Tool for Optimized Design of Integrated RF CMOS Low Noise Amplifiers

by
Papadimitriou Angelos

A DIPLOMA THESIS

Submitted in fulfillment of the requirements for the degree
'Diploma in Electrical and Computer Engineering'

Department of Electrical and Computer Engineering
TECHNICAL UNIVERSITY OF CRETE

January, 2017

Examinatory Committee

Assoc. Prof. Matthias Bucher, Supervisor
Assoc. Prof. Aggelos Bletsas
Prof. Costas Balas

ABSTRACT

In a world where data transmission is one of the most critical aspects of life, there is an increasing trend to produce devices that communicate wirelessly and in the majority of cases are mobile. The most common approach, is to design new wireless devices in the 2.4 GHz and 5.5 GHz bands in order to take advantage of the high transmission speeds that they provide and the fact that a device can operate in these frequency bands without license. Additionally, the demand for low power circuits is a perpetual call for research, where the CMOS technology thrives. The aggressive scaling of CMOS devices also provides the opportunity to integrate numerous digital and analog blocks in a single chip. The system on a chip (SoC) approach also applies in the field of telecommunications, where the different blocks that constitute a transceiver are integrated in a single die. One of those RF blocks is the low noise amplifier. Low noise amplifiers are usually the first blocks in the receiver chain and the ones that first process the incoming signal. Due to the fact that the signal has undergone significant attenuation, the purpose of the LNA is to amplify it while keeping the noise added by the circuit itself, at a minimum. The objective of this thesis is to present a methodology to design optimized LNAs. The term “optimized” translates to finding a design that provides a perfect balance in trade-offs like power consumption, gain and noise performance. This can be achieved with the use of a well defined model of equations that generates an approximation of an optimized LNA in combination with a genetic algorithm for the fine tuning of the circuit parameters.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Radio Frequency (RF) and analog mixed-signal technologies serve the rapidly growing wireless communication market. The growing demand for larger bandwidth, motivates the RF circuit designers to advance to higher frequencies. Additionally, RF circuit design is increasingly taking advantage of the aggressive scaling of submicrometer CMOS technologies that make possible the integration of complete telecommunication systems in a single chip. Although several chips containing RF parts have been appeared in the last few years, the design of RF CMOS integrated circuits remains a challenge due to strong constraints in power consumption and noise.

Low noise amplifiers (LNAs) are one of the most integral parts of any telecommunication system such as the Bluetooth, the global system for mobile communications (GSM), wireless local area networks (WLAN), wide area networks (WAN), global positioning system (GSP), satellite communications along with others. LNAs are also found in medical instruments, such as the electrocardiography system, and other electronic equipment.

Since most of the top selling applications of the low noise amplifiers are mobile, thus powered by battery, the LNA must produce adequate results with low power consumption. Even though there is a wide research in the field, the power constraints adds to the notion that the LNA design is not an easy task. Each application has different protocols that include specific requirements for the RF blocks in the transceiver chain. For example the power levels of a satellite signal received by a GPS receiver may be as low as -160 dBm. This poses a challenge on the sensitivity of the system. A low noise amplifier with low *noise figure* and high *gain* is required to boost the sensitivity of the system. Additionally, in the modern era of mobile phones, the GPS signals are co-habited by strong interfering cellular signals. The cellular signal can mix to produce *intermodulation* products exactly in the GPS receiver frequency band. To enhance interference immunity of the GPS system LNAs with high *linearity* characteristics such as *IP3* and input *P1dB* are required.

The problem with all these design quantities though, is that they are not independent, imposing trade-offs in analog and RF design. Figure 1 shows the RF design hexagon which illustrates that the different figures of merit, such as, gain, noise, linearity, and power consumption are interdependent. Hence there has to be a certain design, where the low noise amplifier operates *optimally*, balancing these figures of merit. The optimization of the performance is one of the most critical parts in RF design. In the past, the most common practice was the trial and error way, where the optimal operating point was a product of many hours spent tweaking the design, and trying to find the best performing one.

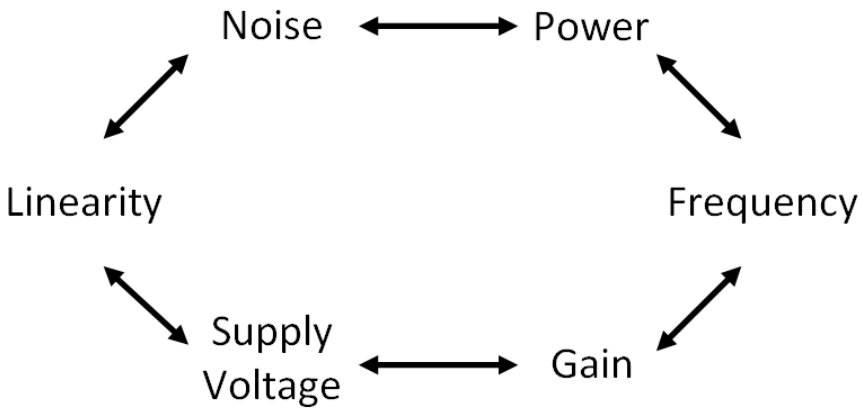

Figure 1: The RF design hexagon

Lately, a more productive approach is used, where the designer follows a methodology that leads to an optimal result. This methodology is usually a set of equations that are produced from a well defined compact transistor characterization model.

Another approach, is based on methods with iterative philosophy. These methods usually rely on computer simulations that use the results of past designs as input and try to output a better design in each iteration, either by a deterministic algorithm, that evaluates the input parameters and acts accordingly, or by non-deterministic (*randomized*) algorithms, such as evolutionary algorithms, that search for the optimum in a wide search space. The iterative approaches, always produce an optimum result, but the convergence time is not finite, i.e. they may be time and (computing) power consuming.

The advantage of randomized algorithms, over the deterministic counterparts is that they do not require a strict modeling of the environment variables (e.g. exact modeling of inductors), because they just try to maximize (or minimize) a single figure of merit that summarizes how close a given design solution is to achieving the set aims, by randomly adjusting the design variables. This figure of merit is often called a *fitness function*. The disadvantage of this type of methodology over a deterministic one with a well defined modeled environment, is that their convergence time is likely to be longer due to their random complexity.

In this thesis we will take advantage of both the model equations and a genetic algorithm, and ultimately use their combination to produce an optimized low noise amplifier design. The overview of the thesis is as follows. Chapter 2 is going to be an introduction to the RF analog design, presenting concepts that a designer must be aware of when designing a circuit. These concepts range from telecommunication theories, to noise in MOSFETs and matching network design. Chapter 3, introduces the circuit of the low noise amplifier and the analysis of the design specifications as well as the topologies that are going to recur throughout this work. Chapter 4 is the presentation of the procedure of defining where the optimum operation can be found, and introduce the notion of the *figure-of-merit*. Later on, chapter 5 uses the combined knowledge of chapter 3 and 4 to propose a methodology to design a low noise amplifier. Afterwards, chapter 5 analyzes an iterative approach that is used to produce the final optimized design. Finally, using the methodology, we show a case study for a 5 GHz LNA. Chapter 6, is the conclusion and the future work.

CHAPTER 2

BASIC CONCEPTS IN RF DESIGN

Radio frequency design draws upon many concepts from a variety of fields, including signals and systems, electromagnetics and microwave theory, and communications. Nonetheless, RF design has developed its own analytical methods and its own language. This chapter deals with general concepts that prove essential to the analysis and design of RF circuits, closing the gaps with respect to other fields.

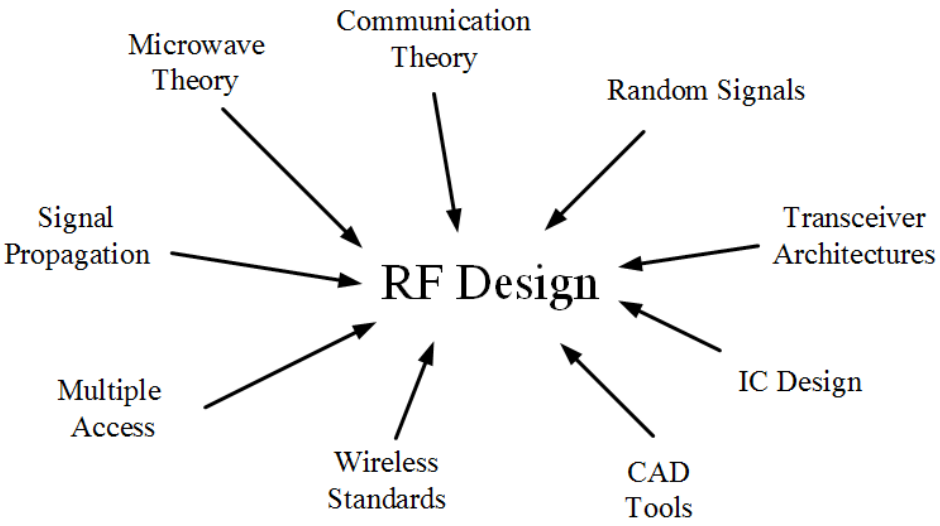

Figure 2: Various Disciplines in RF design

2.1 SCATTERING PARAMETERS

Microwave theory deals mostly with power quantities rather than voltage or current quantities. Two reasons can explain this approach. First, traditional microwave design is based on transfer of power from one stage to the next. Second, the measurement of high frequency voltages and currents in the laboratory proves very difficult, whereas that of average power is more straightforward. Microwave theory therefore models devices, circuits and systems by parameters that can be obtained through

the measurement of power quantities. These are called scattering *parameters* or *S-parameters* and are used extensively to characterize components and also to analyze circuits.

