An FM Chirp Waveform Generator and Detector for Radar

Modulator and Mixer Subsystems

by

Roderick Tapia Barroso,
Dimme de Groot

In partial fulfilment of the requirements for the degree of

Bachelor of Science
In Electrical Engineering

Supervisors:

dr. S.M. Alavi,
dr. M. Babaie

June 29, 2020

Delft University Of Technology
Faculty of Electrical Engineering, Mathematics and Computer Science
Electrical Engineering Programme

TU Delft

Abstract

This thesis presents two submodules which are part of 'an FM Chirp Waveform Generator and Detector for Radar', namely the mixer and the modulator. The goal of this project is to design and simulate these submodules from an educational perspective. The modulator is implemented as a single-ended common-base Colpitts oscillator with modulation capabilities added by means of varactor diodes. The carrier frequency of the presented modulator is tunable from 88 MHz to 108 MHz and the achieved modulation linearity for a sawtooth baseband signal is approximately 34 dB over the tuning range. The presented mixer consist of a double-balanced Gilbert-type topology with 29.5 dB (tunable) gain, 23 mW power consumption, high Input-Output isolation and single-ended output achieved by means of an active load. The project was carried out in the scope of the bachelor graduation project in the bachelor of electrical engineering at the Delft University of Technology.

Preface

During the last three years we have been Bachelor students of Electrical Engineering at the Delft University of Technology. During these past years we have learned a lot and we are happy to say that this thesis allowed us to apply a large amount of that knowledge. The thesis has been written during the COVID-19 pandemic of 2020, which prevented us from realising a physical prototype, but gave us the opportunity to get a better and deeper understanding of the subject.

Writing this thesis has been a very rewarding and enjoyable step into the world of RF microelectronics. That would not have been possible without the help and guidance from our supervisors: Dr. Masoud Babaie and dr. S. Morteza Alavi, whose mailbox was always open. Remarkably, we both started and will end our bachelor in the (virtual) classroom of dr. Morteza.

Furthermore we would like to thank our families for supporting us throughout our studies. We would also like to thank our teammates: Abbas Sabti, Filip Bradarić, Lars Bouman and Youri Blom, for being good friends throughout our studies and engaging us in many fruitful discussions. We would also like to thank dr. Ioan Lager for putting up a great bachelor graduation project even in the current situation.

Roderick G. Tapia Barroso,
Dimme de Groot,
Delft, June 2020

Contents

Part I

System Overview

1. Project Introduction

In this project, an FM chirp generator and detector for radar is developed [1].

RAdio Detection And Ranging (radar) is a technique used to determine range, angle and velocity of objects [2]. Currently, radar is mostly used to detect objects such as aircraft, ships and cars. It has applications in astronomy, geology, and weather forecasting amongst others [2], [3].

A schematic overview of a radar system can be seen in Figure 1.1. The principle on how radar operates is by radiating an Electromagnetic (EM) wave towards an object and measuring the reflected wave.

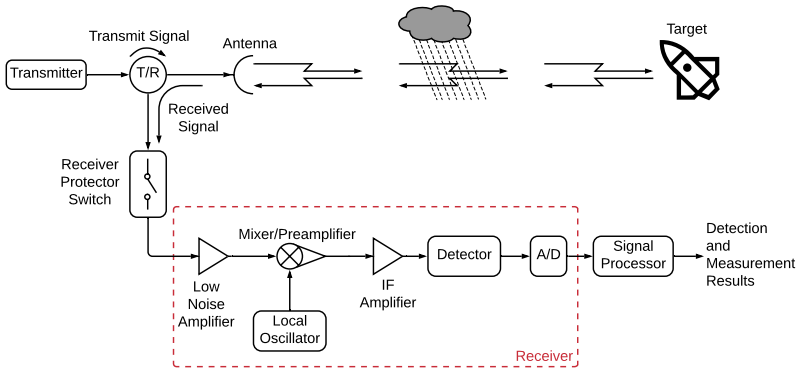


Figure 1.1: Major elements of the radar transmission/reception concept (based on [4]).

With the growing interest for autonomous vehicles, radar has come to play an important role in the automotive industry. Frequency-Modulated Continuous-Wave (FMCW) is often used in these radar systems [5]. State-of-the-art radar systems and a basic explanation on their operation can be found in Section 2.2.

The system designed in this thesis consists of two main parts, the first part is the transmitter (generally referred to as TX), which is used for generating and transmitting the chirp signal. The second part is the receiver (generally referred to as RX), which is used to receive and recover the sawtooth signal by demodulating the chirp signal. A top-level block-diagram of the two parts is given in Figure 1.2.

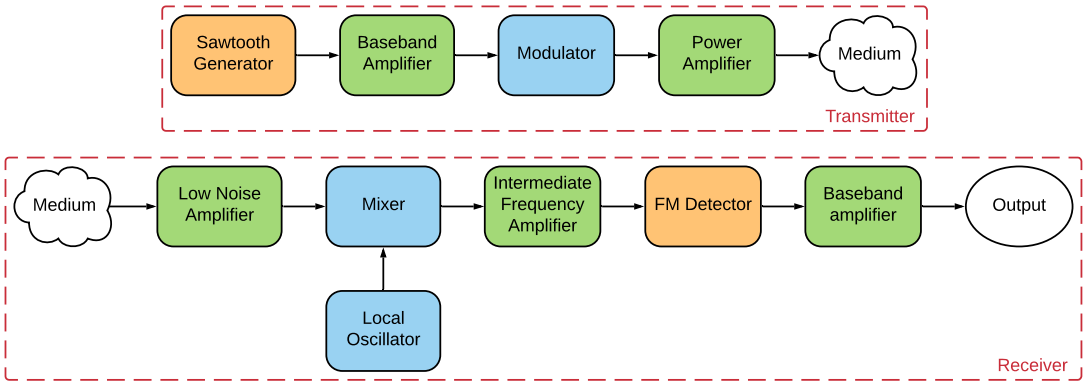


Figure 1.2: A top level block-diagram of the receiver and transmitter architecture.

As can be seen, the system consists of many parts. These parts can be divided in a few subgroups, which are briefly described below.

- **The sawtooth generator** is formed by the sub-block depicted on the upper left of Figure

1.2. As the name states, it is used to generate a sawtooth waveform. The generated signal is also referred to as the baseband signal.

- **The amplifiers** are used to amplify the signal at different places in the system. In Figure 1.2, five amplifiers are depicted. The baseband and the intermediate frequency (IF) amplifier are used to amplify signals in the baseband frequency and in the IF, respectively. Both frequencies are lower than the transmit (or radio) frequency. A Power Amplifier (PA) is used to amplify the FM-signal and also forms the interface with the antenna. At the receiver side, a Low Noise Amplifier (LNA) is used to amplify the received signal. Both the PA and LNA operate at the Radio Frequency (RF). Finally, a baseband amplifier is used to buffer the signal at the output of the detector.
- **The modulator** and the **FM detector** are respectively used to modulate and demodulate the signal. The modulator oscillates at the RF, with the instantaneous frequency being determined by the amplitude of the baseband signal. This ultimately results in a chirp signal at the output. The detector is used to recover the original baseband signal. This is achieved by extracting the message signal from the phase of the IF signal to create a baseband signal.
- **The mixer** and the **Local Oscillator (LO)** are used to shift the frequency of the output of the LNA to an IF. This simplifies the design of subsequent stages in the receiver. It should be noted that, just like the modulator, the LO is a Voltage-Controlled Oscillator (VCO).

1.1 Project Objective and Problem Definition

The rise of autonomous vehicles has sparked the interest in high-end radar sensors from industry. This project aims to introduce students to RF circuit design from an educational perspective by allowing them to design the needed subsystems for a basic FM chirp waveform transmitter and receiver.

As the attentive reader might have noticed, the proposed system architecture does not actually resemble a complete radar system. Because the development of a full radar system lies beyond the scope of a Bachelor thesis, this project constrains itself to the development of a separate receiver and transmitter part. Ultimately, the goal is to be able to modulate a sawtooth signal and transmit this over a range of at least five meters, after which the sawtooth signal should successfully be recovered at the output.

To facilitate designing the system in three subgroups, the previously described modules are split into three parts.

1. The amplifiers
2. The modulator, mixer and local oscillator
3. The FM detector and the sawtooth generator

This thesis focuses on the development and simulations of the mixer, the modulator and the LO.

1.2 Methodology

In order to arrive at a working transceiver design, it is important to have the means to reliably simulate and validate the designed circuits. For this purpose, Advanced Design System (2020) (ADS) is used. ADS is a software package developed by Keysight technologies. Furthermore MATLAB is used for processing the data obtained from ADS. It is also used to generate figures of the data for aesthetic reasons. Finally, Lucidchart is used to make figures with illustrative purposes.

Note that, under normal circumstances, the designed circuits would have been validated by physical prototyping. However, in light of the current COVID-19 pandemic, the circuit validation is also done in ADS.

1.3 Thesis synopsis

The thesis is divided into two main parts. In the first part the design of the modulator is discussed and the design is verified by means of simulation. In the second part the same is done for the mixer. Both parts follow the same structure, starting with a brief introduction after which some theoretical background is discussed. After that, the requirements are given, the design rules are derived and the design is presented. The designs are then simulated using ADS, after which the results are discussed and some recommendations for future work are given. Finally a joint conclusion is presented.

2. Frequency Modulation

In radio technology, signals are typically modulated before transmissions using different modulation techniques. An advantage of using modulation techniques is that a specific bandwidth can be assigned to every user, such that there is little interference. One of the modulation techniques is Frequency Modulation (FM), which is used for this project. An advantage of FM is that the amplitude of the carrier is constant which means that the power is constant. Another advantage is that the information is not present in the amplitude of the signal meaning that it is less susceptible to noise [6]. This is not the case for Amplitude Modulation, where the information signal is incorporated in the amplitude of the carrier.

2.1 The mathematical theory

In FM, the instantaneous frequency of the transmitted signal is changed according to the information signal, as shown in Equation (2.1) and Equation (2.2) found in [7].

$$s(t) = A_c \cos\left[\omega_c t + \theta(t)\right] \tag{2.1}$$

where

$$\theta(t) = D_f \int_{-\infty}^{t} m(\tau) d\tau \tag{2.2}$$

where D_f is the frequency deviation constant, $m(t)$ is the information signal and ω_c is the carrier frequency.

The instantaneous frequency of the signal is $f_i(t) = f_c + \frac{1}{2\pi} D_f m(t)$. To make sure that the frequency range does not interfere with other neighbouring signals, the deviation of the frequency should not be too large. The frequency deviation can be determined by Equation (2.3).

$$\Delta F = \frac{1}{2\pi} D_f V_p \tag{2.3}$$

where

$$V_p = \max\left[m(t)\right] \tag{2.4}$$

A practical example of FM can be seen in Figure 2.1 where a carrier wave is being modulated by a sawtooth signal. The frequency range of the transmitted signal is centred around ω_c. At the receiver, this signal has to be demodulated to obtain the original signal.

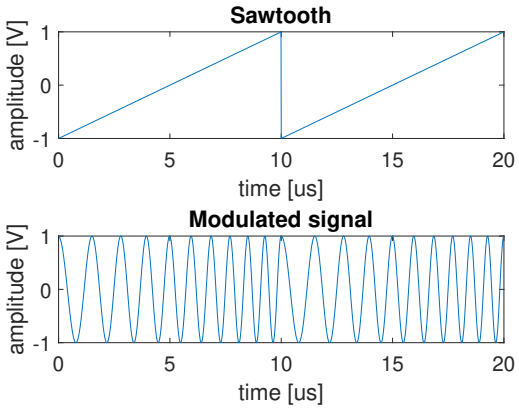


Figure 2.1: The waveform of the sawtooth signal and the modulated signal.

2.2 Frequency-Modulated Continuous-Wave (FMCW)

Frequency-Modulated Continuous-Wave (FMCW) is a special type of FM where the modulation signal is a periodic wave. The reflected signal can be used to estimate the speed and distance to objects. The transmitter and receiver are placed at the same location and the receiver will receive the reflected signal that was sent by the transmitter. When comparing the difference between the received signal and the transmitted signal, the distance can be determined. A visualisation of the relevant parameters is shown in Figure 2.2.

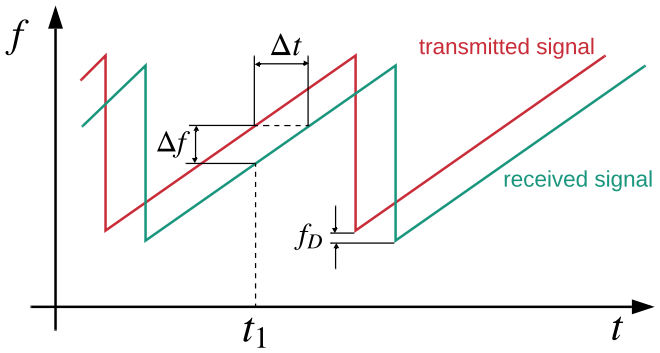


Figure 2.2: The relevant parameters for determining the distance using FMCW.

From Figure 2.2, it can be seen that the received signal has a phase shift compared to the transmitted signal caused by the propagation time. Since the signal is propagating with the speed of light c_0, the distance R can be determined using Equation (2.5) [8].

$$R = \frac{c_0|\Delta t|}{2} = \frac{c_0|\Delta f|}{2\frac{df}{dt}} \tag{2.5}$$

There is a maximal distance that can be measured, given by Equation (2.6), due to the fact that the signal is periodic. This equation is derived in [9, p. 17]. If the phase shift is larger than the period time T, it cannot be distinguished from which period the signal originated.

$$R_{max} = \frac{c_0 T}{2} \tag{2.6}$$

FMCW is not only limited to measuring the distance, but also the speed can be measured. Due to the Doppler effect [10], the frequency spectrum of a signal will shift by an amount of f_D. This frequency shift can be measured and therefore be used to determine the speed. For this project, a sawtooth is used as the modulation signal, for which the Doppler effect has a negligible influence [8] and cannot be distinguished from Δf. Therefore, the transceiver of this project cannot be used to accurately determine the distance of moving objects nor their speed.

3. Programme of Requirements

The goal of the project is to design an FM chirp generator and detector. The application in mind would be to integrate the design in cars for remote sensing applications. The complexity of the design for such an application lies outside the scope of this Bachelor project, since it requires the design to be in the GHz range (FMCW) with IC technology. For educational purposes, the requirements will be simplified. The operational frequency must be in the FM radio band and only discrete components are allowed. Needless to say, the end result of this project will not be near an applicable nor commercially competitive consumer product. The legal limit of power transmission in the FM radio band without a license will therefore also not be considered.

3.1 System requirements

The following specifications and requirements apply to the full transceiver system. The requirements are given by, or derived from, the specifications provided by the project supervisor [1]. Requirements governing the performance of the subsystem are discussed in the respective theses and chapters, and follow from the following:

Functional requirements:

- SYS0: The transceiver must be composed of only discrete components.
- SYS1: Frequency Modulation (FM) must be used.
- SYS2: The modulation signal must be a sawtooth waveform.
- SYS3: The sawtooth waveform must be recoverable over a minimum distance of 5 m.
- SYS4: The receiver must drive the recovered signal over a 50 Ω dummy load.

Non-Functional requirements:

- SYS5: The Radio Frequency (RF) carrier wave must be tunable from 88 MHz to 108 MHz.
- SYS6: The FM modulation bandwidth must be between 180 kHz and 400 kHz.
- SYS7: The receiver noise figure must be lower than 5 dB.
- SYS8: The Signal-to-Noise Ratio (SNR) at the receiver output must be greater than 30 dB.
- SYS9: The transmitted power must be at least 100 mW.
- SYS10: The transmitter efficiency must be higher than 50%.
- SYS11: The total power consumption of the transceiver must be less than 10 W.
- SYS12: The transceiver must operate on a 12 V power supply.
- SYS13: The antennas provide a 50 Ω resistive load or source impedance.

3.2 Assumptions and disclaimers

- The detected waveform is only used to drive a 50 Ω dummy load. Processing the signals to perform distance measurements is outside the scope of the project.
- Antennas are assumed to be 50 Ω purely resistive: This simplifies the project to fit in the available timespan.
- The specified transmission power in the FM radio band is illegal [11]. However, the design will not be physically realized.
- The power consumption limit and supply voltage are chosen with the idea to be suitable for car applications.

Part II

Modulator

4. Introduction and Theoretical Background

4.1 Introduction

In this part, the background theory, design and results of the modulator are discussed. As is evident from the name, the modulator is used to modulate the baseband signal onto a carrier wave by means of FM; concretely, this means that the frequency of the carrier wave is varied based on the amplitude of the baseband signal. For a sawtooth signal, this results in a chirp signal at radio frequency at the output of the modulator. An illustration of this is given in Figure 4.1.

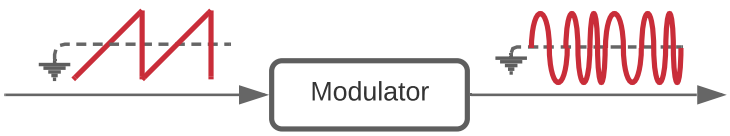


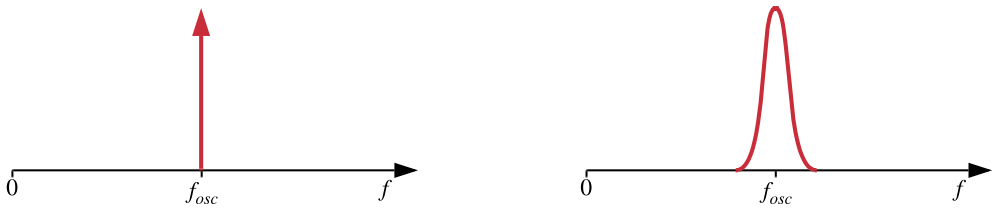
Figure 4.1: A top-level overview of the modulator. The sawtooth signal at the input results in a chirp signal at the output.

Since the modulator is implemented as an oscillator with a variable frequency, some theory behind oscillators is discussed first. Then, in Chapter 5, the requirements for the modulator are discussed. In Chapter 6, the design considerations are derived and the design is presented. In Chapter 7, the design is simulated and verified based on the requirements. Finally, a discussion about the results and future work can be found in Chapter 8.

4.2 Theoretical Background

In this section, first some general theory behind oscillators is discussed, after which some oscillator topologies are briefly introduced and a method to extend the oscillator to a VCO is given. The presented method is used to both make the LO and the modulator. Finally some performance metrics are presented.

An oscillator is a device that is used to convert a DC signal into an AC signal with a certain frequency f_{osc}. This frequency can, among others, be set by using a tuned LC-circuit in the feedback path or by using a piezoelectric crystal. Since in this project the latter is not allowed, we will focus on the former. An ideal oscillator would produce a perfect sine wave in the time domain, and thus be an impulse in the frequency domain. In practice, the spectrum will be 'smeared out' a bit due to noise. This is illustrated in Figure 4.2.



(a) Ideal spectrum

(b) More realistic spectrum

Figure 4.2: An ideal frequency spectrum for an oscillator versus a more realistic spectrum.

Except for the element which can be used to determine the frequency, two other parts must be present in a practical oscillator: an active device with a voltage gain at the frequency of oscillation, and a mechanism that limits and stabilises the output amplitude [12].

4.2.1 Linear Feedback Oscillators

For a circuit to be able to start oscillating it should have two small-signal complex conjugate poles which lie in the right-hand side of the complex plane [12]. When the circuit is now excited by noise or due to a current impulse when the circuit is turned on, oscillations which grow in amplitude will occur: the system is unstable because of the right-hand side poles, and it oscillates because they are complex.

Since we eventually want to obtain a constant amplitude sinewave, the amplitude's growth must stop at a certain point. This is realised by designing the system such that some of its parameters change as the envelope of the sine wave increases. Changing the parameter must ultimately result in the real part of the poles going to zero. The varied parameter is usually the amount of amplification.

A system with the correct behaviour can be realised by using feedback. A general model for a feedback amplifier is given in Figure 4.3.

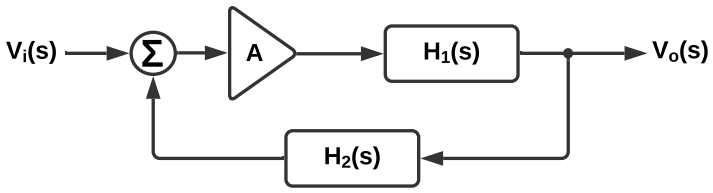


Figure 4.3: A general model of a feedback amplifier.

The transfer function of the feedback amplifier is given by Equation (4.1).

$$\frac{V_o(s)}{V_i(s)} = \frac{AH_1(s)}{1 - AH_1(s)H_2(s)} \tag{4.1}$$

In general, the poles of Equation (4.1) are given by the solution of Equation (4.2).

$$1 - AH_1(s)H_2(s) = 0 \tag{4.2}$$

The term $AH_1(s)H_2(s)$ is called the loop gain $A_L(s)$ of the system. From Equation (4.1), it follows that, if the poles indeed are complex conjugates lying in the right-half plane, $v_0(t)$ will be a growing sinusoid. It is evident that, when designing an oscillator, care must be taken that for *small signals* (e.g. noise) poles must be positioned such that they lie in the right half of the complex plane.

From Equation (4.2), the well known Barkhausen criterion can be derived, this criterion provides conditions that are necessary (but not sufficient) for stable oscillations to occur and is given by Equation (4.3).

$$\mathrm{Re}\{A_L(j\omega_{osc})\} = 1 \tag{4.3a}$$
$$\mathrm{Im}\{A_L(j\omega_{osc})\} = 0 \tag{4.3b}$$

In the equation, $\omega_{osc} = 2\pi f_{osc}$ and $s = j\omega_{osc}$. Intuitively, the Barkhausen criterion tells us that, at steady state, the phase of the signal must not be shifted (or be shifted an integer multiple of 360°) during the traverse of the loop at ω_{osc}. Also the signal must not be attenuated at this frequency.

An extended definition is used by [13], here the first condition is divided into two parts: one part during startup, where the gain is higher than one, and one part for the steady state, where the gain is equal to one. This is summarised in Equation (4.4).

$$\mathrm{Re}\{A_L(j\omega_0)\} > 1, \quad \text{during startup,} \tag{4.4a}$$
$$\mathrm{Re}\{A_L(j\omega_0)\} = 1, \quad \text{while in steady state} \tag{4.4b}$$
$$\mathrm{Im}\{A_L(j\omega_0)\} = 0 \tag{4.4c}$$

The extended criterion is again intuitive: for the oscillations to start, the present small signal must be amplified. After that, the real part of the poles starts moving to zero until equilibrium is reached. In practical circuits, the non-linearity of the amplifier devices (such as a BJT) can be used to make the real part of the poles move to zero.

4.2.2 LC-circuits

The circuit topologies that are considered in Section 4.2.3 make heavy use of a so called LC-tank. An example of such a circuit is given in Figure 4.4. Here R_P is the parallel equivalent of the series resistances that are present in the inductor and the capacitor. For a slightly more complete description of LC-circuits, the reader can refer to Appendix B.

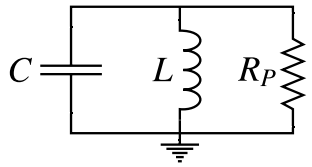


Figure 4.4: A parallel LC-circuit together with its equivalent parallel resistance R_P.

In oscillators, LC-circuits are often used to set the frequency of oscillation. The frequency of oscillation ω_{osc} equals the resonance frequency ω_0 of the tank. It is given by Equation (4.5).

$$\omega_0 = \frac{1}{\sqrt{LC}} \tag{4.5}$$

A commonly used parameter of tank circuits is their Q-factor. This factor is the ratio between the resonance frequency and the bandwidth. Intuitively, this can be interpreted as how rejective the circuit is to frequencies other than the resonance frequency. Q can be calculated using Equation (4.6).

$$Q = R_P\sqrt{\frac{C}{L}} \tag{4.6}$$

For reasons that will become clear in Chapter 6, in oscillators one wants to have the value of Q as high as possible.

4.2.3 Oscillator topologies

There exist many different oscillator topologies. Three of the main types that can be considered are RC-oscillators, ring oscillators and LC-oscillators. The former is not used, since it is less suitable for high frequency applications than the other two, and it is also relatively hard to tune f_{osc} [14]. One of the other options, the ring oscillator, produces a square-like wave, while ideally we want to produce a sinusoid. Hence, the chosen oscillator type is the LC-oscillator.

Three considered LC-oscillators are the Hartley oscillator, the Colpitts oscillator and the Clapp oscillator. The Hartley oscillator uses a tapped inductor to set the feedback, while both the Clapp and Colpitts oscillator use a capacitive divider network to realise the feedback. For all three oscillators, it holds that their frequency can be tuned easily and that their output amplitude is relatively stable over the frequency tuning range. However, the output of the Hartley oscillator is rich in harmonics [15].

The Clapp and Colpitts oscillator do not have the previously mentioned disadvantage [16], [17]. Furthermore, they are largely similar, with the difference being that in the Clapp oscillator a capacitor is placed in series with the inductor. A large advantage of this is that, if chosen correctly,

the capacitive feedback network will have almost no influence on f_{osc}. A disadvantage of the Clapp oscillator with respect to the Colpitts oscillator is that it has more stringent start up requirements [18]. Ultimately, the Colpitts oscillator is chosen, also due to the large amount of information available.

4.2.4 Voltage Controlled Oscillators

As the name states, in a VCO the frequency of oscillation is determined by an applied voltage. The frequency-voltage characteristic of an ideal VCO is described by equation (4.7).

$$\omega_{out} = K_{VCO} V_{cont} + \omega_0 \tag{4.7}$$

In the equation, the constant K_{VCO} is called the sensitivity of the oscillator and V_{cont} is the applied voltage. To implement the VCO, one needs to vary one of the parameters of the LC-tank. In practice, the capacitance is varied, this because it is very difficult to change inductances electronically.

The capacitance is varied using so called "varactors". These varactors can be implemented using reverse-biased pn-junctions. Since the width of the depletion layer changes, so does the capacitance. The capacitance C_j can be estimated by Equation (4.8) [19].

$$C_j = \frac{C_{j0}}{(1 + {}^{V_D}/_{V_0})^m} \tag{4.8}$$

In the equation, C_{j0} is the capacitance at zero bias, V_D is the applied reverse voltage, V_0 is the junction voltage, and m is a device dependent parameter.

4.2.5 Oscillator Performance Metrics

In order to assess the performance of the modulator, metrics are needed. In this thesis, the following metrics are considered [12], [19]:

- **The output voltage swing** is a measure of how large the output amplitude is.
- **The frequency range** is the frequency range to which the oscillator can be tuned.
- **Phase Noise**: The ideal frequency spectrum of an oscillator would be an impulse. In practice this will not be the case, the spectrum is made wider due to noise being added by the different parts of the oscillator. This is measured using the concept of phase noise.
- **The power usage** is a metric of how much power is used by the oscillator.

It should be noted that a lot more metrics can be used and that there are a lot more types of noise than just phase noise. However, to stay inside the scope of a bachelor thesis, the noise type that is considered is phase noise only.

4.2.6 Phase noise

Because the phase noise is often considered the most important performance metric of an oscillator, it is briefly discussed here.

In an ideal oscillator, the period of the output signal is constant. However, in practice a small deviation is present, this is expressed by Equation (4.9).

$$V_{out} = V_0 \cos(\omega_0 t + \phi_n(t)) \tag{4.9}$$

In the equation, $\phi_n(t)$ is called the phase noise, it only refers to random disturbances in the phase [20].

The phase noise is measured at different frequency offsets $\Delta\omega$ with respect to the carrier frequency ω_0. The quantitative value at a certain frequency offset is calculated by measuring the power of the phase noise in a 1 Hz bandwidth at said offset and normalising it with respect to the carrier power. The phase noise is commonly expressed in dBc/Hz.

The (single sided) frequency spectrum of the phase noise typically consists of three regions (ignoring very small frequency offsets) [21]. In the first region (called the $1/f^3$ region) the slope is -30 dB/decade for increasing $\Delta\omega$. In the second region (called the $1/f^2$ region), the slope is -20 dB/decade. In the third region, the spectrum has become flat [21]. This is illustrated in Figure 4.5. In the figure, $L(\Delta\omega)$ refers to the phase noise at an offset $\Delta\omega$.

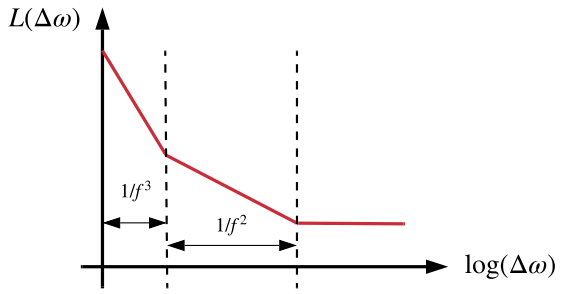


Figure 4.5: Illustration of a typical single sided phase noise spectrum.

A model for the phase noise, called Leeson's model, is given by Equation (4.10) [21]. This model is used in Section 6.1 to obtain some design guidelines.

$$L(\Delta\omega) = 10\log_{10}\left\{\frac{2FkT}{P_{sig}}\left(1+\left(\frac{\omega_0}{2Q\Delta\omega}\right)^2\right)\left(1+\frac{\Delta\omega_{1/f^3}}{|\Delta\omega|}\right)\right\} \tag{4.10}$$

In the equation, F is an empirical fitting parameter, k denotes Boltzmann's constant and T is the temperature. Often $\Delta\omega_{1/f^3}$ is treated as an empirical parameter as well, it relates to the $1/f^3$ corner frequency of the phase noise spectrum. Lastly P_{sig} is the signal power, an equation for P_{sig} is given in Section 6.1.

5. Programme of Requirements

The requirements for the modulator are found in Table 5.1.

Table 5.1: The requirements for the modulator.

Tag	Description	Value
MOD0	Peak-to-peak output voltage swing	> 6 V
MOD1	Power consumption	< 100 mW
MOD2	Higher order harmonics	< -30 dB
MOD3	DC input resistance	1.5 kΩ
MOD4	Output resistance	< 2 kΩ
MOD5	Modulation frequency linearity	> 30 dB
MOD6	Phase noise (at 10 kHz)	< -30 dBc/Hz

Furthermore, three quantitative main requirements (as given in Chapter 3) that are directly applicable to the modulator are SYS5, SYS6 and SYS12. For the sake of completeness, they are listed again below.

Table 5.2: The main system requirements that are directly applicable to the modulator.

Tag	Description	Value
SYS1	FM must be used.	-
SYS5	RF Tuning range	88 MHz to 108 MHz
SYS6	Modulation Bandwidth	between 180 kHz and 400 kHz
SYS12	Power supply voltage	12 V

It must be noted that the value of MOD2 is specified with respect to the value of the first harmonic. Furthermore, for SYS5, a maximum deviation of 0.5% at both sides of the range is considered acceptable.

It should also be noted that, while all requirements must hold for the whole RF tuning range, the actual value can change. For example, MOD0 states that the peak-to-peak output voltage swing must always be larger than 6 V. The exact value can however be ranging from, for example, 6 V to 8 V over the span of the tuning range.

Furthermore, for the reasoning behind requirement MOD1, the reader can refer to Appendix A. In here system level calculations are done to derive a power budget for the separate system components.

Finally, the requirement MOD4 is set such that it is much lower than the input resistance of the next stage. Hence it allows for designing the PA and the modulator relatively independently.

6. Design

In this chapter, the design of the modulator is presented. First, some design considerations are specified. After that, the design itself is started by first designing an oscillator and then tweaking it such that it becomes voltage controlled and so that the user can tune it to the desired center frequency.

As explained in Section 4.2.3, a Colpitts oscillator topology is used. It is configured as a common-base oscillator and an extra tank capacitance with respect to the normal Colpitts oscillator is added. The AC model of this configuration using a BJT-transistor is shown in Figure 6.1.

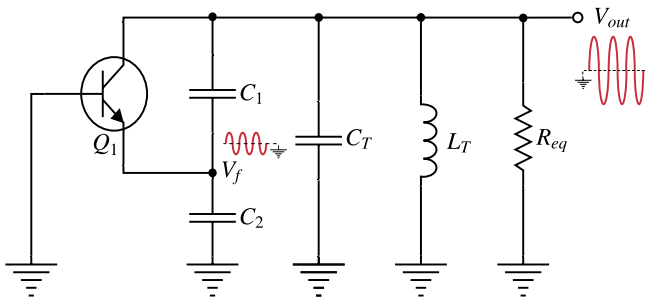


Figure 6.1: The AC-equivalent circuit of a common-base Colpitts oscillator with added tank capacitance.

As can be seen in the figure, the feedback voltage V_f is set through the capacitive divider consisting of capacitances C_1 and C_2. The resonance frequency ω_0 is determined by the reactive elements in the circuit and can be approximated by Equation (6.1).

$$\omega_0 = \frac{1}{\sqrt{L_T\left(\frac{C_1 C_2}{C_1+C_2} + C_T\right)}} \tag{6.1}$$

The transistor Q_1 serves as the amplifying element and the resistor R_{eq} models the equivalent parallel resistance of the resistances in the circuit.

6.1 Design Rules

In order to design the modulator, some design rules must be established. Ultimately, a design must be made that fulfils all requirements. The used design steps are loosely as follows:

1. Design the biasing network for the transistor.
2. Select a value for L_T.
3. Select a value for the series combination C_{eq} of C_1 and C_2.
   - Must be set so small that C_T is the predominant capacitive element in determining the frequency, but
   - Must be set so large that the transistor capacitances do not become dominant.
4. Based on requirement MOD0 and the value of R_{eq}, the ratio of C_1 and C_2 can be determined (and hence also their values).
5. Varactors and a coarse tuning mechanism are added to the design to make the system a VCO.
6. Means are provided to control the oscillation frequency via the baseband voltage.

Now that the design steps have been loosely specified we need to determine how to execute them based on the specified requirements. Since phase noise is often regarded as the most important

parameter of an oscillator, priority is placed there. As it turns out, good phase noise properties do often come with a large voltage swing as well. It is, however, a trade-off with the amount of power that is used. This is assumed not to be a problem, since requirement MOD1 provides not that stringent of a power requirement.

