mean curve.

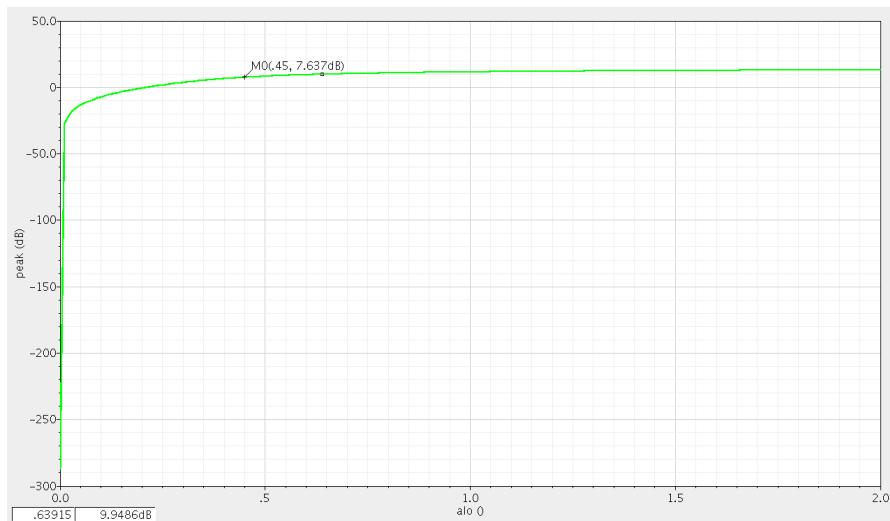


Figure 5.11: Conversion gain for the mixer

The conversion gain, under the assumption that the mixer is perfectly switching, is defined as

$$CG = \frac{2}{\pi} g_m R_L = \frac{4I_D}{\pi V_{ov}} R_L. \quad (5.1)$$

For the mixer designed as in chapter 4.4, that gives

$$CG = \frac{4 \cdot 100 \mu A}{\pi \cdot 0.2 V} \cdot 5 k\Omega = 3.2. \quad (5.2)$$

Converting that to dB, the result is

$$CG_{dB} = 10 \cdot \log(CG) = 10 \cdot \log(3.2) = 5.0 \text{ dB}. \quad (5.3)$$

5 Results and Evaluation

The deviation from the 7.6 dB can be explained by the switching not being perfect and because the overdrive voltage may be slightly different from the desired 0.2 V.

To measure the conversion gain, again spectreRF analyses are used. Firstly a pss analysis similar to the one described in the previous section is done. Before setting up the simulation, the IF port needs to be set from sine wave to DC signal and the small signal parameter for the magnitude of the port, used later during a PAC (periodic AC) analysis, is set to 1. In the pss settings, there should be only one entry under fundamental tones, which is the LO port. The beat frequency is 2.5 GHz, as the expected output of the mixer is close to that value (ideally 2.500250 GHz). The number of harmonics was set to two for this simulation and the amplitude of the LO signal is swept from 0 to 2 V.

Additionally, a pac analysis needs to be set up. The frequency range is there set to single point with the 250 kHz of the IF input. The number of sidebands is the same as the number of harmonics specified in the pss analysis, so 2.

The noise was measured over the amplitude at the LO port and as a relative offset from the output frequency over the frequency range. Figure 5.12 shows the former and Figure 5.13 the latter.

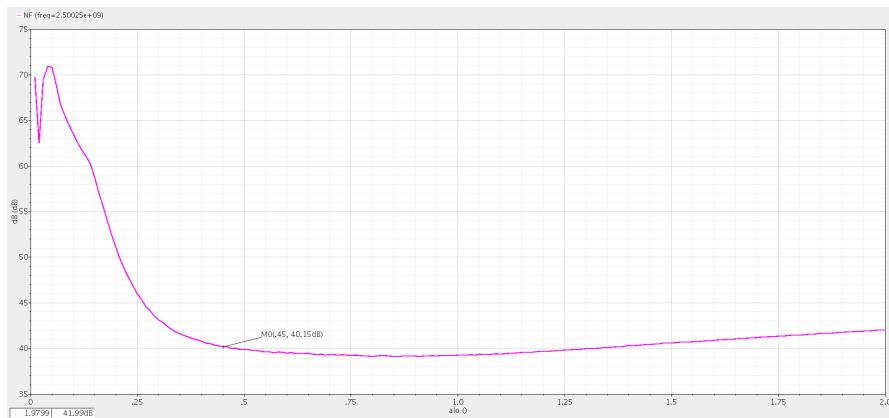


Figure 5.12: Noise Figure depending on the LO input power

In Figure 5.12, it can be seen that the phase noise is very high for amplitudes at the LO port below 0.25 V and then reaches a near steady state around 40 dB. At the chosen LO amplitude of 450 mV, the noise is 40.2 dB.