The S-parameters are members of a family of similar parameters, other examples being: Y-parameters, Z-parameters, H-parameters, T-parameters or ABCD-parameters. They differ from these, in the sense that S-parameters do not use open or short circuit conditions to characterize a linear electrical network; instead, matched loads are used. These terminations are much easier to use at high signal frequencies than open-circuit and short-circuit terminations.

In the S-parameter approach, an electrical network is regarded as a "black box" containing various interconnected basic electrical circuit components or lumped elements such as resistors, capacitors, inductors and transistors, which interacts with other circuits through *ports*. The network is characterized by a square matrix of complex numbers called S-parameter matrix, which can be used to calculate its response to signals applied to the ports.

For the S-parameter definition, it is understood that a network may contain any components, provided that the entire network behaves *linearly* with incident small signals. It may also include many typical communication system components or "blocks" such as amplifiers, attenuators, filters, etc, provided they are also operating under *linear* and defined conditions.

An electrical network to be described by S-parameters may have any number of ports. Ports are the points at which electrical signals either enter or exit the network. Ports are usually pairs of terminals with the requirement that the current into one terminal is equal to the current leaving the other.

The S-parameter matrix describing an N-port network will be square of dimension N and will therefore contain N^2 elements. At the test frequency each element or S-parameter is represented by a unitless complex number that represents magnitude and angle, i.e. *amplitude* and *phase*.

For a generic multi-port network, the ports are numbered from 1 to N, where N is the total number of ports. For port n, the associated S-parameter definition is in terms of incident and reflected waves, a_n and b_n respectively. The incident and reflected waves are defined as :

$$a=\frac{1}{2}k\left(V+Z_P I\right) \text{ (2.1)}, \quad b=\frac{1}{2}k\left(V-Z_P^* I\right) \text{ (2.2)}$$

where Z_P is the diagonal matrix of the complex reference impedance for each port, Z_P* is the element wise complex conjugate of Z_P, V and I are respectively the column vectors of the voltages and currents at each port and

$$k=\frac{1}{|\Re\{Z_P\}|} \text{ (2.3)}$$

Sometimes it is useful to assume that the reference impedance is the same for all ports in which case the definitions of the incident and reflected waves may be simplified to

$$a=\frac{V+Z_0 I}{|\Re\{Z_P\}|} \text{ (2.4) and } b=\frac{V+Z_0 I}{|\Re\{Z_P\}|} \text{ (2.5)}$$

For all ports the reflected power waves may be defined in terms of the S-parameter matrix and the incident power waves by the following matrix equation : $b = Sa$, where S is an NxN matrix. The S-parameter matrix of the 2-port is probably the most commonly used, and serves as the building block for generating the higher order matrices for larger networks. In this case the relationship between the reflected and the incident waves and the S-parameter matrix is given by :

$\begin{pmatrix} b_1 \\ b_2 \end{pmatrix}=\begin{pmatrix} S_{11}S_{12} \\ S_{21}S_{22} \end{pmatrix}\begin{pmatrix} a_1 \\ a_2 \end{pmatrix}$ (2.6), expanding the matrix into equations :

$b_1=S_{11}a_1+S_{12}a_2$ (2.7) and $b_2=S_{21}a_1+S_{22}a_2$ (2.8).

Each equation gives the relationship between the reflected and incident power waves at each of the network ports, 1 and 2, in terms of the network's individual S-parameters. Considering the

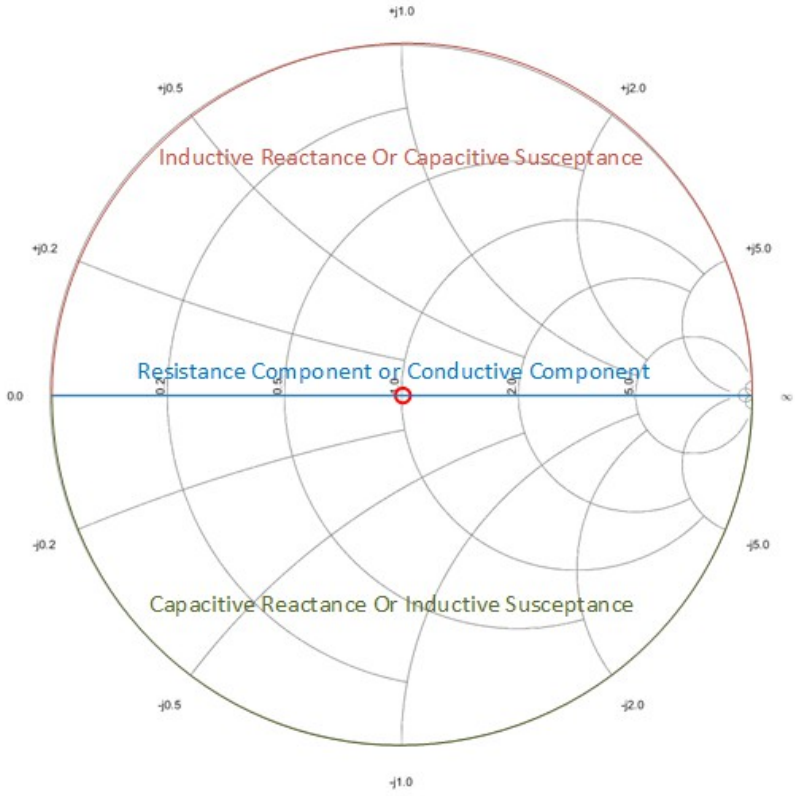

Figure 3: The Smith Chart

definitions of the 2-port scattering matrix the S_{ij} parameters are calculated in 2.9-2.12 as :

$$S_{11}=\frac{b_1}{a_1}\Big|_{a_2=0},\ S_{21}=\frac{b_2}{a_1}\Big|_{a_2=0},\ S_{12}=\frac{b_1}{a_2}\Big|_{a_1=0},\ S_{22}=\frac{b_2}{a_2}\Big|_{a_1=0}$$

If we define the incident waves as $a_1=V_1^+$, $a_2=V_2^-$, and the reflected waves as $b_1=V_1^-$ and $b_2=V_2^+$ then the S-parameters take a more intuitive meaning such as :

S_{11} : is the input port reflection coefficient
S_{12} : is the reverse voltage gain
S_{21} : is the forward voltage gain
S_{22} : is the output port voltage reflection coefficient

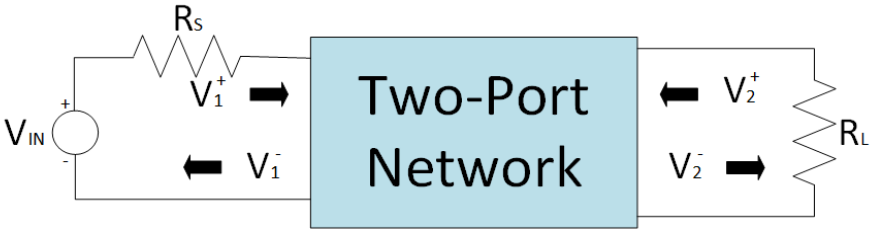

Figure 4: A two-port network with incident and return signals

Any 2-port S-parameter may be displayed on a *Smith Chart* using polar co-ordinates, but the most meaningful would be S_{11} and S_{22} since either of these may be converted directly into an equivalent normalized impedance (or admittance) using the characteristic Smith Chart impedance (or admittance) scaling appropriate to the system impedance.

2.2 THE SMITH CHART

The Smith chart shown in figure 3, is a graphical aid that is very useful when solving transmission line problems. Although there are a number of other impedance and reflection coefficient charts that can be used for such problems, the Smith chart is probably the best known and most widely used. Besides being an integral part of any current computer-aided-design (CAD) software and test equipment for microwave design, the Smith chart provides an extremely useful way of visualizing transmission line phenomena.

The Smith chart is essentially a polar plot of the voltage reflection coefficient Γ. Let the reflection coefficient be expressed in magnitude and phase (polar) form as $\Gamma = |\Gamma| e^{j\theta}$. Then the magnitude $|\Gamma|$ is plotted as a radius ($|\Gamma| \leq 1$) from the center of the chart, and the angle θ ($-180^o \leq \theta \leq 180^o$) is measured from the right-hand side of the horizontal diameter. Any passively realizable ($|\Gamma| \leq 1$) reflection coefficient can then be plotted as a unique point on the Smith chart.
The real utility of the Smith chart though, lies in the fact that it can be used to convert from reflection coefficients, to normalized impedances (or admittances), and vice versa, using the impedance (or admittance) circles printed on the chart. When dealing with impedances on a Smith chart, normalized quantities are generally used. The normalization constant is usually the characteristic impedance of the line.

A locus of points on a Smith chart covering a range of frequencies can be used to visually represent :

- How capacitive or how inductive a load is across the frequency range.
- How difficult matching is likely to be at various frequencies.
- How well matched a particular component is.