In order to get a slightly more quantitative feeling for the design, we first take a look at Leeson's phase noise model (given in Section 4.2.6). For the sake of completeness, the model is again given below:

$$L(\Delta\omega) = 10\log_{10}\left\{\frac{2FkT}{P_{sig}}\left(1+\left(\frac{\omega_0}{2Q\Delta\omega}\right)^2\right)\left(1+\frac{\Delta\omega_{1/f^3}}{|\Delta\omega|}\right)\right\} \tag{6.2}$$

Since we can only directly influence Q and P_{sig}, Equation (6.2) is rewritten into Equation (6.3), thereby dropping properties that are assumed to be beyond our control. To denote that the equation does not actually resemble the phase noise anymore, $L(\Delta\omega)$ is dropped in favour of $\bar{L}(\Delta\omega)$.

$$\bar{L}(\Delta\omega) = 10\log_{10}\left\{\frac{1}{P_{sig}}\left(1+\left(\frac{\omega_0}{2Q\Delta\omega}\right)^2\right)\right\} \tag{6.3}$$

The expression for Q is given by Equation (4.6). Substituting it in Equation (6.3) yields Equation (6.4).

$$\bar{L}(\Delta\omega) = 10\log_{10}\left\{\frac{1}{P_{sig}}\left(1+\frac{1}{R_{eq}^2}\frac{L_T}{C_{tot}}\left(\frac{\omega_0}{2\Delta\omega}\right)^2\right)\right\} \tag{6.4}$$

In the equation, C_{tot} is the total tank capacitance. It is calculated using Equation (6.5).

$$C_{tot} = C_{eq} + C_T = \frac{C_1C_2}{C_1+C_2} + C_T \tag{6.5}$$

We do not yet have an expression for P_{sig}. As it turns out, the amplitude V_{out} of the voltage at the output of the oscillator can be estimated using Equation (6.6) [22].

$$V_{out} \approx 2I_{C,b}R_{eq}\left(1-\frac{1}{N}\right) \tag{6.6}$$

In the equation, $I_{C,b}$ is the collector bias current of the transistor. Furthermore, N is specified by Equation (6.7). The reason for making it a specific constant is because it is used extensively later on.

$$N = \frac{C_1+C_2}{C_1} \tag{6.7}$$

Based on Equation (6.6) and the notion that the output signal is a sinusoid, P_{sig} can be estimated. This is shown in Equation (6.8).

$$P_{sig} = \left(\frac{V_{out}}{\sqrt{2}}\right)^2\frac{1}{R_{eq}} = 2I_{C,b}^2R_{eq}\left(1-\frac{1}{N}\right)^2 \tag{6.8}$$

Substituting Equation (6.8) in Equation (6.4) yields Equation (6.9).

$$\bar{L}(\Delta\omega) = 10\log_{10}\left\{\frac{1}{2I_{C,b}^2R_{eq}\left(1-\frac{1}{N}\right)^2}\left(1+\frac{1}{R_{eq}^2}\frac{L_T}{C_{tot}}\left(\frac{\omega_0}{2\Delta\omega}\right)^2\right)\right\} \tag{6.9}$$

From Equation (6.9) some design rules to minimise phase noise can be derived:

- L_T should be chosen to be small.
- $I_{C,b}$ should be made large.

Of course, there are practical limits to how good we can adhere to the above list. Since the output voltage swing cannot be too large (since than both the limit of the supply voltage and the breakdown region of the transistor would be reached), $I_{C,b}$ cannot be made too large for a given R_{eq}. This would also have negative consequences on the amount of power drawn from the source. Furthermore, as becomes clear in Section 6.2.2, the value of L_T cannot be chosen too small because otherwise the capacitance of the considered varactors is not high enough to keep the amount of varactors down to a reasonable number.

Now that some general rules have been derived, also a method of choosing the feedback ratio N must be derived. This is done by considering the small-signal open-loop gain. When deriving the equation, a low-frequency version of the hybrid-pi model is used (see [23]). This derivation yields Equation (6.10). For the derivation, the reader can refer to Appendix C.

$$N = \pm\sqrt{\left(\frac{g_m R_{eq}}{2A_L}\right)^2 - g_m R_{eq}} + \frac{g_m R_{eq}}{2A_L} \tag{6.10}$$

In the equation, g_m is the small signal transconductance and A_L is the magnitude of the loop gain. The small signal transconductance is given by Equation (6.11) [24].

$$g_m = \frac{I_{C,b}}{V_T} \tag{6.11}$$

Where V_T is the thermal voltage. At room temperature, it equals approximately 26 mV.
From Barkhausens criterion during startup (see Equation (4.4)), it is known that A_L should be larger than one. It should be noted that the equation yields two values for N. In general, it is best to choose the largest value. This because it yields a larger value for C_2, which in turn means that the capacitive voltage divider will do a better job approximating an ideal capacitive voltage divider. For a short discussion on capacitive voltage dividers, the reader can refer to Appendix D.

From the value of N and a chosen value for C_{eq}, the capacitances C_1 and C_2 can be calculated using Equation (6.12).

$$C_1 = \frac{NC_{eq}}{N-1} \tag{6.12a}$$

$$C_2 = C_1(N-1) \tag{6.12b}$$

6.2 The design

In this section the design is presented. It is based on the common-base Colpitts topology as given in [25]. The symbolic circuit, still without means to control f_{osc}, is shown in Figure 6.2. After designing the oscillator itself, C_T will be replaced with a set of varactors. In the circuit, $R_{B,1}$, $R_{B,2}$ and R_E are used to bias the transistor. The capacitors $C_{S,1}$ and $C_{S,2}$ serve as AC shorts.

The biasing network

In this section, the design of the bias network is done. From the previously found results, it is clear that the biasing must be set such that the transistor can afford a large voltage swing. The used transistor is the BFU550A from NXP (see [26] for the datasheet), for reasons explained in Appendix E. The chosen type of biasing network is divider biasing with emitter degeneration because it provides good temperature stability. The equations used for the biasing are explained in Appendix F.

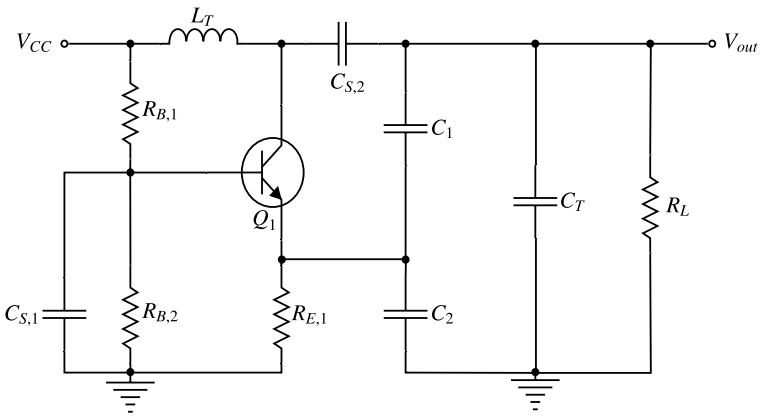
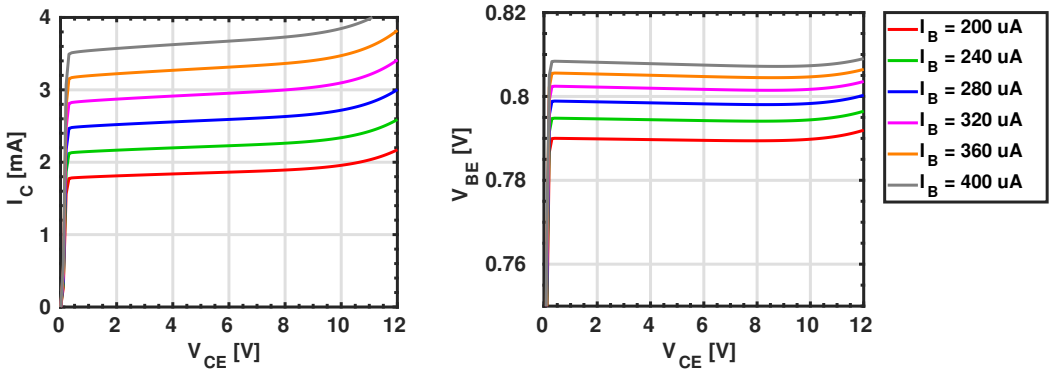


Figure 6.2: The complete symbolic circuit for the designed Colpitts oscillator.

Since current spikes are present in a Colpitts oscillator, the transistor must be biased comfortably below the maximum limit of 50 mA, hence a biasing collector current $I_{C,b}$ of 2 mA is chosen. The transistor curves at this current level are shown in Figure 6.3.



(a) I_C vs V_{CE} (b) V_{BE} vs V_{CE}

Figure 6.3: The I_C vs V_{CE} and the V_{BE} vs V_{CE} curve of the BFU550A in the neighbourhood of the chosen biasing point.

From the figures, it can be seen that the base-emitter voltage V_{BE} is approximately 0.8 V at the chosen biasing point. Furthermore, it should be avoided that the collector-emitter voltage V_{CE} exceeds approximately 10.5 V. Hence, the biasing voltage $V_{CE,b}$ is set to 5.5 V. At this point, the common-emitter current gain (h_{FE}) equals approximately 90. The emitter resistance R_E is now calculated using Equation (6.13).

$$R_E = \frac{V_{CC} - V_{CE,b}}{I_{C,b}} = \frac{12 - 5.5}{2 * 10^{-3}} = 3.2 \text{ k}\Omega \tag{6.13}$$

To calculate the ratio $R_{B,1}$ over $R_{B,2}$, Equation (6.14) is used.

$$\frac{R_{B,1}}{R_{B,2}} = \frac{V_{CC} - I_{C,b}R_E - V_{BE}}{I_{C,b}R_E + V_{BE}} = \frac{5.5 - 0.8}{12 - 5.5 + 0.8} \approx 0.64 \tag{6.14}$$

The actual values for $R_{B,1}$ and $R_{B,2}$ are chosen such that Equation 6.15 holds. This is needed to actually get the advantages corresponding to the used type of biasing.

$$R_{B,1} \parallel R_{B,2} \ll h_{FE} R_E \approx 288 \text{ k}\Omega \tag{6.15}$$

Setting the parallel combination of $R_{B,1}$ and $R_{B,2}$ equal to approximately 12 kΩ, we find $R_{B,1} =$ 18 kΩ and $R_{B,2} = 30$ kΩ. Both resistors are rounded down to the nearest E-24 series value. An ADS simulation reveals that the actual obtained collector current is indeed 2.0 mA. The collector emitter voltage is 5.53 V, which is close enough to the estimated value.

6.2.1 Choosing the inductor and capacitive divider values

In this section, the value of L_T is chosen and the values for the capacitive divider are calculated. For the inductor, a small value is preferred. However, it also needs to have a low series resistance and it should not be so small that a lot of varactors are needed to get the system oscillating at the tuning frequency band. A 27 nH inductor is chosen, in Section 6.2.2 it is verified that this is a reasonable value. The chosen inductor has a series resistance of 4 mΩ.

From the chosen inductor value and based on requirement SYS5, the minimum and maximum value of C_{tot} can be estimated.

$$C_{tot,min} = \frac{1}{L_T(2\pi f_{osc,max})^2} \approx 80.4 \text{ pF},$$
$$C_{tot,max} = \frac{1}{L_T(2\pi f_{osc,min})^2} \approx 121 \text{ pF}$$

Due to linearity considerations, C_{tot} should be primarily formed by C_T. Hence, the "capacitance budget" for C_{eq} is about 10 pF. Furthermore, under the assumption that the series resistance of the used capacitors is low, the load R_L is assumed to equal R_{eq} and is set to 1.4 kΩ.

Using Equation (6.10), the value of N is determined. A value $A_L = 4$ is used, furthermore $g_m \approx 40 * 2 * 10^{-3} = 0.08$ S. The found values for N are 23.2 and 4.8.

By now using the large value for N, setting $C_{eq} = 7$ pF and filling in Equation (6.12), we find $C_1 = 7.3$ pF and $C_2 = 162$ pF. Rounding down to the nearest E-24 value, C_1 is found to be 6.8 pF and C_2 equals 160 pF.

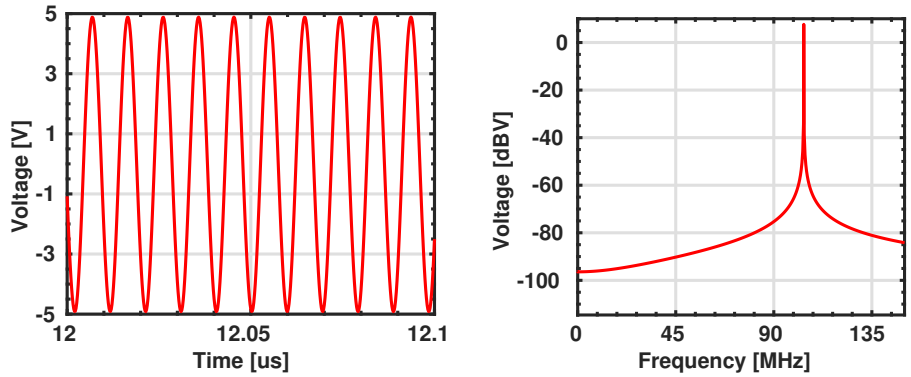
To verify that the design works and to get an estimate of C_{eq} when the transistor capacitances are present, the design is simulated in ADS. The shorting capacitors $C_{S,1}$ and $C_{S,2}$ are set to 3.3 μF, and C_T is set to 80 pF. The output and *part of* the corresponding Fourier transform are shown in Figure 6.4.

From the spectrum, it is found that f_{osc} equals 103.8 MHz, hence C_{eq} is estimated to be 7.1 pF.

6.2.2 Adding the Varactor Diodes

As the oscillator has been designed, it is time to extend it so that it can be used as a modulator. This is done using varactor diodes. When adding the varactor diodes, care must be taken that they are reverse biased at all times. Furthermore, multiple topologies exist to add them to the circuit.

The chosen topology is to add multiple anti-series configured varactors in parallel. They are placed in anti-series to reduce the influence of the present RF-signal: when the reverse voltage over the upper varactor goes up, the reverse voltage over the lower varactor goes down. A disadvantage of this setup is that the total capacitance is about half of that when using a single varactor diode. To overcome this issue, multiple anti-series configured varactor diodes are placed in parallel. Far more advanced topologies exist (see [27]), however those are beyond the scope of the thesis. The used topology is schematically shown in Figure 6.5, it is added in place of the tank capacitor C_T.



(a) Time domain (b) Frequency domain

Figure 6.4: Part of the time and frequency domain signal of the output of the oscillator.

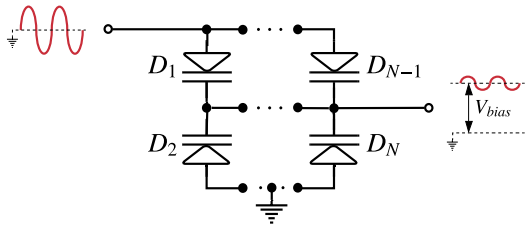


Figure 6.5: A schematic representation of the used topology for placing the varactors.

The varactor that is chosen is the BB201 from NXP Semiconductors, this for reasons explained in Appendix G.

To estimate the amount of varactors needed, the frequency is plotted as a function of the reverse voltage for a given number of varactors. The script used for this is given in Appendix H.1. The results are shown in Figure 6.6.

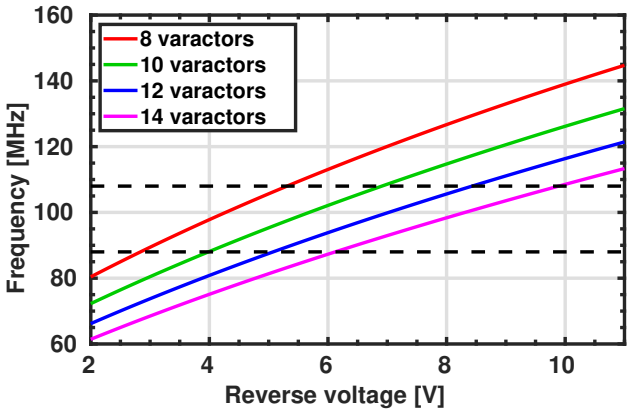


Figure 6.6: The resonance frequency versus the applied reverse voltage when 8, 10, 12 or 14 BB201 varactors are used. The dashed lines indicate the upper and lower carrier frequency.

Looking closely at the figure, it can be noted that (1) the behaviour is more linear when a larger

number of varactors is used and (2) that the slope is smaller when a larger number of varactors is used. Furthermore, when a higher reverse voltage is applied, it is easier to ensure that the varactors will not accidentally become forward biased due to the RF signal at the output. Ultimately, it is chosen to use twelve varactors, since more will also increase the system's price, but do not yield a much better linearity (especially not on the scale of the modulation bandwidth). Furthermore, the model provided by NXP (see [28]) holds for zero to ten volts, and at 108 MHz that reverse voltage is approached closely by the fourteen varactors.

Adding the twelve varactors to the circuit and measuring the oscillation frequency for a set of reverse voltages yields Figure 6.7. Here, the value of R_L has been increased to 1.7 kΩ, this because the added circuit parts reduce the total equivalent load resistance. The value is determined empirically.

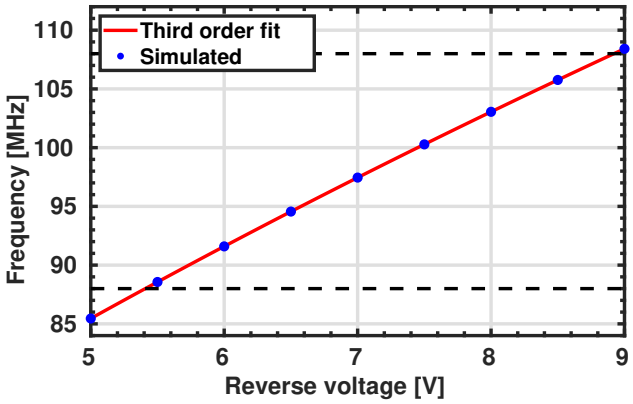


Figure 6.7: The simulated oscillation frequency as a function of the reverse voltage. The dashed lines indicate the upper and lower carrier frequency.

A third-order polynomial is fitted to the simulated datapoints. The intersections of the fitted polynomial with the 88 MHz and 108 MHz line are at a reverse voltage of respectively 5.4 V and 8.9 V. The script used to generate the figure and calculate the intersections is given in Appendix H.2.

6.2.3 Adding the coarse tuning

Since the user must be able to tune the resonance frequency from 88 MHz to 108 MHz, means must be provided to do so. From the previous section, it is known that the minimum reverse voltage over the varactors needed to achieve this is 5.4 V and the maximum reverse voltage is 8.9 V. The tuning is implemented using a resistive divider, from which the user can tune the lower resistor. This is schematically shown in Figure 6.8.

The reason for making the lower resistor tunable is that the graph of the transfer function is concave, this is the opposite of the curve followed by the frequency as function of reverse voltage. While it cannot completely reduce the error introduced, it will be less than when the upper resistor is tunable.

For now it is assumed that, in the design of the divider, R_{choke} and L_{choke} can be ignored. This is a reasonable assumption because they are connected to the varactors, which are an open circuit for the DC signal present at the divider. The transfer function of the resistive divider is given by

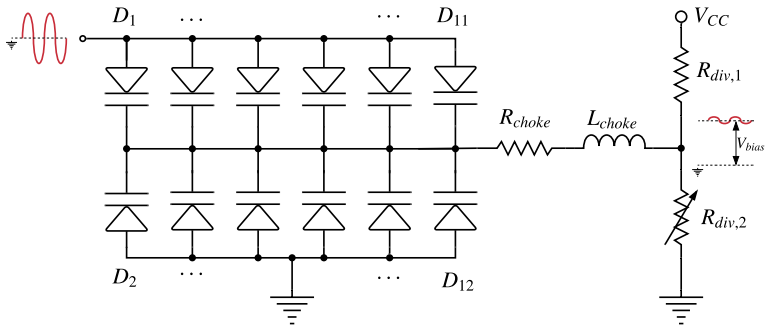


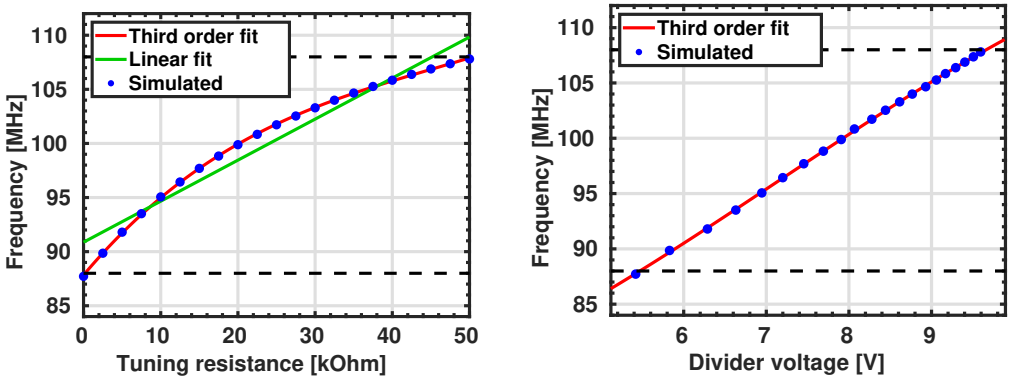
Figure 6.8: The twelve varactors together with the voltage divider used for coarse tuning.

Equation (6.17).

$$V_{bias} = \frac{R_{div,2}}{R_{div,1} + R_{div,2}} V_{CC} \tag{6.17}$$

Setting $R_{div,1}$ to 24.5 kΩ, we find that $R_{div,2}$ must range from 20 kΩ to 70 kΩ. This can be approximately achieved by using a resistor R_{var} that is tunable from 0 kΩ to 50 kΩ in series with a resistor of 20 kΩ. In Figure 6.9a, the resonance frequency as a function of $R_{div,2}$ can be seen. Ideally, the green line would be followed, since that is a first-order polynomial fitted to the simulated datapoints. However, this clearly is not the case, which is due to the non-linearity of the voltage divider. Since this is the coarse tuning it is not considered to be a problem.

In Figure 6.9b, the frequency is again given, but now as a function of the average divider voltage V_{bias}. The third-order polynomial fit is used to select the voltage swing that should be presented by the sawtooth signal coming from the baseband amplifier.



(a) (b)

Figure 6.9: The resonance frequency as a function of (a) R_{var}, and the resonance frequency as a function of (b) the average divider voltage V_{bias}. The dashed lines indicate the upper and lower carrier frequency.

6.2.4 Adding the modulation

To be able to add the sawtooth signal to the bias voltage, another resistive divider is used. The total value of this divider equals 1.5 kΩ by requirement MOD3. It is connected to the other resistive divider via a decoupling capacitor. This allows for independently designing the two.

Since by requirement SYS6 the modulation bandwidth should be between 180 kHz and 400 kHz, it is selected to be approximately 350 kHz at the lower carrier frequency. This leaves room for some error, which is needed since it is only a crude estimate. Using the third-order fit given in Figure 6.9b, it is found that voltage swing due to the sawtooth should be approximately 64 mV. This voltage swing yields a bandwidth of approximately 270 kHz at the upper resonance frequency.

The script used to plot the figures and to calculate the needed voltage swing is given in Appendix H.3.

Since the input voltage from the baseband amplifier is 3 V_{PP}, the lower resistance $R_{div,4}$ of the new divider must be set to 32 Ω. The upper resistance $R_{div,3}$ should be 1500-32 = 1468 Ω. In practice, this will be hard to come by. However taking into consideration that the output resistance of the baseband amplifier (see [29]) is 16.4 Ω, $R_{div,3}$ can be reduced to 1450 Ω.

The last three parts that need to be given a value are $C_{S,4}$, L_{choke} and R_{choke}. Considering that $C_{S,4}$ is simply used as a DC blocking capacitor, a value of 3.3 μF is chosen. The value for L_{choke} and R_{choke} is a bit less trivial. They are used to block the RF signal but let through the sawtooth. Hence, the value for L_{choke} must be selected such that its reactance is very small over the bandwidth of the sawtooth, but high for the RF signal. R_{choke} is used to complement L_{choke}, so that resonance occurring after a sharp sawtooth transition does not significantly influence the varactors and so that the value seen by the varactors is relatively constant over the bandwidth of the sawtooth.

The considered bandwidth is the 98% power bandwidth. A MATLAB simulation, given in Appendix H.4, reveals that this is about 600 kHz for an ideal sawtooth wave with a 10 kHz period. By setting L_{choke} equal to 10 μH, its reactance ranges from about 0.63 Ω (at 10 kHz) to 38 Ω (at 610 kHz). At the lowest RF frequency, the reactance equals 5.5 kΩ. For the resistor R_{choke} a value of 120 Ω is chosen, so that, over the bandwidth of the sawtooth, the impedance seen by the varactors is not changing that much.

This finalises the design of the modulator. The full symbolic circuit is given in Figure 6.10.

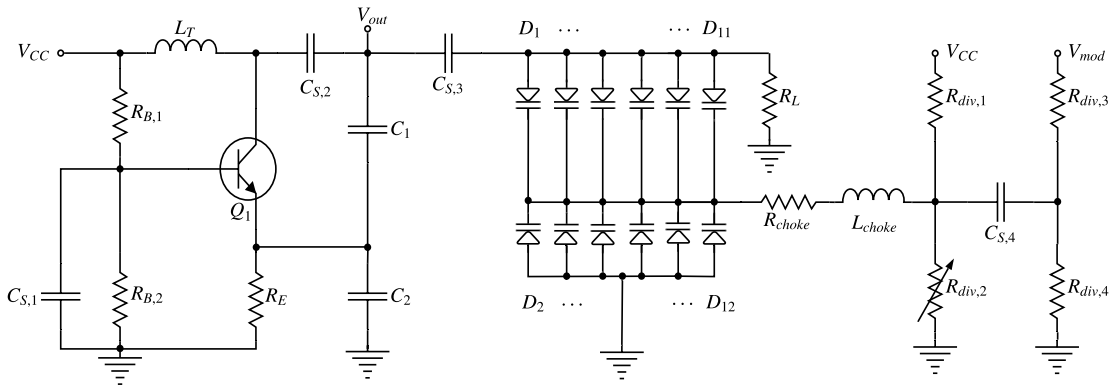


Figure 6.10: The final design of the modulator.

In Table 6.1, an overview of all the component values is found. The corresponding bill of materials is found in Appendix V.1.

Table 6.1: The values of the components used in the modulator. Note that, in a hardware implementation, some resistances need to be formed by a series combination of multiple resistors.

Component	**Value**	**Component**	**Value**	**Component**	**Value**
$C_{s,1}$ ... $C_{s,4}$	3.3 μF	$R_{B,1}$	18 kΩ	$R_{div,1}$	24.5 kΩ
D_1 ... D_{12}	BB201	$R_{B,2}$	30 kΩ	$R_{div,2}$	20 kΩ to 70 kΩ
L_T	27 nH	R_E	3.2 kΩ	$R_{div,3}$	1450 Ω
Q_1	BFU550A	R_L	1.7 kΩ	$R_{div,4}$	32 Ω
C_1	6.8 pF	R_{choke}	120 Ω	L_{choke}	10 μH
C_2	160 pF				

7. Results

In this chapter, the results of the modulator are discussed. Each requirement is verified for a carrier frequency of 88 MHz, 98 MHz and 108 MHz. It is assumed that, if the requirement holds at these three points, it will also hold at the other carrier frequencies in the desired range.

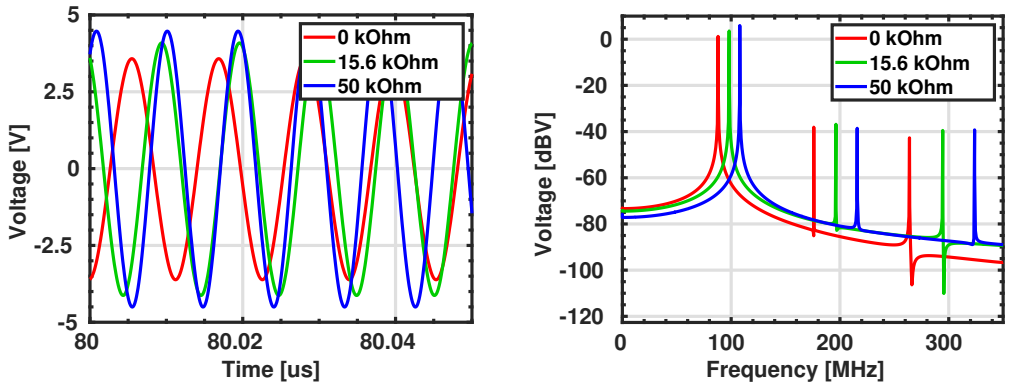
A general version of the ADS circuit and the settings that are used to verify the requirements is given in Appendix J. The simulation settings can be different than shown from requirement to requirement. Furthermore, it should be noted that the shown frequency spectra are always part of a two sided spectrum unless mentioned otherwise.

7.1 Voltage swing, power consumption and harmonics

In this section, requirements SYS5, MOD0, MOD1 and MOD2 are verified.

During the verification of these requirements, the sawtooth signal is set to zero volts and the transient simulation is run from 50 μs to 110 μs. Furthermore, the maximum timestep is 0.1 ns and the used values of R_{var} are 0 kΩ, 15.6 kΩ and 50 kΩ. These values are chosen based on Figure 6.9a.

The results for the output voltage in time and frequency domain can be found in Figure 7.1.



(a) Time domain

(b) Frequency domain

Figure 7.1: The simulation results of the output voltage in the (a) time and (b) frequency domain for the three different tuning resistances.

From the figures, the resonance frequency, peak-to-peak voltage and level of the second harmonic with respect to the first can be determined. The results are summarised in Table 7.1. The MATLAB script used to generate the figures is given in Appendix H.5.

Table 7.1: The results for the resonance frequency, output voltage, power consumption and harmonics.

Resistance [kΩ]	**f_{osc} [MHz]**	**V_{pp} [V]**	**Power Consumption [mW]**	**2nd harmonic [dB]**
0	88.0	7.2	31	-39.3
15.6	98.2	8.2	30	-40.4
50	107.9	9.0	27	-44.5

From the results, it is concluded that requirements MOD0, MOD1 and MOD2 are fulfilled. Also SYS5 is assumed to be fulfilled, since the discrepancy of 100 kHz falls inside the 0.5% margin.

7.2 The modulation bandwidth and linearity

In this section, requirement SYS6 and MOD5 are considered. For FM signals, the modulation bandwidth B_{FM} is defined as the bandwidth in which 98% of the energy resides [30].

For the verification of these requirements, the results are generated by running the transient simulation for 410 μs, of which the last 200 μs are stored to obtain two baseband periods. The frequency spectrum of the obtained output voltage is shown in Figure 7.2. For a closer look at the peaks of the spectra, the reader can refer to Appendix K.1.

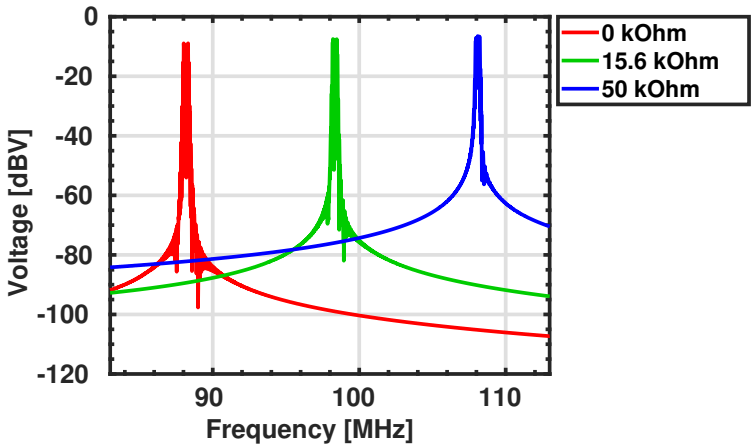


Figure 7.2: The frequency spectrum for the three different tuning resistance values over the course of two full baseband periods.

As can be seen, the spectra are wide at the tops. The bandwidth is now estimated using the MATLAB script given in Appendix H.6. A built-in function is used, which estimates the bandwidth by integrating the spectrum both from the left and the right until 1% of the total power is reached on each side. The results are given in Table 7.2. As can be seen, requirement SYS6 is satisfied.

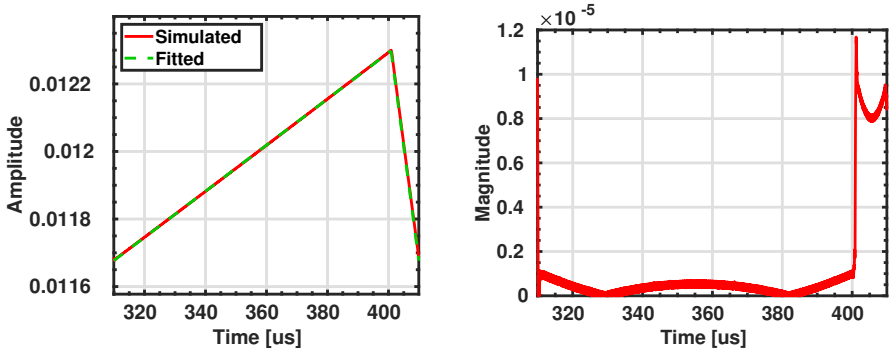
Table 7.2: The modulation bandwidth B_{FM} for three different tuning resistances.

Resistance [kΩ]	**Bandwidth [kHz]**
0	399
15.4	364
50	352

The linearity of the modulation is estimated by doing a numerical demodulation to recover the baseband signal from the chirp. The used baseband signal, a sawtooth where the rise time is ten times the fall time, is fitted against the obtained signal after which the linearity k is calculated using Equation (7.1).

$$k = 20 \log_{10} \left\{ \frac{V_{pp}}{\max(|\text{error}|)} \right\} \tag{7.1}$$

The methods used to do the demodulation and the subsequent curve fitting are respectively explained in Appendix I.4 and Appendix I.5. The corresponding code is also explained there. The resulting waveform for a tuning resistance of 0 kΩ is shown in Figure 7.3a. The code used to plot the figures is given in Appendix H.7.



(a) The fitted sawtooth plotted over the recovered sawtooth

(b) The absolute error between the measured and fitted waveform

Figure 7.3: The resulting waveform for the linearity measurement and the corresponding error. The used tuning resistance is 0 kΩ.

From Figure 7.3b, it can be seen that the error is largest at the transition from rising to falling. This is to be expected, since here the finetune circuitry needs to be able to follow the sharp transition in the waveform. The figures containing the results for $R_{var} = 15.6$ kΩ and $R_{var} = 50$ kΩ are very similar and can be found in Appendix K.2.

The results for the linearity are found in Table 7.3. As can be seen, requirement MOD5 is satisfied.

Table 7.3: The modulation linearity when applying a 10 kHz sawtooth baseband signal for three different tuning resistances.

Resistance [kΩ]	**Linearity [dB]**
0	34.5
15.4	34.1
50	33.8

7.3 Phase noise

In this section, the last remaining requirement, MOD6, is verified. This is done using a Harmonic Balance (HB) simulation in ADS. The sawtooth signal is turned off. In Figure 7.4, the results for the three different tuning resistances are given. The frequency offset ranges from 4 kHz to 10 GHz.

As can be seen, the noise spectrum for all three carrier frequencies follows a very similar curve, where the harmonics can be distinguished. Most notably, the peak corresponding to the third harmonic is quite large. Furthermore, the three different regions from Leeson's model can be distinguished.