The relative noise figure is 36 dB at 1 MHz offset from the centre frequency under standard circumstances as shown in Figure 5.13. Figure 5.14 points out the PVT variations of the noise. The red traces show the cases for 125 °C, the green traces for 27 °C and the blue traces for -40 °C. The process cases SF and FS were omitted because there are no PMOS transistors in the circuit and therefore SS and FF are sufficient, as the second letter always refers to PMOS type transistors. The lowest of each block of traces is always for the FF case, the middle trace for the standard process variation setting and the upper trace for the SS case. For better

5.2 Gilbert Cell Characteristics

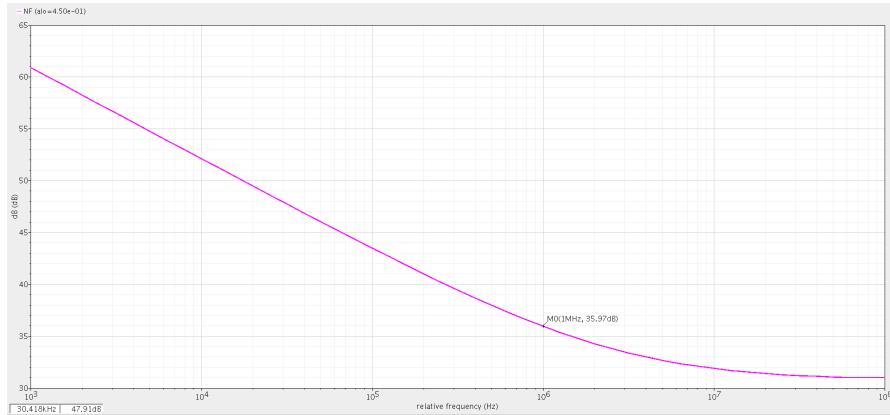


Figure 5.13: Realitive Noise Figure over frequency

readability, voltage variations were not plotted except for the black trace, which was measured for 125 °C at 1.62 V and with slow transistors and a magnitude of 47.7 dB at an offset of 1 MHz. This is the worst case noise. All other voltage variations were somewhere in between best and worst case and always quite close to the corresponding 1.8 V trace. The best case is for fast transistors at -40 °C and 1.8 V supply voltage with a value of 28.8 dB at an offset of 1 MHz from the output frequency of the mixer.

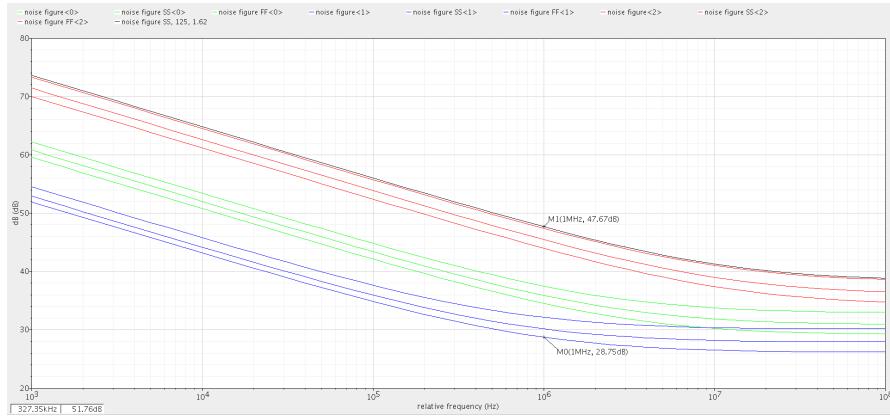


Figure 5.14: Worst case relative Noise Figure with PVT variations

Noise is measured with the same pss setup as the conversion gain. Additionally a pnoise analysis is needed, which is specified with a frequency sweep range from 1 kHz to 100 MHz. The number of sidebands is again the same as the number of harmonics in the pss analysis, so 2. The output is set to "voltage" and the RF differential output nodes are chosen from the schematic. The input is given by the IF port. The reference sideband is the first and the noise type is set to sources.

As with these settings, either the LO amplitude and the frequency are swept, it can be chosen over which the noise is plotted.