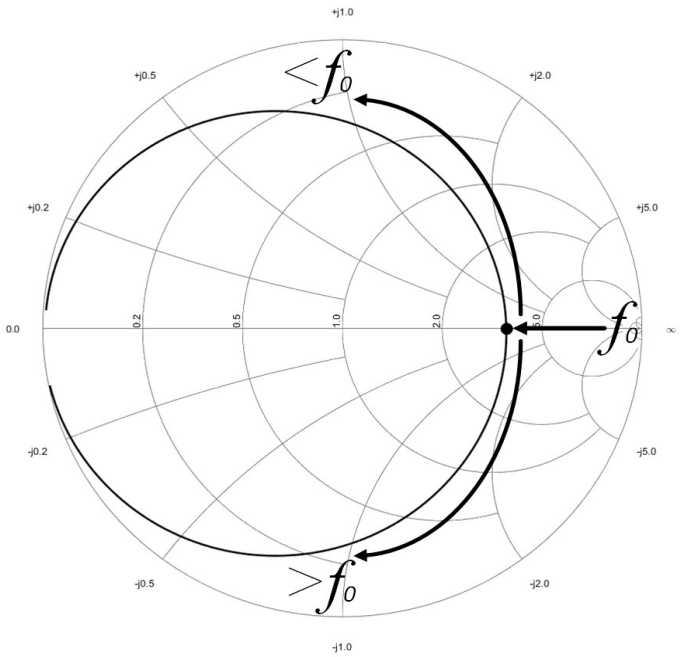

Figure 5: Example of an S-parameter response on the Smith Chart

Now we will explain the different regions on the Smith chart. If a polar diagram is mapped on to a Cartesian coordinate system it is conventional to measure angles relative to the positive x-

axis using a counterclockwise direction for positive angles. The magnitude of a complex number is the length of a straight line drawn from the origin to the point representing it. The Smith chart uses the same convention, noting that, in the normalized impedance plane, the positive x-axis extends from the center of the Smith chart at $z_T = 1 \pm j0$ to the point $z_T = \infty \pm j\infty$. The region above the x-axis represents inductive impedances (positive imaginary parts) and the region below the x-axis represents capacitive impedances (negative imaginary parts).

If the termination is perfectly matched, the reflection coefficient will be zero, represented effectively by a circle of zero radius or in fact a point at the center of the Smith chart. If the termination was a perfect open circuit or short circuit the magnitude of the reflection coefficient would be unity, all power would be reflected and the point would lie at some point on the unity circumference circle.

In the example of figure 5, the system shows a capacitive behavior for frequencies lower than the center frequency f_0, and inductive behavior for frequencies greater than f_0. In the center frequency, the S-parameter response do not lie in the center of the Smith chart therefore, the system is not matched at the characteristic impedance.

2.3 IMPEDANCE MATCHING

The impedance matching is often a part of the larger design process for a microwave system. The basic idea of impedance matching is illustrated in figure 6, which shows an impedance matching network placed between a load impedance and a transmission line. The matching network is ideally lossless, to avoid unnecessary loss of power, and is usually designed so that the impedance seen looking into the matching network is Z_0.Then reflections are eliminated on the transmission line to the left of the matching network, although there will be multiple reflections between the matching network and the load. This procedure is also referred to as *tuning*. Impedance matching or tuning, is important for the following reasons :

- Maximum power is delivered when the load is matched to the line.
- Impedance matching sensitive receiver components (antenna, low noise amplifier, etc.) improves the signal-to-noise ratio of the system
- Impedance matching in a power distribution network will reduce amplitude and phase errors.

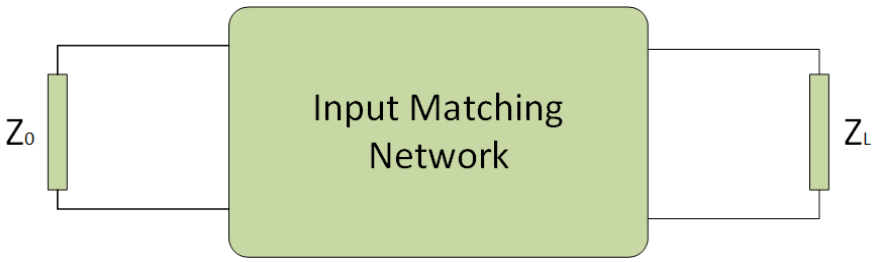

Figure 6: General impedance matching network

As long as the impedance, Z_L, has some nonzero real part a matching network can always be found. Many choices are available. The factors that may be important in the selection of a particular matching network include the following :

- Complexity : A simpler matching network is usually cheaper, more reliable and less lossy than a more complex design.
- Bandwidth : Any type of matching network can ideally give perfect match at a single

frequency. In many applications, however, it is desirable to match a load over a band of frequencies.

- Implementation : Depending on the transmission line being used, one type of matching network may be preferable compared to another.
- Adjustability : In some applications the matching network may require adjustment to match a variable load impedance.

2.3.1 L-SECTIONS

An L-section is a two-element matching network. The four possible configurations are shown in figure 7. Depending on the position of the first component (as viewed from the load), the load resistance can be transformed upwards or downwards with an L-section.
When the first reactive component is a series component, the transformation is upward; and when it is parallel, the transformation is downward.
The second element in the L-section is used to remove the residual reactance caused by the transformation element (i.e. the first element). The second element is therefore the compensating element.

When a reactive element (X_1) is added in series with a resistor (R) and the equivalent parallel combination is considered (series to shunt transformation), the resistance increases by a factor $D_1 = 1 + Q_1^2$ (2.13), where $Q_1 = X_1 / R$ (2.14).

When a reactive element (X_1) is added in parallel with a resistor (R) and the equivalent series combination is considered (parallel to series transformation), the resistance decreases by the same factor D_1. In this case however, the Q-factor is defined as : $Q_1 = -R / X_1 = B_1 / G$ (2.15).

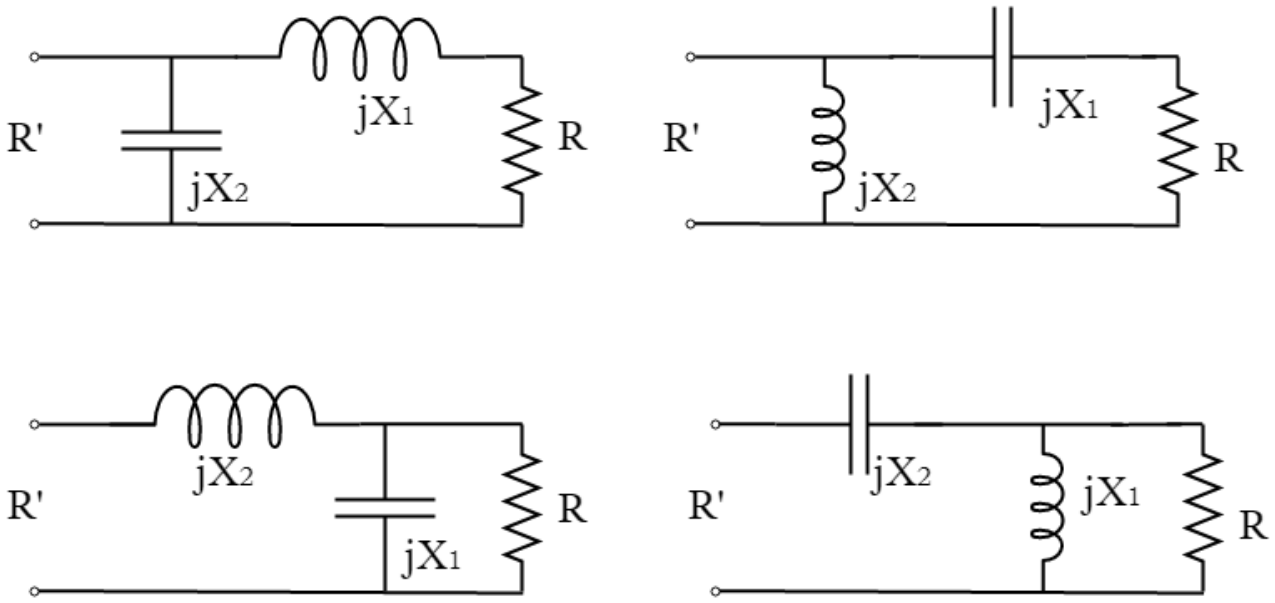

Figure 7: The four possible configurations for an L-section

The ratios defined in (2.14) and (2.15) are similar in form to the Q-factors of the series or parallel resonant circuits, respectively. These ratios are referred to as *transformation Qs*.

As in the case with resistance, the reactance increases after a series to shunt transformation and decreases when a shunt to series transformation is considered. The reactance of the first element used in an L-section is determined by the transformation Q required to transform the load resistance

(R) to the value required (R'). The Q value can be calculated using the relationship $R'=D_1 R=(1+Q_1^2)R$ (2.16), note that a positive or negative sign can be assigned to the transformation Q. The second element in the L-section is used to achieve the desired reactance level. If a purely resistive input impedance is required, the reactance of this element is given by

$X_2=-X_1(1+\frac{1}{Q_1^2})=\frac{R'}{Q_1}$ (2.17), if the first element is a series element and by

$X_2=\frac{-X_1}{(1+1/Q_1^2)}=R'\cdot Q_1$ (2.18), if the first element is a shunt element.

The procedure outlined can be extended easily to the general case where the load and source impedances are complex. Additionally the transformation Q of the matching circuit determines the bandwidth of the circuit. The bandwidth is calculated by : $Q=\frac{f_0}{BW}$ (2.19), where f_0 is the frequency of interest.