The values at a frequency offset of 10 kHz are given by Table 7.4. As can be seen, requirement MOD6 is satisfied.

In order to get an estimate for the signal to phase noise ratio SNR_{PN} over the modulation bandwidth, the phase noise spectrum is integrated over half the bandwidth (where the half bandwidth is due to the phase noise spectrum being single sided). The starting point of the integration is 4 kHz. The MATLAB script used for the integration is given in Appendix H.8. The results are given in Table 7.5.

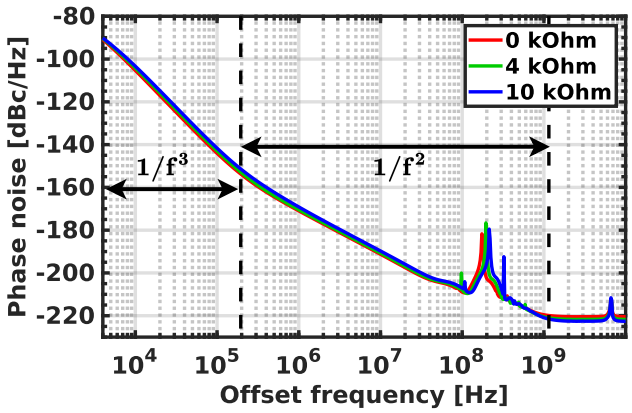


Figure 7.4: The single-sided phase noise spectrum for the three different tuning resistance values with a frequency offset from 4 kHz to 10 GHz. The dashed lines indicate the transitions between the different regions.

Table 7.4: The phase noise at 10 kHz for the three different tuning resistances.

Resistance [kΩ]	**Phase noise at 10 kHz [dBc/Hz]**
0	-105.3
15.4	-104.2
50	-103.3

Table 7.5: The SNR_{PN} for the three different resistances.

Resistance [kΩ]	**$\mathbf{SNR_{PN}}$ [dB]**
0	59.0
15.4	58.2
50	57.7

7.4 Additional Figures

At this point, all requirements have been verified. There are however more assumptions that need to be verified, namely that (1) the collector current does not exceed the limit of 50 mA, that (2) the collector-emitter voltage stays below 10.5 V so that the breakdown region is not reached and (3) that the varactor diodes do not become forward biased. Since these are not directly related to the requirements, they are not considered explicitly here. However, the reader can refer to Appendix K for the corresponding figures. The output waveform of the modulator during startup is found there as well.

8. Discussion and future work

In this chapter, the results of the modulator are discussed and some possibilities for future work are presented.

As shown in Chapter 7, the design adheres to the requirements. This provides confidence in the feasibility of a hardware implementation of the design. Sometimes, however, the expected values deviate slightly from the simulation results. This can often be attributed to approximations which are made during the design. An example of this is that the bandwidth at the lowest carrier frequency was approximated to be 350 kHz, but a simulation revealed that it is actually 399 kHz. This can largely be attributed to making the approximation while treating the frequency spectrum as an ideal brickwall spectrum.

The simulations in Chapter 7 are all carried out using ideal components. In practice, components have (1) parasitics and (2) their actual value might deviate slightly from their nominal value. Hence, a practical circuit can (and will) behave different from the results presented here. While this problem can largely be relieved by using high-quality components, that also increases manufacturing costs. Even if the modulator, in general, is expected to be relatively robust against small changes, it is worth verifying whether this is indeed the case. This has not yet been done due to time constraints. One point where tolerances are expected to influence the circuit significantly is at the coarse tuning, this since the tunable resistors in general have large tolerances.

Furthermore, in the current implementation, the coarse tuning is not linear. For example, the linearity could be improved by placing a less stringent requirement on the linearity of the resonance frequency as a function of the applied reverse voltage. When this is done, the frequency curve can be chosen such that it, more or less, follows the 'opposite' curve with respect to the bias voltage curve. Here, the bias voltage curve is considered as a function of the dividers tuning resistance. Note that this approach would influence the modulation linearity. However, if the modulation bandwidth is chosen small, the effect might actually be negligible. This of course has to be verified.

Another recommendation on the coarse tuning is replacing the tunable resistor for a MOSFET transistor biased in the triode region. By changing the gate voltage, the "resistance" of the MOSFET will also change linearly. Hence it can be used to tune the resonance frequency. The advantage of using the MOSFET compared to the tunable resistor is that it is easier to make the resonance frequency electronically controlled in future implementations. Note that, when the MOSFET would be added to the current implementation, a tunable resistance would still be needed to control the gate voltage.

The presented design uses many varactors to make the resonance frequency relatively independent from the feedback capacitances, this increases the cost of the system. While the Clapp oscillator was disregarded early, its main advantage is that the capacitive feedback can be made such that it has almost no influence on the resonance frequency when using a small tank capacitance. It is worth doing a more thorough investigation of the Clapp oscillator to see if it might be the more appropriate choice.

The implemented oscillator produces a sinewave, however ideally the PA should receive a square wave [29]. In future work, it is worth to investigate high-frequency square wave voltage-controlled oscillators as well.

Furthermore, in order to get a wider view of the modulator performance, more aspects need to be simulated. For example, the modulation linearity can be characterised for multiple types of input waveforms. As can be seen from the results, the modulator has a hard time following the sharp transitions present in a sawtooth. If, for example, the linearity was calculated for a 10 kHz sinewave, it is likely that a different value would have been obtained. Another thing that can be characterised is, for example, the jitter noise.

Finally, methods should be investigated which can be used to keep the amplitude of the output voltage more stable over the output frequency range.

Part III

Mixer and Local Oscillator

9. Introduction and Theoretical Background

9.1 Introduction

In the second part of this thesis, the design of the mixer is discussed. The mixer is the stage between the LNA and the IF amplifier. The goal of the mixer is to translate the frequency of the RF input signal to a fixed intermediate frequency (IF). The motivation to do this is that it simplifies the design of the subsequent stages in the receiver [12]. Instead of tuning all stages in the receiver for a particular frequency, one only needs to tune the LO. A schematic view of the mixer and its function is shown in Figure 9.1.

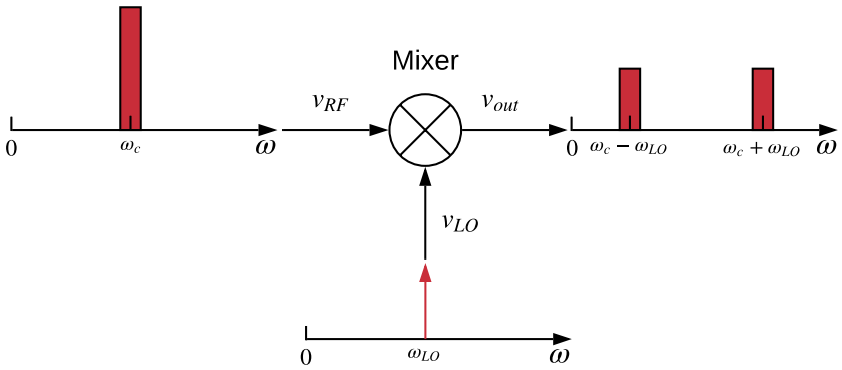


Figure 9.1: A top-level overview of the mixer.

In order to understand their operation, a background on mixers is presented in section 9.2. Subsequently, the programme of requirements is discussed in Chapter 10. In Chapter 11 the design process of the mixer is explained and the final design is presented. The results of this mixer can be then found in Chapter 12 and finally in Chapter 13 the results are discussed and recommendations for future work are provided.

Since the design steps and the results of the LO are very similar to those of the modulator, they are given in Appendix T.

9.2 Theoretical Background

As previously mentioned, in receivers, mixers are used to translate the frequency of the RF input signal to a fixed IF. The RF signal gets essentially multiplied by the signal of the LO and produces an output signal with frequency components on both the sum and difference of the frequencies of the input signals. This is shown in Figure 9.1. The operation of the mixer relies on the trigonometric identity depicted in Equation (9.1).

$$A_{RF}\cos(\omega_c t)A_{LO}\cos(\omega_{LO}t) = \frac{A_{RF}A_{LO}}{2}[\cos((\omega_c - \omega_{LO})t) + \cos((\omega_c + \omega_{LO})t)] \tag{9.1}$$

From this equation, it can be seen that both high- and low-frequency components will be present on the output signal. When a mixer is used in a receiver, the interest lies on the low-frequency component. This is called a downconversion mixer and is achieved by filtering the output with a bandpass filter or in the case that the desired IF is zero with a lowpass filter. Receivers which convert the RF signal directly to a baseband signal are also called homodyne or zero-IF receivers. When the high-frequency component is the desired one, the mixer is called an upconversion mixer.

This is often employed in transmitters where the input signal is transmitted at a higher frequency. The motivation to use downconversion mixers at the receiver is that with a low IF, higher values of quality factor Q can be attained where the IF filters will be able to provide steep attenuation using practical circuit elements [7].

9.2.1 Performance metrics

In order to assess the performance of the mixer, metrics are needed. Common performance metrics used are the following [19]:

- **Voltage conversion gain**: this is the ratio between the rms voltage at the desired IF and the input RF signal.
- **Port-to-Port Feedthrough**: Unwanted coupling between ports.
- **Noise Figure**: this is the ratio between the SNR at the input RF port and the SNR at the output IF port [25].
- **Harmonic Intermodulation Distortion**: Distortion due to harmonics of the input signals produced by the mixer [25].

A more elaborate explanation and many other metrics can be found in [31, Sec. 23.4].

9.2.2 Topologies

In this section, different mixer topologies are presented. Mixers are often classified in three categories: unbalanced, single balanced or double balanced [7]. In an unbalanced mixer the output signal will contain feed through from both inputs, a single balanced suppresses one of the inputs but lets the other feed through and finally a double balanced mixer suppresses both input signal and the output is ideally solely given by the product of both input signals. Port isolation is important since feedthrough can cause LO radiation by the antenna [19].

Double balanced mixers

Mixers can be both passive and active. Active mixers provide amplification. In this context, we discuss two common double-balanced mixer topologies, one active and one passive. In Figure 9.2 a diode ring mixer can be seen. This is a passive mixer. It operates by using the diodes as switches which are driven by the LO signal. From the baluns B1 and B2, it can be seen that both inputs are differential hence making this a double-balanced mixer.

This circuit has been commonly used for many years [31]. It has a low footprint and can easily be made with discrete components. However, this topology requires the LO signal to be larger than the largest RF signal so that the LO is the one switching the diodes [31]. Additionally, as this is a passive mixer there is no conversion gain and in the ideal case, while driven by an LO square wave between +1 and -1, there will be a conversion loss of $\frac{2}{\pi}$ [31].

In the case of an active mixer, a commonly used topology is the Gilbert Cell. The basic topology for this mixer is shown in Figure 9.3. As can be seen, both inputs are differential and the output is differential as well. In essence, the current provided by I_{EE} is split between Q5 and Q6 depending on the signal v_{RF}. The LO signal then steers the current by having two transistors on and two off in the top two pairs. If there is no differential signal available at the RF input port then the current is split evenly between Q5 and Q6, this causes the output to be zero irrespective of the LO input. This is due to the cross connection in the top four transistors, also referred to as the switching quad. In the case the differential LO signal is zero, the output is zero as well irrespective of the RF input signal.

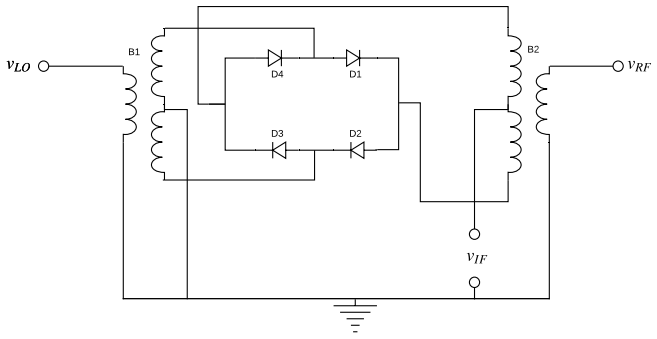


Figure 9.2: Diode ring mixer.

Single balanced mixers

Besides double-balanced mixers, single-balanced mixers are also often used. As in the previous section, single-balanced mixers may be active or passive. A passive single balanced mixer can be realized in the same manner as a diode ring mixer, however using only two diodes. This circuit is shown in Figure 9.4. Here it can be seen that the output is not isolated from the RF input thus this component will be present at the output. However, the LO is made differential and thus this component will not appear at the output.

In the case of an active mixer, it is also possible to implement it in a single-balanced fashion. As the reader might expect, it has a lot of resemblance to the Gilbert Cell. Instead of using a differential signal for the RF input the single balanced mixer shown in Figure 9.5a uses a single-ended signal. In this case the RF voltage signal is converted to a current and then is split by the differential pair driven by the LO signal. Subsequently, it is reconverted in a voltage using resistors as a load. A block diagram is shown in Figure 9.5b.

Unbalanced mixers

Finally, mixers can also be unbalanced. As perhaps the reader might expect, it is possible to implement a passive mixer using a single diode. Some drawbacks are the feed through but this can be compensated using filters, however this requires large board area and can be expensive. Due to their poor performance compared to other topologies, they are not further discussed.

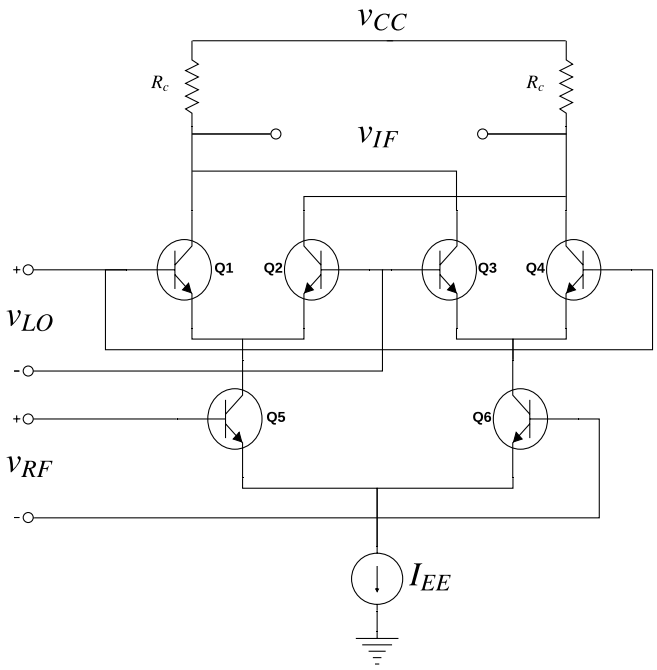


Figure 9.3: Gilbert Cell mixer.

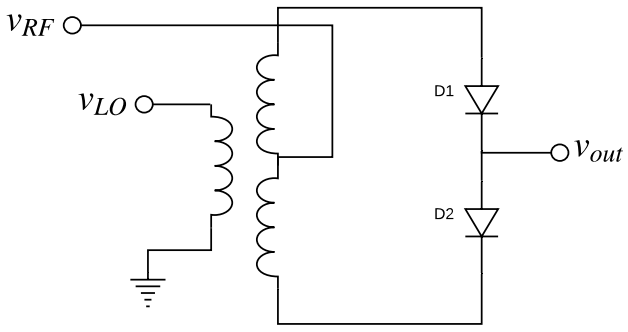
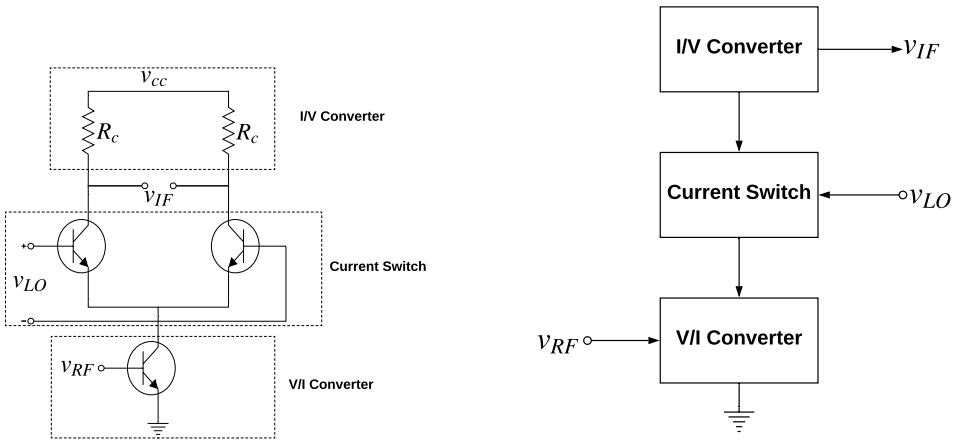


Figure 9.4: Single balanced diode mixer.



(a) Circuit implementation. (b) Block diagram.

Figure 9.5: Single-balanced active mixer [19, Sec. 6.3]

10. Programme of Requirements

The requirements for the mixer can be found in Table 10.1.

Table 10.1: The requirements for the mixer

Tag	Description	Value
MIX0	Power consumption	< 50 mW
MIX1	Amplification	> 29.54 dB
MIX2	LO-Output isolation	> 50 dB
MIX3	RF-Output isolation	> 50 dB
MIX4	The mixer must have a single-ended output	-
MIX5	The mixer must have single-ended input for the RF port	-
MIX6	The mixer must have a single-ended input for the LO port	-
MIX7	Supply Voltage	12 V
MIX8	The circuit shall contain discrete components only	-

These requirements are the result of system-level design as well as constraints from preceding and succeeding stages. The supply voltage and power budget are defined in the system-level design. The amplification and requirements for single-ended ports are requirements coming from the LNA and IF amplifier. The isolation requirements have been chosen to relax the requirements for the filter in the IF amplifier. It is important to note that no IF frequency has been stated as a requirement. This is due to the fact that the mixer acts as a frequency translation device and the filter to select the IF frequency is located in the IF amplifier and is not discussed in this thesis. However, it is known that the IF will be located at 10 MHz so this should be kept in mind when designing the mixer.

11. Design

In this chapter the steps towards the final design of the mixer will be shown. First of all, a mixer topology should be chosen. According to Requirement MIX1 the mixer should provide amplification. This means that a passive mixer cannot be considered. Furthermore, according to Requirement MIX3 and MIX2 the mixer should provide high LO-output and RF-output suppression. This means that the mixer should be double balanced, which means that the design should begin from the Gilbert Cell. The basic topology for the Gilbert Cell has been shown in Figure 9.3, in here it can be seen that it consists of a current source and three differential pairs. In order to design and understand the operation of the Gilbert Cell first closed attention is paid to the operation of the differential pair and subsequently practical implementations are discussed for the current source.

11.1 The Differential Pair

The basic differential pair is shown in Figure 11.1. In the differential pair the differential input voltage is converted to a differential current output which is then reconverted to a voltage using the load resistors R_c.

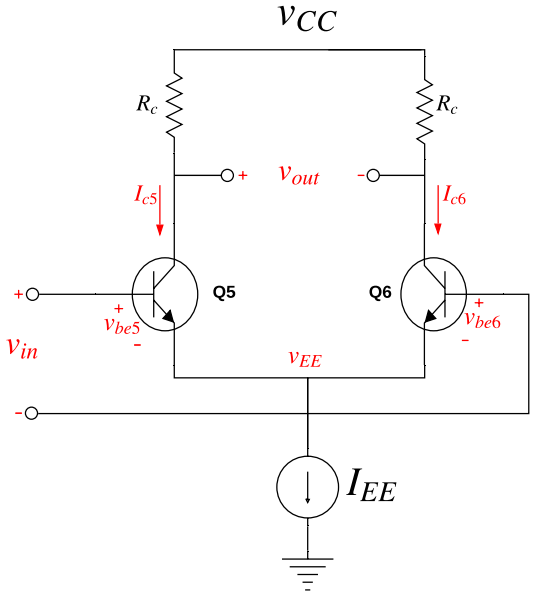


Figure 11.1: Differential pair.

The output voltage of the differential pair can be calculated by first looking at each of its output nodes and then taking the difference between V_{out+} and V_{out-}.

$$V_{out+} = V_{cc} - R_c I_{c5} \tag{11.1}$$

$$V_{out-} = V_{cc} - R_c I_{c6} \tag{11.2}$$

$$V_{out} = V_{out+} - V_{out-} = -R_c(I_{c5} - I_{c6}) \tag{11.3}$$

According to [24], if the differential input is "small" then the voltage v_{EE} can be assumed constant. Therefore we can write:

$$v_{in+} - v_{be5} = v_{in-} - v_{be6} \tag{11.4}$$

$$V_{be} = V_T \ln\left(\frac{I_c}{I_s}\right) \tag{11.5}$$

Using Equation (11.5) gives the following result:

$$\begin{aligned} v_{in+} - v_{in-} &= v_{be5} - v_{be6} \\ &= V_T \ln\left(\frac{I_{c5}}{I_s}\right) - V_T \ln\left(\frac{I_{c6}}{I_s}\right) \\ &= V_T \ln\left(\frac{I_{c5}}{I_{c6}}\right) \end{aligned} \tag{11.6}$$

The collector current through Q5, I_{c5}, can then be written as shown in Equation (11.7).

$$I_{c5} = I_{c6} e^{\frac{v_{in+} - v_{in-}}{V_T}} \tag{11.7}$$

Assuming now that $I_{c5} + I_{c6} = I_{EE}$, I_{c5} can be rewritten as:

$$I_{c5} = \frac{I_{EE}}{1 + e^{\frac{v_{in-} - v_{in+}}{V_T}}} \tag{11.8}$$

A similar equation can be derived for I_{c6}. Now using Equation (11.3) results in Equation (11.9).

$$V_{out} = -R_c I_{EE} \tanh\left(\frac{v_{in+} - v_{in-}}{2V_T}\right) \tag{11.9}$$

It can be seen from Figure N.1 in Appendix N.1 and Equation (11.9) that for small differential input voltage the transfer will be linear and the gain will be given by Equation (11.10). This linearization holds if $|v_{in+} - v_{in-}|$ remains well below $4V_T$ [24].

$$A_v = -R_C \frac{I_{EE}}{2V_T} \tag{11.10}$$

11.2 The Current Source

As can be seen from Equation (11.10), the gain can be adjusted by varying the tail current I_{EE}. However, natural current sources do not exist, therefore one should be designed using active components. In this section we look at different current source topologies in order to finally implement one in the Gilbert Cell.

11.2.1 Topologies

In order to achieve Constant Current Source (CCS) operation a current mirror can be used. This is a circuit that copies a current reference and produces an identical output. In modern systems, such current references can be implemented using a bandgap reference circuit, which can create a supply- and temperature-independent current [24]. However, in this thesis the time-span of the project does not allow for such a design. In this case, a resistor is considered to create a current reference. For the actual implementation of the current source four different topologies are considered, namely:

Simple current mirror, cascode current mirror, Wilson current mirror and Widlar current mirror. The circuits for these four topologies can be found in Appendix O. When comparing these circuits it is clear that the first three circuits mirror the current with minimal error, this systematic gain error is given by the following equations [32, Sec. 4.2]:

$$\epsilon_{simple} \approx \frac{V_{CE2} - V_{CE1}}{V_A} - \frac{1 + \frac{I_{s1}}{I_{s2}}}{h_{FE}} \tag{11.11}$$

$$\epsilon_{Cascode} \approx -\frac{4}{h_{FE} + 4} \text{ given that } h_{FE} \gg 1 \tag{11.12}$$

$$\epsilon_{Wilson} \approx -\frac{2}{h_{FE}^2 + 2h_{FE} + 2} \tag{11.13}$$

Where V_A is the early voltage, I_s the saturation current and h_{FE} the base-collector current gain of the transistor. Therefore to reduce the systematic error the Wilson current mirror should be chosen out of the three. Finally, when comparing the Wilson current mirror with the Widlar current mirror it is important to note that in the Widlar topology, the output current will be lower than the input current [33, Sec. 6.3], increasing power consumption. Therefore, the chosen topology is the Wilson current mirror in order to be able to satisfy Requirement MIX0 more easily.

11.2.2 Current mirror choice

The implementation of the Wilson current mirror is shown in Figure 11.2. The current I_{EE} can be set by varying the resistor R_W, the current through this resistor is then mirrored to Q1. The output current is given by Equation (11.14). In a practical implementation a potentiometer can be used to set the current I_{EE}, this will prove beneficial for tuning the gain of the Gilbert Cell. When using this circuit for the Gilbert Cell care should be taken to avoid saturation. According to [32], the minimum voltage at the collector of Q2 with respect to ground is given by Equation 11.15. In this equation, $V_{BE(on)}$ refers to the base-emitter voltage at a given collector current and $V_{CE2(sat)}$ refers to the collector-emitter voltage when Q2 enters saturation.

$$I_{EE} = \frac{V_{CC} - 2V_{be}}{R_W} \tag{11.14}$$

$$V_{C2(min)} = V_{CE1} + V_{CE2(sat)} = V_{BE(on)} + V_{CE2(sat)} \tag{11.15}$$

11.3 The Gilbert Cell

In Subsection 9.2.2 the Gilbert Cell was introduced. In this section, we take a deeper look into its performance and show the mathematics associated with this mixer.

11.3.1 Mathematical Analysis

As already mentioned the Gilbert Cell uses three differential pairs. Therefore, the equations derived in Section 11.1 can be used to analyze the Gilbert Cell. As can be seen from Figure 11.3 Q5 and Q6 form a differential pair with a tail current I_{EE}, according to the derivations shown in Section 11.1 the collector currents of Q5 and Q6 can be written as:

$$I_{c5} = \frac{I_{EE}}{1 + e^{\frac{-V_{RF}}{V_T}}} \tag{11.16}$$

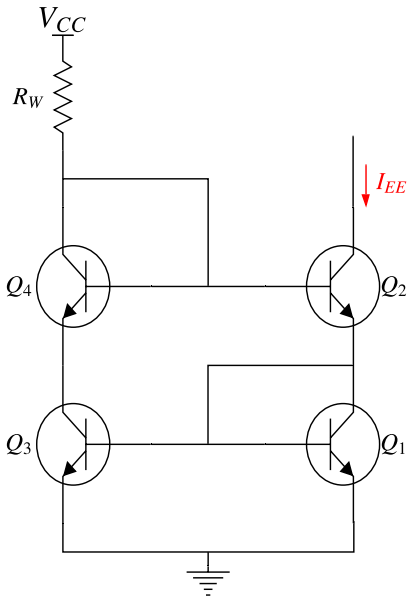


Figure 11.2: The Wilson current mirror.

$$I_{c6} = \frac{I_{EE}}{1 + e^{\frac{V_{RF}}{V_T}}} \tag{11.17}$$

When looking at the switching quad, which consists of Q1-4, it can be seen that Q1 and Q2, as well as Q3 and Q4 form differential pairs. These differential pairs have tail currents given by I_{c5} and I_{c6} respectively. The collector currents of the switching quad can then be written as:

$$\begin{aligned} I_{c1} &= \frac{I_{c5}}{1 + e^{\frac{-V_{LO}}{V_T}}} \\ &= \frac{I_{EE}}{\left(1 + e^{\frac{-V_{RF}}{V_T}}\right)\left(1 + e^{\frac{-V_{LO}}{V_T}}\right)} \end{aligned} \tag{11.18}$$

$$\begin{aligned} I_{c2} &= \frac{I_{c5}}{1 + e^{\frac{V_{LO}}{V_T}}} \\ &= \frac{I_{EE}}{\left(1 + e^{\frac{-V_{RF}}{V_T}}\right)\left(1 + e^{\frac{V_{LO}}{V_T}}\right)} \end{aligned} \tag{11.19}$$

$$\begin{aligned} I_{c3} &= \frac{I_{c6}}{1 + e^{\frac{V_{LO}}{V_T}}} \\ &= \frac{I_{EE}}{\left(1 + e^{\frac{V_{RF}}{V_T}}\right)\left(1 + e^{\frac{V_{LO}}{V_T}}\right)} \end{aligned} \tag{11.20}$$

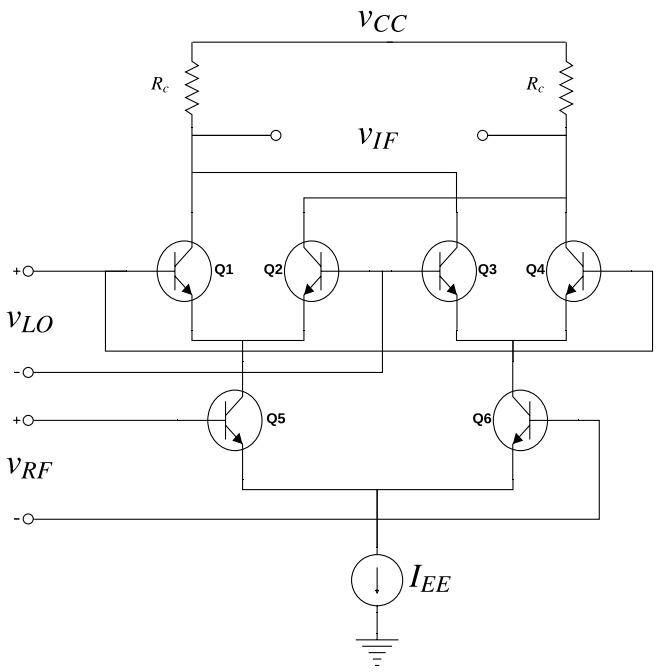


Figure 11.3: Gilbert Cell mixer.

$$\begin{aligned} I_{c4} &= \frac{I_{c6}}{1+e^{\frac{-V_{LO}}{V_T}}} \\ &= \frac{I_{EE}}{\left(1+e^{\frac{V_{RF}}{V_T}}\right)\left(1+e^{\frac{-V_{LO}}{V_T}}\right)} \end{aligned} \tag{11.21}$$

The differential output current is then [32]:

$$\begin{aligned} \Delta I &= I_{c1-3} - I_{c2-4} = I_{c1} + I_{c3} - (I_{c2} + I_{c4}) \\ &= (I_{c1} - I_{c4}) - (I_{c2} - I_{c3}) \\ &= I_{EE} \tanh\left(\frac{V_{LO}}{2V_T}\right) \tanh\left(\frac{V_{RF}}{2V_T}\right) \end{aligned} \tag{11.22}$$

In the case the LO signal is large compared to V_T and V_{RF} is small Equation (11.22) can be rewritten as:

$$\Delta I \approx I_{EE} \frac{V_{RF}}{2V_T} \tanh\left(\frac{V_{LO}}{2V_T}\right) \tag{11.23}$$

This equation shows that the mixer acts as a modulator where the input RF signal is multiplied by a square wave. This can be explained as follows: Assume the signal from the LO is given by Equation (11.24). The function $tanh(x)$ will convert this into a square wave with amplitude 1, frequency ω_{LO} and 50% duty cycle. This can be seen in Appendix N.2.

$$V_{LO} = |V_{LO}| \cos(\omega_{LO} t) \tag{11.24}$$

The fourier expansion of this square wave is given in Equation (11.25) and the derivation can be found in Appendix M.

$$f(t) = \frac{4}{\pi} \sum_{n=1,3,5,..}^{\infty} \frac{1}{n} \cos(n\omega_{LO} t) \tag{11.25}$$

Now assume the signal at the RF port is given by Equation (11.26).

$$V_{RF} = |V_{RF}| \cos(\omega_{RF} t) \tag{11.26}$$

$$\therefore \Delta I \approx I_{EE} \frac{|V_{RF}|}{2V_T} \cos(\omega_{RF} t) \left[\frac{4}{\pi} \sum_{n=1,3,5,...}^{\infty} \frac{1}{n} \cos(n\omega_{LO} t) \right] \tag{11.27}$$

Assuming the output of the Gilbert Cell is purely resistive the output voltage will be the product of ΔI and R_l and thus directly proportional to Equation (11.27). The output voltage is then given by Equation (11.28).

$$V_{out} \approx I_{EE} R_l \frac{|V_{RF}|}{2V_T} \cos(\omega_{RF} t) \left[\frac{4}{\pi} \sum_{n=1,3,5,...}^{\infty} \frac{1}{n} \cos(n\omega_{LO} t) \right] \tag{11.28}$$

11.4 Active load

In the previous section close attention has been paid to the Gilbert Cell. In this section the design is modified to include an active load. An active load refers to the use of a current mirror as a load. This serves as a differential to single-ended converter whilst not halving the voltage gain in the case of taking only one of the outputs in the fully differential pair. This is needed in order to satisfy Requirement MIX4, which states that the output should be single-ended.

In order to analyze the operation of the active load we consider a differential pair for the sake of simplicity. The circuit is shown in Figure 11.4. As has been explained before, the transistor Q7 mirrors its collector current onto transistor Q8. Therefore, $I_{c5} = I_{c8}$. In the case of the differential pair this current is equal to $\frac{I_{EE}}{2}$ if no differential input signal is present. Now lets assume there is a small differential input, this will induce a change in the currents I_{c5} and I_{c6}.

$$I_{c5} = \frac{I_{EE}}{2} + \Delta I = I_{c8} \tag{11.29}$$

$$I_{c6} = \frac{I_{EE}}{2} - \Delta I \tag{11.30}$$

$$\therefore I_{out} = 2\Delta I = I_{c8} - I_{c6} \tag{11.31}$$

Both the change in current I_{c8} as I_{c6} will cause the output voltage to rise. Essentially, there are two signal paths, one through Q5 and Q6 and another one through Q7 and Q8, both causing the output voltage to rise. It is important to note that the resistor is added to make the analysis more intuitive, but is not necessary for the actual operation [24]. This circuit provides the same gain as the fully differential amplifier shown in Figure 11.1. For a small signal analysis of this circuit the reader is referred to [24, Sec. 10.6].

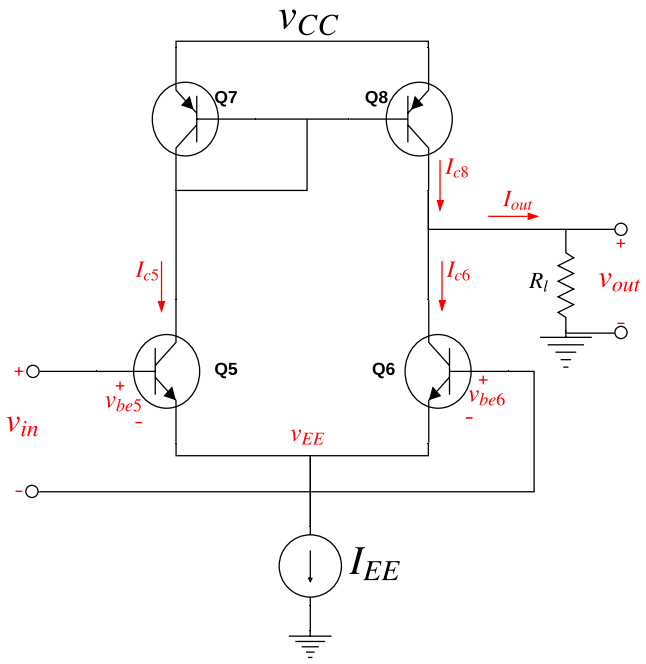


Figure 11.4: Differential pair with active load.