5 Results and Evaluation

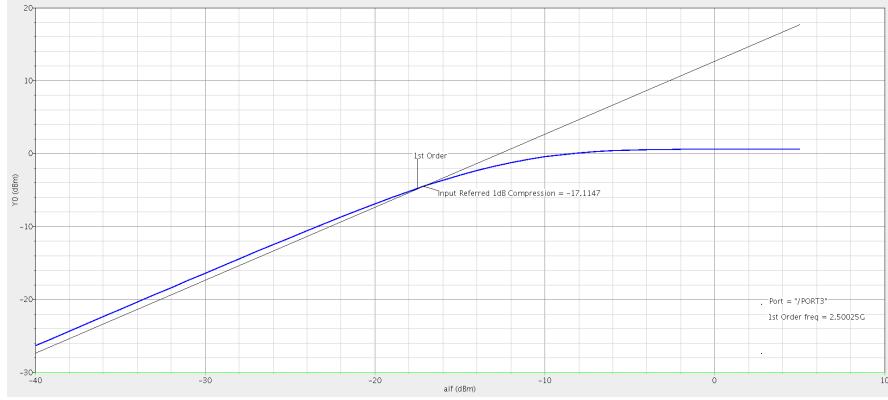


Figure 5.15: 1 dB compression point for the mixer

The linearity of the mixer is expressed by the 1 dB compression point and the input referred third order intercept point. For this, the mixer output is expressed as a Taylor series with the third polynomial as the highest polynomial for simplification. The intersection of the first and the third polynomial forms the third order intercept point and is a comparable metric for linearity of different mixers.

Figure 5.15 shows the plot of the mixer RF output power over the IF input power in blue. The linear region of the trace is extrapolated and shown in black for a 1 dB offset. The intersection of both traces gives the 1 dB compression point which is at -17 dB. This is the point up to which the mixer is deemed to be working linearly.

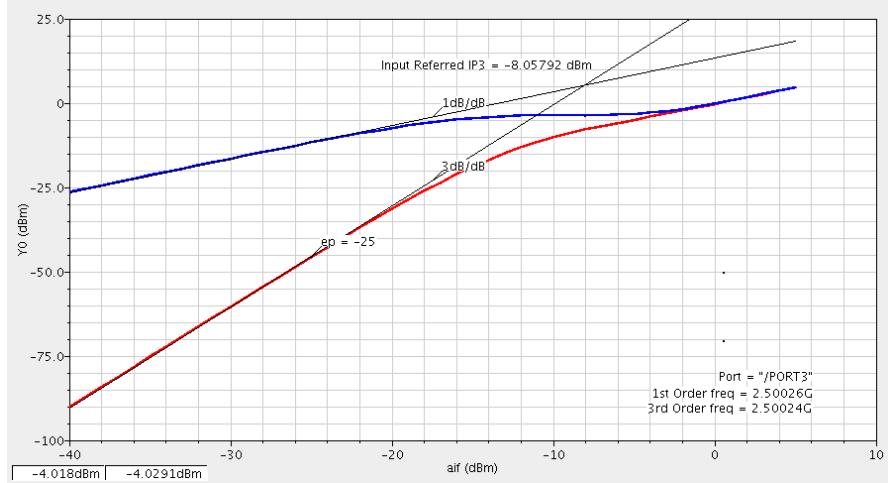


Figure 5.16: Input third order intercept point for the mixer

The construction of the third order intercept point is illustrated in Figure 5.16. The blue trace shows the first order approximation and the red trace the third order approximation. The extrapolation of the linear regions of both gives an intercept point at an input power of -8 dBm.

5.2 Gilbert Cell Characteristics

To simulate the IIP3 and 1 dB-compression point, a qpss (= quasi-periodic steady state) analysis is run together with a qpac (= quasi-periodic AC) analysis. The difference to the pss and pac analysis is that with the quasi-periodic analyses two tone simulations can be done. That means one can simulate the mixer without setting the IF input to DC.

For this simulation, the amplitude of the IF port is set as a variable "aif" and specified in dB, not as before in Volts. The small signal PAC parameter is set to the same variable "aif", so that when the amplitude is swept, the small signal amplitude is swept as well. In the qpss settings window, the LO port is specified as a large tone with 2 harmonics, which is sufficient for displaying the desired output. The IF port is specified as a moderate tone with also 2 harmonics. The variable "aif" is swept from -40 to 5 dB. The qpac analysis is set to a single point frequency range at a frequency close to the IF frequency, in this case 260 kHz. The sidebands are chosen to be $2.5 \text{ GHz} + (250 \text{ kHz} \pm (260 \text{ kHz} - 250 \text{ kHz}))$, so 2.50024 GHz and 2.50026 GHz. These are the first and third order approximations, which are displayed in Figure 5.16.

As could already be seen in Figure 4.13, the amplitude of the converter is not constant, as it would be in the ideal case. That means that the IF frequency is leaking through. The relation of the leaking through signal to the overall signal is called carrier suppression. The carrier suppression is mathematically defined as:

$$\text{CarrierSuppression}_{dB} = 20 \cdot \log \left(\frac{A}{B} \right) \quad (5.4)$$

where A is the maximum peak-to-peak amplitude of the overall output signal and B is the peak-to-peak amplitude of the leaking through signal [27].