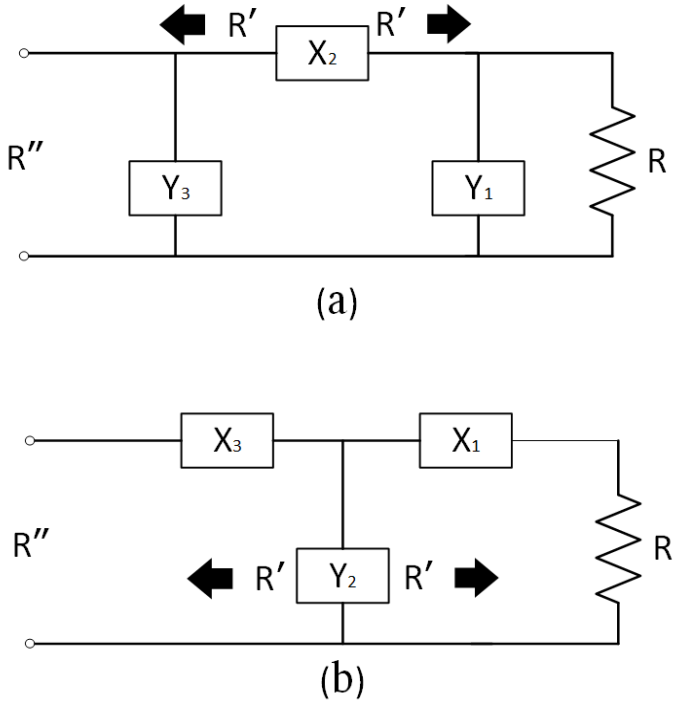

Figure 8: Topology for (a) a PI-section and (b) a T-section

2.3.2 Π-SECTIONS AND T-SECTIONS

The two configurations for the PI- (Π) and T- networks are shown in figure 8. The main reason to employ a T-network or a PI-network is to get control of the circuit Q. In designing an L-network the transformation Q is a function of the input and output impedances. This means that the matching is accompanied by a fixed Q that may not meet your design specs. In most cases the Q is low, and may not be appropriate for applications where a limited bandwidth is required, in order to reduce harmonics or help filter out adjacent signals, without the use of additional filters. The T-networks and PI-networks provide enough variety to fit almost any situation.

The first two elements in these sections are transforming elements. One of these elements causes the resistance to increase while the other causes it to decrease. The reactance level is set by

the last element in the section (the compensating element).
Because the resistance is transformed twice, there are two transformation *Qs* in these sections. The highest transformation *Q* can be chosen to have any value higher than the required in an equivalent L-section.

As in the case of L-sections, the bandwidth of PI- and T-networks are also determined by the transformation *Qs*. Where the two Q-factors are different, the *Q* of the network will be approximately equal to half of the highest transformation *Q*.

Π-SECTION

The resistance transformations cased by a PI-section are illustrated in figure 9. The resistance is first transformed downwards by a factor $(1+Q_1^2)$ and then upwards with a factor $(1+Q_2^2)$.
Q_1 is the first transformation *Q* and is associated with the load resistance and the first element of the network, while Q_2 is the second transformation *Q*.

The value of the highest transformation *Q* is determined by the required bandwidth of the network. The *Q* of the network is approximately equal to one-half of the highest transformation *Q* when the transformation *Q* factors are sufficiently different.

Equations 2.20- 2.25 can be used to design a PI-network when the load resistance must be transformed downward (figure 9 (a)), while equations 2.26-2.28 should be used with 2.23-2.25 when the load resistance must be transformed upwards (figure 9 (b)).

$$Q_1 = Q_{max} = 2Q \quad (2.20)$$
$$R' = R/(1+Q_1^2) \quad (2.21)$$
$$1+Q_2^2 = R''/R' \quad (2.22)$$
$$Y_1 = Q_1/R \quad (2.23)$$
$$X_2 = R'(Q_1+Q_2) \quad (2.24)$$
$$Y_3 = Q_2/R'' \quad (2.25)$$
$$Q_2 = Q_{max} = 2Q \quad (2.26)$$
$$R' = R''/(1+Q_2^2) \quad (2.27)$$
$$1+Q_1^2 = R/R'' \quad (2.28)$$

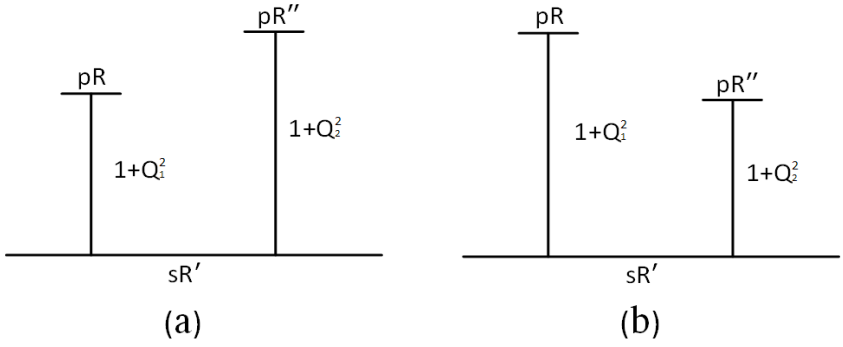

Figure 9: (a) upward transformation of the load resistance with a PI-section (b) downward transformation of the load resistance with a PI-section

T-SECTION

The dual of a PI-section is the T-section. Therefore, the formulas for designing a PI-section can also be used to design a T-section. In order to do so, it is necessary to replace the resistance and reactance in these formulas with conductance and susceptance, respectively. The terminations used must also be inverted (*1/R*).

The reactance results of the PI-section apply directly to the T-section if these are interpreted to be susceptances. To illustrate this, if the components required for a PI-section are *j10Ω*, *-j5.1Ω*, and *j5.3Ω*, the components required in the T-section are *j10Ω*, *-j5.1Ω*, and *j5.3Ω*.
Equations 2.29-2.34 can be used when the load resistance must be decreased, while 2.35-2.37 and 2.32-2.34 can be used when an upward transformation is required :

$$Q_2 = Q_{max} = 2Q \quad (2.29)$$

$$R = R''\left(1+Q_1^2\right) \quad (2.30)$$
$$1+Q_1^2 = R'/R \quad (2.31)$$
$$X_1 = Q_1 R \quad (2.32)$$
$$Y_2 = \left(Q_1+Q_2\right)/R' \quad (2.33)$$
$$X_3 = Q_2 R'' \quad (2.34)$$
$$Q_1 = Q_{max} = 2Q \quad (2.35)$$
$$R' = R\left(1+Q_1^2\right) \quad (2.36)$$
$$1+Q_2^2 = R'/R'' \quad (2.37)$$

As in the L-section, this procedure can be applied to the general case, where the load and source impedances are complex.

2.4 LINEARITY

Considering a single-input, single-output system, for input *x(t)* the output is *y(t)*. This system is considered to be *linear* if its input can be expressed as a linear combination (superposition) of responses to individual inputs. More specifically, if the outputs in response to inputs $x_1(t)$ and $x_2(t)$ can be expressed as $y_1(t) = f\{x_1(t)\}$, and $y_2(t) = f\{x_2(t)\}$, respectively, then :
$a \cdot y_1(t) + b \cdot y_2(t) = f\{a \cdot x_1(t)\} + f\{b \cdot x_2(t)\}$, for arbitrary values of α and *b*. Any system that not satisfy this condition is non-linear.

Furthermore, a system is called *time-invariant* if a time shift in its input results in the same time shift in its output. That is if, $y(t) = f\{x(t)\}$, then $y(t-\tau) = f\{x(t-\tau)\}$, for arbitrary τ. A linear system can generate frequency components that do not exist in the input signal, if only it is time-variant.

A system is called *memoryless* or *static* if its output does not depend on the past values of its input or the past values of the output itself. For a memoryless linear system the input/output characteristic is :
$y(t) = a \cdot x(t)$, where α is a function of time if the system is time-variant. For a memoryless nonlinear system, the input/output characteristic can be approximated with the polynomial :

$$y(t) = a_0 + a_1 x(t) + a_2 x^2(t) + a_3 x^3(t) + \cdots \quad (2.38)$$

where α_j *may* be functions of time if the system is time-variant. If a sinusoidal input $x(t) = A\cos(\omega t)$, is applied to a non-linear 3rd order approximation system, the output becomes :

$$y(t) = a_0 + a_1 A\cos(\omega t) + a_2 A^2 \cos^2(\omega t) + a_3 A^3 \cos^3(\omega t)$$
$$= a_0 + a_1 A\cos(\omega t) + \frac{a_2 A^2}{2}\left(1+\cos(2\omega t)\right) + \frac{a_3 A^3}{3}\left(3\cos(\omega t) + \cos(3\omega t)\right)$$
$$= a_0 + \frac{a_2 A^2}{2} + \left(a_1 A + \frac{3 a_3 A^3}{4}\right)\cdot\cos(\omega t) + \frac{a_2 A^2}{2}\cdot\cos(2\omega t) + \frac{a_3 A^3}{4}\cdot\cos(3\omega t) \quad (2.39)$$

This result shows that, the output introduces frequency components (*harmonics*) that are multiples of the input frequency, and a dc quantity arising from the second order non-linearity. Generally, dc offsets are introduced by even-order non-liniearities.