11.5 Single-ended to Differential conversion

According to Requirement MIX5 and MIX6 the mixer should have single-ended inputs. In the Gilbert Cell the inputs are differential by default. Therefore a subcircuit to convert the single-ended output of the LO to differential is needed. At first sight it might seem beneficial to use a differential amplifier. However, including this circuit to the design would add at least four transistors for the current source, two load resistors and two transistor for the differential pair. Additionally, we must stay within the power budget. In order to avoid these issues, a simpler solutions was used: A passive balun. This is a circuit that can convert an unbalanced signal to a balanced signal. This may be achieved with a toroidal transformer or as a microstrip on the Printed Circuit Board (PCB) [19]. The issues with baluns, is that they have a loss associated with them. However, low loss baluns are available. In this project, an integrated balun is out of the scope. Therefore, a passive balun implemented with a transformer is chosen. The circuit is shown in Figure 11.5.

11.6 Final Design Mixer

In the previous sections different subcircuits have been discussed, the goal is to combine these subcircuits to create the final design. In this section the final circuit will be presented.

First of all, we take a step back and make sure all requirements can be fulfilled with previously discussed circuits. Requirement MIX0 suggests that the power consumption should be less than 50 mW in order to satisfy the power budget of the receiver. In order to satisfy this requirement the circuit will be designed, simulated and if the design exceeds the power budget, power reduction techniques can be applied [34]. According to Requirement MIX1 the system must provide at least

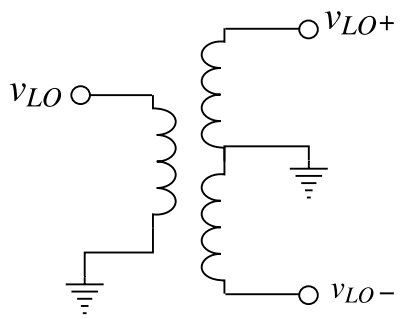


Figure 11.5: Passive balun.

29.54 dB of amplification. The Gilbert Cell is an active mixer and can provide amplification. This amplification can be tuned by adjusting the tail current which in turn can be tuned by adjusting the resistor R_W in Figure 11.2. Therefore, Requirement MIX1 can be satisfied. In order to satisfy Requirement MIX2 and MIX3 the mixer must be double balanced, the Gilbert Cell satisfies these criteria. It has also been shown that the active load can be used for converting the Gilbert Cell's output from differential to single-ended output which satisfies Requirement MIX4. Furthermore, in the previous section it was shown how to convert a single-ended signal to a differential signal by using a transformer. This satisfies Requirements MIX5 and MIX6. Finally, a supply voltage of 12 V should be used to satisfy Requirement MIX7. Since all requirements can be fulfilled the subcircuits can be combined into a final design, this is shown in Figure 11.6.

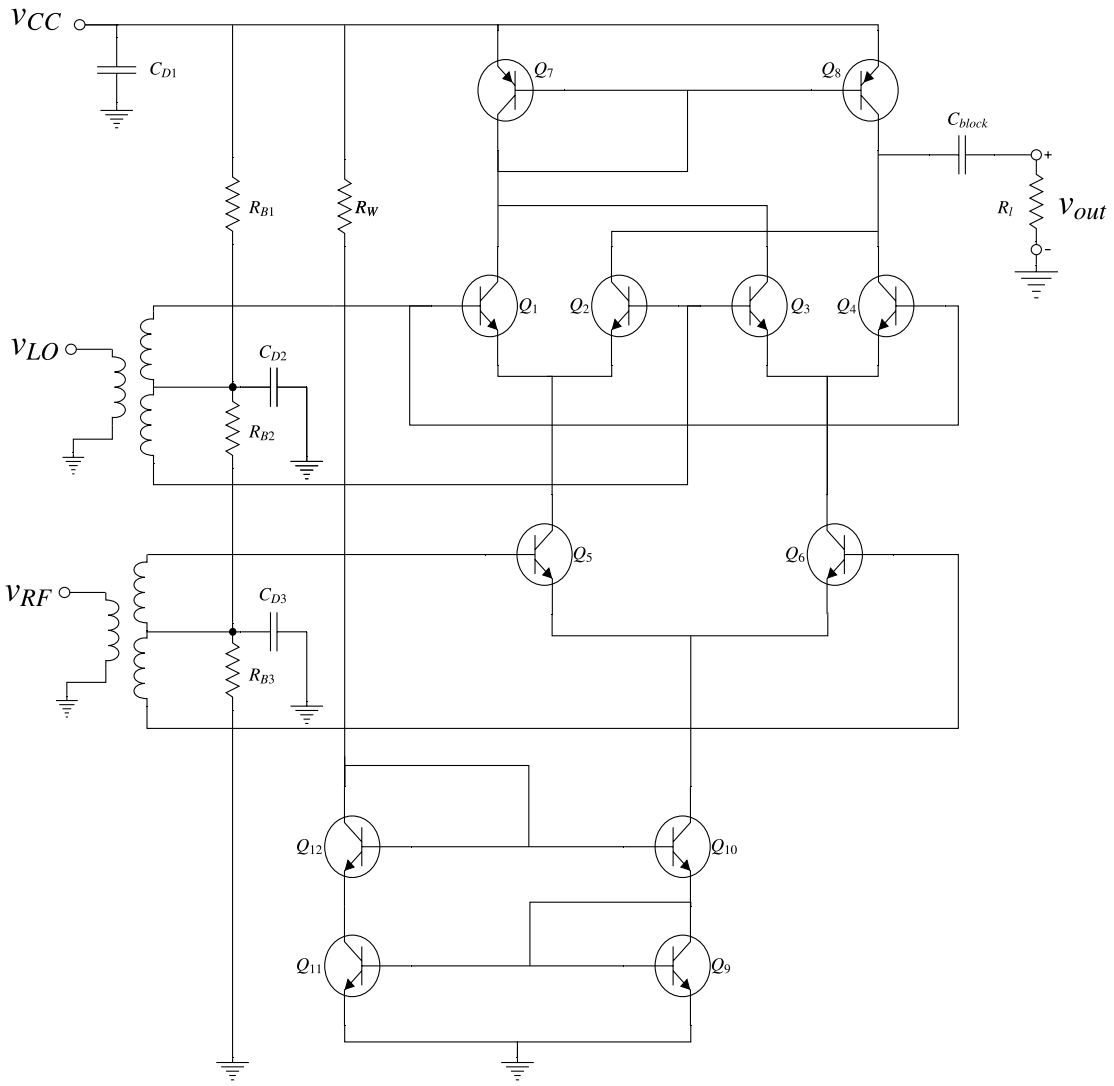


Figure 11.6: Final design of the mixer.

11.6.1 Component selection

In this section the components for the final design are calculated. The choice of transistors is explained in Appendix E. The choice of all other components is discussed in this section.

Conversion Gain

According to Requirement MIX1, the conversion gain should be at least 29.54 dB. This is the gain from the RF input to the down-converted output. Therefore, in order to calculate the conversion gain, Equation (11.27) can be rewritten into Equation (11.32). If the load resistor is set to 3 kΩ the tail current I_{EE} can be calculated for the necessary conversion gain. The calculation is shown

in Equation

$$CG = \frac{|V_{out}(\omega = \omega_{RF} - \omega_{LO})|}{|V_{RF}|} = \frac{I_{EE}}{2V_T} R_l \frac{2}{\pi} \tag{11.32}$$

$$I_{EE} = CG \frac{V_T}{R_l} \pi = 10^{29.54/20} \frac{26 \cdot 10^{-3}}{3 \cdot 10^3} \pi \approx 0.82 \text{ mA} \tag{11.33}$$

Wilson current mirror

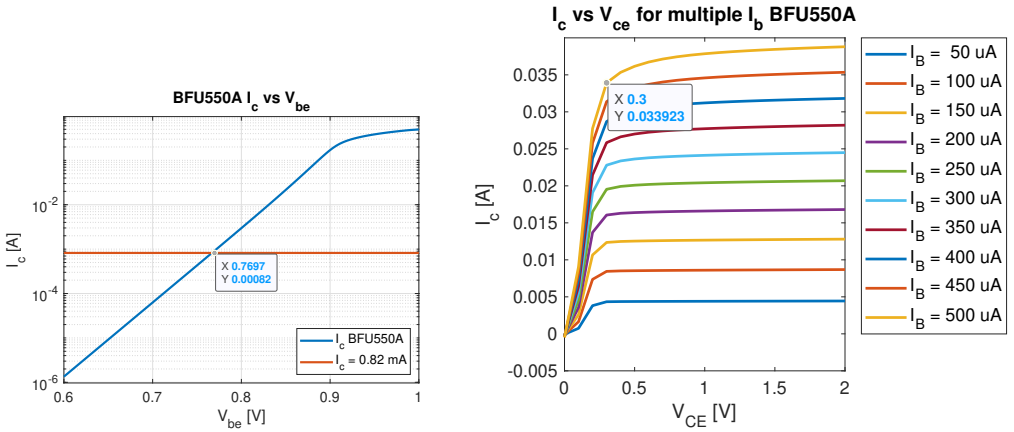
In order to set the current output to the desired value, Equation (11.14) can be used. In order to obtain the desired R_W value, this equation can be rewritten into Equation (11.34).

$$R_W = \frac{V_{CC} - 2V_{be}}{I_{EE}} \tag{11.34}$$

Before this equation can be used, we have to determine V_{be}. This can be done from Figure 11.7a. Using the shown value of V_{be} the value of R_W is:

$$R_W = \frac{12 - 2 \cdot 0.77}{0.82 \cdot 10^{-3}} \approx 12.8 \text{ k}\Omega \tag{11.35}$$

In the practical implementation, this value can change slightly due to non-idealities. Therefore, a potentiometer should be chosen in order to tune the gain to the correct value. The final component choice can be found in the Bill of Materials in Appendix V.



(a) Simulation of I_c vs V_{be} at $V_{CE} = 8V$

(b) I-V characteristic curves.

Figure 11.7: Simulations of BFU550A in ADS plotted in MATLAB.

Another important parameter to take into account is the voltage at the output node of the wilson current mirror. In section 11.2 it was stated that the voltage at the output node should remain above $V_{be(on)} + V_{ce(sat)}$. The value of $V_{ce(sat)}$ can be obtained from Figure 11.7b. In here it can be seen that the transistor enters saturation when V_{ce} drops below 0.3 V. Therefore the output voltage should always remain above $1.5(0.77 + 0.3) \approx 1.6$ V, this includes a safety factor of 1.5.

Biasing circuit

The biasing circuit is done using a string of resistors. For the resistor values E-24 resistors were chosen. As mentioned previously the voltage at the output of the wilson current mirror should not

fall below 1.6 V. Therefore, the DC base voltage of Q5 and Q6 should be chosen such as to prevent this from happening. In a similar way as done previously the V_{be} was estimated at a collector current of $\frac{I_{EE}}{2}$, this resulted in $V_{be} = 0.75$ V. This means that the base voltage of Q5 and Q6 should be at least $0.75 + 1.6 = 2.35$ V. In order to prevent any errors a base voltage of 3 V was chosen.

In order to choose the DC base voltage of Q1-Q4, care should be taken to not let Q5 and Q6 enter saturation. This is done by making sure that the collector voltage of Q5 and Q6 remain above 3 V. Therefore, the bias voltage at Q1-Q4 should be at least $3 + 0.75 = 3.75$ V. For safety margin a value of 6 V was chosen. The chosen resistor values are shown in Table 11.1. Further elaboration on resistive biasing can be found in Appendix F.

Table 11.1: Chosen resistor values for biasing scheme.

Name	Value
R_{B1}	2 kΩ
R_{B2}	1 kΩ
R_{B3}	1 kΩ

Decoupling and coupling capacitors

In this design three decoupling capacitors and one coupling capacitor are used. The decoupling capacitors become a short circuit in the AC analysis creating a low impedance path to ground for high frequency signals. These capacitors are implemented to filter noise in the supply voltage and in the biasing string. The coupling capacitor, C_{block} is used to block any DC offset from being present at the output. In RF applications, Surface Mount Device (SMD) ceramic capacitors are popular due to their RF performance, especially Multilayer Ceramic Capacitor (MLCC) technology [35]. When choosing the capacitors, it is important to pay attention to the voltage rating, the dielectric and the capacitor value. For this particular application 22 uF capacitors were chosen with X7R dielectric, 16 V maximum rating, low Equivalent Series Inductance (ESL) and low Equivalent Series Resistance (ESR). The value was chosen based on a trade-off between price and performance. The part number can be found in the Bill of Materials in Appendix V.

12. Results

In this chapter the results for the designed mixer are presented. Since the design and results of the LO are largely similar to those of the modulator they are not given here, they can however be found in Appendix T. The simulations are done in ADS but both MATLAB and ADS are used to visualize the data for aesthetic reasons.

12.1 Mixer

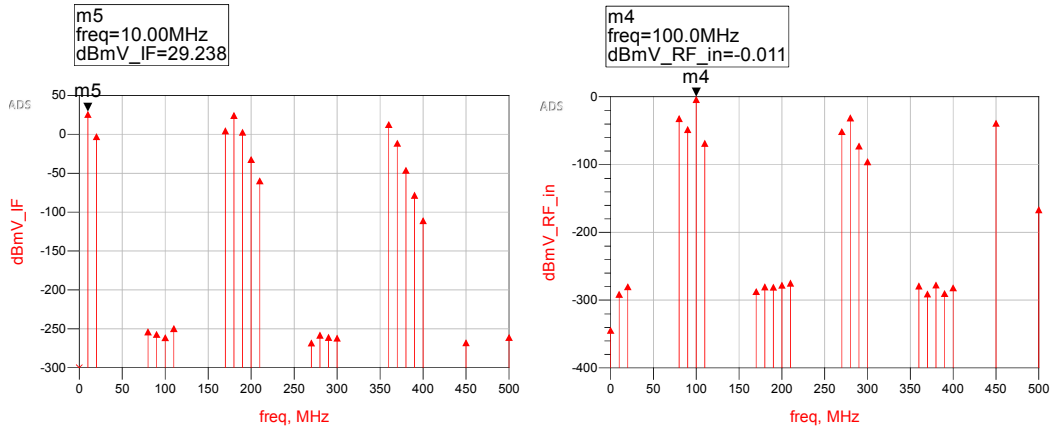
In the first section the results are shown for the mixer. The circuit used for the simulation is shown in Appendix L.

12.1.1 Conversion Gain

As previously mentioned the conversion gain should be at least 29.54 dB. In order to validate this the mixer was simulated in ADS using an HB simulation with parameters as shown in Table 12.1. In this table Freq[1] refers to the RF signal and Freq[2] refers to the LO signal. The order refers to the amount of harmonics that are analyzed.

Table 12.1: Harmonic Balance simulation setup for Conversion Gain determination.

Parameter	Value
Freq[1]	100 MHz
Freq[2]	90 MHz
Order[1]	5
Order[2]	5



(a) Frequency spectrum of the output of the mixer.

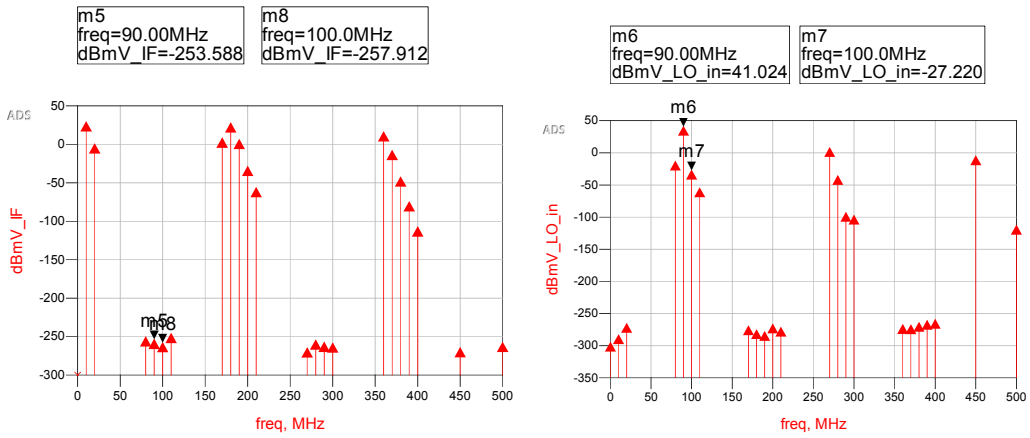
(b) Frequency spectrum of the signal at the RF port of the mixer.

Figure 12.1: Simulations for Conversion Gain calculation.

The resulting Conversion Gain is equal to $CG = 29.238 - -0.011 = 29.249$ dB. This is slightly lower than calculated but can be easily tuned by changing R_W in the current source. In practice this should be mechanically tuned to achieve the exact value, as non-idealities introduce an error margin into our calculations. The desired CG was achieved with $R_W = 12.4$ kΩ instead of the calculated 12.8 kΩ.

12.1.2 Input-Output Isolation

Another important specification of the mixer is the input-output isolation. The results for the RF-to-IF, LO-to-IF, RF-to-LO and LO-to-RF isolation are presented in Table 12.2, this is based on the simulation results shown in Figure 12.2a, Figure 12.2b and Figure 12.3. These simulations have been performed using the circuit in Appendix L where ideal AC voltage sources are used in series with their approximate output impedance at the inputs of the mixer.



(a) Frequency spectrum of the output of the mixer.

(b) Frequency spectrum of the signal at the LO port of the mixer.

Figure 12.2: Simulations for input to output isolation calculations.

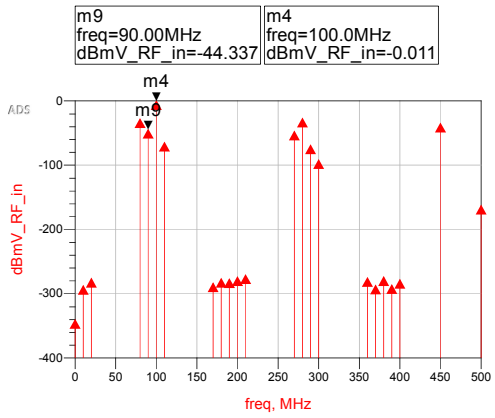


Figure 12.3: Frequency spectrum of the signal at the RF port of the mixer.

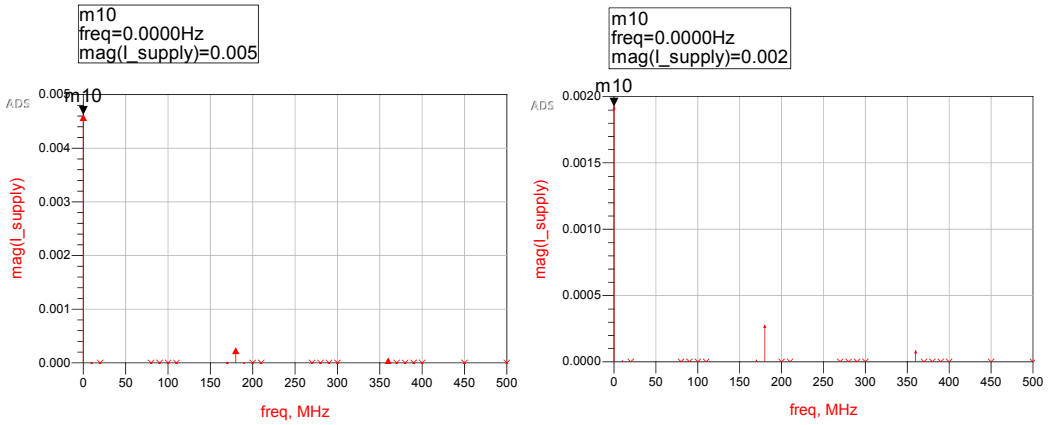
12.1.3 Power consumption

The power consumption of the mixer has been simulated in ADS. This has been done by measuring the current from the supply voltage and multiplying the DC component of this with the supply voltage. The spectrum of the output current is given in Figure 12.4a. The DC current supply is estimated to be 3.6 mA meaning the consumed power $P_{mix} = 12 \cdot 4.6 \cdot 10^{-3} \approx 55$ mW. This power consumption is not within specification thus the mixer needs to be improved. This can be done by

Table 12.2: Isolation results for the mixer.

Isolation type	Value
RF-to-IF	-0.011+257.912 = 257.901 dB
LO-to-IF	41.024+253.588 = 294.612 dB
RF-to-LO	-0.011+27.220 = 27.209 dB
LO-to-RF	41.024+44.337 = 85.361 dB

increasing all biasing resistors in the bias string by a factor ten, the resulting DC supply current can be seen in Figure 12.4b. This reduces the power consumption to $12 \cdot 1.9 \cdot 10^{-3} \approx 23$ mW, without degrading the gain. Therefore, when manufacturing this circuit the bias string can be adapted to allow this power reduction.



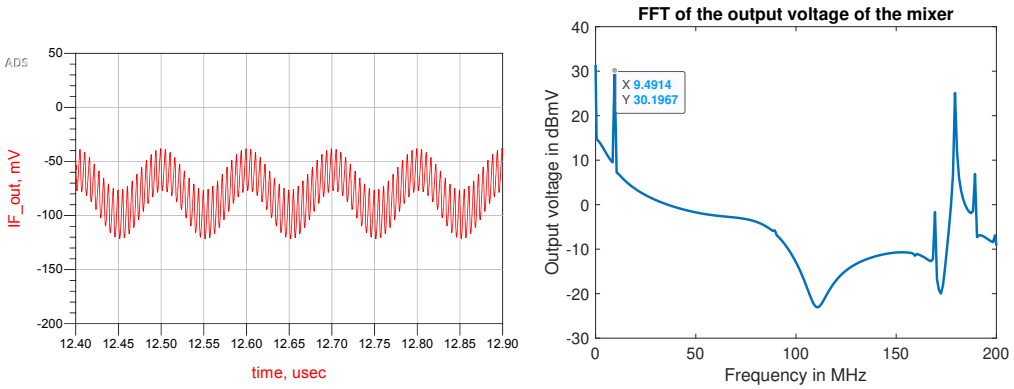
(a) Frequency spectrum of the supply current for designed circuit.

(b) Frequency spectrum of the supply current for improved circuit.

Figure 12.4: Simulations for power consumption.

12.1.4 Transient simulation

The previous simulations have been performed using a Harmonic Balance simulation. However, for final validation a transient simulation is performed as sanity check. The time-domain result is plotted in ADS, this data is then exported into MATLAB where an FFT analysis is performed. The simulation was performed using the original $R_W = 12.8$ kΩ, the time-domain result is shown in Figure 12.5a and the frequency spectrum is shown in Figure 12.5b.



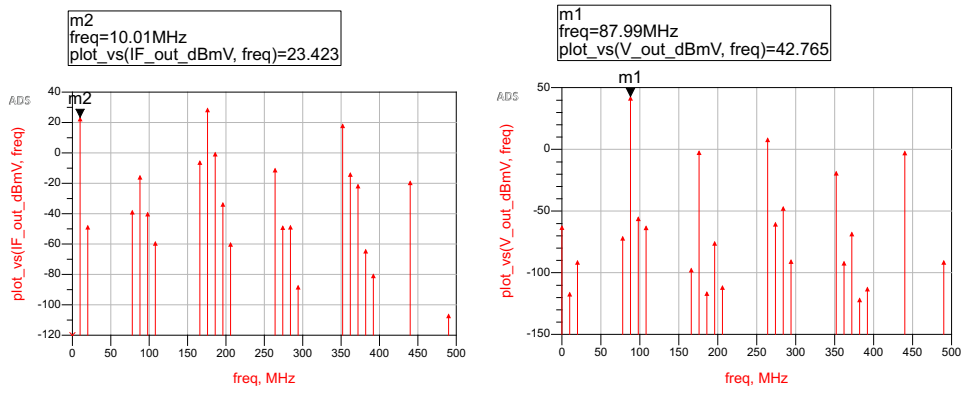
(a) Time domain signal at the output of the mixer.

(b) FFT of the output signal of the mixer performed in MATLAB using data from ADS.

Figure 12.5: Simulations for final validation.

12.2 Results full subsystem

In this section the results of the full subsystem are presented. This is the combination of the LO and the mixer. The simulations are done using an HB simulation in ADS. The simulation setup, as well as the circuit, are shown in Appendix P. In order to validate the performance of the mixer we analyse the conversion gain and the output frequency spectrum of the subsystem. The output frequency spectrum for a 98 MHz RF input and a 88 MHz LO input is given in Figure 12.6a. First of all, it can be seen that the frequency translation property of the mixer is performing well since it produces an output at the difference frequency between both inputs. Furthermore, in Figure 12.7 it can be seen that the RF input is -5.961 dBmV and the output at the IF is 23.423 dBmV which means there is a gain of 29.384 dB. This is slightly lower than calculated but can be easily tuned using R_W. This simulation has been performed for multiple input RF frequencies and the results can be found in Appendix Q.



(a) Frequency spectrum of signal at the output of the mixer.

(b) Frequency spectrum of signal at the LO input.

Figure 12.6: Simulation for the mixer including LO input.

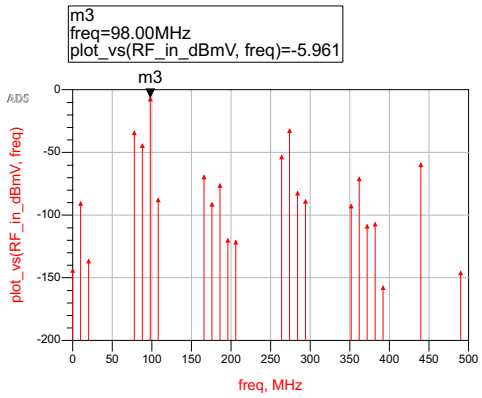


Figure 12.7: Frequency spectrum of signal at the RF input of the mixer.

13. Discussion and future work

In this chapter the results are discussed and some possibilities for future work are presented.

The initial results of the mixer provide confidence in the design. First of all, it can be seen that the peak at the output always occurs at (or near) 10 MHz, which is the desired IF. It has also been shown that the gain is close to the calculated value and can be tuned using the current mirror. The discrepancies between calculated values and actual simulated values may be due to approximations made during the design and calculations. Furthermore, relatively good suppression of the LO and RF signal is achieved at the output. The relatively low suppression from RF to LO should not be a problem since this value might appear low with respect to the other values, however, 27 dB equals a suppression factor of about 500. A possible problem with this circuit is the relatively large frequency component at 20 MHz, which is presumably caused by a combination of feedthrough and mixing of the harmonics. However, more research is needed to confirm this is indeed the case. This problem could degrade the performance of the receiver, but this will depend on the performance of the filter at the input of the IF amplifier. In the case the filter does not suppress this component enough the design of this Gilbert Cell can be easily modified to operate as a linear multiplier. This can be achieved by lowering the output voltage swing of the LO and increasing the tail current in order to reach the desired gain. This would relieve the necessity of increasing the order of the IF filter. The modifications needed to achieve this are shown in Appendix R. Another interesting result is the presence of a DC output voltage while there is a capacitor blocking any DC signal at the output. However, this can be explained by realizing that the capacitor needs to charge and the time constant is relatively large compared to the simulation duration.

As a recommendation for future work we highly recommend performing a transient simulation for the mixer combined with the local oscillator and including an FM input. Furthermore, it would be ideal to develop the mixer, LNA and LO more closely since the topology choice of the mixer can influence the topology choice of the other circuits. In a future project we recommend the use of an LNA with differential output as well as an LO with differential output. This would improve the performance of the circuit by discarding passive baluns as used in this thesis. This can improve the conversion gain of the circuit. Due to time constraints this has not been done in this thesis. However, it is highly encouraged for future projects. In the case of a future hardware implementation of this circuit, we recommend placing the transistors of the Gilbert Cell close to each other to achieve better temperature stability, since the thermal voltage is temperature dependent and appears in the equations that have been derived for the mixer. Another improvement to the circuit would be to add inductive degeneration which improves the linearity of the mixer. Other possible improvement for noise performance, linearity and power can be found in [34]. Furthermore it should be considered to replace the coarse tuning resistance of the LO by a MOSFET biased in the triode region. This allows for an easy implementation of electronic frequency control in future implementations.

14. Conclusion

In this thesis the background, design and simulation of a modulator and mixer have been presented. The modulator is based on the standard Colpitts topology where the resonance frequency is controlled by a set of varactor diodes which are placed in parallel with the other tank components. The mixer consists of a Gilbert-type topology with an active load for a single-ended output. Both circuits are implemented using only discrete components and for the transistors BJT devices are chosen.

In Chapter 7 and Chapter 12 it is verified, by means of simulation, that both the modulator and mixer satisfy all of their respective requirements. Since the result of the simulations agree with predictions, it is expected that both subsystems would work in an actual hardware implementation.

Ultimately, a modulator has been implemented which achieves a linearity of approximately 34 dB for an applied 10 kHz sawtooth baseband signal. The modulator's carrier frequency is tunable from 88 MHz to 108 MHz. Furthermore, it has an output voltage swing which is larger than 7.2 V over the whole tuning range. The final design of the mixer is a double-balanced Gilbert cell with a (tunable) gain of 29.5 dB at an RF frequency of 108 MHz. The power consumption is of 23 mW and the circuit provides high port isolation.

Part IV

Appendices

A. Power Budget

The total system has the power requirements SYS9, SYS10 and SYS11. In order to meet these requirements, some system level calculations on the power budget have to be done.

The total transmitted power must be greater than 20 dBm (100 mW). To determine the amount of power that reaches the receiver, Friis formula for free-space transmission is used, obtained from [36]:

$$\frac{P_r}{P_t} = D_t D_r \left(\frac{\lambda}{4\pi d}\right)^2 \tag{A.1}$$

For this purpose, the equation is rewritten into its dB counterpart:

$$P_{r,dB} = P_{t,dB} + D_{r,dB} + D_{r,dB} + 20\log_{10}\left(\frac{\lambda}{4\pi d}\right) \tag{A.2}$$

In this bachelor project, isotropic antennas with a 50 Ω impedance are used. For isotropic antennas, the directivity is 0 dB and can therefore be removed from Equation (A.2). The carrier wave of the transceiver varies is in the range of 88 MHz until 108 MHz. The free space loss increases for higher frequencies, hence the worst case, in terms of power at the receiver, occurs when the carrier frequency equals 108 MHz.

The minimal transmission distance should be 5 m, which results in a free-space loss of 47 dB, hence a received power of -27 dBm. This is enough to be detected at the receive antenna, but when transmitting 20 dBm, requirement SYS10 can not be met due to the power consumption of the signal generation in the transmitter. To calculate the transmit power required to achieve a 50% efficiency, Equation (A.3) is used.

$$\eta_t = \frac{P_t}{P_t/\eta_{PA} + P_{budget}} \tag{A.3}$$

With η_{PA} the efficiency of the power amplifier, η_t the efficiency of the complete transmitter, and P_{budget} the power used by the other stages in the transmitter. The power amplifier is assumed to have a 70% efficiency, and P_{budget} is assumed to be 300 mW. Solving for a transmission efficiency of 50% results in a required minimum transmission power of 525 mW, 27.2 dBm. Since this requirement is higher than the previous one, this one is considered. Each component in the transmitter will have their own power requirements, based on the total power budget available. The power division for transmitter and receiver can be seen in Table A.1.

Table A.1: Estimated power consumption for each component

Component	**Estimated power**	**Component**	**Estimated power**
LNA	100 mW	Sawtooth generator	150 mW
Mixer	50 mW	Modulator	100 mW
IF amplifier	200 mW	BB1	50 mW
FM Detector	150 mW	Power amplifier	(4.5 W)
BB2	200 mW		
Local Oscillator	100 mW		
Total	**800 mW**	**Total**	**300 mW (4.8 W)**

In this table, the power amplifier is assumed to have a transmit power of 3 W, and an efficiency of 70%. See [29] for the reasoning behind these values.

The total power consumption of the transceiver results in 5.6 W, which satisfies requirement SYS11.

B. LC-circuits

Oscillators often use an LC-tank as the frequency determining element. An example of such a circuit is given in Figure B.1a. Here the values $r_{s,c}$ and $r_{s,l}$ are the respective equivalent series resistances (ESR's) of the capacitor and of the inductor. Often, the two resistances can be written as a parallel equivalent R_P. This is also shown in Figure B.1b

(a) Actual parasitic resistances

(b) Equivalent parallel resistance

Figure B.1: A practical LC-tank circuit with the inductor and capacitor series resistances, and its parallel equivalent.

The value of R_P is the parallel combination of the equivalent parallel resistance R_{LP} of the inductor and the equivalent parallel resistance R_{CP} of the capacitor. R_{LP} and R_{CP} are found using Equation (B.1).

$$R_{LP} \approx \frac{(\omega_0 L)^2}{r_{s,l}} \tag{B.1a}$$

$$R_{CP} \approx \frac{1}{(\omega_0 C)^2 r_{s,c}} \tag{B.1b}$$

Where ω_0 is the resonance frequency. At this frequency the total reactance of the circuit equals zero because the inductive and capacitive reactance are equal in magnitude [37]. Using this we find Equation (B.2).

$$\omega_0 = \frac{1}{\sqrt{LC}} \tag{B.2}$$

Since R_P is the parallel combination of R_{LP} and R_{CP}, it can be found using Equation (B.3).

$$R_P = \frac{R_{LP} R_{CP}}{R_{LP} + R_{CP}} \tag{B.3}$$

An RLC-circuit can be used as a filter. The bandwidth of this filter with respect to its center frequency ω_0 is expressed by the quality factor Q of the filter. The quality factor is defined as:

$$Q = 2\pi \frac{\text{Maximum energy stored}}{\text{Total energy lost per cycle at resonance}}$$

The relation between the bandwidth B, the resonance frequency ω_0, and the quality factor Q is given by Equation (B.4) [37].

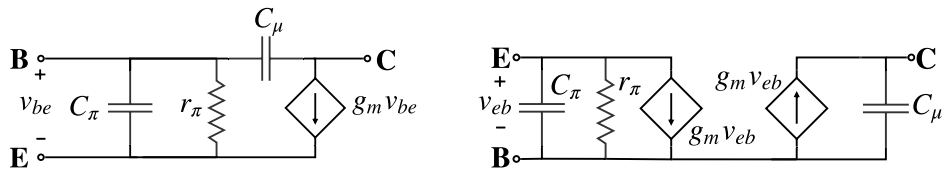
$$B = \frac{\omega_0}{Q} \tag{B.4}$$

Finally, Q can be calculated using Equation (B.5).

$$Q = R_P \sqrt{\frac{C}{L}} \tag{B.5}$$

C. Small-Signal Analysis Open-loop Gain

In this appendix the equation used to calculate N is derived, the derivation is based on the result presented in [38]. The analysis starts with the low-frequency version of the hybrid-π model [23]. Note that a *small-signal* model is an appropriate choice since at startup the oscillator amplifies noise. The model is shown in Figure C.1. Here also the first step in the analysis, namely a current split is given.