In this design it gives:

$$\text{CarrierSuppression}_{dB} = 20 \cdot \log \left(\frac{1114.2 \text{ mV}}{79.6 \text{ mV}} \right) = 20 \cdot \log(14) = 22.9 \text{ dB}. \quad (5.5)$$

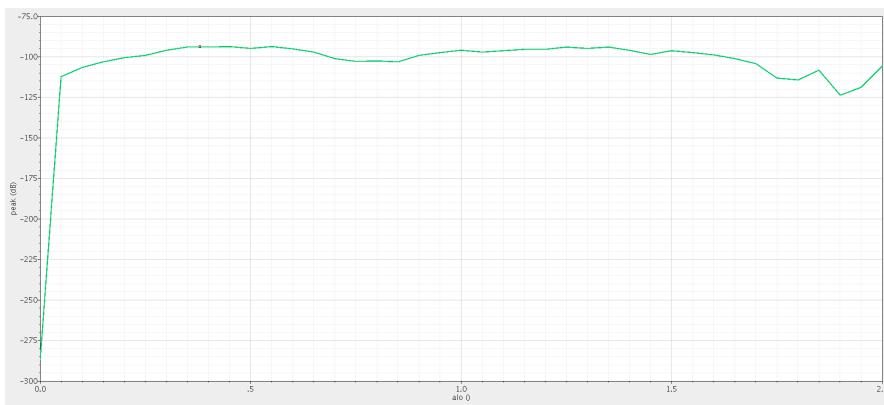


Figure 5.17: Isolation from the IF port to the RF port depending on the LO input amplitude

5 Results and Evaluation

The isolation from the IF port to the RF port over the LO amplitude is shown in Figure 5.17. The value at 450 mV is -94 dB, which is the maximum of the curve. This is a fairly small value and explains, why the IF signal is still visible on the output.

Table 5.2 summarises the simulation results described in this section and compares them to recent publications introduced in Chapter 3.2. The difference from the mixers from publications is, that they are down-converting to a low frequency, which means that their inputs are in the same high frequency range. This design has a high and a low frequency input, which is more complex to design and is also reflected in the results. The noise figure is higher than for the publications. The conversion gain is lower, but still there is no conversion loss. The IIP3 is also lower, which means the designed mixer is less linear. For [18], there are no noise values available, but in terms of gain and linearity, the values are very close to this design. The overall results achieved in this project are reasonable and can be worked with. Improvement might be done in the future.

Design	f_{RF}	f_{LO}	NF	Δf	Conv. Gain	IIP3
[15]	2.120 GHz	2.119 GHz	8.96 dB	1 MHz	20.55 dB	-0.54 dBm
[16]	2.450 GHz	2.449 GHz	16 dB	1 MHz	13.36 dB	-1 dBm
[18]	2.4 GHz	2.3 GHz	n.a.	n.a.	6.7 dB	-7.5 dBm
This	≤ 250 kHz	2.5 GHz	35.97 dB	1 MHz	7.64 dB	-8.06 dBm

Table 5.2: Comparison of published mixer designs to this design

6 Conclusion and Outlook

6.1 Summary

In this project, a novel PLL was developed, first as an ideal VerilogA model with parts later replaced by more realistic models or circuits on transistor level. The single blocks which form the PLL are based on established designs, which have been used for years in integrated circuits. The novelty is in the interaction of these blocks. A $\Delta\Sigma$ -modulator followed by a quadrature frequency converter replace the conventional VCO. A signal processor translates the phase-frequency relation of the input reference of the PLL and the fed back output into a digital word, which then controls the system output frequency to be exactly 2.5 GHz. These 2.5 GHz are composed of a by two divided 5 GHz LC oscillator with a certain variation which is compensated by a frequency generated by the $\Delta\Sigma$ -DAC in a range of 1 mHz up to 250 kHz. The main implementation in this part of the project focused on the LC oscillator and the quadrature frequency converter. Another project related to this system implemented the $\Delta\Sigma$ -modulator and a low-pass filter.

A testbench was developed, showing that the overall system works. However, the simulation time was very high and the computer lab PCs could not handle the amount of data produced. Therefore, the values for characterising the single blocks such as sample rates and resolutions were set to values other than the specified ones, to produce an output showing the PLL working.