The small signal gain of circuits is usually obtained with the assumption that harmonics are negligible. However, the formula of harmonics (2.39), indicates that the gain experienced by $A\cos(\omega t)$ is equal to $a_1 A + \frac{3 a_3 A^3}{4}$ (2.40), and hence varies appreciably as *A* becomes larger.

2.4.1 1-DB COMPRESSION

Most linear amplifiers have a fixed gain for a specific frequency range. Figure 10 shows this linear behavior in an input power vs output power plot. The slope of this line is the gain. As the input power continues to increase, at some point the gain begins to decrease. The amplifier goes into *compression* where no further output increase occurs for an input increase. The gain flattens,

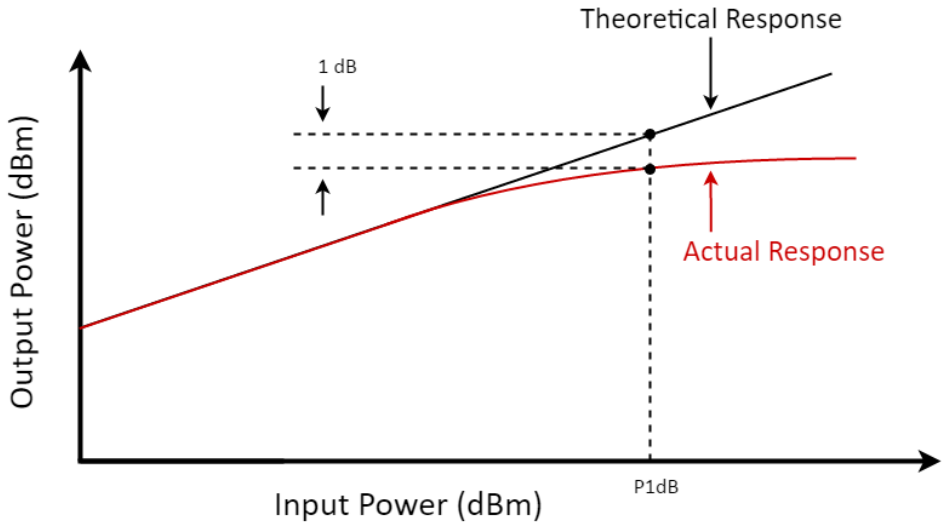

Figure 10: 1dB compression point

meaning that at high signal levels, the amplifier becomes *saturated*. Its response becomes non-linear and produces signal distortion, harmonics and potentially intermodulation products, i.e. the modulation of signals that contain two or more different frequencies.

It is important to know at what point compression begins to occur so input levels can be restricted to prevent distortion. That point is usually the input power that causes the gain to decrease 1 dB from the normal linear gain specification. The 1-dB decrease may be specified as the input level that produces it, or the output power where 1-dB drop occurs.

To calculate the input 1-dB compression point, we equate the compressed gain as seen in (2.40), to 1 dB less than the ideal gain, α_1 :

$$20\log\left|a_1+\frac{3a_3A_{\text{in},1dB}^2}{4}\right|=20\log|a_1|-1\,dB \quad (2.41)$$

It follows that

$$A_{\text{in},1dB}=\sqrt{0.145\left|\frac{a_1}{a_3}\right|} \quad (2.42)$$

While gain compression by 1 dB seems arbitrary, the 1-dB compression point represents 10% reduction in the gain and is widely used to characterize RF circuits and systems. Additionally, the amplifier should operate below the 1-dB compression point, in the linear region.

2.4.2 INTERMODULATION

Another scenario of interest in RF design occurs if two interferers accompany the desired signal. If two interferers at ω_1 and ω_2 are applied to a nonlinear system, the output generally exhibits

components that are not harmonics of these frequencies. Called *intermodulation* (IM), this phenomenon arises from mixing (multiplication) of the two components as their sum is raised to a power greater than unity. Considering again the 3rd order approximation of equation (2.39), and assuming the input is $x(t)=A_1\cos(\omega_1 t)+A_2\cos(\omega_2 t)$. Thus,

$$y(t)=a_1(A_1\cos(\omega_1 t)+A_2\cos(\omega_2 t))+a_2(A_1\cos(\omega_1 t)+A_2\cos(\omega_2 t))^2 + a_3(A_1\cos(\omega_1 t)+A_2\cos(\omega_2 t))^3 \quad (2.43)$$

Expanding the right-hand side and discarding the dc terms, harmonics, and components at $\omega_1\pm\omega_2$, we obtain the following *intermodulation products* :

$$\text{At } \omega=2\omega_1\pm\omega_2 : \frac{3a_3A_1^2A_2}{4}\cos[(2\omega_1+\omega_2)t]+\frac{3a_3A_1^2A_2}{4}\cos[(2\omega_1-\omega_2)t] \quad (2.44)$$

$$\text{At } \omega=2\omega_2\pm\omega_1 : \frac{3a_3A_1^2A_2}{4}\cos[(2\omega_2+\omega_1)t]+\frac{3a_3A_1^2A_2}{4}\cos[(2\omega_2-\omega_1)t] \quad (2.45)$$

Plus these fundamental components :

At $\omega=\omega_1,\omega_2$:

$$\left(a_1A_1+\frac{3}{4}a_3A_1^3+\frac{3}{2}a_3A_1A_2^2\right)\cos(\omega_1 t)+\left(a_1A_2+\frac{3}{4}a_3A_2^3+\frac{3}{2}a_3A_2A_1^2\right)\cos(\omega_2 t) \quad (2.46)$$

Figure 11 illustrates the results. Among these the third order IM products at $2\omega_1-\omega_2$ and $2\omega_2-\omega_1$ are of particular interest. This is because, if ω_1 and ω_2 are close to each other, then $2\omega_1-\omega_2$ and $2\omega_2-\omega_1$, appear in the vicinity of ω_1 and ω_2.

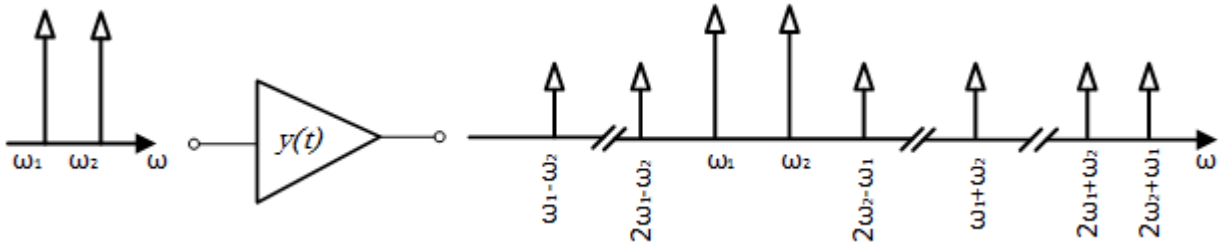

Figure 11: Generation of various intermodulation components in a two-tone test

This statements is of utmost significance. Suppose an antenna receives a small desired signal at ω_0 with two large interferers at ω_1 and ω_2, providing a combination to a low noise amplifier. If the interferer frequencies happen to satisfy $2\omega_1-\omega_2=\omega_0$, consequently, the intermodulation product at $2\omega_1-\omega_2$ falls onto the desired channel, corrupting the signal (figure 10).

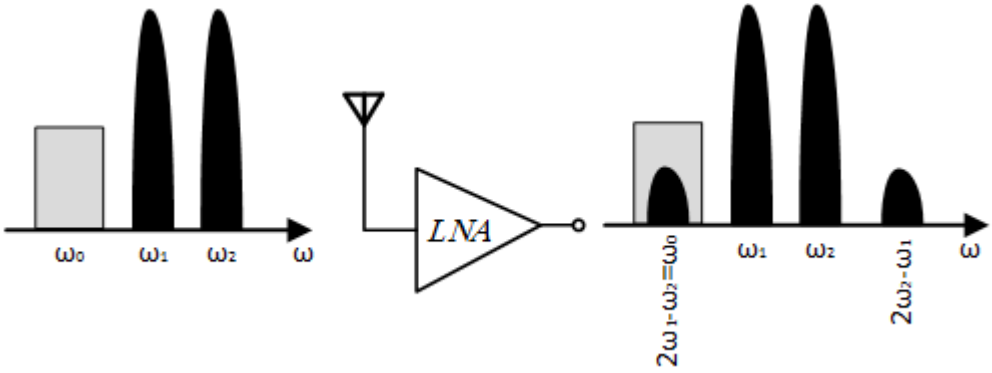

Figure 12: Corruption due to third-order intermodulation

2.4.3 THIRD INTERCEPT POINT

Figure 12 shows two signals ω_1 and ω_2 occurring within the amplifier bandwidth. With distortion, the third order product $2\omega_1 - \omega_2$, appears in the bandwidth of the desired signal and corrupts the output.

Figure 13 shows an output versus input power plot, that is similar to the case of the 1-dB compression. This is the first-order plot. Note that the flattening of the gain curve shows compression. Also plotted on the same graph are the third-order product signal levels. Figure 13 also shows the extended linear portions of the two gain curves. They meet at a point where the third order signals equal the first order (or input) signals in amplitude. This is the third order intercept point *IP3*. It is a theoretical point that is never achieved in practice, however it is useful in determining the linearity condition of an amplifier.

The *IP3* value can be read with reference to the input or the output. Reading the value from the output axis, one gets the *OIP3*, while reading the value from the input axis, the value is *IIP3*. The higher the output at the intercept, the better the linearity, and the lower the intermodulation distortion. The *IP3* value essentially indicates how large a signal the amplifier can process before intermodulation distortion occurs.