(a) Low frequency hybrid-π (b) Redrawn using current split

Figure C.1: The hybrid-π model for low frequencies and a redrawn verison of it.

Substituting the transistor for the redrawn model in the AC-equivalent circuit of the common base Colpitts (see Figure 6.1) yields Figure C.2.

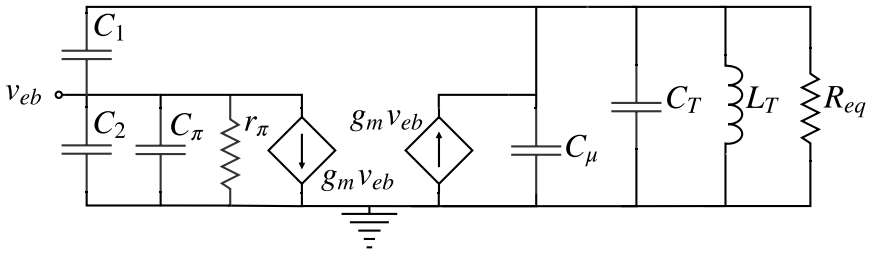
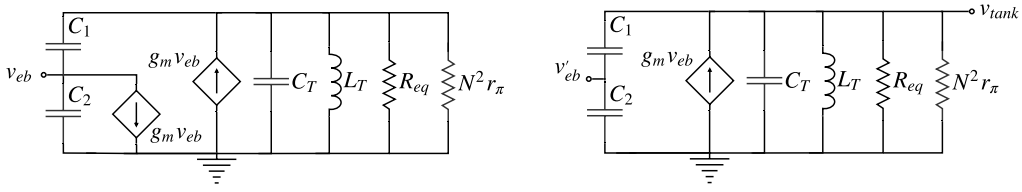


Figure C.2: Small signal model for the common base Colpitts oscillator.

We will now assume that (1) $C_2 \gg C_\pi$, (2) $C_T \gg C_\mu$ and (3) that the capacitors C_1 and C_2 work as an ideal capacitive divider (see Appendix D). This yields the circuit shown in Figure C.3a, recall that $N = {}^{(C_1 + C_2)}/_{C_1}$. By now opening the loop, the circuit shown in Figure C.3b is obtained.



(a) Simplified model (b) Open loop

Figure C.3: The simplified model based on the assumptions and the simplified model with the loop opened.

We will now start calculating the open loop gain. This is done by first defining the impedance

Z_{tot}:

$$Z_{tot} = \left(\frac{1}{sC_1} + \frac{1}{sC_2}\right) \parallel \frac{1}{C_T s} \parallel L_T s \parallel R_{eq} \parallel N^2 r_\pi \tag{C.1}$$

Equivalently:

$$Z_{tot} = \left(C_T s + \frac{1}{R_{eq}} + \frac{1}{L_T s} + \frac{1}{N^2 r_\pi} + \frac{C_1 C_2 s}{C_1 + C_2}\right)^{-1} \tag{C.2}$$

Substituting $s = j\omega_0$, with ω_0 given by Equation (C.3), yields Equation (C.4).

$$\omega_0 = \left\{\left(\frac{C_1 C_2}{C_1 + C_2} + C_T\right) L_T\right\}^{-1/2} \tag{C.3}$$

$$Z_{tot} = \frac{N^2 R_{eq} r_\pi}{R_{eq} + N^2 r_\pi} \tag{C.4}$$

From Figure C.3b, Equations (C.5a) and (C.5b) are derived.

$$v'_{eb} = \frac{1}{N} v_{tank} \tag{C.5a}$$

$$v_{tank} = g_m v_{eb} Z_{tot} \tag{C.5b}$$

By using the two equations and noting that $r_\pi = {}^1/_{g_m}$, the magnitude of the loop gain A_L is determined. It is given by Equation (C.6).

$$A_L = \frac{v_{eb'}}{v_{eb}} = \frac{N R_{eq}}{R_{eq} + {}^{N^2}/_{g_m}} \tag{C.6}$$

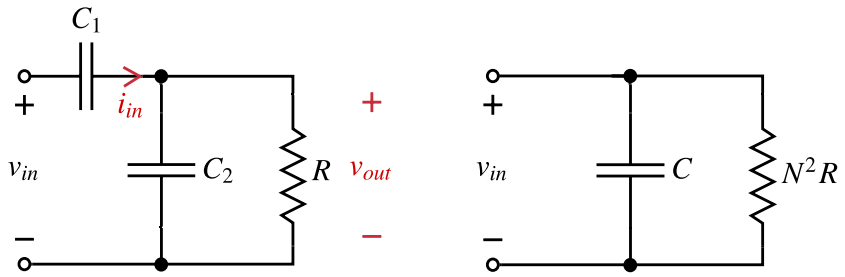
Solving for N yields Equation (C.7).

$$N = \pm\sqrt{\left(\frac{R_{eq} g_m}{2A_L}\right)^2 - g_m R_{eq}} + \frac{R_{eq} g_m}{2A_L} \tag{C.7}$$

From Barkhausens criterion (see Equation (4.4)) it is known that during startup the value for A_L must be larger than one. Hence Equation (C.7) can be used to estimate the needed ratio of C_1 and C_2.

D. The Capacitive Voltage Divider

In this appendix a small discussion about treating the capacitive divider as a capacitive transformer is given. The appendix is based on the analysis presented in [39]. The corresponding circuit is depicted in Figure D.1a. It will be shown that under certain conditions the circuit can be rewritten into the circuit shown in Figure D.1b. Here N, sometimes called the 'turns ratio', is defined as $N = {}^{(C_1+C_2)}/_{C_1}$. The capacitor C equals ${}^{C_1C_2}/_{(C_1+C_2)}$.



(a) Actual circuit (b) Equivalent circuit

Figure D.1: The (a) capacitive transformer network with a load R and (b) its equivalent circuit. The current i_{in} and the voltage v_{out} are indicated because they are used in the calculations.

The impedance seen by the source V_{in} is given by Equation (D.1).

$$\frac{V_{in}(s)}{I_{in}(s)} = Z_{in}(s) = \frac{1}{Y_{in}(s)} \tag{D.1}$$

The current I_{in} is calculated using Equation (D.2).

$$I_{in}(s) = \{V_{in}(s) - V_{out}(s)\}\, sC_1 \tag{D.2}$$

We first need to calculate $V_{out}(s)$ as a function of $V_{in}(s)$. To do this, the impedance of C_2 parallel to R is calculated using Equation (D.3).

$$\frac{1}{sC_2} \parallel R = \frac{R}{1+sC_2R} \tag{D.3}$$

Using the result of (D.3) and by seeing the network as a voltage divider, Equation (D.4) is obtained.

$$\begin{aligned} V_{out}(s) &= V_{in}(s)\frac{R}{1+sC_2R}\left(\frac{R}{1+sC_2R} + \frac{1}{sC_1}\right)^{-1} \\ &= V_{in}(s)\frac{sRC_1}{sR(C_1+C_2)+1} \end{aligned} \tag{D.4}$$

Substituting Equation (D.4) in Equation (D.2) yields Equation (D.5).

$$\begin{aligned} I_{in}(s) &= V_{in}(s)\left\{sC_1 - \frac{s^2C_1^2R}{sRC_1+1+sC_2R}\right\} \\ &= V_{in}(s)\frac{sC_1+s^2C_1C_2R}{sR(C_1+C_2)+1} \end{aligned} \tag{D.5}$$

Now $Y_{in}(s)$ can easily be calculated. Setting $s = j\omega$ yields Equation (D.6).

$$Y_{in} = \frac{j\omega^3 R^2 C_1 C_2 \left(C_1 + C_2\right) + \omega^2 R C_1^2 + j\omega C_1}{1 + \omega^2 R^2 \left(C_1 + C_2\right)^2} \tag{D.6}$$

By now assuming that $\omega^2 R^2 (C_1 + C_2)^2 \gg 1$ and by splitting the equation in a real and imaginary part, Equation (D.7) is obtained.

$$Y_{in} = \frac{C_1^2}{R\left(C_1 + C_2\right)^2} + j\frac{\omega^2 R^2 C_1 C_2 \left(C_1 + C_2\right) + C_1}{\omega R^2 \left(C_1 + C_2\right)^2} \tag{D.7}$$

By making the additional assumption that $\omega^2 R^2 C_2 \left(C_1 + C_2\right) \gg 1$, Equation D.8 is obtained.

$$Y_{in} = \frac{C_1^2}{R\left(C_1 + C_2\right)^2} + j\omega\frac{C_1 C_2}{C_1 + C_2} \tag{D.8}$$

From the above equation it can be concluded that, if the two assumptions hold, the circuit from Figure D.1a can be redrawn into the circuit depicted in Figure D.1b.

E. Transistor Choice

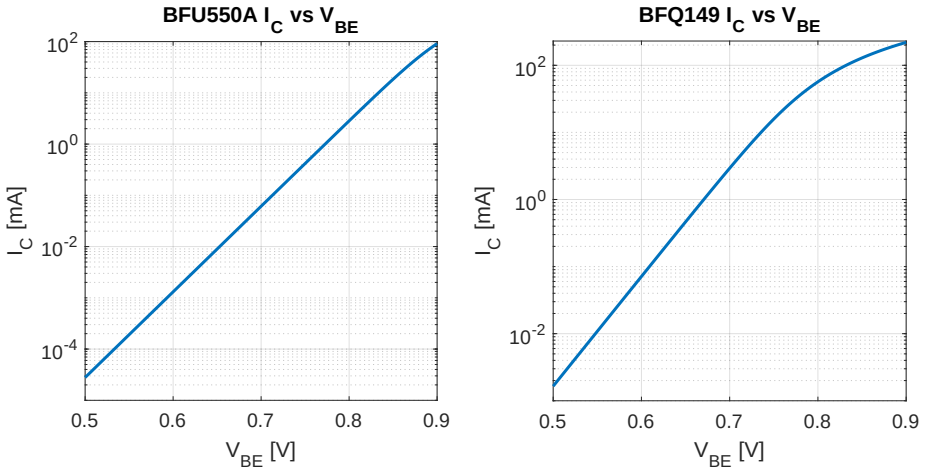
In this project, two types of transistors are being used: a PNP bipolar transistor and NPN bipolar transistors. For the choice of the PNP transistor, only one working PNP bipolar transistor was found in the libraries available, which is the BFQ149 [40]. Its parameters are shown in Table E.1. In this project, BJTs are used, since they offer a wide selection for many applications and are more suitable for high speed circuits compared to MOSFETs [41]. A disadvantage of BJTs compared to MOSFETs is that BJTs typically use more power, since they need a base current, whereas MOSFETs use a base voltage. Therefore the MOSFET uses almost no current [41].

For the choice of the NPN transistor, three different transistors have been considered and compared in Table E.1. The h_{FE} is defined as the common-emitter current gain while the f_T is the transistor cut-off frequency [23, Ch. 12]. The CE and CB in Table E.1 refer to the collector-emitter and collector-base voltage respectively.

Table E.1: Comparison of different transistors. Above are the NPN transistors [42], [43], [26] and below is the PNP transistor [40].

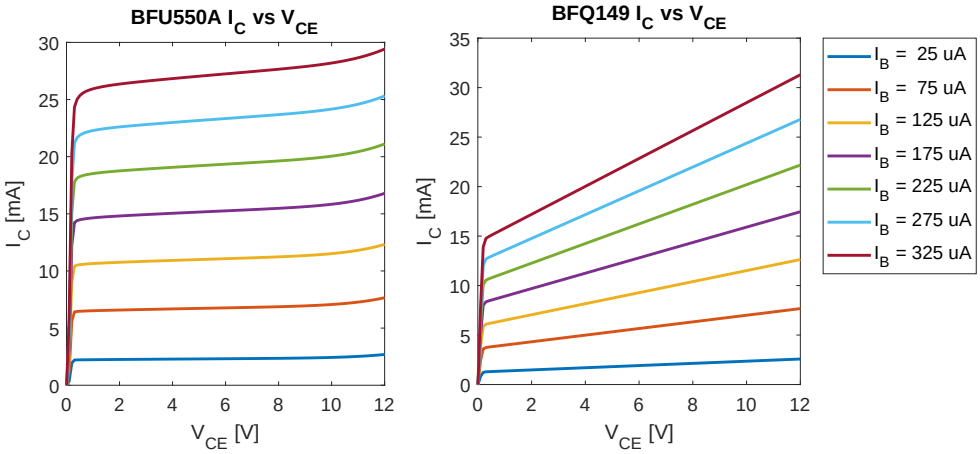
Type	$\mathbf{h_{FE}}$	$\mathbf{f_T}$ **[GHz]**	**Max Current [mA]**	**Max Voltage [V]**	**Price [€]**
BFR92P	100	5	$I_C = 45$, $I_B = 5$	CE = 15, CB = 20	0.46
BFR35AP	100	5	$I_C = 45$, $I_B = 5$	CE = 15, CB = 20	0.37
BFU550A	95	11	$I_C = 50$	CE = 12, CB = 24	0.09
Type	$\mathbf{h_{FE}}$	$\mathbf{f_T}$ **[GHz]**	**Max Current [mA]**	**Max Voltage [V]**	**Price [€]**
BFQ149	50	5	$I_C = 100$	CE = -15, CB = -20	1.35

Since the BFU550A has the highest current rating and it is the cheapest, this transistor is chosen. The maximum voltage ratings are higher than it could encounter during operation, so the lower voltage rating compared to the other two does not influence the decision. A higher current rating is desired, since this provides less restrictions during the design. A cheaper transistor price is also desired, since it will lower the total manufacturing costs.



(a) (b)

Figure E.1: The I_C vs V_{BE} curve of the: (a) BFU550A and (b) BFQ149 for $V_{CE} = 8$ V.

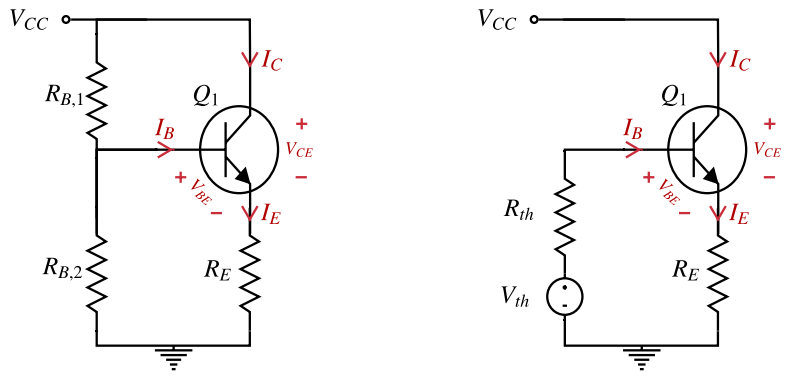


(a) (b)

Figure E.2: The I_C vs V_{CE} curve of the: (a) BFU550A and (b) BFQ149.

F. Transistor Biasing

In this appendix, some background is given on the resistive divider with emitter degeneration biasing network. The advantage of such a network is that the operating point is not very dependent on changes in h_{FE} due to for example temperature variations. The used circuit is shown in Figure F.1a.



(a) The biasing network

(b) Divider redrawn to Thevenin equivalent

Figure F.1: The used biasing network (a) and the same circuit but with the divider redrawn into its Thevenin equivalent (b).

Throughout the analysis, it is assumed that the base current I_B is much smaller than the current flowing through the resistive divider network, this is described by Equation (F.1).

$$\frac{V_{CC}}{R_{B,1}+R_{B,2}} \gg I_B = \frac{I_C}{h_{FE}} \tag{F.1}$$

If Equation (F.1) indeed holds, the circuit can be redrawn using the Thevenin equivalent. The resulting circuit is shown in Figure F.1b. The values R_{th} and V_{th} are given by the equations below.

$$V_{th} = \frac{R_{B,2}}{R_{B,1}+R_{B,2}} V_{CC} \tag{F.2a}$$

$$R_{th} = R_{B,1} \parallel R_{B,2} \tag{F.2b}$$

Using KVL over the loop containing R_{th} yields:

$$V_{th} = I_B R_{th} + V_{BE} + I_B(h_{FE}+1)R_E \tag{F.3}$$

Rewriting Equation (F.3) we find:

$$I_B = \frac{V_{th}-V_{BE}}{R_{th}+(h_{FE}+1)R_E} \approx \frac{V_{th}-V_{BE}}{h_{FE}R_E} \tag{F.4}$$

From Equation (F.4) two conditions for the biasing network follow:

$$R_{B,1}||R_{B,2} \ll h_{FE}R_E \tag{F.5a}$$

$$h_{FE} \gg 1 \tag{F.5b}$$

If the conditions given by Equation (F.5) hold, we find that I_C indeed becomes independent of h_{FE}. This is illustrated in Equation (F.6).

$$I_B \propto \frac{1}{h_{FE}}, \quad I_C = h_{FE} I_B \propto h_{FE} \frac{1}{h_{FE}} \tag{F.6}$$

The value of the resistor R_E is calculated by noticing that the total voltage drop V_{CE} from the collector to the emitter and over R_E should be equal to V_{CC}. We find Equation (F.7).

$$R_E = \frac{V_{CC} - V_{CE}}{(h_{FE} + 1) I_B} \approx \frac{V_{CC} - V_{CE}}{I_C} \tag{F.7}$$

Finally, we can rewrite Equation (F.4) to give us the ratio $R_{B,1}/R_{B,2}$. This yields Equation (F.8).

$$\frac{R_{B,1}}{R_{B,2}} = \frac{V_{CC} - I_C R_E - V_{BE}}{I_C R_E + V_{BE}} \tag{F.8}$$

G. Choosing the Varactor Diode

In this appendix, the process behind choosing the varactor diode is discussed. First recall that the equation for the varactor capacitance is as follows:

$$C_j = \frac{C_{j0}}{(1 + {}^{V_D}/_{V_0})^m} \tag{G.1}$$

All considered varactors are from NXP semiconductors, because they provide easy to use ADS models. The considered varactors are the BB135, BB201, BB207 and BB208, their datasheets are respectively given by [44], [45], [46] and [47].

Their main parameters are summarised in G.1. Here r_s is the series resistance.

Table G.1: Comparison of the BB135, BB201, BB207 and BB208 varactor diodes.

Type	**m**	**C_{j0} [pF]**	**V_0 [V]**	**r_s [Ω]**
BB135	1.0368	23.179	2.4894	0.7
BB201	1.3251	133.21	3.4103	0.23
BB207	0.81153	122.50	1.4881	0.3
BB208	2.2125	30.186	5.7502	0.25

Since multiple sets of anti-series varactors are used, and the total capacitance must reach up to more or less 121 pF (see Section 6.2.1), the BB208 and BB135 are not very practical. Hence, that leaves the BB201 and BB207. Assuming that the varactor capacitances are the dominant capacitive element in setting the frequency, Equation (G.2) is obtained.

$$\omega_0 \overset{\propto}{\sim} \left(1 + \frac{V_D}{V_0}\right)^{m/2} \tag{G.2}$$

From the above equation, it is found that the frequency changes most linearly with linearly changing V_D if ${}^{V_D}/_{V_0} \gg 1$ and if m approaches two. Since we are able to control V_D relatively well but are not able to influence m the BB201 varactor is chosen for its larger m.

H. Modulator Scripts

H.1 Varactor Curves

The script below is used to estimate the frequency of oscillation versus the applied reverse voltage for a given number of varactor diodes.

```
%Author:    Dimme
%Date:      13/05/2020
%Descr:     Varactor frequency curve plotter
%           Plot based on script provided by dr. S. M. Alavi, TU Delft

clear all;
close all;

%Some constants
L        = [27e-9:1e-9:27e-9];  %[H],   tank inductance
C        = 7.1e-12;             %[F],   intrinsic capacitance of osc.

V = 2.0:0.001:11;               %[V],   applied reverse voltage
freq_min = [88, 88];            %[MHz], lower tuning frequency
freq_max = [108, 108];          %[MHz], higher tuning frequency

Cj0_main = 127.21e-12;

%one BB201 contains two varactors)
%Varactor data: 8*BB201, 10*, 12*, 14*
Cj0 = [2*Cj0_main, 2.5*Cj0_main, 3*Cj0_main, 3.5*Cj0_main];   %[F], small-signal junction capacitance
M   = [1.3251, 1.3251, 1.3251, 1.3251, 1.3251];
Vj  = [3.4103, 3.4103, 3.4103, 3.4103];         %[V], build in junction potential

%calculate capacitances of varactors
for(i = 1:length(Cj0))
    Cj(i,:) = Cj0(i) * (V/Vj(i) + 1).^(-M(i));
end

%calculate frequency curve resulting from each varactor
for(i = 1:length(Cj0))
    f(:,i) = ( 2*pi * sqrt( L .* (C + Cj(i,:)))).^(-1);
end

f = f/(10^6);   %[MHz] --> convert Hz to MHz

color_vec = [1 0 0; 0 0.8 0; 0 0 1; 1 0 1; 1 0.5 0; 0.5 0.5 0.5];
plot(V, f(:,:), 'Linewidth', 3)
set(gca,'FontSize',21,'FontName','Arial','FontWeight',...
    'bold','LineWidth',3,...
    'XMinorTick','On',...
    'YMinorTick','On')
set(0,'DefaultAxesColorOrder',color_vec)
hold on
plot([0, 12], freq_max, 'k', 'Linewidth', 3, 'linestyle','--')
hold on
plot([0, 12], freq_min, 'k', 'Linewidth', 3, 'linestyle', '--')
```

```
ylabel("Frequency [MHz]")
xlabel("Reverse voltage [V]")
xlim([min(V) max(V)])
legend(["8 varactors", "10 varactors", "12 varactors", "14 varactors"], 'location', 'northwest')
grid;
Hfig9 = findall(gcf,'type','text');
set(Hfig9,'FontSize',21,'FontName','Arial')
set(gcf, 'Position',  [100, 100, 800, 500])
```

H.2 Frequency vs reverse voltage

The script below is used to extract the maximum frequencies from the frequency domain data generated by ADS. It fits a third-order polynomial to said points and calculates the intersections of the fitted curve with points specified by the user. For calculating the intersections, the function 'intersect' is used, which is given in Appendix I.1.

```
%Author:    Dimme
%Date:      04/05/2020
%Descr:     Script to read frequency data exported from ADS.
%           And estimate the curve using a third order polynomial
%           Plot based on script provided by dr. S. M. Alavi, TU Delft
clear all
close all

fid = fopen('../data_report/modulatorStep3-f_out-sweep.txt','rt');
data = textscan(fid, '%f%f%f%f','Delimiter','/', 'HeaderLines',1);
fclose(fid);

V_axis      = unique(data{1,1});            %used for axis
V_axis      = V_axis';
nbr_curves  = length(V_axis);               %amount of curves
datapoints  = length(data{1,1})/nbr_curves; %datapoints per curve

%frequency limits and y-axis limits
freq_min = [84 84];                         %lower freq threshold
freq_max = [112, 112];                      %upper freq threshold
freq_min_act = [88, 88];                    %lower freq threshold
freq_max_act = [108, 108];                  %upper freq threshold

tmp_amp  = data{1,3};                       %amplitude data
tmp_freq = data{1,2};                       %frequency data

freq = zeros(nbr_curves, datapoints);       %frequency vector
amp  = zeros(nbr_curves, datapoints);       %amplitude vector

for (i = 1:nbr_curves)
    freq(i,:) = tmp_freq((i-1)*datapoints+1:i*datapoints);
    amp(i,:)  =  tmp_amp((i-1)*datapoints+1:i*datapoints);
end

freq = freq';
```

```
36
37 %extract resonance frequencies from data
38 [~, index] = max(amp');
39 f_osc_meas = freq(index) / (10^6);
40
41 %"exact" result [3rd order polynomial]
42 p_2 = polyfit(V_axis, f_osc_meas, 3);
43 V_axis_long = min(V_axis):min(V_axis)/600:max(V_axis);
44 f_osc_fit = p_2(4) + p_2(3)*V_axis_long+p_2(2)*V_axis_long.^2 + p_2(1)*V_axis_long.^3;
45
46 %calculate intersections
47 Vr_max_index = intersect(f_osc_fit, 108);
48 Vr_min_index = intersect(f_osc_fit, 88);
49 Vr_max       = V_axis_long(Vr_max_index);
50 Vr_min       = V_axis_long(Vr_min_index);
51
52 %plot results
53 color_vec = [1 0 0; 0 0 1; 0 0 1; 1 0 1; 1 0.5 0; 0.5 0.5 0.5];  %plot colors
54 set(0,'DefaultAxesColorOrder',color_vec) %plot colors
55 plot(V_axis_long, f_osc_fit, 'Linewidth', 3)
56 hold on;
57 scatter(V_axis, f_osc_meas, 80 ,'filled');
58 hold on;
59 plot([0 100], freq_max_act,'--k', 'Linewidth', 3)
60 hold on;
61 plot([0 100], freq_min_act,'--k', 'Linewidth', 3)
62 ylim([freq_min(1) freq_max(1)]);
63 xlim([min(V_axis), max(V_axis)]);
64 xlabel("Reverse voltage [V]")
65 ylabel("Frequency [MHz]")
66 legend("Third order fit", "Simulated", 'location', 'northwest')
67 grid on;
68 set(gca,'FontSize',21,'FontName',...
69     'Arial','FontWeight',...
70     'bold','LineWidth',3,...
71     'XMinorTick','On',...
72     'YMinorTick','On')
73 Hfig9 = findall(gcf,'type','text');
74 set(Hfig9,'FontSize',21,'FontName','Arial')
75 set(gcf, 'Position',  [100, 100, 800, 500])
```

H.3 Frequency vs tuning resistance

This is the script used to plot the frequency of oscillation as a function of the tuning resistance. Furthermore it used to calculate the required voltage swing from the sawtooth signal at a bandwidth specified by the user. To check if the requirements are satisfied, the script can also be used to verify if the voltage swing does not make the bandwidth at the higher resonance frequency too small. For finding the voltages based on the bandwidth, the function 'find_point' is used. This function is given in Appendix I.2.

```
%Author:    Dimme
%Date:      04/05/2020
%Descr:     Script to read frequency data exported from ADS.
%           And estimate the curve using a third order polynomial
%           Two plots are made, one with R as x-axis and one with the
%           corresponding voltage at the divider at the x-axis.
%           Plot based on script provided by dr. S. M. Alavi, TU Delft

clear all
close all

fid = fopen('../data_report/modulatorStep3-f_out-sweep_R.txt','rt');
data = textscan(fid, '%f%f%f%f','Delimiter','/', 'HeaderLines',1);
fclose(fid);

R_axis      = unique(data{1,1});            %used for axis
R_axis      = R_axis'/10^3;                 %transpose, convert to kOhm
nbr_curves  = length(R_axis);               %amount of curves
datapoints  = length(data{1,1})/nbr_curves; %datapoints per curve

%frequency limits and y-axis limits
freq_min = [84 84];                         %lower freq threshold
freq_max = [112, 112];                      %upper freq threshold
freq_min_act = [88, 88];                    %lower actual freq threshold
freq_max_act = [108, 108];                  %upper actual freq threshold

tmp_amp  = data{1,3};                       %amplitude data
tmp_freq = data{1,2};                       %frequency data

freq = zeros(nbr_curves, datapoints);       %frequency vector
amp  = zeros(nbr_curves, datapoints);       %amplitude vector

%split long arrays into the corresponding set of curves
for (i = 1:nbr_curves)
    freq(i,:) = tmp_freq((i-1)*datapoints+1:i*datapoints);
    amp(i,:)  =  tmp_amp((i-1)*datapoints+1:i*datapoints);
end

freq = freq';

%extract resonance frequencies from data
[~, index] = max(amp');
f_osc_meas = freq(index) / (10^6);

%limit linear fit to data in FM range
tmp1 = find(f_osc_meas>freq_min_act(1)-2);
f_osc_meas = f_osc_meas(tmp1);
R_axis = R_axis(tmp1);
tmp1 = find(f_osc_meas<freq_max_act(1)+2);
f_osc_meas = f_osc_meas(tmp1);
R_axis = R_axis(tmp1);

```

```
%linear result
p_1 = polyfit(R_axis, f_osc_meas, 1);
f_osc_ideal = p_1(2) + p_1(1)*R_axis;

%"exact" result [3rd order polynomial]
p_2 = polyfit(R_axis, f_osc_meas, 3);
R_axis_long = min(R_axis):max(R_axis)/1000:max(R_axis);
f_osc_fit = p_2(4) + p_2(3)*R_axis_long+p_2(2)*R_axis_long.^2 + p_2(1)*R_axis_long.^3;

%calculate intersections
R_max_index = intersect(f_osc_fit, 108);
R_min_index = intersect(f_osc_fit, 88);
R_max       = R_axis_long(R_max_index);
R_min       = R_axis_long(R_min_index);

%plot results
figure
color_vec = [1 0 0; 0 0.8 0; 0 0 1; 1 0 1; 1 0.5 0; 0.5 0.5 0.5];  %plot colors
set(0,'DefaultAxesColorOrder',color_vec) %plot colors
plot(R_axis_long, f_osc_fit,'Linewidth', 3)
hold on;
plot(R_axis, f_osc_ideal, 'Linewidth', 3)
hold on;
scatter(R_axis, f_osc_meas, 80, 'filled');
hold on;
plot([0 100], freq_max_act,'--k', 'Linewidth', 3)
hold on;
plot([0 100], freq_min_act,'--k', 'Linewidth', 3)
grid on;
ylim([freq_min(1) freq_max(1)]);
xlim([min(R_axis), max(R_axis)]);
xlabel("Tuning resistance [kOhm]")
ylabel("Frequency [MHz]")
legend("Third order fit","Linear fit","Simulated", 'Location', 'northwest')
set(gca,'FontSize',21,'FontName',...
    'Arial','FontWeight',...
    'bold','LineWidth',3,...
    'XMinorTick','On',...
    'YMinorTick','On')
Hfig9 = findall(gcf,'type','text');
set(Hfig9,'FontSize',21,'FontName','Arial')
set(gcf, 'Position',  [100, 100, 650, 500])

%extra if also the average divider voltage has been calculated
fid = fopen('../data_report/modulatorStep3-f_out-sweep_R_Vdiv_avg.txt','rt');
data2 = textscan(fid, '%f%f%f%f','Delimiter','/', 'HeaderLines',1);
fclose(fid);
V_axis = data2{1,2};

%"exact" result [3rd order polynomial]
p_3 = polyfit(V_axis', f_osc_meas, 3);
V_axis_long = min(V_axis)-0.3:max(V_axis)/600:max(V_axis)+0.3;
f_osc_fit2 = p_3(4) + p_3(3)*V_axis_long+p_3(2)*V_axis_long.^2 + p_3(1)*V_axis_long.^3;
```

```

%plot results
figure
color_vec = [1 0 0; 0 0 1; 0 0 1; 1 0 1; 1 0.5 0; 0.5 0.5 0.5];  %plot colors
set(0,'DefaultAxesColorOrder',color_vec) %plot colors
plot(V_axis_long, f_osc_fit2,'Linewidth', 3)
hold on;
scatter(V_axis, f_osc_meas, 80, 'filled');
hold on;
plot([0 100], freq_max_act,'--k', 'Linewidth', 3)
hold on;
plot([0 100], freq_min_act,'--k', 'Linewidth', 3)
grid on;
ylim([freq_min(1) freq_max(1)]);
xlim([min(V_axis)-0.3, max(V_axis)+0.3]);
xlabel("Divider voltage [V]")
ylabel("Frequency [MHz]")
legend( "Third order fit", "Simulated", 'Location', 'northwest')
grid on;
set(gca,'FontSize',21,'FontName',...
    'Arial','FontWeight',...
    'bold','LineWidth',3,...
    'XMinorTick','On',...
    'YMinorTick','On')
Hfig9 = findall(gcf,'type','text');
set(Hfig9,'FontSize',21,'FontName','Arial')
set(gcf, 'Position',  [100, 100, 650, 500])

%find the intersection points resulting in 350 kHz bandwidth at the lower
%carrier frequency
bw = 350e-3;
[vrL_low, vrL_mid, vrL_high]  = find_point(f_osc_fit2, f_osc_meas(1), bw, V_axis_long, 1);
[fH_low, fH_mid, fH_high]  = find_point(f_osc_fit2, f_osc_meas(end), vrL_high-vrL_low, V_axis_long,
↪  2);

"Voltage swing for specified bandwidth: "
vrL_high - vrL_low

"Minimal bandwidth for given voltage swing: "
fH_high - fH_low
```

H.4 Bandwidth Sawtooth

The function used to estimate the 98% power bandwidth of a 10kHz sawtooth wave is given below.

```
%Author:    Dimme
%Date:      12/06/2020
%Descr:     script used to estimate the 98% power bandwidth of a
%           10kHz sawtooth signal

Fc = 10e3;          %10kHz carrier
```

```
Fs = 30e6;          %sample rate of 100MHz
N  = 10000;         %number of periods
t  = 0:1/Fs:N/Fc;
F = 0:Fs/(length(t)-1):Fs;
F = F - Fs/2;

wave = sawtooth(2*pi*Fc*t);
bw3 = obw(wave, Fs, [], 98);
```