The LC oscillator developed in this project is based on the cross-coupled NMOS-PMOS architecture. A variation was tested with two extra transistors and compared to the original architecture. Under best case circumstances, the new architecture gives the best results in terms of noise. Compared to recent papers, the results are competitive and show even better values. The best case phase noise achieved was measured at 1 MHz offset from the centre frequency with a magnitude of -133.3 dBc/Hz, the worst case at the same offset was -92.9 dBc/Hz, but for different geometries in the design. The oscillator was tested for PVT variations and gave a maximum deviation of 0.4 % in the worst case. This result does not include process variations of the inductor.

The quadrature frequency converter is composed of two Gilbert Cell mixers sharing their resistive load to produce the difference of both individual mixer outputs. The mixer design was challenging in a way, that a very low frequency, which is close to DC, is mixed with a very high frequency. This case is rarely discussed in publications. Most mixers are used for down-converting to a low

6 Conclusion and Outlook

frequency, this project's mixer does both up- and down-conversion, depending on the direction of the deviation of the LC oscillator's frequency. The mixer has a slightly higher noise figure than comparable designs, which however are down-converting to a low frequency. The linearity is compared to some cases given over a smaller range and the gain is a bit lower than for other designs.

The implementations satisfy the needs of the system for a first design step. Improvements which could be worked on in the future are discussed in the next section.

6.2 Future work

To gain more realistic results for the oscillator over process variations, an inductor model needs to be developed, which reflects its realistic behaviour, in case the layers are not perfectly produced and where parasitics are present. This model would help understand how good the design really is in terms of PVT variations, as this beside others characterises the quality.

The overall noise of the quadrature frequency converter could not be simulated due to limitations of spectreRF. For example, it can't simulate a circuit with more than two different ports in sine-mode. The current testbench for the converter needs four. Another issue is, that models written in VerilogA can't be simulated. When implementing further blocks of the PLL on transistor level, the whole design can be simulated. A first block would be the D-Flip-Flop divider used to divide down the oscillator frequency of 5 GHz and producing quadrature related streams.

The Gilbert Cell produces an output with the IF frequency leaking through. Improving the isolation and noise by further analysing the circuit on a very detailed level could help improve the output in terms of leakage.

It was not possible simulating the whole system with its specified values over a reasonable simulation time. Using a strong PC over several days, the system might be simulated.

On a later stage, the LC oscillator will be replaced by another PLL producing a very accurate frequency so that only very low frequencies will be needed to compensate it and gain the exact 2.5 GHz output.

References

- [1] Howard Luong and Gerry Leung. *Low-Voltage CMOS RF Frequency Synthesizers*. Cambridge University Press, 2004.
- [2] Behzad Razavi. *RF Microelectronics*. Prentice Hall, 1998.
- [3] Adel Sedra and Kenneth Smith. *Microelectronic Circuits*. Oxford University Press, 1998.
- [4] Ed da Silva. *High Frequency and Microwave Engineering*. Butterworth-Heinemann, 2001.
- [5] Jimmin Chang, A.A. Abidi, and C.R. Viswanathan. Flicker noise in cmos transistors from subthreshold to strong inversion at various temperatures. *Electron Devices, IEEE Transactions on*, 41(11):1965 –1971, 1994.
- [6] R. Sarpeshkar, T. Delbruck, and C.A. Mead. White noise in MOS transistors and resistors. *Circuits and Devices Magazine, IEEE*, 9(6):23 –29, 1993.
- [7] Thomas Lee. *The Design of CMOS Radio-Frequency Integrated Circuits*. Cambridge University Press, 2004.
- [8] Barrie Gilbert. A precise four-quadrant multiplier with subnanosecond response. *IEEE Journal of Solid-State Circuits*, 3(4):365–373, 1968.
- [9] John Silver. Gilbert cell mixer design tutorial. http://www.odysseus.nildram.co.uk/RFIC_Circuits_Files/MOS_Gilbert_Cell_Mixer.pdf. [Online; retrieved 27/09/2012].
- [10] M. M. Mansour. On the design of low phase-noise CMOS LC-tank oscillators. *Proc. Int. Conf. Microelectronics ICM 2008*, pages 407–412, 2008.
- [11] C. C. Boon, M. A. Do, K. S. Yeo, J. G. Ma, and R. Y. Zhao. Parasitic-compensated quadrature LC oscillator. *IEEE Proceedings - Circuits, Devices and Systems*, 151(1):45–48, 2004.
- [12] Zhongtao Fu, A. Pappu, and A. Apsel. Beating the power limit of LC oscillators. *Proc. 50th Midwest Symp. Circuits and Systems MWSCAS 2007*, pages 441–444, 2007.