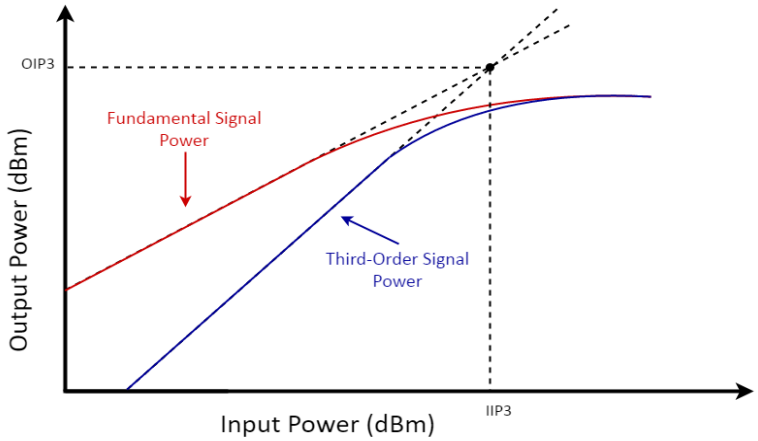

Figure 13: The third intercept point

2.5 NOISE

The performance of RF systems is limited by noise. Without noise, an RF receiver would be able to detect arbitrarily small signals, allowing communication across arbitrarily long distances. The problem with the noise is its random nature. The random nature means that, the instantaneous value of the noise cannot be predicted. Let's assume a simple circuit of a resistor with a battery (fig. 14). The current flowing through the resistor R should be equal to V_B/R, but due to ambient temperature, each electron experiences thermal agitation, thus following a somewhat random path, while on average moving towards the positive terminal of the battery. As a result the average current flowing through the resistor R is equal to V_B/R, but the instantaneous current displays random values. In the graph of figure 14, the red line displays the random fluctuations of the current while the black line is the average value of current flowing through the resistor. Since the instantaneous value of the noise cannot be determined, we use the average value which is a statistical approach to express the effects of the noise in the time-domain. If the waveform of the noise is represented by n(t) then the average power of the noise is :

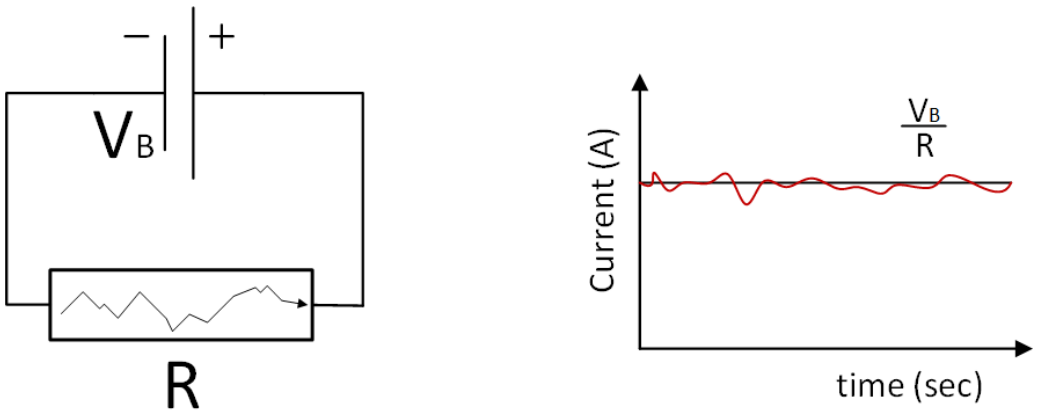

Figure 14: Noise induced from a resistance

$$P_n = \lim_{T\to\infty} \frac{1}{T} \int_0^T n^2(t)\,dt \quad (2.47)$$

where T represents the time. Due to randomness, noise consists of different frequencies. Thus, the sampling time T should be long enough to accommodate several cycles of the lowest frequency.

However, the time-domain approach, provides limited information, i.e. the average power. The frequency-domain view however, yields greater insight and proves more useful in RF design. To provide that, the concept of *power spectral density* (*PSD*) is introduced. Power spectral density is calculated by expressing the average power of a signal x(t), in every frequency of the spectrum.

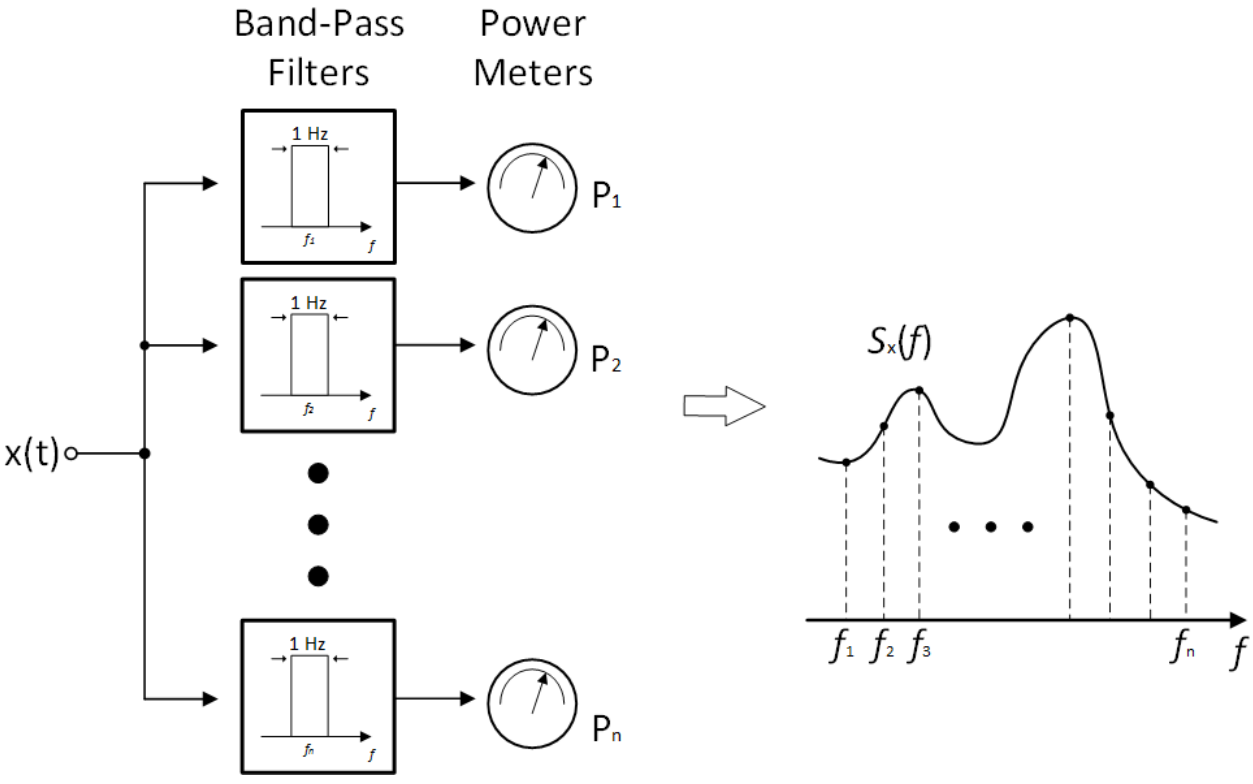

Figure 15: Power Spectral Density Calculation process

Figure 15, illustrates this procedure. Band-pass filters of 1-Hz bandwidth single out each frequency and then a power measurement takes place. After all frequencies on the spectrum have been measured, the resulting plot of the power spectral density displays the average power across the spectrum as seen in the graph of figure 15 and it is denoted by $S_x(f)$. The total area under $S_x(f)$ represents the average power carried by x(t) :

$$\int_0^\infty S_x(f)\,df = \lim_{T\to\infty}\frac{1}{T}\int_0^T x^2(t)\,dt \quad (2.48)$$

2.5.1 THERMAL NOISE OF RESISTORS

As mentioned previously, the current in a physical resistor experiences thermal agitation, due to the random collision of the electrons with lattice atoms which leads to noise. This noise can be modeled by a series voltage source with power spectral density of $S_{v_n}(f)=\overline{V_n^2}=4kTR_1$ (Thevenin equivalent), or with a parallel current source with PSD of $\overline{I_n^2}=\overline{V_n^2}/R_1=4kT/R_1$ (Norton equivalent, figure 16), where k is the Boltzmann's constant and T the absolute temperature. The units for each representation are V^2/Hz and A^2/Hz, respectively. The quantity kT is called the *available noise power* having a dimension of power per unit bandwidth and is independent of the resistor's value. As indicated by the above equations, the resistor's thermal noise is *white,* i.e it does not depend on frequency. In reality $S_{v_n}(f)$ if flat up to frequency of 100 THz, dropping at higher frequencies (figure 17).