H.5 Voltage Swing and Harmonics

This is the code used to plot the frequency and time domain data of the output voltage. It also returns the peak-to-peak output voltage by using the average of both the negative and positive peaks in the data. Note that this method is only valid when the data is free of significant ripple.

```
%Date:      10/06/2020
%Descr:     Script to read frequency data exported from ADS.
%           Since the time-axis data of ADS is not always constant,
%           seperate data sets are used. The function ads_to_fft is used
%           to sample the ads datasets and return the FFT but also the
%           sampled data if needed
%           The frequency spectrum and time domain signal are plotted
%           Plot based on script provided by dr. S. M. Alavi, TU Delft

clear all
close all

fid = fopen('../data_report/RESULTS1modulatorV_out-R0.txt','rt');
data1 = textscan(fid, '%f%f%f%f','Delimiter','/', 'HeaderLines',1);
fclose(fid);

fid = fopen('../data_report/RESULTS1modulatorV_out-R15_6.txt','rt');
data2 = textscan(fid, '%f%f%f%f','Delimiter','/', 'HeaderLines',1);
fclose(fid);

fid = fopen('../data_report/RESULTS1modulatorV_out-R50.txt','rt');
data3 = textscan(fid, '%f%f%f%f','Delimiter','/', 'HeaderLines',1);
fclose(fid);

Fs = 2e9;

tmp_t1 = data1{1,1};   %time data
tmp_a1 = data1{1,2};   %amplitude data
tmp_t2 = data2{1,1};   %time data
tmp_a2 = data2{1,2};   %amplitude data
tmp_t3 = data3{1,1};   %time data
tmp_a3 = data3{1,2};   %amplitude data

[tmp_f1, tmp_f1a, ~, ~, ~] = ads_to_fft(tmp_t1, tmp_a1, Fs, 'spline', 1);
[tmp_f2, tmp_f2a, ~, ~, ~] = ads_to_fft(tmp_t2, tmp_a2, Fs, 'spline', 1);
[tmp_f3, tmp_f3a, ~, t, a] = ads_to_fft(tmp_t3, tmp_a3, Fs, 'spline', 1);
```

```
37
38 %plot voltage results
39 figure
40 color_vec = [1 0 0; 0 0.8 0; 0 0 1; 1 0 1; 1 0.5 0; 0.5 0.5 0.5];  %plot colors
41 set(0,'DefaultAxesColorOrder',color_vec) %plot colors
42 plot(tmp_t1*10^6, tmp_a1, 'linewidth', 3);
43 hold on;
44 plot(tmp_t2*10^6, tmp_a2, 'linewidth', 3);
45 plot(tmp_t3*10^6, tmp_a3, 'linewidth', 3);
46 ylim([-5 5]);
47 xlim([80.0, 80.05]);
48 ylabel("Voltage [V]")
49 xlabel("Time [us]")
50 legend("0 kOhm", "15.6 kOhm", "50 kOhm")
51 grid on;
52 set(gca,'FontSize',21,'FontName',...
53     'Arial','FontWeight',...
54     'bold','LineWidth',3,...
55     'XMinorTick','On', 'XTick', [80:0.02:80.05],...
56     'YMinorTick','On',...
57     'YTick',[-5:2.5:5]);
58 Hfig9 = findall(gcf,'type','text');
59 set(Hfig9,'FontSize',21,'FontName','Arial')
60 set(gcf, 'Position',  [100, 100, 650, 500])
61
62 %voltage swing
63 mean(findpeaks(tmp_a1)) + mean(findpeaks(-tmp_a1))
64 mean(findpeaks(tmp_a2)) + mean(findpeaks(-tmp_a2))
65 mean(findpeaks(tmp_a3)) + mean(findpeaks(-tmp_a3))
66
67 %plot frequency results
68 figure
69 plot(tmp_f1, tmp_f1a, 'linewidth', 3);
70 hold on;
71 plot(tmp_f2, tmp_f2a, 'linewidth', 3);
72 plot(tmp_f3, tmp_f3a, 'linewidth', 3);
73 xlim([0, 350]);
74 ylim([-inf 10])
75 ylabel("Voltage [dBV]")
76 xlabel("Frequency [MHz]")
77 legend("0 kOhm", "15.6 kOhm", "50 kOhm")
78 grid on;
79 set(gca,'FontSize',21,'FontName',...
80     'Arial','FontWeight',...
81     'bold','LineWidth',3,...
82     'XMinorTick','On',...
83     'YMinorTick','On')
84 Hfig9 = findall(gcf,'type','text');
85 set(Hfig9,'FontSize',21,'FontName','Arial')
86 set(gcf, 'Position',  [100, 100, 650, 500])
```

H.6 Bandwidth

The MATLAB code used to estimate the 98% power bandwidth occupied by the output modulator is given below.

```
%Date:      10/06/2020
%Author:    Dimme
%Descr:     Script to read frequency data exported from ADS.
%           Since the time-axis data of ADS is not always constant,
%           seperate data sets are used. The function ads_to_fft is used
%           to sample the ads datasets and return the FFT but also the
%           sampled data if needed
%           This script is used to print the bandwidths of the signal
%           for three different carrier frequencies
%           Plot based on script provided by dr. S. M. Alavi, TU Delft

clear all
close all

fid = fopen('../data_report/RESULTS2modulatorV_out-R0-2period.txt','rt');
data1 = textscan(fid, '%f%f%f%f','Delimiter','/', 'HeaderLines',1);
fclose(fid);

fid = fopen('../data_report/RESULTS2modulatorV_out-R15_6-2period.txt','rt');
data2 = textscan(fid, '%f%f%f%f','Delimiter','/', 'HeaderLines',1);
fclose(fid);
%
fid = fopen('../data_report/RESULTS2modulatorV_out-R50-2period.txt','rt');
data3 = textscan(fid, '%f%f%f%f','Delimiter','/', 'HeaderLines',1);
fclose(fid);

Fs = 5e9;

tmp_t1 = data1{1,1};   %time data
tmp_a1 = data1{1,2};   %amplitude data
tmp_t2 = data2{1,1};   %time data
tmp_a2 = data2{1,2};   %amplitude data
tmp_t3 = data3{1,1};   %time data
tmp_a3 = data3{1,2};   %amplitude data

[tmp_f1, tmp_f1a, ~, t1, a1] = ads_to_fft(tmp_t1, tmp_a1, Fs, 'spline', 1);
[tmp_f2, tmp_f2a, ~, t2, a2] = ads_to_fft(tmp_t2, tmp_a2, Fs, 'spline', 1);
[tmp_f3, tmp_f3a, ~, t3, a3] = ads_to_fft(tmp_t3, tmp_a3, Fs, 'spline', 1);

%calculate 98% power bandwidth
bw1 = obw(a1, Fs, [], 98);
bw1 = bw1*10^(-3);
bw2 = obw(a2, Fs, [], 98);
bw2 = bw2*10^(-3);
bw3 = obw(a3, Fs, [], 98);
bw3 = bw3*10^(-3);

%plot frequency results
```

```
figure
color_vec = [1 0 0; 0 0.8 0; 0 0 1; 1 0 1; 1 0.5 0; 0.5 0.5 0.5];  %plot colors
set(0,'DefaultAxesColorOrder',color_vec) %plot colors
plot(tmp_f1, tmp_f1a, 'linewidth', 3);
hold on;
plot(tmp_f2, tmp_f2a, 'linewidth', 3);
plot(tmp_f3, tmp_f3a, 'linewidth', 3);
grid on;
xlim([83, 113]);
ylim([-120 0])
ylabel("Voltage [dBV]")
xlabel("Frequency [MHz]")
legend("0 kOhm", "15.6 kOhm", "50 kOhm", 'Location', 'northeastoutside')
set(gca,'FontSize',21,'FontName',...
    'Arial','FontWeight',...
    'bold','LineWidth',3,...
    'XMinorTick','On',...
    'YMinorTick','On')
Hfig9 = findall(gcf,'type','text');
set(Hfig9,'FontSize',21,'FontName','Arial')
set(gcf, 'Position',  [100, 100, 860, 500])
```

H.7 Modulator linearity

In this appendix the code used to calculate the linearity of the modulation for a baseband sawtooth signal is given. It makes use of the two functions 'demodulator' and 'fitter_fnc'. The first demodulates the signal, while the second is used to fit the sawtooth. Since they require a bit more explanation, they are respectively given in Appendix I.4 and Appendix I.5. The code used to calculate the linearity is given below.

```
%Date:      10/06/2020
%Author:    Dimme
%Descr:     Script to read frequency data exported from ADS.
%           Since the time-axis data of ADS is not always constant,
%           seperate data sets are used. The function ads_to_fft is used
%           to sample the ads datasets and return the FFT but also the
%           sampled data if needed
%           The signal is demodulated using the function 'demodulator' and
%           one signal period is selected by 'fitter_fnc', here the used
%           sawtooth is estimated and fitted against the signal
%           Plot based on script provided by
%           dr. S. M. Alavi, TU Delft

clear all
close all

fid = fopen('../data_report/RESULTS3modulatorV_out-R0-linearity.txt','rt');
data1 = textscan(fid, '%f%f%f%f','Delimiter','/', 'HeaderLines',1);
fclose(fid);

fid = fopen('../data_report/RESULTS3modulatorV_out-R15_6-linearity.txt','rt');
```

```
data2 = textscan(fid, '%f%f%f%f','Delimiter','/', 'HeaderLines',1);
fclose(fid);

fid = fopen('../data_report/RESULTS3modulatorV_out-R50-linearity.txt','rt');
data3 = textscan(fid, '%f%f%f%f','Delimiter','/', 'HeaderLines',1);
fclose(fid);

Fs = 10e9;
period = 10/11; %percentage of time sawtooth is rising

tmp_t1 = data1{1,1};   %time data
tmp_a1 = data1{1,2};   %amplitude data
tmp_t2 = data2{1,1};   %time data
tmp_a2 = data2{1,2};   %amplitude data
tmp_t3 = data3{1,1};   %time data
tmp_a3 = data3{1,2};   %amplitude data

%sample ADS data to get uniform axisses
[~, ~, ~, t1, a1] = ads_to_fft(tmp_t1, tmp_a1, Fs, 'spline');
[~, ~, ~, t2, a2] = ads_to_fft(tmp_t2, tmp_a2, Fs, 'spline');
[~, ~, ~, t3, a3] = ads_to_fft(tmp_t3, tmp_a3, Fs, 'spline');

[MOD1, TMOD1, ~]   = demodulator(a1, t1, Fs);
TMOD1 = TMOD1*10^6;
[rise1, fall1, fit1, signal1]  = fitter_fnc(MOD1, TMOD1, Fs, period);
edge_rise1 = fit1(1,2) + rise1(1,:)*fit1(1,1); %construct polynomial curve
edge_fall1 = fit1(2,2) + fall1(1,:)*fit1(2,1); %construct polynomial curve

[MOD2, TMOD2, ~]   = demodulator(a2, t2, Fs);
TMOD2 = TMOD2*10^6;
[rise2, fall2, fit2, signal2]  = fitter_fnc(MOD2, TMOD2, Fs, period);
edge_rise2 = fit2(1,2) + rise2(1,:)*fit2(1,1); %construct polynomial curve
edge_fall2 = fit2(2,2) + fall2(1,:)*fit2(2,1); %construct polynomial curve

[MOD3, TMOD3, ~]   = demodulator(a3, t3, Fs);
TMOD3 = TMOD3*10^6;
[rise3, fall3, fit3, signal3]  = fitter_fnc(MOD3, TMOD3, Fs, period);
edge_rise3 = fit3(1,2) + rise3(1,:)*fit3(1,1); %construct polynomial curve
edge_fall3 = fit3(2,2) + fall3(1,:)*fit3(2,1); %construct polynomial curve

figure % 0 kOhm
color_vec = [1 0 0; 0 0.8 0; 0 0.8 0; 1 0 1; 1 0.5 0; 0.5 0.5 0.5];  %plot colors
set(0,'DefaultAxesColorOrder',color_vec) %plot colors
plot(signal1(1,:), signal1(2,:), 'linewidth', 3)
hold on
grid on
plot(rise1(1,:), edge_rise1,  'linewidth', 3, 'linestyle', '--')
plot(fall1(1,:), edge_fall1,  'linewidth', 3, 'linestyle', '--')
xlim([signal1(1,1) signal1(1,end)]);
ylim([min(signal1(2,:))-0.0001 0.0001+max(signal1(2,:))])
ylabel("Amplitude")
xlabel("Time [us]")
legend(["Simulated", "Fitted"], 'location', 'northwest')
```

```
set(gca,'FontSize',21,'FontName',...
    'Arial','FontWeight',...
    'bold','LineWidth',3,...
    'XMinorTick','On',...
    'YMinorTick','On')
Hfig9 = findall(gcf,'type','text');
set(Hfig9,'FontSize',21,'FontName','Arial')
set(gcf, 'Position',  [100, 100, 650, 500])

figure %15 kOhm
plot(signal2(1,:), signal2(2,:), 'linewidth', 3)
hold on
grid on
plot(rise2(1,:), edge_rise2,  'linewidth', 3, 'linestyle', '--')
plot(fall2(1,:), edge_fall2,  'linewidth', 3, 'linestyle', '--')
xlim([signal2(1,1) signal2(1,end)]);
ylim([min(signal2(2,:))-0.0001 0.0001+max(signal2(2,:))])
ylabel("Amplitude")
xlabel("Time [us]")
set(gca,'FontSize',21,'FontName',...
    'Arial','FontWeight',...
    'bold','LineWidth',3,...
    'XMinorTick','On',...
    'YMinorTick','On')
Hfig9 = findall(gcf,'type','text');
set(Hfig9,'FontSize',21,'FontName','Arial')
legend(["Simulated", "Fitted"], 'location', 'northwest')
set(gcf, 'Position',  [100, 100, 650, 500])

figure %50 kOhm
plot(signal3(1,:), signal3(2,:), 'linewidth', 3)
hold on
grid on
plot(rise3(1,:), edge_rise3, 'linewidth', 3, 'linestyle', '--')
plot(fall3(1,:), edge_fall3, 'linewidth', 3, 'linestyle', '--')
xlim([signal3(1,1) signal3(1,end)]);
ylim([min(signal3(2,:))-0.0001 0.0001+max(signal3(2,:))])
ylabel("Amplitude")
xlabel("Time [us]")
set(gca,'FontSize',21,'FontName',...
    'Arial','FontWeight',...
    'bold','LineWidth',3,...
    'XMinorTick','On',...
    'YMinorTick','On')
Hfig9 = findall(gcf,'type','text');
set(Hfig9,'FontSize',21,'FontName','Arial')
legend(["Simulated", "Fitted"], 'location', 'northwest')
set(gcf, 'Position',  [100, 100, 650, 500])

%calculate error
error_vectora = abs(edge_rise1 - rise1(2,:));
error_vectorb = abs(edge_fall1 - fall1(2,:));
error_vector(1,:)  = [error_vectora error_vectorb];
```

```
128
129 error_vectora = abs(edge_rise2 - rise2(2,:));
130 error_vectorb = abs(edge_fall2 - fall2(2,:));
131 error_vector(2,:)  = [error_vectora error_vectorb];
132
133 error_vectora = abs(edge_rise3 - rise3(2,:));
134 error_vectorb = abs(edge_fall3 - fall3(2,:));
135 error_vector(3,:)  = [error_vectora error_vectorb];
136
137 pkpk(1) = max([rise1(2,:) fall1(2,:)]) - min([rise1(2,:) fall1(2,:)]);
138 pkpk(2) = max([rise2(2,:) fall2(2,:)]) - min([rise2(2,:) fall2(2,:)]);
139 pkpk(3) = max([rise3(2,:) fall3(2,:)]) - min([rise3(2,:) fall3(2,:)]);
140 error = max(error_vector(:,5:end-5)');
141
142 figure %0 kOhm
143 plot(signal1(1,:), error_vector(1,:), 'linewidth',3)
144 grid on
145 xlim([signal1(1,1) signal1(1,end)]);
146 ylabel("Magnitude")
147 xlabel("Time [us]")
148 set(gca,'FontSize',21,'FontName',...
149     'Arial','FontWeight',...
150     'bold','LineWidth',3,...
151     'XMinorTick','On',...
152     'YMinorTick','On')
153 Hfig9 = findall(gcf,'type','text');
154 set(Hfig9,'FontSize',21,'FontName','Arial')
155 set(gcf, 'Position',  [100, 100, 650, 500])
156
157 figure %15 kOhm
158 plot(signal2(1,:), error_vector(2,:),  'linewidth',3)
159 grid on
160 xlim([signal2(1,1) signal2(1,end)]);
161 ylabel("Magnitude")
162 xlabel("Time [us]")
163 set(gca,'FontSize',21,'FontName',...
164     'Arial','FontWeight',...
165     'bold','LineWidth',3,...
166     'XMinorTick','On',...
167     'YMinorTick','On')
168 Hfig9 = findall(gcf,'type','text');
169 set(Hfig9,'FontSize',21,'FontName','Arial')
170 set(gcf, 'Position',  [100, 100, 650, 500])
171
172 figure %50 kOhm
173 plot(signal3(1,:), error_vector(3,:), 'linewidth',3)
174 grid on
175 xlim([signal3(1,1) signal3(1,end)]);
176 ylabel("Magnitude")
177 xlabel("Time [us]")
178 set(gca,'FontSize',21,'FontName',...
179     'Arial','FontWeight',...
180     'bold','LineWidth',3,...
```

```
    'XMinorTick','On',...
    'YMinorTick','On')
Hfig9 = findall(gcf,'type','text');
set(Hfig9,'FontSize',21,'FontName','Arial')
set(gcf, 'Position',  [100, 100, 650, 500])

linearity = 20*log10(pkpk./error);
```

H.8 Phase Noise over Bandwidth

The code used to estimate the signal to phase noise ratio inside the modulation bandwidth is given below.

```
%date:      18/06/2020
%author:    Dimme de Groot
%Descr:     Calculate power of phase noise inside bandwidth
%`          Note: noise spectrum is single sided
%           hence use bandwidth/2

clear all;

fid = fopen('../data_report/RESULTS5modulator-PN_R0-int.txt','rt');
data1 = textscan(fid, '%f%f','Delimiter','\t', 'HeaderLines',1);
fclose(fid);
fid = fopen('../data_report/RESULTS5modulator-PN_R156-int.txt','rt');
data2 = textscan(fid, '%f%f','Delimiter','\t', 'HeaderLines',1);
fclose(fid);
fid = fopen('../data_report/RESULTS5modulator-PN_R50-int.txt','rt');
data3 = textscan(fid, '%f%f','Delimiter','\t', 'HeaderLines',1);
fclose(fid);

tmp_1a = data1{1,1};%noise freq
tmp_1b = data1{1,2};
tmp_2a = data2{1,1};%noise freq
tmp_2b = data2{1,2};
tmp_3a = data3{1,1};%noise freq
tmp_3b = data3{1,2};

bw = [399, 364, 352]*10^3; %bandwidth [Hz]

%convert back from dBc/Hz
tmp_1c = 10.^(tmp_1b./10);
tmp_2c = 10.^(tmp_2b./10);
tmp_3c = 10.^(tmp_3b./10);

%find indices corresponding to half bandwidth
index1 = intersect(tmp_1a, bw(1)/2);
index2 = intersect(tmp_2a, bw(2)/2);
index3 = intersect(tmp_3a, bw(3)/2);

n_1a = tmp_1c(1:index1);
```

```
n_2a = tmp_2c(1:index2);
n_3a = tmp_3c(1:index3);
f_a1 = tmp_1a(1:index1);
f_a2 = tmp_2a(1:index2);
f_a3 = tmp_3a(1:index3);

%perform the integration using matlab built-in
P1 = trapz(f_a1, n_1a);
P2 = trapz(f_a2, n_2a);
P3 = trapz(f_a3, n_3a);

%convert to dBc and print
10*log10(1/P1)
10*log10(1/P2)
10*log10(1/P3)
```

I. Helper Functions

In this appendix, the functions used by the main codes in Appendix H are given.

I.1 The function 'intersect'

The function 'intersect' is used to calculate the point in a vector which is closest to a point specified by the user.

```
%Author:        Dimme de Groot
%Date:          25/05/20202
%Description:   Used to find the index where the data is closest to a point
%               "index = intersect(data, point)"

function index = intersect(data, point)
    data_tmp = abs(data - point);
    [~, index] = min(data_tmp);
end
```

I.2 The function 'find_point'

This function is imilar to 'intersect', however here also a bandwidth is specified. The intersections corresponding to 'point $\pm\frac{\text{bandwidth}}{2}$' are also returned.

```
%Author:        Dimme de Groot
%Date:          10/06/20202
%Description:   Used to find the index where the data is closest to a point

function [res_l, res_m, res_h] = find_point(data, point, bw, axis, option)
    if option == 1
        index_m = intersect(data, point);
        index_h = intersect(data, point+bw/2);
        index_l = intersect(data, point-bw/2);
        res_m   = axis(index_m);
        res_h   = axis(index_h);
        res_l   = axis(index_l);
    else
        index_m = intersect(data, point);
        tmp     = axis(index_m);
        index_l = intersect(axis, tmp - bw/2);
        index_h = intersect(axis, tmp + bw/2);
        res_m   = data(index_m);
        res_l   = data(index_l);
        res_h   = data(index_h);
    end;
end
```

I.3 The function 'ads_to_fft'

This function is used to sample the non-uniformly spaced data obtained from ADS. After this, it performs an FFT on the data. Both the sampled data, a frequency axis, and the magnitude and phase of the FFT is returned. The returned axis units are MHz. By default spline interpolation is used and no normalisations are done.

```
%date:      25/05/2020
%author:    Dimme de Groot
%Descr:     Function used to obtain the FFT from the ads time-domain data
%           "[X, Y, x, y] = ads_to_fft(time_axis, ampl_axis, Fs)"
%           with X,Y the Frequency counterpart of x, y
%           FFt returned in MHz!!!

function [X, Y, Y_phase, x, y] = ads_to_fft(time_axis, ampl_axis, Fs, type, option1, option2)
    if nargin < 6
        option2 = 0;
    end

    if nargin < 5
        option1 = 0;
    end

    if nargin < 4
        type = 'spline';
    end

    %define sample axis and sample data
    x = time_axis(1):1/Fs:time_axis(end);
    y = interp1(time_axis,ampl_axis,x, type);

    %transform to frequency domain
    X = [0:1/length(y):1-1/(length(y))]*Fs;     %define x-axis (1)
    X = (X-Fs/2)/(10^6);                        %define x-axis (2)
    Y = fft(y);

    if option1 ~= 0
        Y = Y/length(Y);    %approx. same amplitude for all sample freq
    end
    Y = 20*log10(abs(Y));   %do FFT, convert to dB

    if option2 ~= 0
       Y = Y - max(Y);      %normalise
    end

    Y_phase = angle(Y)*180/pi;
    Y = fftshift(Y);
end
```

I.4 The function 'demodulation'

In this section, the code which is used to perform the demodulation is given. The code is based on the blockdiagram given in Figure I.1.

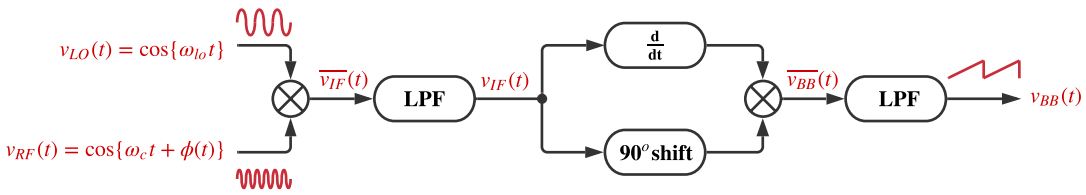


Figure I.1: Blockdiagram describing the method used to perform the demodulation.

At the inputs of the multiplier, the RF signal and the LO signal are present. Multiplying them yields the result given in Equation (I.1). Here, ω_{IF} is the intermediate frequency, it equals $\omega_C - \omega_{LO}$.

$$\overline{v_{IF}}(t) = v_{RF}(t)v_{LO}(t) = \frac{1}{2}\left[\cos\left\{\omega_{IF}t + \phi(t)\right\} + \cos\left\{(\omega_C + \omega_{LO})t\right\}\right] \tag{I.1}$$

The obtained signal is now passed through a Low Pass Filter (LPF), hence only the IF component $v_{IF}(t)$ remains:

$$v_{IF}(t) = \frac{1}{2}\cos\left\{\omega_{IF}t + \phi(t)\right\} \tag{I.2}$$

The signal is now split in two, of one part the derivative is taken and the other part is passed through the Hilbert transform resulting in a 90° phaseshift. This is given in Equation I.3. Note that the baseband signal is given by $\frac{d\phi(t)}{dt}$.

$$\frac{dv_{IF}(t)}{dt} = -\frac{1}{2}\left(\omega_{IF} + \frac{d\phi(t)}{dt}\right)\sin\left\{\omega_{IF}t + \phi(t)\right\} \tag{I.3a}$$

$$v_{IF,shift}(t) = -\frac{1}{2}\sin\left\{\omega_{IF}t + \phi(t)\right\} \tag{I.3b}$$

By now multiplying Equations (I.3a) and (I.3b), we obtain Equation (I.4).

$$\overline{v_{BB}}(t) = \frac{1}{8}\left(\omega_{IF} + \frac{d\phi(t)}{dt}\right)\left(1 - \cos\left\{2\omega_{IF}t + 2\phi(t)\right\}\right) \tag{I.4}$$

Passing the result through the second LPF yields Equation (I.5).

$$v_{BB}(t) = \frac{1}{8}\left(\omega_{IF} + \frac{d\phi(t)}{dt}\right) \tag{I.5}$$

The found results is a constant added to a scaled version of the baseband signal and as such can be used to calculate the linearity. The code that is used to perform the demodulation is given below.

```
1 %Author:    Dimme
2 %Date:      12/06/2020
```

```
%Descr:     Downconvert signal to the IF and subsequently demodulate it
%           make sure there area at least two full periods + some edge
%           information in the signal

function [signal_out, time_out, Fc] = demodulator(signal_in, time_in, Fs, IF)
    if nargin<4
        IF = 10e6; %10 MHz
    end
    if nargin<3
        %time_in can easily be constructed from Fs
    end

    %get carrier, construct LO
    Fc = meanfreq(signal_in, Fs);
    Fd = Fc - IF;
    mixer_signal = sin(2*pi*Fd*time_in);

    %downconvert to IF, apply LPF
    signal_int1 = signal_in.*mixer_signal;
    signal_int2 = lowpass(signal_int1, 2*IF, Fs, 'ImpulseResponse','iir','Steepness',0.95);

    %the intermediate signal is (a) differentiated, (b) hilbert transformed
    signal_intD = abs(diff(signal_int2));
    signal_intH = abs((hilbert(signal_int2)));
    signal_intH = signal_intH(2:end);
    time_in     = time_in(2:end);

    %multiplying the two intermediate signals
    signal_int3 = signal_intD.*signal_intH;
    signal_int4 = lowpass(signal_int3, 0.08*IF, Fs, 'ImpulseResponse','iir','Steepness',0.95);

    %delete outer parts of data
    [peak, index_main] = findpeaks(-signal_int4); %negative peaks

    if(peak(1) > peak(2)*1.01)  %take care of weird effects at first edge
        start = index_main(2)+1;
        finish = index_main(end)-1;
    else
        start = index_main(2)+1;
        finish = index_main(end)-1;
    end

    signal_out = signal_int4(start:finish);
    time_out = time_in(start:finish);
end
```

I.5 The function 'fitter_fnc'

This function function used to fit a sawtooth on the demodulated signal. The function first selects a full period of the input signal. After that, the function splits the full period into a rising and a falling part. The falling and rising parts are then extended with additional points until the total period equals the period as specified by the user, and until the ratio between the rise time and the

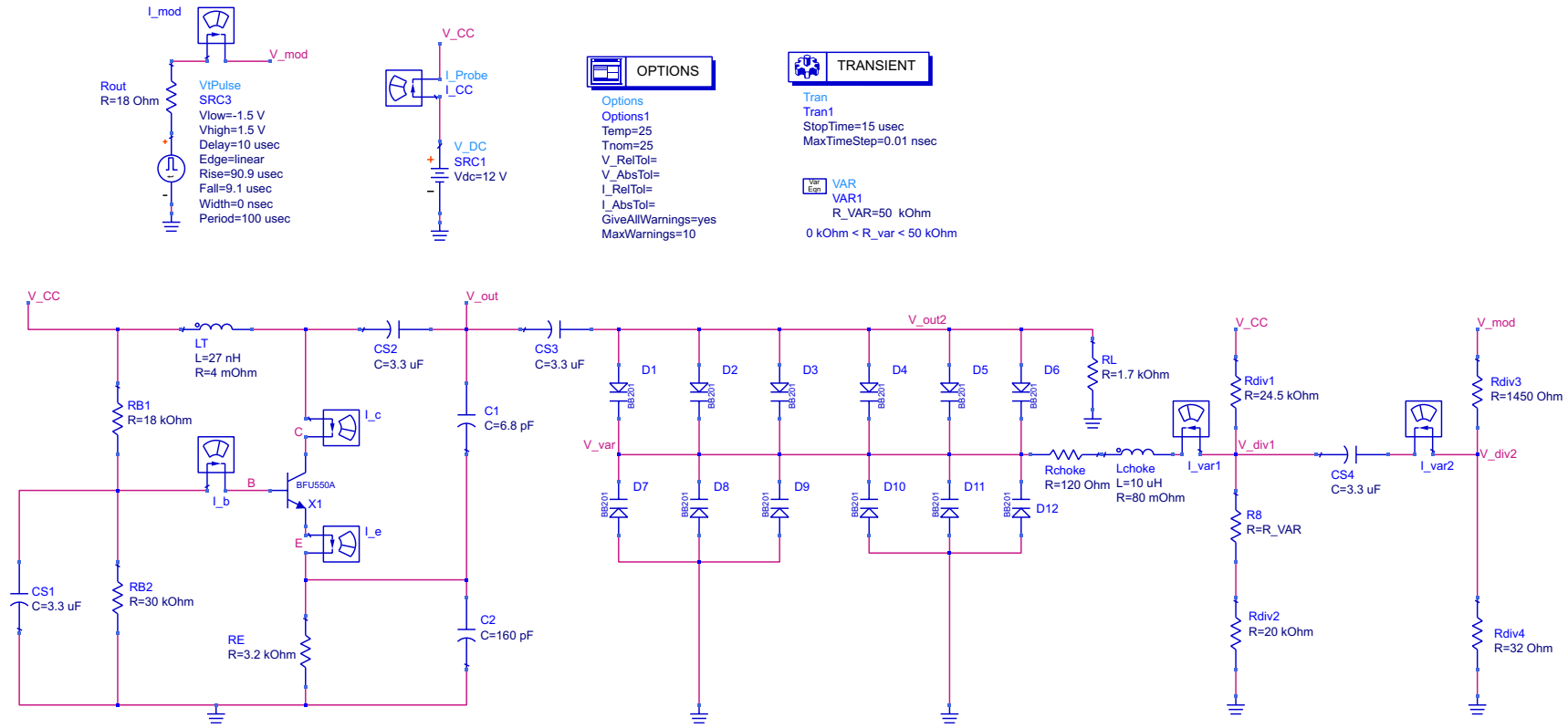
fall time equals the value as specified from the duty cycle. After that, a first-order polynomial is fitted to the rising part. By using the first-order polynomial estimate of the rising signal, the duty cycle and the time at which the peak of the sawtooth occurs, the falling signal is determined.

The code which is used to do the sawtooth fitting is given below.

```
%Author:    Dimme
%Date:      12/06/2020
%Descr:     This function fits a first order polynomial to the rising part
%           part of the sawtooth baseband input signal. Using this polynomial and
%           the duty cycle, the second fit is estimated.
%           "[rise, fall, fit, data] = fitter_fnc(signal_in, time_in, Fs, period, T)"
%           With 'period' the duty cycle, T the lenght of a full period, Fs
%           the sample rate (could just as easily be constructed from the
%           time axis), 'time_in' the time axis and 'signal_in'
%           the corresponding amplitude levels.

function [rise, fall, fit, data] = fitter_fnc(signal_in, time_in, Fs, period, T)
    if nargin < 5
        T = 100;    %note: this is the actual 'period'
    end
    if nargin < 4
       period = 0;  %note: this is the duty cycle
    end

    T_bin    = 1/Fs * 10^6;

    %find maxima
    N = length(signal_in);
    [~, index_m(1)] = min(signal_in(floor(0.3*N):floor(0.7*N)));
    [~, index_m(2)] = min(signal_in(floor(0.7*N):end));
    index_m(1) = index_m(1) + floor(0.3*N) - 1;
    index_m(2) = index_m(2) + floor(0.7*N) - 1;

    %first select only one period of signal
    signal = signal_in(index_m(1):index_m(2));
    time   = time_in(index_m(1):index_m(2));

    %split signal in rising and falling part.
    [~, index] = max(signal);
    rise_sig = signal(1:index(1));
    rise_tim = time(1:index(1));
    fall_sig= signal(index(1):end);
    fall_tim= time(index(1):end);

    if period ~= 0
        if(rise_tim(end) - rise_tim(1) < period*T)
            diff = period*T - (rise_tim(end) - rise_tim(1));
            r_off= -floor(diff/T_bin);
        elseif(rise_tim(end) - rise_tim(1) > period*T)
            diff = (rise_tim(end) - rise_tim(1)) - period*T;
            r_off= floor(diff/T_bin);
        else
            r_off = 0;
```

```
49          end
50
51          if(fall_tim(end) - fall_tim(1) < (1 - period)*T)
52              diff = (1-period)*T - (rise_tim(end) - rise_tim(1));
53              f_off= -floor(diff/T_bin);
54          elseif(fall_tim(end) - fall_tim(1) > (1-period)*T)
55              diff  = (fall_tim(end) - fall_tim(1)) - (1-period)*T;
56              f_off = floor(diff/T_bin);
57          else
58              f_off = 0;
59          end
60      else
61          f_off = 0;
62          r_off = 0;
63      end
64
65      %first select only one period of signal
66      signal = signal_in(index_m(1)+r_off:index_m(2)-f_off);
67      time   = time_in(index_m(1)+r_off:index_m(2)-f_off);
68
69      %split signal in rising and falling part.
70      [~, index] = max(signal);
71      rise_sig = signal(1:index(1));
72      rise_tim = time(1:index(1));
73      fall_sig= signal(index(1)+1:end);
74      fall_tim= time(index(1)+1:end);
75
76      %store the selected data
77      data = zeros(2, length(signal));
78      data(1,:) = time;
79      data(2,:) = signal;
80
81      %vectors containing the fitted data
82      rise = zeros(2,length(rise_sig));
83      fall = zeros(2,length(fall_sig));
84      rise(1,:) = rise_tim;
85      rise(2,:) = rise_sig;
86      fall(1,:) = fall_tim;
87      fall(2,:) = fall_sig;
88
89      %fit curve for rising and falling
90      fit = zeros(2,2);
91      fit(1,:) = polyfit(rise_tim, rise_sig, 1); %fit for rise
92
93      if period ~= 0
94          fit(2,1) = fit(1,1)*period / (period - 1);
95          fit(2,2) = fit(1,2) - time(index(1))*fit(1,1) / (period - 1);
96      else
97          fit(2,:) = polyfit(fall_tim, fall_sig, 1);
98      end
99  end
```

J. ADS Circuit Modulator

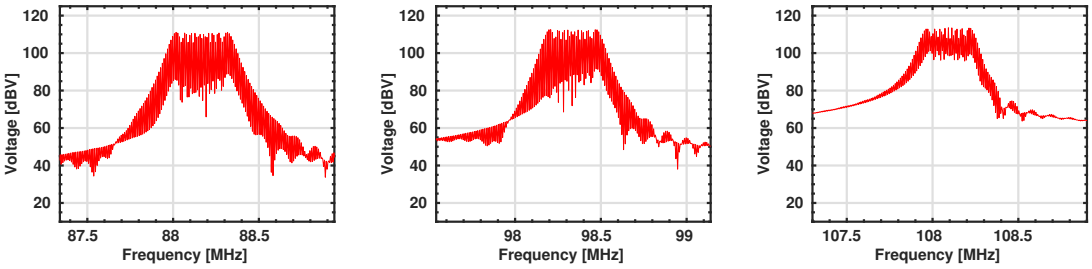


K. Additional Figures Modulator

In this appendix, some additional figures with information concerning the modulator are given.