References

- [13] D. Solanki, R. Chandel, T. Alam, and A. Nishad. Design of LC-VCO for low power narrowband electronic applications. *Proc. 2nd Int Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (Wireless VITAE) Conf*, pages 1–4, 2011.
- [14] Joonhong Park, Junyoung Park, Youngwan Choi, Kweebo Sim, and Donghyun Baek. A fully-differential complementary hartley VCO in 0.18 μm CMOS technology. *Microwave and Wireless Components Letters, IEEE*, 20(2):91 –93, 2010.
- [15] S. K. Alam. A 2 GHz low power down-conversion quadrature mixer in 0.18- μm CMOS. *Proc. the Int VLSI Design Held jointly with 6th Int. Conf. Embedded Systems*, pages 146–154, 2007.
- [16] Chun Geik Tan. A high-performance low-power CMOS double-balanced IQ down-conversion mixer for 2.45-GHz ISM band applications. *Radio Frequency Integrated Circuits (RFIC) Symposium, 2003 IEEE*, pages 457 – 460, 2003.
- [17] Wang-Chi Cheng, Cheong-Ft Chan, Chiu-Sing Choy, and Kong-Pang Pun. A 1.2 V 900 MHz CMOS mixer. *IEEE International Symposium on Circuits and Systems*, 5:365 – 368, 2002.
- [18] Chih-Chun Tang, Wen-Shih Lu, Lan-Da Van, and Wu-Shiung Feng. A 2.4-GHz CMOS down-conversion doubly balanced mixer with low supply voltage. *IEEE International Symposium on Circuits and Systems, 2001*, 4:794 –797, 2001.
- [19] X. Pu, A. Kumar, S. Goldman, and K. Nagaraj. Low-noise low-spur architecture for a fully integrated analog PLL working from a low-frequency reference. *Circuits and Systems II: Express Briefs, IEEE Transactions on*, PP(99):1 –5, 2012.
- [20] Zhipeng Ye, Wenbin Chen, and M.P. Kennedy. A novel dual-loop multi-phase frequency synthesizer. In *18th European Conference on Circuit Theory and Design, 2007*, pages 567 –570, 2007.
- [21] Ching-Yuan Yang, Chih-Hsiang Chang, Jun-Hong Weng, and Hsin-Ming Wu. A 0.5/0.8-v 9-GHz frequency synthesizer with doubling generation in 0.13- μm CMOS. *IEEE Transactions on Circuits and Systems*, 58(2):65–69, 2011.
- [22] Wenting Wang and H.C. Luong. A 0.8-V 4.9-mW 1.2-GHz CMOS fractional-N frequency synthesizer for UHF RFID readers. *IEEE Transactions on Circuits and Systems I*, 55(9):2505 –2513, 2008.
- [23] Hsieh-Hung Hsieh, Chung-Ting Lu, and Liang-Hung Lu. A 0.5-V 1.9-GHz low-power phase-locked loop in 0.18- μm CMOS. In *IEEE Symposium on VLSI Circuits*, pages 164 –165, 2007.

- [24] Shih-An Yu and P. Kinget. A 0.65-V 2.5-GHz fractional-N synthesizer with two-point 2-Mb/s GFSK data modulation. *IEEE Journal of Solid-State Circuits*, 44(9):2411 –2425, 2009.
- [25] Cadence. *Virtuoso®SpectreRF Simulation Option User Guide*, product version 5.1.41 edition, November 2005.
- [26] Cadence. *VCO Design Using SpectreRF Application Note*, product version 5.0 edition, September 2004.
- [27] Joseph J. Carr. *Practical Radio Frequency Test and Measurement: A Technician's Handbook*. Newnes, 1999.
- [28] Shruti Gudi. Research review: Low noise, high frequency synthesiser, part 1. Technical report, University of Bristol, Department of Computer Science, 2012.
- [29] Wikipedia: Delta-sigma modulation. http://en.wikipedia.org/wiki/Delta-sigma_modulation. [Online; retrieved 27/09/2012].

References

Appendix A

A.1 $\Delta\Sigma$ -digital-to-analogue-modulator

The $\Delta\Sigma$ -modulator in the PLL developed in this project translates the 32-bit digital input from the signal processor to a pulse-density modulated signal, which is then low-pass-filtered to result in a sinusoidal signal. The $\Delta\Sigma$ -modulator acts as a quantiser, which translates the high resolution input from the signal processor into a lower-resolution signal. Due to the decreasing resolution, the signal becomes more inaccurate. The added inaccuracy is known as quantisation error and adds to the noise on the output signal. The quantisation error is negatively fed back to compensate itself in the next cycle. Oversampling is used as a technique to move the noise out of the band of interest to a higher frequency, which is called noise shaping. Usually, an oversampling rate of 64 or 128 is used, which is much higher than the Nyquist rate, being only twice the baseband frequency. [28, 29]

The $\Delta\Sigma$ -modulator specified for this project's PLL is a second-order type as shown in Figure A.2, which has the advantage of lowering the noise, however, it is more likely to be unstable. The oversampling rate was chosen to be 64 and the quantiser is 1-bit wide.