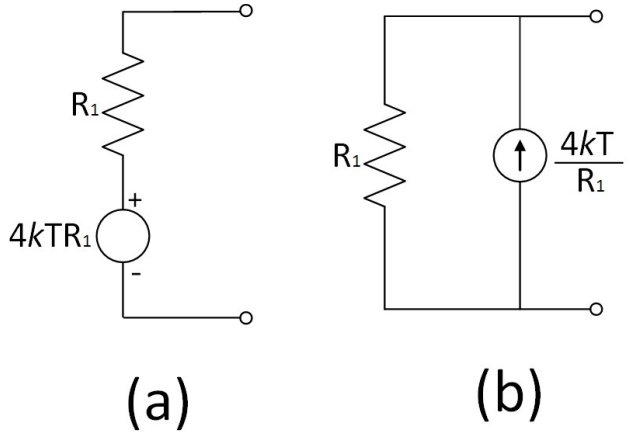

Figure 16: Thermal noise modeling of resistor as (a) voltage source (b) current source

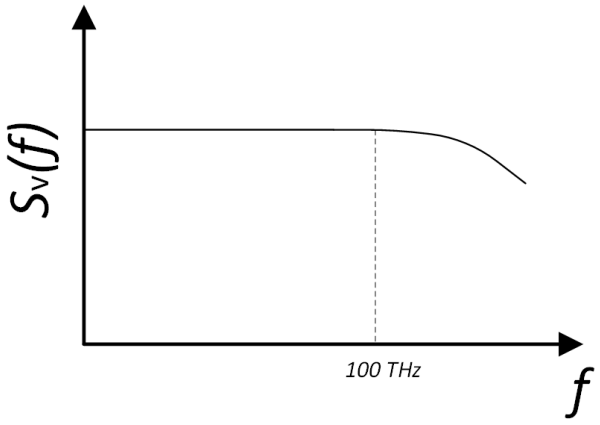

Figure 17: Power spectral density of White noise

2.5.2 NOISE IN MOS TRANSISTORS

There are three main noise contributors in MOS transistors, thermal channel noise, flicker noise and induced gate noise. The flicker noise is inversely proportional to the frequency of operation, thus in high frequency systems, such as in RF low noise amplifiers, flicker is not as relevant as the other two noise contributors, and it is not taken account in the noise calculations.

The thermal channel noise can be modeled as a current source as seen in figure 18. The power spectral density of the thermal noise is [4]:

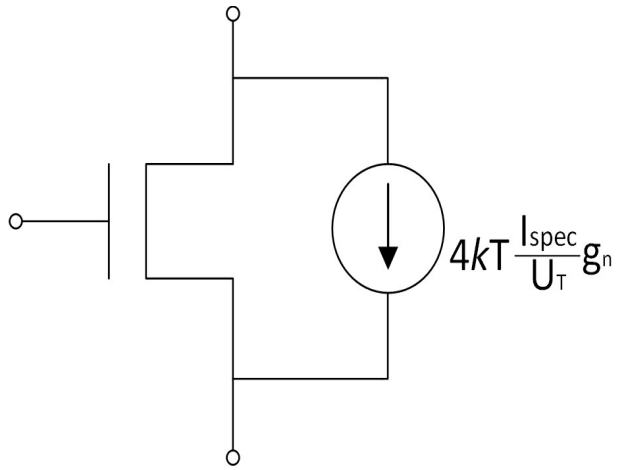

Figure 18: Thermal channel noise of a MOSFET modeled as a current source

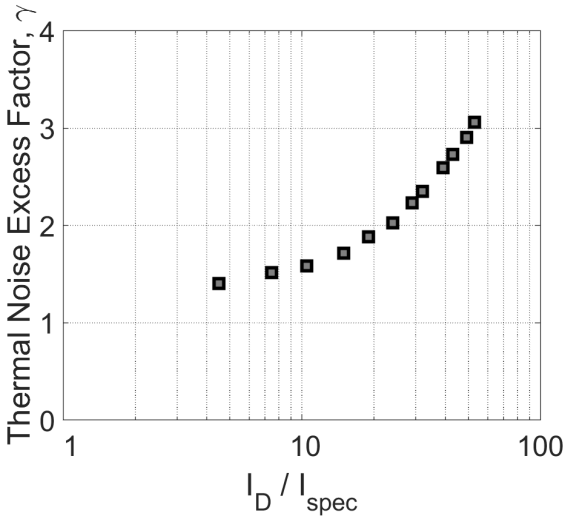

Figure 19: Thermal excess noise factor of 100nm devices as a function of the normalized drain current measured in [4]

$$S_{id} = 4\,k\,T\,\frac{I_{spec}}{U_T}\,g_n \quad (2.49)$$

where g_n is the normalized thermal noise conductance, and is expressed in terms of the normalization current I_{spec}, and the thermal voltage U_T:

$$I_{spec} = 2nU_T^2\,\mu\,C'_{ox}\,\frac{W}{L} = I_0\,\frac{W}{L} \quad (2.50)$$

The specific current I_{spec} depends on the slope factor n [3], and the low-field mobility μ, while I_0 is called the technology current. The parameter C'_{ox} is the gate oxide capacitance per unit area and it is defined as $C'_{ox} = \varepsilon_{ox}/\tau_{ox}$, where ε_{ox} is a constant that defines the permittivity of silicon dioxide, and τ_{ox} is the gate oxide thickness. Finally W and L are the transistor's width and length respectively.

The thermal noise conductance g_n is can be expressed as $g_n = \delta \cdot g_{d0}$, where the parameter δ is called thermal noise parameter. g_{d0} is the output conductance g_{ds} at $V_{DS}=0$. Another critical noise parameter in literature is the excess noise factor γ that is defined as $\gamma = g_n/g_m$. From a circuit design point of view the excess noise factor is of major importance for the noise performance of circuits, since it represents the noise that is generated at the drain of the transistor, for a given transconductance. In the low noise amplifier design, for example, γ is used to determine the minimum noise factor of a common source topology[1]. The value of γ is typically 2/3 for long channel devices, and may rise up to 3 in short channel devices as seen in figure 19.

At high frequency, the local channel voltage fluctuations due to thermal noise couple to the gate through the oxide capacitance and cause an induced gate current to flow. In saturation, most of the channel charge is located on the source side, and hence, the noise current can be modeled by a single noisy current source S_{ing} connected in parallel with the MOSFET's gate to source capacitance with a power spectral density given by :

1 The definition of noise factor can be seen in chapter 3

$$S_{ing} = \frac{4}{5} k T \gamma \delta_g \frac{(C_{gs} \cdot \omega)^2}{g_m} \quad (2.51)$$

Expression (2.51) shows the strong correlation of the induced gate noise with the gate to source capacitance and the frequency of operation ($\omega = 2\pi f$). The parameter δ_g is introduced which is defined as the coefficient of gate noise, and is typically two times the value of γ.
The gate noise is partially correlated with the drain noise due to the white noise, with a correlation coefficient c given by

$$c = \frac{S_{ing} S_{id}^*}{\sqrt{S_{ing} S_{id}}} \quad (2.52)$$

with *c* approximately equal to *j0.4*.

Finally another noise contributor is introduced to the design from the gate resistance. A (distributed) gate resistance R_g generates a noise voltage $S_{vg} = 4kT\ R_g$, according to section 2.5.1, which transfers to the drain current via the transconductance g_m, adding a contribution to the drain current noise [8]. The value of the gate resistance is given by

$$R_g = \frac{W}{L} R_{sh} \quad (2.53)$$

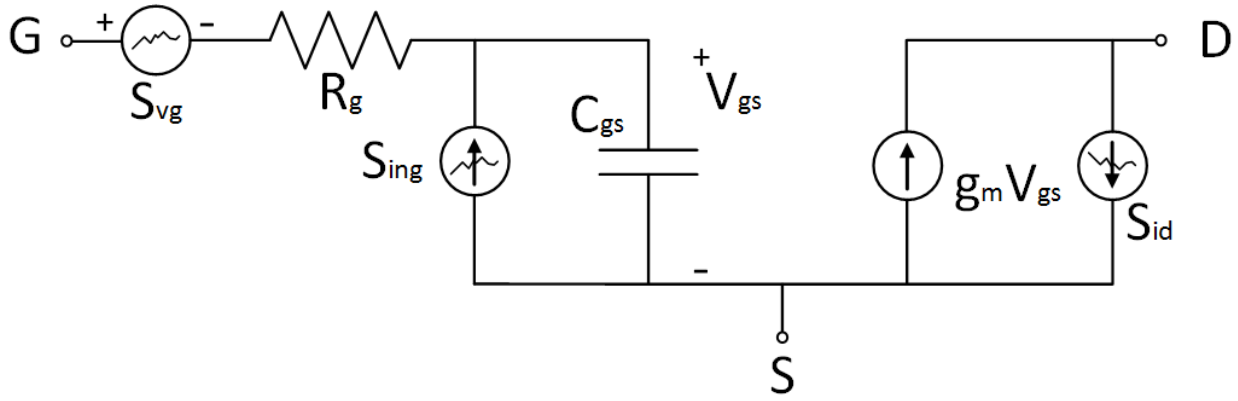

Figure 20: Noise sources in the MOS transistor in saturation

where R_{sh} denotes the sheet resistance of the gate. Figure 20 shows a simplified small signal equivalent of a MOS transistor in saturation, with the noise sources that have been introduced in this chapter. The gate resistance can be lowered by using transistors with multiple *fingers,* as well as double-sided gate contacting.

2.5.3 NOISE DUE TO OTHER COMPONENTS

While designing an RF building block, one must have in mind that each component in the circuit contributes to the overall noise factor. Inductors, capacitors, transmission lines etc, are not ideal, thus, each of them is accompanied by a parasitic resistance. In section 2.5.1, it can be seen that resistors produce noise due to temperature.