K.1 The frequency spectra

Part of the frequency spectra of the chirp signal simulated over two full sawtooth waves for a tuning resistance of 0 kΩ, 15.6 kΩ and 50 kΩ are given in Figure K.1.

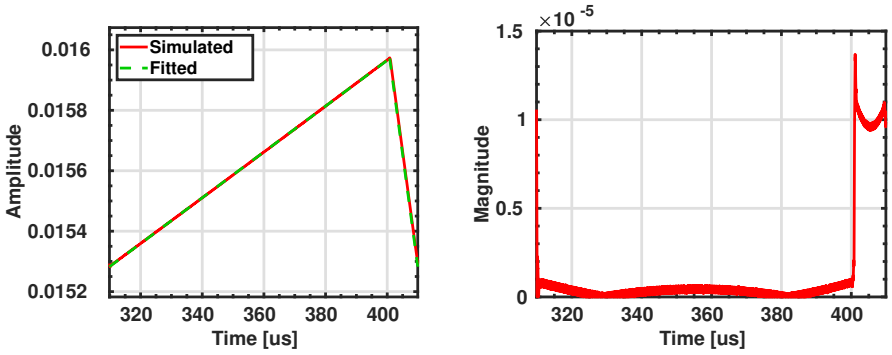


(a) 0 kΩ (b) 15.6 kΩ (c) 50 kΩ

Figure K.1: The frequency spectra around the center frequency corresponding to a 0 kΩ, 15.6 kΩ and 50 kΩ tuning resistance calculated over two full sawtooth periods.

K.2 The linearity results

The linearity results for a tuning resistance of 15.6 kΩ are given in Figure K.2.

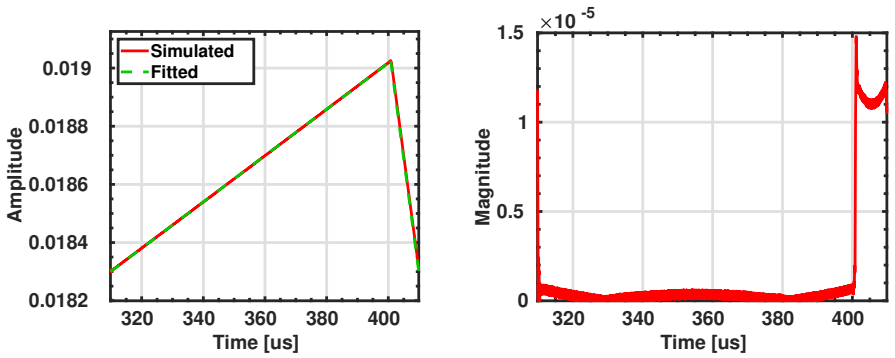


(a) The normalised simulated waveform and the fitted sawtooth

(b) The error between the measured and fitted waveform

Figure K.2: The resulting waveform for the linearity measurement and the corresponding error. The used tuning resistance is 15.6 kΩ.

The linearity results for a tuning resistance of 50 kΩ are given in Figure K.3.



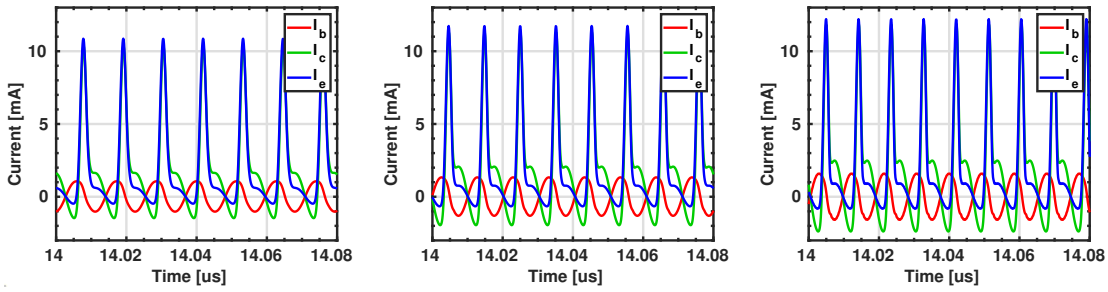
(a) The normalised simulated waveform and the fitted sawtooth

(b) The error between the measured and fitted waveform

Figure K.3: The resulting waveform for the linearity measurement and the corresponding error. The used tuning resistance is 50 kΩ.

K.3 Transistor Currents

The transistor base current, collector current and emitter current during steady state oscillation are given for the three different tuning resistances in Figure K.4. Note that, as expected for the Colpitts oscillator, the collector and emitter currents form large spikes. The spikes stay below the collector current limit of 50 mA of the used BFU550A transistor.



(a) 0 kΩ

(b) 15.6 kΩ

(c) 50 kΩ

Figure K.4: The collector, emitter and base current through the BFU550A transistor during steady state oscillation. The results are given for the three different tuning resistances of 0 kΩ, 15.6 kΩ and 50 kΩ.

K.4 Base-Emitter Voltage

The voltage swing over the transistor is chosen such that the collector-emitter voltage V_{BE} does not become larger than 10.5 V, this in order to avoid reaching the breakdown region. Using Figure K.5 it can be verified that the 10.5 V is not exceeded.

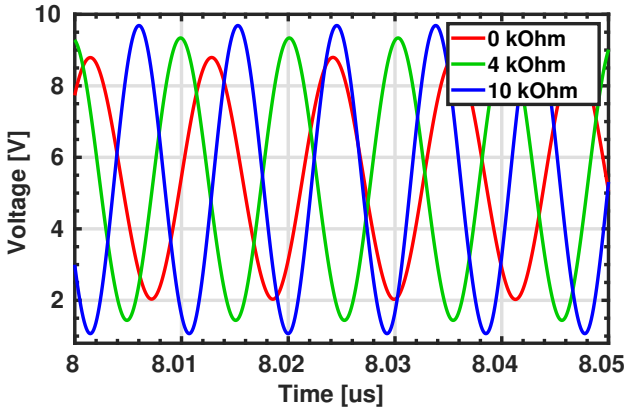
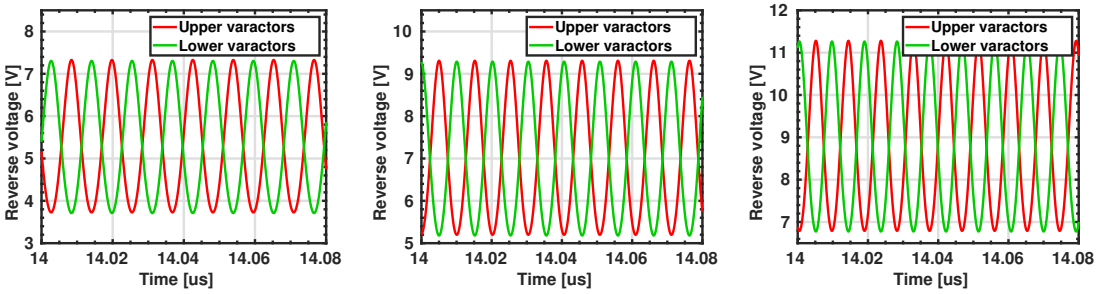


Figure K.5: The collector-emitter voltage during steady state oscillations for the three different tuning resistances.

K.5 Reverse Voltages

In Figure K.6 the reverse voltage over the varactors is shown. Note that, due to the used topology, the reverse voltage over the upper set of varactors increases when that over the lower set decreases and vice versa. The effect of this is that the change in capacitances due to the RF signal becomes less.

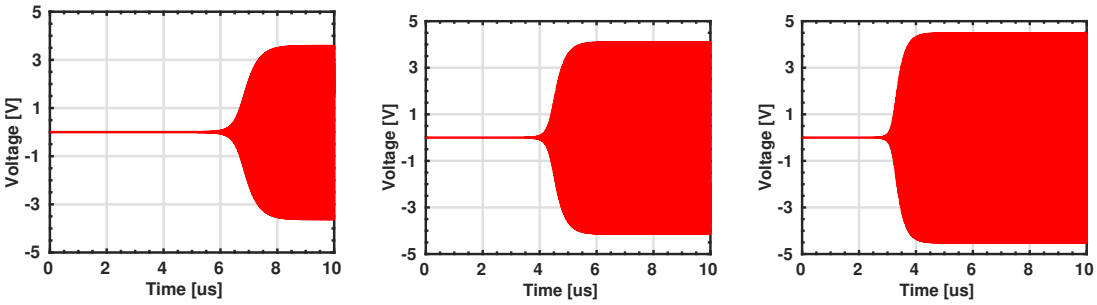


(a) 0 kΩ (b) 15.6 kΩ (c) 50 kΩ

Figure K.6: The reverse voltage over the lower and upper set of varactors in the modulator during steady-state oscillations for the three different tuning resistances.

K.6 Startup waveform

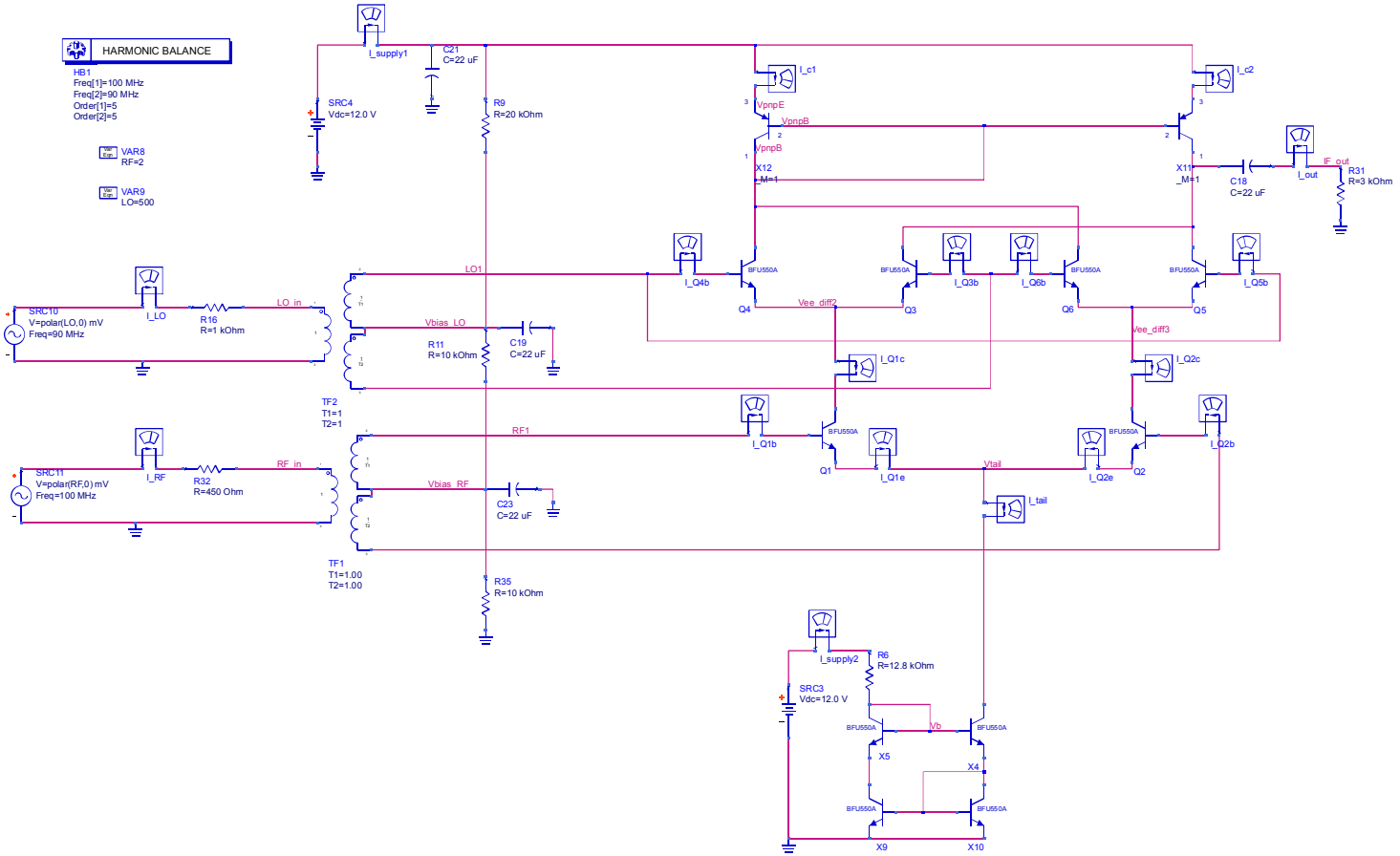
The waveforms during startup for the three different tuning resistances are given in Figure K.7. Note that the time it takes for the oscillation to start differs between the three different tuning resistances.



(a) 0 kΩ (b) 15.6 kΩ (c) 50 kΩ

Figure K.7: The waveforms during start up of the oscillation for the three different tuning resistances.

L. ADS Circuit Mixer



M. Fourier Expansion Square wave

In this appendix we derive the fourier expansion of a square wave with amplitude 1 and frequency ω_{LO}. A plot of this square wave is shown in Figure M.1.

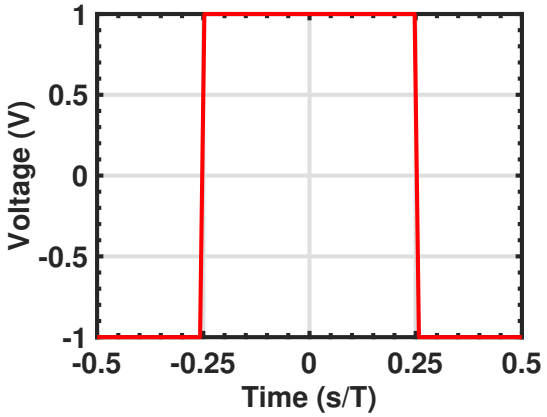


Figure M.1: Square wave for a normalized period T = $\frac{2\pi}{\omega_{LO}}$.

This square wave can be written as in Equation (M.1).

$$f(x) = \begin{cases} -1, & -\frac{1}{2}T \leq x < -\frac{1}{4}T \\ 1, & -\frac{1}{4}T \leq x < \frac{1}{4}T \\ -1, & \frac{1}{4}T \leq x < \frac{1}{2}T \end{cases} \tag{M.1}$$

This function can be decomposed into its fourier series using the Euler-Fourier formulas. The standard fourier series is given in Equation (M.2).

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{n\pi x}{L}\right) + b_n \sin\left(\frac{n\pi x}{L}\right) \right]. \tag{M.2}$$

With a_n and b_n given by:

$$a_n = \frac{1}{L} \int_{-L}^{L} f(x) \cos\left(\frac{n\pi x}{L}\right) dx.\ n \in \mathbb{N} \tag{M.3}$$

$$b_n = \frac{1}{L} \int_{-L}^{L} f(x) \sin\left(\frac{n\pi x}{L}\right) dx.\ n \in \mathbb{N} \setminus \{0\} \tag{M.4}$$

In this case L = $\frac{1}{2}$T and T = $\frac{2\pi}{\omega_{LO}}$. We also note that since $f(x) = f(-x)$ the function is even, thus $b_n = 0$ for all n. We can then calculate a_n in the following way:

$$a_n = \frac{2}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} f(x) \cos\left(\frac{2n\pi x}{T}\right). \tag{M.5}$$

In the special case when n= 0, $a_0 = 0$. In order to calculate the remaining values of a_n Equation (M.5) can be split into three integrals as shown in Equation (M.6):

$$a_n = \frac{2}{T} \left[-\int_{-\frac{T}{2}}^{-\frac{T}{4}} \cos\left(\frac{2n\pi x}{T}\right) dx + \int_{-\frac{T}{4}}^{\frac{T}{4}} \cos\left(\frac{2n\pi x}{T}\right) dx - \int_{\frac{T}{4}}^{\frac{T}{2}} \cos\left(\frac{2n\pi x}{T}\right) dx \right]. \tag{M.6}$$

By exploiting symmetry this expression can be simplified into Equation (M.7).

$$a_n = \frac{2}{T}\left[4\int_0^{\frac{T}{4}} \cos\left(\frac{2n\pi x}{T}\right)\right]. \tag{M.7}$$

This evaluates to:

$$a_n = \frac{4}{n\pi}\sin\left(\frac{2n\pi x}{T}\right)\Bigg|_0^{\frac{T}{4}}. \tag{M.8}$$

Which results in:

$$a_n = \frac{4}{n\pi}\sin\left(\frac{1}{2}n\pi\right) = \frac{4}{n\pi}. \tag{M.9}$$

Where n is an odd number and not zero. Therefore Equation (M.1) can be written as:

$$f(x) = \frac{4}{\pi}\sum_{n=1,3,5,..}^{\infty}\frac{1}{n}\cos\left(\frac{2n\pi x}{T}\right) = \frac{4}{\pi}\sum_{n=1,3,5,..}^{\infty}\frac{1}{n}\cos\left(n\omega_{LO}x\right). \tag{M.10}$$

N. Simulations in MATLAB for mixer

N.1 Linearization of tanh

In the following figure the behaviour of the tanh function and its linearization can be seen.

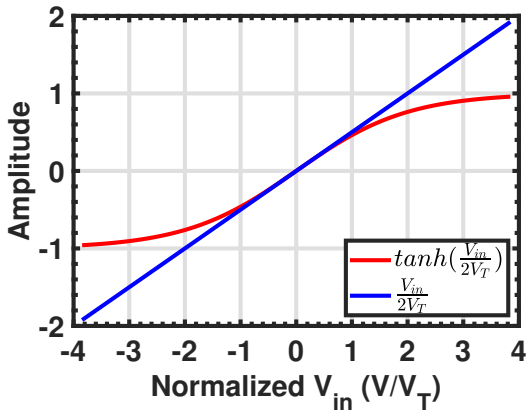


Figure N.1: Output of the tanh($\frac{V}{2V_T}$) function and its linearization for small input signals.

The MATLAB code for generating this plot is given below.

```
%Author: Roderick Tapia
%Date: 16/06/2020
%Description: Plot of output of tanh(x/2VT) for a small input and
%linearization of tanh
%             Plot layout by Dr. S.M. Alavi, TU Delft.

close all;
clear all;

Vt = 26; %Vt in mV

x = linspace(-100,100); %input in mV
y = tanh(x/(2*Vt));
f = x/(2*Vt);

figure;
plot(x/Vt,y,'color',[1,0,0],'LineWidth',3);hold on
plot(x/Vt,f,'color',[0,0,1],'LineWidth',3);hold on

xlabel('Normalized V_{in} (V/V_T)'),ylabel('Amplitude');
grid;
%ylim([-1.0 1.0]);
legend('$tanh(\frac{V_{in}}{2V_T})$ ','$\frac{V_{in}}{2V_T}$','Location','SE','Interpreter','latex');
set(gca,'FontSize',21,'FontName','Arial','FontWeight',...
    'bold','LineWidth',3,...
    'XMinorTick','On','XTick',[-4 -3 -2 -1  0  1 2 3 4],...
    'YMinorTick','On','YTick',...
    [-2 -1 0 1 2]);%figure setting
```

```
Hfig9 = findall(gcf,'type','text');
set(Hfig9,'FontSize',21,'FontName','Arial');
```

N.2 Tanh(x) output for LO input

The output of the tanh(x) function for a cosinus input can be seen in Figure N.2.

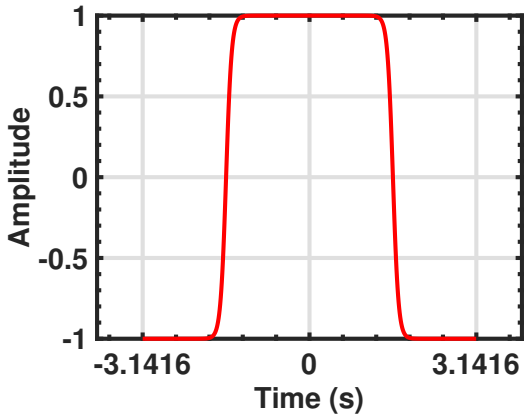
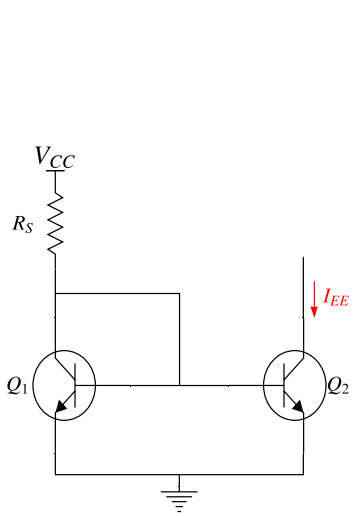


Figure N.2: Output of the tanh(x) function for a cosinus input with period of 2π and amplitude of 500 mV.

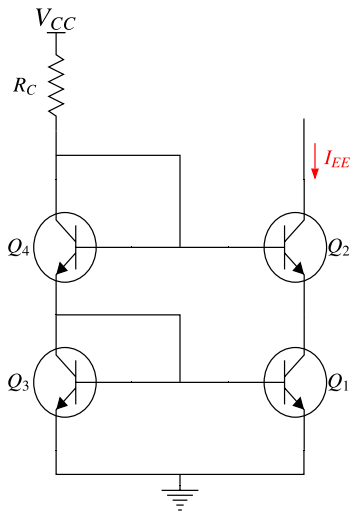
The MATLAB code for generating this plot is given below.

```
%Author: Roderick Tapia
%Date: 16/06/2020
%Description: Plot of output of tanh(x) for a cosinus input
%             Plot layout by Dr. S.M. Alavi, TU Delft
close all;
clear all;

x = (-pi:0.01:pi);
LO=9.6*cos(x); %LO input 500*10^-3 -> 500/(2*26) = 9.6
y = tanh(LO);

figure;
plot(x,y,'color',[1,0,0],'LineWidth',3);hold on
xlabel('Time (s)'),ylabel('Amplitude');
grid;
set(gca,'FontSize',21,'FontName','Arial','FontWeight',...
    'bold','LineWidth',3,...
    'XMinorTick','On','XTick',[-pi 0 pi],...
    'YMinorTick','On','YTick',...
    [ -1 -1/2 0 0.5 1 ]);%figure setting
Hfig9 = findall(gcf,'type','text');
set(Hfig9,'FontSize',21,'FontName','Arial');
```

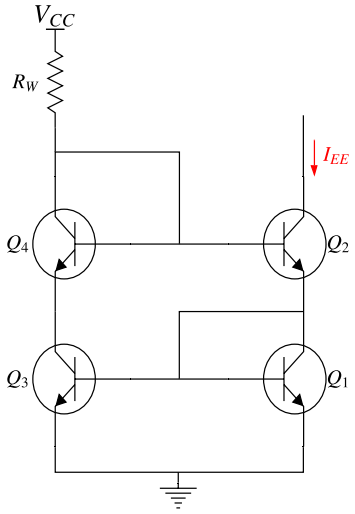
O. Current mirror topologies



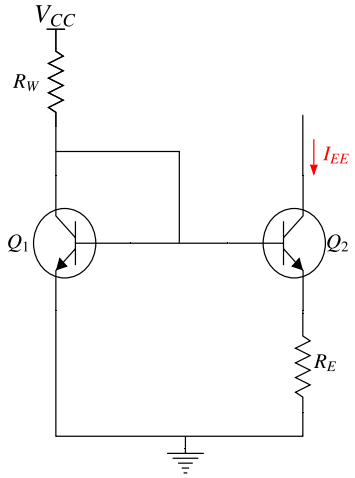
(a) Simple current mirror.



(b) Cascode current mirror.



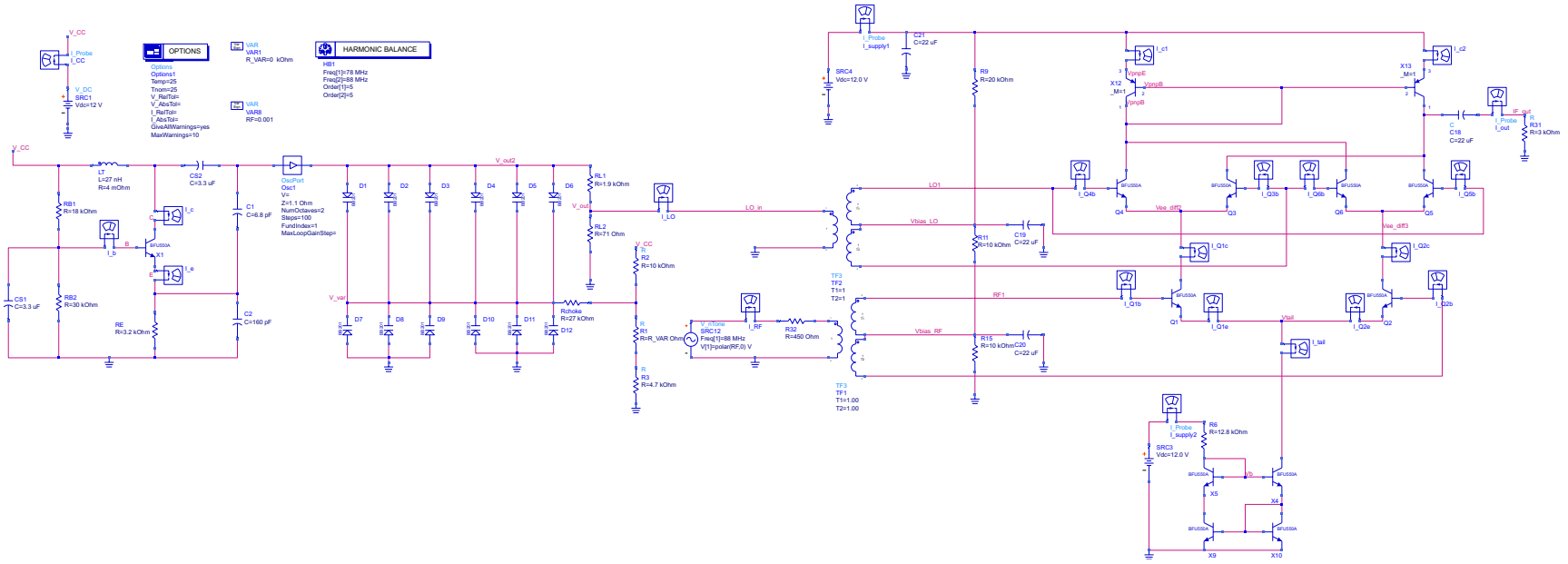
(c) Wilson current mirror.



(d) Widlar current mirror.

Figure O.1: Different current mirror topologies.

P. ADS Circuit Mixer and LO

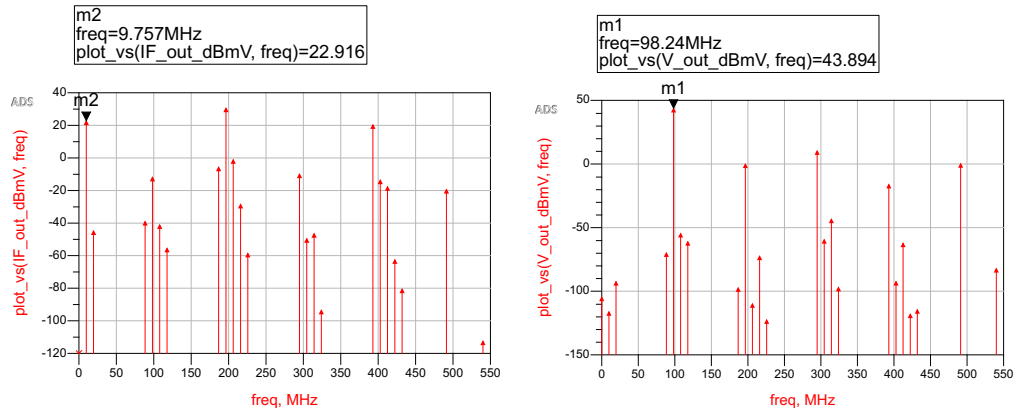


Q. Additional results LO and Mixer

In this Appendix we provide additional results for the combination of the designed LO and the designed mixer. These simulations have been performed using the circuit show in Appendix P.

Q.1 HB simulations

The simulations result for an RF input at 108 MHz and an LO input of 98 MHz is given in Figure Q.1a. The input spectrum of the LO signal is given in Figure Q.1b and the input spectrum of the RF signal is given in Figure Q.2.



(a) Frequency spectrum of signal at the output of the mixer.

(b) Frequency spectrum of signal at the LO input.

Figure Q.1: Simulation for the mixer including LO input at 98 MHz and RF input at 108 MHz.

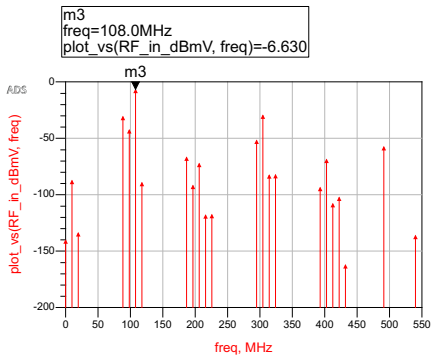
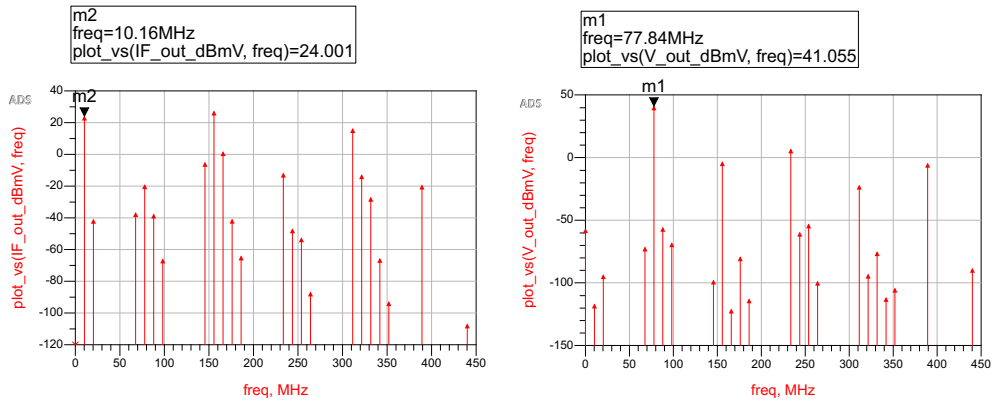


Figure Q.2: Frequency spectrum of signal at the RF input of the mixer

The conversion gain in this case is $22.916 + 6.630 = 29.546$ dB. Simulations have also been performed using a RF input at 88 MHz and an LO input of 78 MHz. The result for this simulation is shown in Figure Q.3a, the spectrum of the LO input is shown in Figure Q.3b and the spectrum of the RF input is shown in Figure Q.4.

From Figure Q.3a, together with Figure Q.4, it can be seen that the gain is $24.001 + 5.296 = 29.297$ dB. This is slightly lower than calculated but can be tuned using the resistor in the wilson current source. The frequency peak in the output occurs at exactly the difference frequency between the two inputs in both simulations.



(a) Frequency spectrum of signal at the output of the mixer.

(b) Frequency spectrum of signal at the LO input.

Figure Q.3: Simulation for the mixer including LO input at 78 MHz and RF input at 88 MHz.

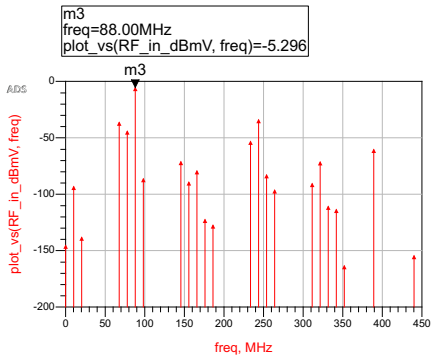
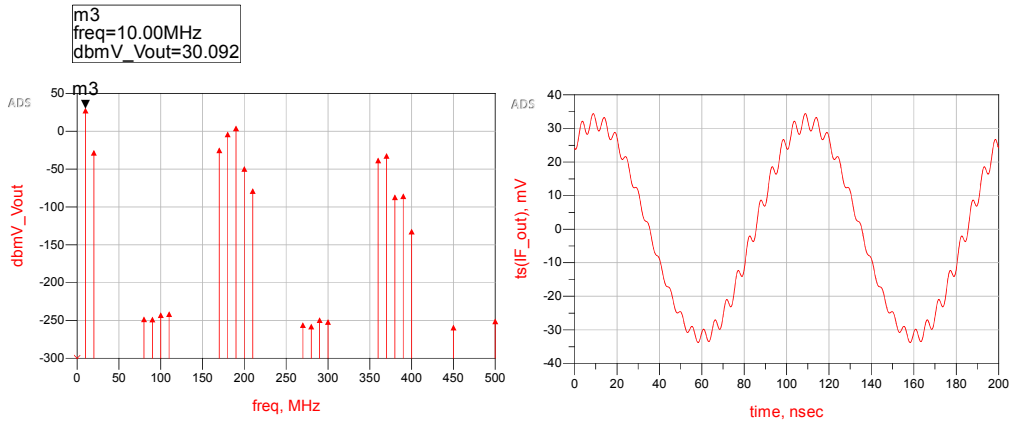


Figure Q.4: Freqeuncy spectrum at the RF input of the mixer.

R. Modified Mixer

In this appendix a modified mixer is presented in which the LO signal is tuned to a low amplitude which allows the Gilbert Cell to operate as a linear multiplier. In order to achieve the desired gain the current output of the Wilson current source can be tuned. The circuit for this modified Gilbert Cell can be found in Appendix S. The resulting output spectrum can be seen in Figure R.1a and the time domain signal is shown in Figure R.1b. It can be seen that the suppression of the component at 20 MHz is better in this modified circuit. The main reason for this is not fully understood yet and more research is needed. However, considering the time constraint within the project, this modified circuit could prove useful in case the 20 MHz component presents a problem.