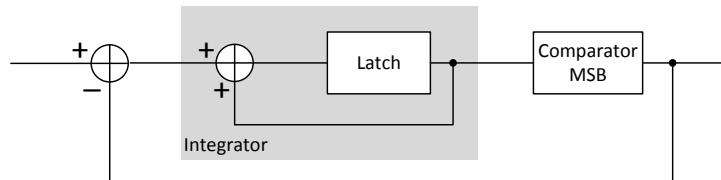


Figure A.1: 1st order Delta Sigma digital to analogue converter

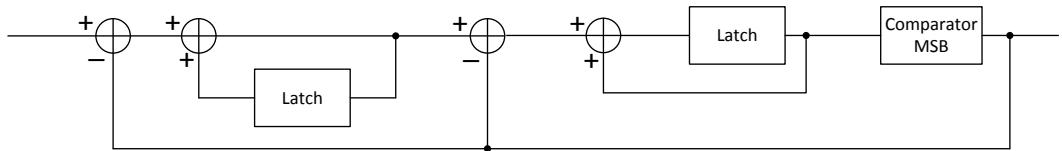


Figure A.2: 2nd order Delta Sigma digital to analogue converter

Appendix A

A low-pass filter transforms the pulse-density-modulated signal into a sine wave. The architecture of the filter was chosen to be a second-order Sallen-Key type, illustrated in Figure A.3.

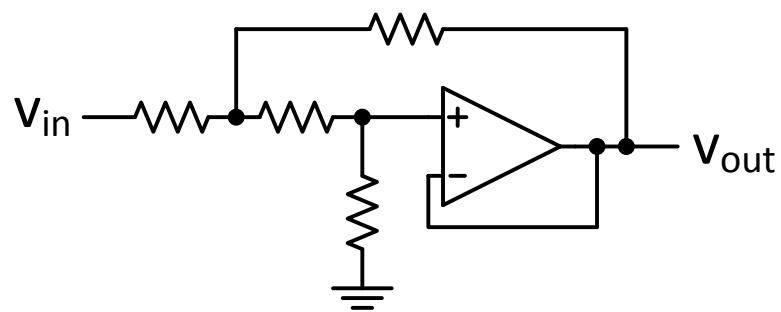


Figure A.3: Sallen-Key second order filter topology

A.2 VerilogA Code

A.2.1 Phase Detector

```
'include "constants.vams"
#include "disciplines.vams"

module PD_2(out, f_fb, f_ref);
output out;
electrical out;
input f_fb;
electrical f_fb;
input f_ref;
electrical f_ref;

integer level;

analog begin

@(cross(V(f_ref)-0.5, 1))
if (level > -1) level = level - 1;
@(cross(V(f_fb)-0.5, 1))
if (level < 1) level = level + 1;

V(out) <+ transition(level, 0, 1p, 1p);

end
endmodule
```

A.2.2 Signal Processor

```
'include "constants.vams"
#include "disciplines.vams"
```

```
module signalproc(out, phaseoffset);
output [15:0] out;
electrical [15:0] out;
input phaseoffset;
electrical phaseoffset;

real vin;
real vin_prev;
integer vout;
integer vout_mod;
// 10 GHz sampling frequency:
parameter sample_freq = 10.016e9;
integer counter;
parameter sample_range = 16;

analog begin

@(timer(0, 1/sample_freq))
begin
counter = counter + 1;

if(counter <= sample_range)
vin = (vin < 32768) ? vin + V(phaseoffset) : 32768;

else begin
if(vin_prev > vin) vout = vout + 1;
else if(vin_prev < vin) vout = vout - 1;
counter = 0;
vin_prev = vin;
vin = 0;
end
end
```

```

if(vout < 0)
    vout_mod = (vout > -32767) ? (-vout + 32768) :
        (32767 + 32768);
else
    vout_mod = (vout < 32767) ? vout : 32767;

V(out[0]) <+ transition(vout_mod & 1,0,1p,1p);
V(out[1]) <+ transition((vout_mod & 2)>>1,0,1p,1p);
V(out[2]) <+ transition((vout_mod & 4)>>2,0,1p,1p);
V(out[3]) <+ transition((vout_mod & 8)>>3,0,1p,1p);
V(out[4]) <+ transition((vout_mod & 16)>>4,0,1p,1p);
V(out[5]) <+ transition((vout_mod & 32)>>5,0,1p,1p);
V(out[6]) <+ transition((vout_mod & 64)>>6,0,1p,1p);
V(out[7]) <+ transition((vout_mod & 128)>>7,0,1p,1p);
V(out[8]) <+ transition((vout_mod & 256)>>8,0,1p,1p);
V(out[9]) <+ transition((vout_mod & 512)>>9,0,1p,1p);
V(out[10]) <+ transition((vout_mod & 1024)>>10,0,1p,1p);
V(out[11]) <+ transition((vout_mod & 2048)>>11,0,1p,1p);
V(out[12]) <+ transition((vout_mod & 4096)>>12,0,1p,1p);
V(out[13]) <+ transition((vout_mod & 8192)>>13,0,1p,1p);
V(out[14]) <+ transition((vout_mod & 16384)>>14,0,1p,1p);
V(out[15]) <+ transition((vout_mod & 32768)>>15,0,1p,1p);

end
endmodule

```

A.2.3 DAC

```

`include "constants.vams"
`include "disciplines.vams"

module DAC(f_I, f_Q, in);
output f_I;

```

```

electrical f_I;
output f_Q;
electrical f_Q;
input [15:0] in;
electrical [15:0] in;

integer vin;
integer vin_prev;
real f_sample;
parameter resolution = 12; // 12 Hz/bit resolution
real freq;
integer n;
real offset;

analog begin

@(initial_step) freq = 623e6;
@(initial_step) offset = 3.8e6;
@(initial_step) f_sample = 626000000;

@(timer(0, 1/f_sample))
begin
    vin_prev = vin;
    vin = V(in[0]) + 2*V(in[1]) + 4*V(in[2]) + 8*V(in[3])
        + 16*V(in[4]) + 32*V(in[5]) + 64*V(in[6])
        + 128*V(in[7]) + 256*V(in[8]) + 512*V(in[9])
        + 1024*V(in[10]) + 2048*V(in[11]) + 4096*V(in[12])
        + 8192*V(in[13]) + 16384*V(in[14]);
    if(V(in[15]) == 1) vin = -vin;
    freq = offset + (vin)*resolution ;
end

V(f_I) <+ sin(`M_PI*2*idtmod(freq,0,1,-0.5));
V(f_Q) <+ cos(`M_PI*2*idtmod(freq,0,1));

```

```
end
endmodule
```

A.2.4 Quadrature Frequency Converter

```
'include "constants.vams"
'include "disciplines.vams"

module mul(RF, IF_I, IF_Q, LO_I, LO_Q);
output RF;
electrical RF;
input IF_I;
electrical IF_I;
input IF_Q;
electrical IF_Q;
input LO_I;
electrical LO_I;
input LO_Q;
electrical LO_Q;

real i_var;
real q_var;

analog begin

    i_var = V(IF_I)*V(LO_I);
    q_var = V(IF_Q)*V(LO_Q);

    V(RF) <+ transition(i_var-q_var, 0, 1p, 1p);

end
endmodule
```

A.2.5 Feedback Divider

```
'include "constants.vams"
'include "disciplines.vams"

module FB_Div(out, in);
output out;
electrical out;
input in;
electrical in;

integer counter;
parameter real divisor = 4;
integer amplitude;

real previous;
real curr;
real period;

analog begin

@(initial_step) amplitude = 1;
@(cross(V(in)-0,1))
begin
    counter = counter + 1;
    previous = curr;
end

if(counter>= divisor/2)
begin
    counter = 0;
    amplitude = amplitude ^ 1;
end
```

```

V(out) <+ transition(amplitude, 0, 10p, 10p);
end
endmodule

```

A.2.6 Oscillator divider

```

'include "constants.vams"
'include "disciplines.vams"

module Divider_DFF_final(f_I_p, f_Q_p, f_in_m, f_in_p);
output f_I_p;
electrical f_I_p;
output f_Q_p;
electrical f_Q_p;
input f_in_m;
electrical f_in_m;
input f_in_p;
electrical f_in_p;

parameter real v0=-1;
parameter real v1=1 from (v0:inf);
parameter integer dir=0 from [-1:1];
parameter real td=0 from [0:inf);

```

```

parameter real tt=0 from [0:inf);
integer actNow, out;
real thresh;
real D1, D2, temp;

analog begin
@(initial_step) begin
D1 = 1;
D2 = 1;
end

actNow = 0;
@(initial_step or cross(V(f_in_p) - V(f_in_m), dir)) begin
temp = D2;
D2 = D1;
D1 = -temp;
end

V(f_I_p) <+ transition(D2, td, tt);
V(f_Q_p) <+ transition(D1, td, tt);

end

endmodule

```