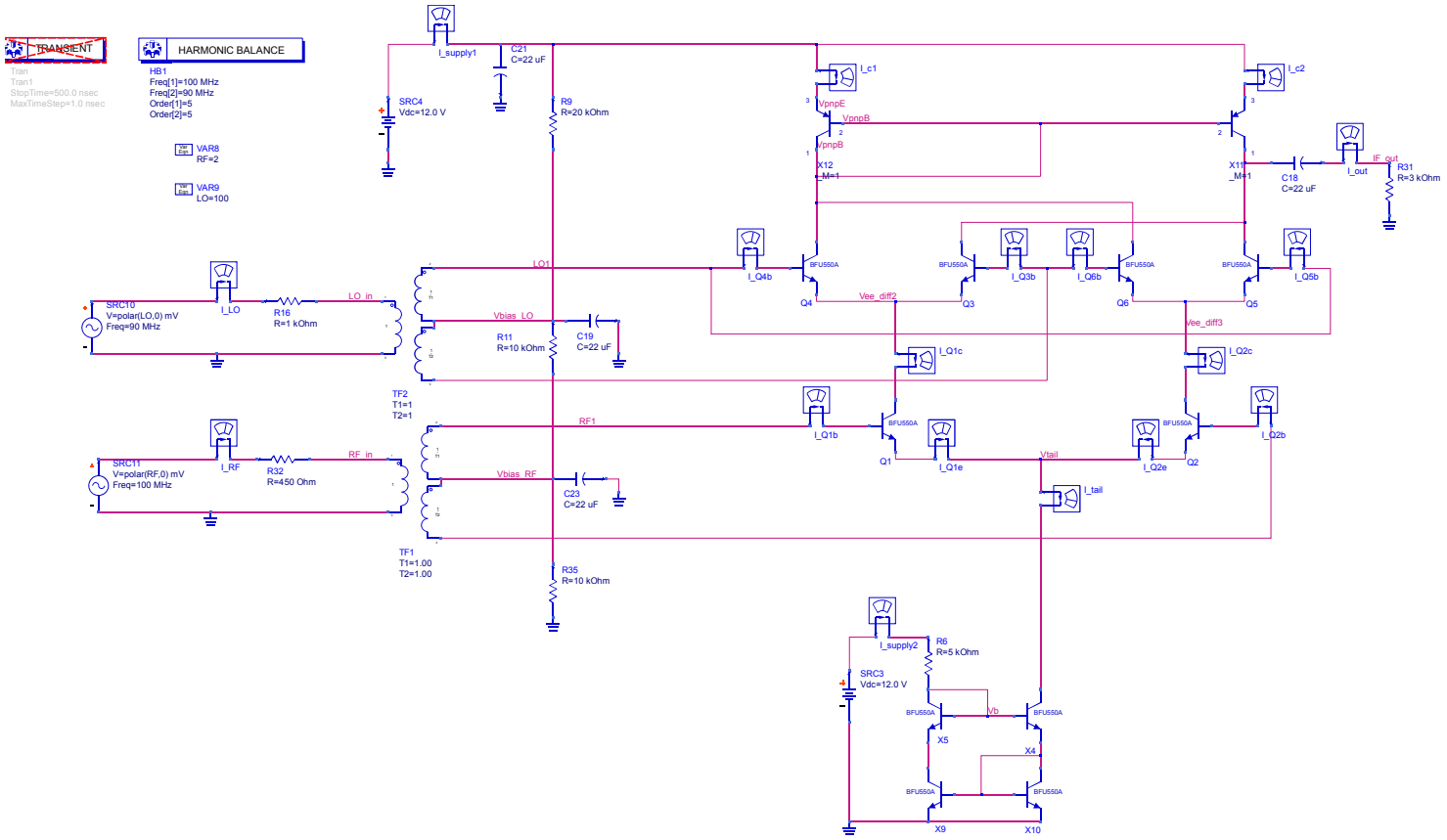


(a) Frequency spectrum of signal at output of the mixer.

(b) Time domain signal at output of the mixer.

Figure R.1: Simulation for the modified Gilbert Cell for better spurs suppression.

S. ADS circuit mixer in linear multiplier mode



T. The Local Oscillator

In this appendix, the programme of requirements, design and results of the LO are described briefly.

T.1 Requirements

The requirements for the LO are given in Table T.1.

Table T.1: The requirements for the local oscillator.

Tag	Description	Value
LO0	Tuning range	78 MHz to 98 MHz
LO1	Peak-to-peak output voltage swing	Between 200 mV and 400 mV
LO2	Power consumption	< 100 mW
LO3	Higher order harmonics	< -30 dB
LO4	Output resistance	$< 150\ \Omega$
LO5	Phase noise (at 10 kHz)	$<$ -30 dBc/Hz

The requirements for the LO are largely similar to those of the modulator. The tuning frequency range is chosen such that it allows for mixing to the IF. For the corresponding requirement, LO0, a 0.5% margin is used. Furthermore the output resistance is chosen such that it is not significantly influenced by the input resistance of the mixer.

T.2 Design of the Local Oscillator

In this chapter, the design of the LO is presented. Since it is very similar to the design of the modulator (see Chapter 6), most design steps are not explained explicitly. The tank inductor L_T is kept at 27 nH, also C_1 and C_2 are kept at their values of respectively 6.8 pF and 160 pF. Furthermore the bias network is not changed. The load R_L is increased to 2 kΩ, and the inductor L_{choke} is removed in favour of a large value of R_{choke}. Plotting the estimated frequency as a function of the number of varactors yields Figure T.1.

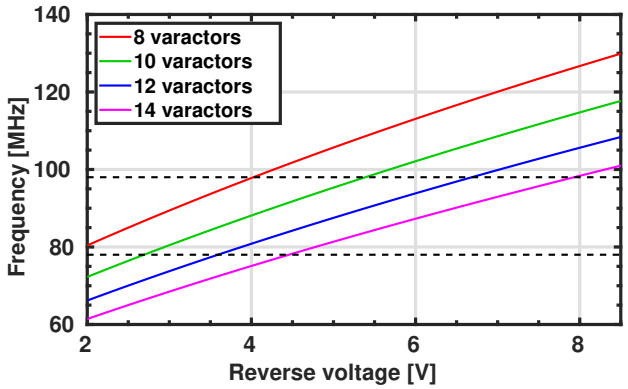


Figure T.1: The oscillation frequency versus the applied reverse voltage when 8, 10, 12 or 14 BB201 varactors are used. The dashed lines indicate the upper and lower carrier frequency.

It is chosen to use twelve varactors. Simulating the frequency as a function of the voltage applied to the varactor diodes terminals yields Figure T.2.

From the figure it is estimated that the required tuning voltage at the divider ranges from 3.8 V to 7.1 V. This is implemented using a voltage divider with a variable resistor ranging from 0 kΩ

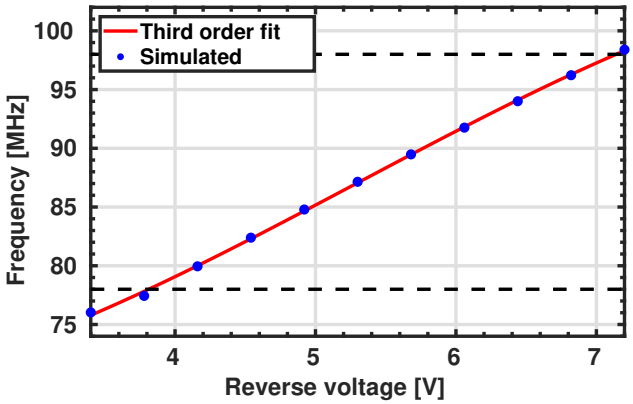


Figure T.2: The simulated resonance frequency as a function of the reverse voltage.

to 10 kΩ in series with a 4.7 kΩ resistor. The upper resistor of the voltage divider is 10 kΩ. The resulting frequency plot for different tuning resistances is given in T.3. In the figure, it can be seen that the lowest frequency equals 77.8 MHz and the maximum frequency equals 98.1 MHz. This is slightly larger than the band specified by LO0, however falls inside the 0.5% margin.

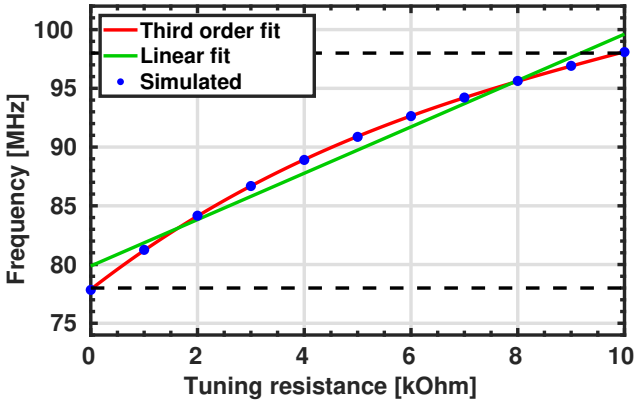


Figure T.3: The simulated resonance frequency as a function of the tuning resistance.

The used value for R_{choke} is 27 kΩ, it is large because it serves as an RF choke. The choke inductance is removed. This is possible because in the LO, f_{osc} does not need to follow a change in the applied varactor bias quickly.

Only one more design step needs to be done, which is converting the output to a signal which adheres to LO1. This is simply achieved by replacing the load resistor with a voltage divider. An ADS simulation reveals that at 78 MHz the voltage swing over R_L is 6.7 V, while at 98 MHz the swing is 9.3 V.

By now setting the output voltage at 98 MHz equal to 250 mV, while keeping total load at about 2 kΩ, we find that the lower divider resistance ($R_{L,2}$ in Figure T.4) must be approximately 71 Ω. For $R_{L,1}$, a resistor of 1900 Ω is used. For 98 MHz this results in a swing of 335 mV at the output, which is inside the range specified by LO1. The used values also allow for some margin which is needed when the circuit is loaded by the mixer, this is verified in Section T.3.3.

The design of the local oscillator is now finalised, the full symbolic circuit is given in Figure T.4.

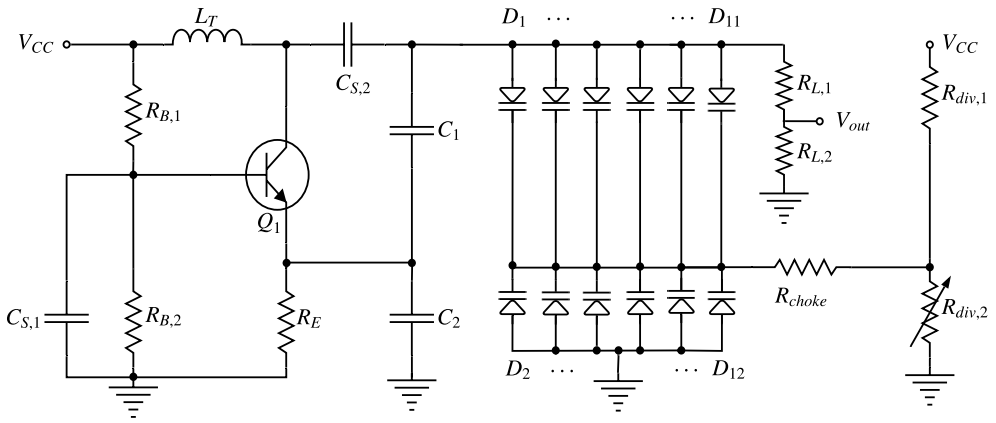


Figure T.4: The complete symbolic circuit of the local oscillator.

In Table T.2 the values for each of the components are given. The corresponding bill of materials is found in Appendix V.2.

Table T.2: The values of the components used in the local oscillator.

Component	Value	Component	Value	Component	Value
$C_{s,1}$, $C_{s,2}$	3.3 μF	$R_{B,1}$	18 kΩ	$R_{div,1}$	10 kΩ
D_1 ... D_{12}	BB201	$R_{B,2}$	30 kΩ	$R_{div,2}$	4.7 kΩ to 14.7 kΩ
L_T	27 nH	R_E	3.2 kΩ	$R_{L,1}$	1900 Ω
Q_1	BFU550A	R_{choke}	27 kΩ	$R_{L,2}$	71 Ω
C_1	6.8 pF	C_2	160 pF		

T.3 Results Local Oscillator

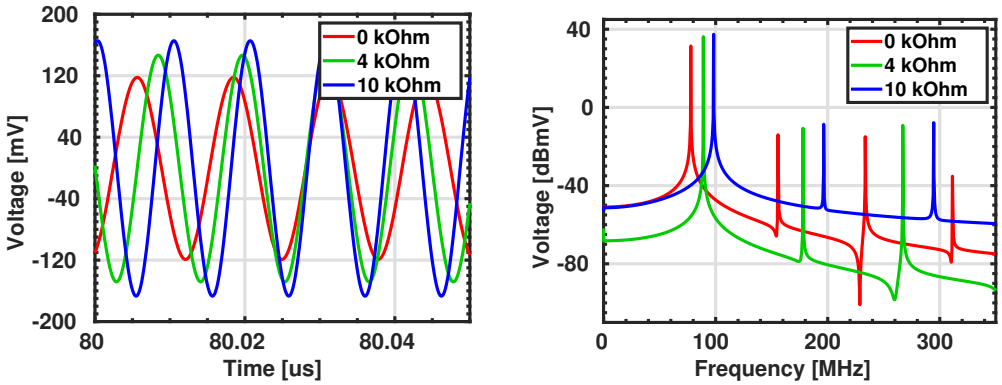
In this section, the result of the LO are presented. As was the case for the design, the approach is similar to that of the modulator. The used ADS circuit is given in Appendix U. The used tuning resistances are 0 kΩ, 4 kΩ and 10 kΩ, which roughly corresponds to an oscillation frequency of 78 MHz, 88 MHz and 98 MHz.

T.3.1 Voltage swing, power consumption and harmonics

The first requirements that are verified are LO1, LO2 and LO3. The result of the corresponding ADS simulation is shown in Figure T.5.

From the figures, the resonance frequency, peak-to-peak voltage and level of the second order harmonic with respect to the first can be determined. The results are summarised in Table T.3. The power is taken directly from the result calculated by ADS.

From the results it is found that LO1, LO2 and LO3 are satisfied.



(a) Time domain (b) Frequency domain

Figure T.5: The simulation results in the (a) time and (b) frequency domain for the three different tuning resistances.

Table T.3: The results for the oscillation frequency, output voltage, power consumption and second harmonic.

Resistance [kΩ]	**f_{osc} MHz**	**V_{pp} [mV]**	**Power Consumption [mW]**	**2nd harmonic [dB]**
0	77.8	237	37	-45.4
4	89.0	295	35	-46.9
10	98.2	332	33	-46.0

T.3.2 Phase noise

The last requirement that needs to be verified is LO5. A plot of the simulated phase noise is given in Figure T.6. As is the case for the modulator, the spectra of the three different resistances follow a similar curve.

The phase noise values at 10 kHz are given in Table T.3. As can be seen, requirement LO5 is satisfied.

Table T.4: The phase noise at 10 kHz for the three different tuning resistances.

Resistance [kΩ]	**Phase noise at 10 kHz [dBc/Hz]**
0	-103.0
4	-102.2
10	-103.2

T.3.3 Output voltage when loaded by the mixer

Since the previously shown results for the output voltage are without the additional loading of the mixer, it needs to be verified that requirement LO1 still holds when the local oscillator is loaded with the mixer. The output voltage under this condition is shown in Figure T.7. The used simulation type is a HB simulation.

From the figure, it is found that the output voltages for a tuning resistance of 0 kΩ, 4 kΩ and 10 kΩ are respectively given by 229 mV, 284 mV and 318 mV. As expected, this is less than without the mixer, but requirement LO1 still holds.

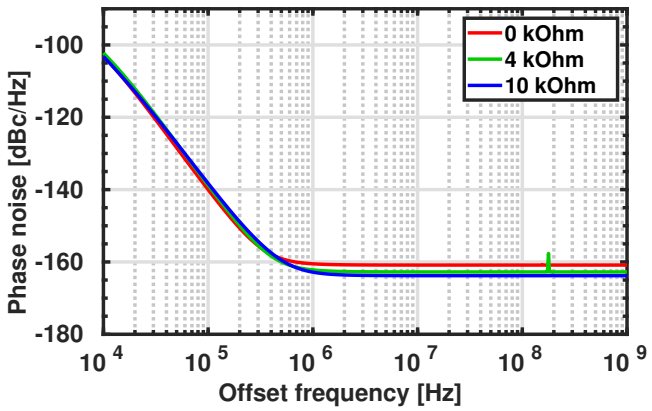


Figure T.6: The phase noise spectrum for the three different tuning resistance values with a frequency offset from 10 kHz to 1 GHz.

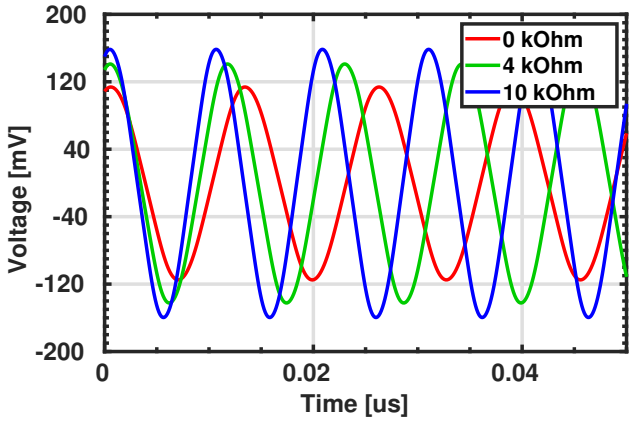
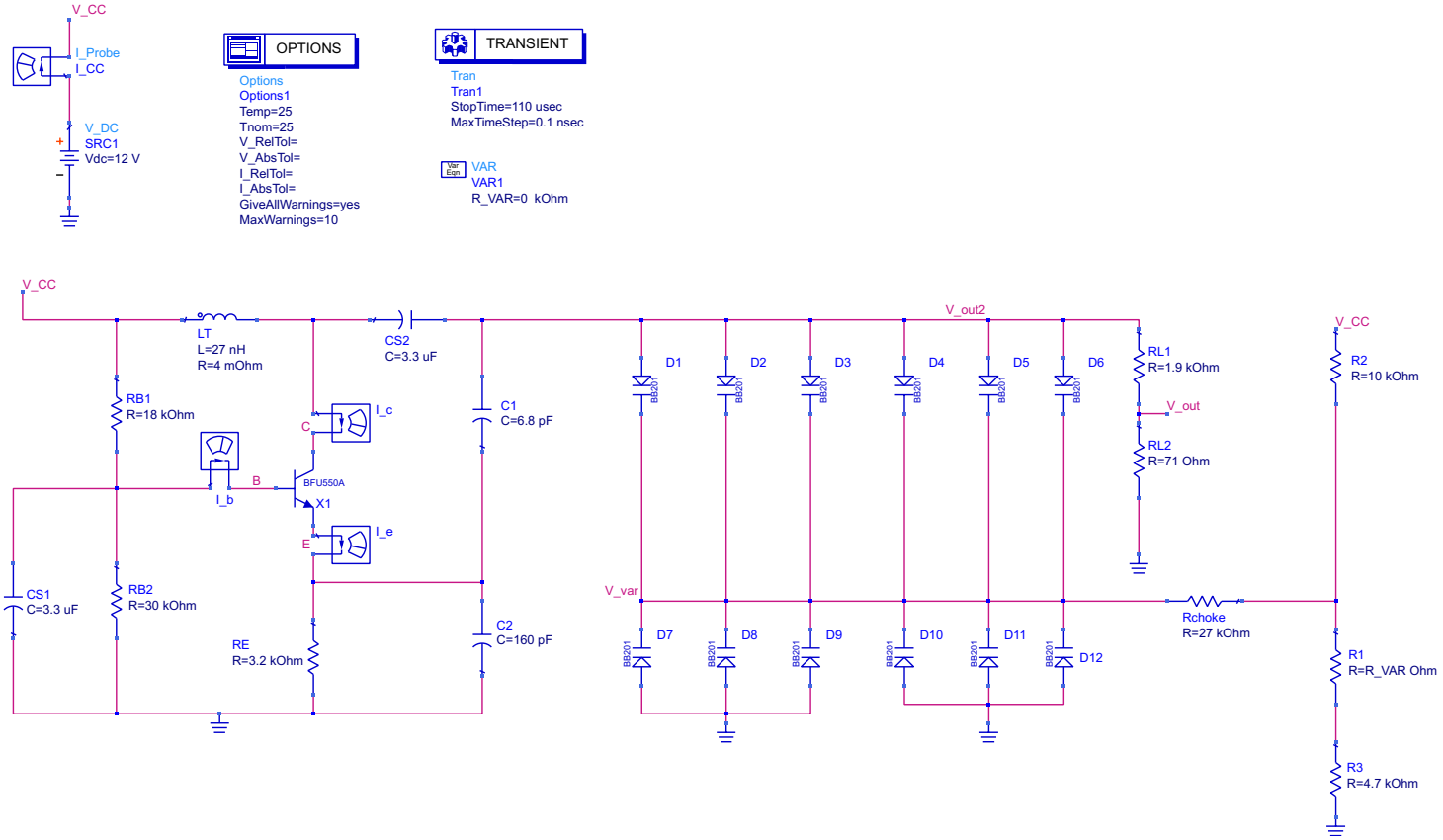


Figure T.7: The simulation results in the time domain for the output voltage of the local oscillator with three different tuning resistances. The local oscillator is loaded by the mixer.

U. ADS Circuit Local Oscillator



V. Bill of Materials

In this appendix, the bill of materials for the three different modules is given. Note that sometimes multiple components are placed in series. This is indicated by square brackets after the component name.

V.1 Modulator

Table V.1: A list of the values of the components used in the modulator and the actual parts corresponding to them. Note that, to form some resistor values, a series combination of multiple resistors is used.

Component	**Value**	**Part Number**	**Tolerance**	**Price [€]**	**amount**
$R_{B,1}$	18 kΩ	CRCW120618K0FKEAC	1%	0.09	1
$R_{B,2}$	30 kΩ	CR1206-FX-3002ELF	1%	0.09	1
R_E [1]	3 kΩ	ERJ-8ENF3001V	1%	0.09	1
R_E [2]	200 Ω	RC1206FR-07200RL	1%	0.09	1
$R_{div,1}$ [1]	24 kΩ	CRCW120624K0FKEA	1%	0.09	1
$R_{div,1}$ [2]	499 Ω	CRCW1206499RFKEAC	1%	0.09	1
$R_{div,2}$ [1]	0 to 50 kΩ	PT6KV-503A1010	10%	0.58	1
$R_{div,2}$ [2]	20 kω	ERJ-S08F2002V	1%	0.48	1
$R_{div,3}$ [1]	1300 Ω	CRCW12061K30FKEAC	1%	0.09	1
$R_{div,3}$ [2]	150 Ω	ERJ-8ENF1500V	1%	0.10	1
$R_{div,4}$	32 Ω	CRCW1206100RFKEAC	0.5%	0.40	1
R_L [1]	1.6 kΩ	ERJ-8ENF1601V	1%	0.09	1
R_L [2]	100 Ω	CRCW1206100RFKEAC	1%	0.09	1
R_{choke}	120 Ω	CRCW1206120RFKEAC	1%	0.09	1
C_1	6.8 pF	VJ1206A6R8CXXCW1BC	0.25 pF	0.13	1
C_2	160 pF	C1206C161G5HACAUTO	2%	0.24	1
$C_{s,1}...C_{s,4}$	3.3 μF	C1206C335K3RACAUTO	10%	0.95	4
L_T	27 nH (4.4 mΩ)	1812SMS-27NGLC	2%	1.74	1
L_{choke}	10 μH (80 mΩ)	SRF0905-100Y	30%	0.97	1
Q_1	-	BFU550AR	-	0.40	1
D_1 ... D_{12}	-	BB201,215 [dual package]	-	0.68	6
Total				13.82	

V.2 Local Oscillator

The bill of materials of the LO is given in Table V.2.

V.3 Mixer

In this section we present the Bill of Materials for the mixer. It can be found in Table V.3.

Table V.2: A list of the values of the components used in the LO and the actual parts corresponding to them. Note that, to form some resistor values, a series combination of multiple resistors is used.

Component	**Value**	**Part Number**	**Tolerance**	**Price [€]**	**amount**
$R_{B,1}$	18 kΩ	CRCW120618K0FKEAC	1%	0.09	1
$R_{B,2}$	30 kΩ	CR1206-FX-3002ELF	1%	0.09	1
R_E [1]	3 kΩ	ERJ-8ENF3001V	1%	0.09	1
R_E [2]	200 Ω	RC1206FR-07200RL	1%	0.09	1
$R_{div,1}$	10 kΩ	CRCW120610K0FKEAC	1%	0.09	1
$R_{div,2}$ [1]	4.7 kΩ	CRGCQ1206F4K7	1%	0.09	1
$R_{div,2}$ [2]	0 to 10 kΩ	PTC10LV10-103A1010	10%	1.06	1
$R_{L,1}$	1.9 kΩ	ERJ-8ENF1911V	1%	0.09	1
$R_{L,2}$ [1]	56 Ω	CRCW120656R0FKEA	1%	0.09	1
$R_{L,2}$ [2]	15 Ω	CRCW120615R0FKEAC	1%	0.09	1
R_{choke}	27 kΩ	CRCW120627K0FKEAC	1%	0.09	1
C_1	6.8 pF	VJ1206A6R8CXXCW1BC	0.25 pF	0.13	1
C_2	160 pF	C1206C161G5HACAUTO	2%	0.24	1
$C_{s,1}$, $C_{s,2}$	3.3 μF	C1206C335K3RACAUTO	10%	0.95	2
L_T	27 nH (4.4 mΩ)	1812SMS-27NGLC	2%	1.74	1
Q_1	-	BFU550AR	-	0.40	1
D_1 ... D_{12}	-	BB201,215 [dual package]	-	0.68	6
Total				10.45	

Table V.3: A list of the values of the components used in the mixer.

Component	**Value**	**Part Number**	**Tolerance**	**Price [€]**	**amount**
R_{B1}	20 kΩ	CRCW120620K0FKEA	1%	0.09	1
R_{B2}	10 kΩ	CRCW120610K0FKEA	1%	0.09	1
R_{B3}	10 kΩ	CRCW120610K0FKEA	1%	0.09	1
R_l	3 kΩ	ERJ-8ENF3001V	1%	0.09	1
R_{W1}	10 kΩ	2002111855	-	0.40	1
R_{W2}	3.9 kΩ	CRCW12063K90FKEAC	1%	0.09	1
Q_1 - Q_6 & Q_9 - Q_{12}	-	BFU550AR	-	0.27	10
Q_7 & Q_8	-	BFQ149	-	0.67	2
C_{D1} - C_{D3}	22 uF	GCM32ER71C226ME19K	20%	1.04	3
C_{Block}	22 uF	GCM32ER71C226ME19K	20%	1.04	1
Balun	-	458PT-1566=P3	-	1.27	2
Total				11.59	

Acronyms

f_T transistor cut-off frequency.

h_{FE} common-emitter current gain.

AC Alternating Current.

ADS Advanced Design System (2020).

AM Amplitude Modulation.

BJT Bipolar Junction Transistor.

CCS Constant Current Source.

DC Direct Current.

EM Electromagnetic.

ESL Equivalent Series Inductance.

ESR Equivalent Series Resistance.

FFT Fast Fourier Transform.

FM Frequency Modulation.

FMCW Frequency-Modulated Continuous-Wave.

HB Harmonic Balance.

IF intermediate frequency.

KVL Kirchoff's Voltage Law.

LNA Low Noise Amplifier.

LO Local Oscillator.

LPF Low Pass Filter.

MLCC Multilayer Ceramic Capacitor.

PA Power Amplifier.

PCB Printed Circuit Board.

radar RAdio Detection And Ranging.

RF Radio Frequency.

SMD Surface Mount Device.

SNR Signal-to-Noise Ratio.

VCO Voltage-Controlled Oscillator.

Bibliography

[1] L. De Vreede, M. Pelk, M. Babaie, and M. Alavi, "An FM Chirp Waveform Generator and Detector for Radar A final bachelor project proposal," tech. rep., TU Delft, Delft, 2019.

[2] B. R. Mahafza, *Radar Systems Analysis and Design Using MATLAB*. Chapman and Hall/CRC, 3rd ed., 2016.

[3] N. Bhatta and M. Geethapriya, "RADAR and its Applications," in *International Journal of Circuit Theory and Applications*, International Science Press, 2016.

[4] M. A. Richards, J. A. Scheer, W. A. Holm, and W. L. Melvin, *Principles of modern radar*. Scitech Publishing, 2010.

[5] S. Tanis, "Automotive Radar Sensors and Congested Radio Spectrum: An Urban Electronic Battlefield? — Analog Devices." Accessed on: Apr. 28, 2020. [Online]. Available: `https://www.analog.com/en/analog-dialogue/articles/automotive-radar-sensors-and-congested-radio-spectrum-an-urban-electronic-warfare.html#`, 2018.

[6] J. S. Chitode, *Principles Of Communication*. Technical Publications Pune, first ed., 2008.

[7] L. W. Couch, M. Kulkarni, and U. S. Acharya, *Digital and analog communication systems*. Pearson, 2013.

[8] C. Wolff, "Radartutorial." Accessed on: June 05, 2020. [Online]. Available: `https://www.radartutorial.eu/02.basics/FrequencyModulatedContinuousWaveRadar.en.html`, 2013.

[9] P. Koivumäki, "Triangular and Ramp Waveforms in Target Detection with a Frequency Modulated Continuous Wave Radar," tech. rep., Aalto University, 2017.

[10] F. B. Berger, "The Nature of Doppler Velocity Measurement," *IRE Transactions on Aeronautical and Navigational Electronics*, vol. ANE-4, no. 3, pp. 103–112, 1957.

[11] European Communication Office, "The European table of frequency allocations and applications in the frequency range 8.3 kHz to 3000 GHz (ECA table)," tech. rep., European Communication Office, 2019.

[12] K. K. Clarke and D. T. Hess, *Communication circuits : analysis and design*. Addison-Wesley Pub. Co, 1971.

[13] A. Tasic, "Oscillators." Accessed on: Apr. 28, 2020. [Online]. Available: `https://ocw.tudelft.nl/wp-content/uploads/Oscillators.pdf`, 2006.

[14] P. Chattopadhyay, D. Rakshit, *Electronics (fundamentals And Applications)*. New Age International, 7 ed., 2006.

[15] "Hartley Oscillator - Tutorialspoint." Accessed on: Apr. 29, 2020. [Online]. Available: `https://www.tutorialspoint.com/sinusoidal_oscillators/sinusoidal_hartley_oscillator.htm`.

[16] "Clapp Oscillator - Tutorialspoint." Accessed on: Apr. 29, 2020. [Online]. Available: `https://www.tutorialspoint.com/sinusoidal_oscillators/sinusoidal_clapp_oscillator.htm`.

[17] "Colpitts Oscillator - Tutorialspoint." Accessed on: Apr. 29, 2020. [Online]. Available: `https://www.tutorialspoint.com/sinusoidal_oscillators/sinusoidal_colpitts_oscillator.htm`.

[18] V. Ulansky, M. Fituri, and I. Machalin, "Mathematical Modeling of Voltage-Controlled Oscillators With the Colpitts and Clapp Topology," *Electronics and Control Systems*, vol. 1, no. 19, 2009.

[19] B. Razavi, *RF microelectronics*. Prentice Hall, 2 ed., 2011.

[20] B. Razavi, *Design of CMOS Phase-Locked Loops*. Cambridge University Press, 2020.

[21] T. H. Lee and A. Hajimiri, "Oscillator phase noise: A tutorial," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 3, pp. 326–335, 2000.

[22] T. H. Lee, *The Design of CMOS Radio-Frequency Integrated Circuits*. Cambridge University Press, 12 2003.

[23] D. A. Neamen, *Semiconductor Physics and Devices Basic Principles Fourth Edition*. McGraw-Hill, 4 ed., 2012.

[24] B. Razavi, *Fundamentals of Microelectronics, 2nd Edition*. Wiley, 2013.

[25] H. L. Krauss, C. W. Bostian, and F. H. Raab, *Solid state radio engineering*. Wiley, 1980.

[26] NXP Semiconductors, "BFU550A Datasheet." Accessed on: Apr. 28, 2020. [Online]. Available: `https://www.nxp.com/docs/en/data-sheet/BFU550A.pdf`, 2014.

[27] C. Huang, *Ultra Linear Low-loss Varactors & Circuits for Adaptive RF Systems*. Delft University Of Technology, 2010.

[28] "BB201_BB207: Low Voltage Variable Capacitance Double Diode — NXP." Accessed on: June 10, 2020. [Online]. Available: `https://www.nxp.com/products/rf/rf-discrete-components-low-power/rf-diodes/varicap-diodes/vco-and-fm-radio-tuning-/low-voltage-variable-capacitance-double-diode:BB201_BB207?fpsp=1&tab=Design_Tools_Tab`.

[29] L. Bouman and A. Sabti, "An FM Chirp Waveform Generator and Detector for Radar; Baseband, intermediate frequency, low noise and RF power amplifiers," tech. rep., Delft University Of Technology, 2020.

[30] M. Alavi, "Communication Concepts , Part II," 2019.

[31] B. Gilbert, "Design considerations for BJT active mixers," in *Low-power HF Microelectronics: a unified approach*, pp. 837–928, IET, 10 2011.

[32] P. R. Gray, P. J. Hurst, S. H. Lewis, and R. G. Meyer, *Analysis and Design of Integrated Circuits*. New York: Wiley, fifth ed., 2009.

[33] C. Toumazou, F. Lidgey, and D. Haigh, "Analogue IC Design-The Current-Mode Approach," tech. rep., 1993.

[34] R. Sharma, A. Chaturvedi, and M. Kumar, "A Comparative Study of Different Types of Mixer Topologies," *ICTACT Journal on Microelectronics*, vol. 02, no. 01, pp. 182–187, 2016.

[35] "Capacitor Uses and Applications Electronics Notes." Accessed on: June 14, 2020. [Online]. Available: `https://www.electronics-notes.com/articles/electronic_components/capacitors/capacitor-uses.php`.

[36] H. T. Friis, "A Note on a Simple Transmission Formula," *Proceedings of the IRE*, vol. 34, no. 5, pp. 254–256, 1946.

[37] C. K. Alexander and M. N. O. Sadiku, *Fundamentals of Electric Circuits*. McGraw-Hill, 6 ed., 2017.

[38] S. Long, "Oscillators." Accessed on: May 25, 2020. [Online]. Available: `http://www.ece.ucsb.edu/~long/ece145b/`, 2007.

[39] R. V. Patron, "Analysis and Applications of the Capacitive Transformer," 2007.

[40] NXP Semiconductors, "BFQ149 Datasheet." Accessed on: Apr. 28, 2020. [Online]. Available: `https://www.nxp.com/docs/en/data-sheet/BFQ149_N.pdf`, 2007.

[41] A. S. Sedra and K. C. Smith, *Microelectronic Circuits, Seventh Edition*. OXFORD UNIVERSITY PRESS, 2014.

[42] Infineon Technologies, "Infineon LNA BFR92P." Accessed on: Apr. 28, 2020. [Online]. Available: `https://www.infineon.com/dgdl/Infineon-BFR92P-DS-v01_01-en.pdf?fileId=db3a30431400ef68011426fd178506a6`, 2013.

[43] Infineon Technologies, "Infineon LNA BFR35AP." Accessed on: Apr. 28, 2020. [Online]. Available: `https://www.infineon.com/dgdl/Infineon-BFR35AP-DS-v01_01-en.pdf?fileId=db3a30431400ef68011426b8f826067e`, 2013.

[44] NXP Semiconductors, "BB137 UHF variable capacitance diode," 2004.

[45] NXP Semiconductors, "BB201 Low-voltage variable capacitance double diode," 2001.

[46] NXP Semiconductors, "BB207 FM variable capacitance double diode," 2011.

[47] NXP Semiconductors, "BB208-02; BB208-03 Low voltage variable capacitance diode," 2011.