An ideal inductor would have no resistance or energy losses. However, real inductors have winding resistance from the metal wire. Since the winding resistance appears as a resistance in series with the inductor, it is often called the *series resistance*. The inductor's series resistance converts electric current through the coils into heat, thus causing a loss of inductive quality. The *quality factor* or *Q* of an inductor is the ratio of its inductive reactance to its resistance at a given

frequency, and is a measure of its efficiency. The higher the Q factor of the inductor, the closer it approaches the behavior of an ideal, lossless, inductor. High Q inductors are used with capacitors to make resonant circuits in radio transmitters and receivers. The Q factor of an inductor can be found through the following formula, where L is the inductance, R_{ind} is the inductor's effective series resistance, ω is the radian operating frequency (also called angular frequency), and the product ωL is called the inductive reactance:

$$Q = \frac{\omega L}{R_{ind}} \quad (2.54)$$

The expression (2.54) shows that the series resistance of an inductor is $R_{ind} = \omega L / Q$, and the noise contribution is $S_{ind} = 4kT\, R_{ind}$.

The same principle applies to the capacitors as well. The Q factor of a capacitor can be found through the following formula:

$$Q = \frac{1}{\omega C R_{cap}} \quad (2.55)$$

where C is the capacitance. The series resistance of a capacitor is $R_{cap} = 1/\omega CQ$, and the noise contribution is $S_{cap} = 4kT\, R_{cap}$.

2.5.4 RF NOISE PARAMETERS

The fundamental noise performance parameter is the *noise factor* (F), which is defined as the ratio of the total output noise power to the output noise due to input source. If the noise factor is expressed in decibels it is called *noise figure* (NF). A two port noisy amplifier can be represented by the same noiseless two port network with an input referred noise voltage source v_n having a PSD

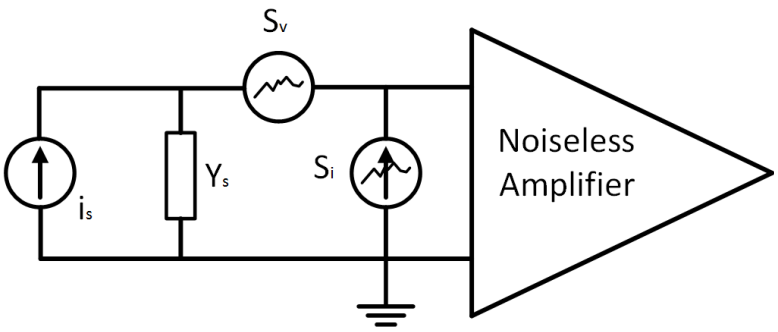

Figure 21: Input referred voltage and current noise sources of a noiseless amplifier

of $S_v = 4\,k\,T\,R_n$ where R_n is a fictitious equivalent noise resistance that represents $\overline{v_n^2}$, and an input referred noise current source in having a PSD of $S_i = 4\,k\,T\,G_n$ (figure 21), and just like R_n, G_n is an equivalent noise conductance that represents $\overline{i_n^2}$. The general representation of the noise factor is :

$$F = F_{min} + \frac{R_n}{G_s}\left[(G_s - G_{opt})^2 + (B_s - B_{opt})^2\right] \quad (2.56),$$

where G_s is the source conductance, F_{min}, is the lowest possible noise factor and Y_{opt} is the optimum

source admittance for minimum noise. Normalizing (2.56) to the input impedance Z_0 leads to :

$$F = F_{min} + \frac{r_n}{g_s}|y_s - y_{opt}|^2 \quad (2.57)$$

Using the reflection coefficients the parameters y_s and y_{opt} are :

$$y_s = \frac{1-\Gamma_s}{1+\Gamma_s} \quad (2.58) \text{ and } y_{opt} = \frac{1-\Gamma_{opt}}{1+\Gamma_{opt}} \quad (2.59)$$

The minimum noise factor is achieved only when a particular reflection coefficient, Γ_{opt} is presented to the input. So $\Gamma_s=\Gamma_{opt}$, leads to the minimum noise figure for the amplifier.

CHAPTER 3

LOW NOISE AMPLIFIER DESIGN

The purpose of this chapter is to introduce essential definitions of the LNA design, and to provide an overview of the essential topologies as well as outline their strengths and weaknesses. The low noise amplifier (LNA) is an integral part of the transceiver chain. It is usually the first component in the receiver chain, only preceded by a band-pass filter in some applications. The main purpose of the low noise amplifier is to increase the power of the attenuated input signal, while at the same time minimizing the noise added to the chain by the circuit itself. In other words, the scope of a LNA is to provide high enough gain to the input signal to enable the signal to tolerate the noise of the subsequent stages while contributing as little noise as possible. Therefore, one of the key design goals for the LNA is a low noise contribution to the input signal, together with a good impedance matching to the signal source, a sufficiently large output signal dynamic range and certainly, a low power consumption. Since LNAs are being used as the input stage of the receivers, they must be tuned, or be tunable, to the carrier frequency of the transmitter. For that reason, low noise amplifiers are inherently considered to be *tuned* amplifiers.

The figure 22 shows a typical transceiver front-end architecture, and the position of the LNA in the chain. According to "Friis equation" the total noise figure in a cascade of stages, like the receiver of figure 22, is given by :

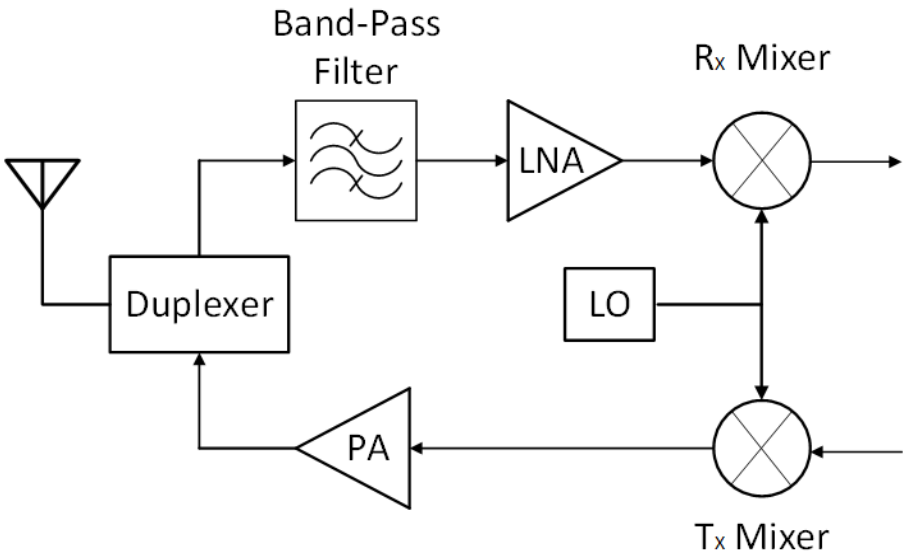

Figure 22: A typical transceiver front-end with homodyne (zero-IF) receiver

$$NF = 1 + (NF_1 - 1) + \frac{NF_2 - 1}{A_{P1}} + \ldots + \frac{NF_m - 1}{A_{P1} \ldots A_{P(m-1)}} \quad (3.1)$$

N_{Fm} and A_{Pm} are the noise figure and power gain of stage m. Equation (3.1) suggests that the noise contributed by each stage decreases as the gain of the preceding stages increases. Thus, the first stage plays the most critical role in a receiver chain.

3.1 LNA REQUIREMENTS

3.1.1 NOISE

The noise figure *(NF)* of the LNA directly adds to that of the receiver. The attenuated signal received at the input not only consist of the signal sent from the transmitter but also of the noise signal generated by the antenna. To obtain a sufficiently high level of signal power with reasonable signal-to-noise ratio (SNR), the noise produced by the amplifier should be kept as low as possible. The noise performance of an amplifier is expressed by its noise factor *(F)*, which is the signal-to-noise ratio at its input divided by the signal-to-noise ratio at its output. The noise figure is the noise factor expressed in decibels ($NF = 10 \times \log F$). Ideally the noise factor of a noiseless LNA equals unity. The low noise required for LNAs, limits the choice of circuit topology. This means that in cases of very strict noise specifications, some existing LNA topologies become inappropriate.

3.1.2 MATCHING

There are three different types of matching in an RF building block, namely impedance matching, power matching, and noise matching. Impedance matching is a term that is frequently used in the area of transmission lines. A transmission line is characterized by an impedance Z_c. Suppose the line is terminated with an impedance Z. A voltage wave V^+ traveling across the line will be partially reflected at the end of the line depending on the termination impedance. The reflected wave V^- is given by $V^- = \Gamma \cdot V^+$, where

$$\Gamma = \frac{Z - Z_c}{Z + Z_c} \quad (3.2)$$

Note that Γ is a complex number comprising both the amplitude ratio and the phase turn. If $Z = Z_c$ then $\Gamma = 0$ and no reflection occurs. The parameter Γ is called input return loss and expresses the quality of the input match.

Power matching is in essence not related to impedance matching. The origin of power matching lies in the fundamental quest for energy efficiency. Suppose a voltage source (voltage V_S), with a source impedance Z_S, drives a load impedance Z_L. From basic circuit theory, it can be shown that the value of Z_L that maximizes the power dissipation in the load is given by $Z_L = Z_S^*$.

Noise matching is completely unrelated to both previous types of matching. The origin here is the quest for good *SNR* and hence low noise figure. For a given two-port, a noise match is obtained when the impedance of the source driving the two-port minimizes the noise figure of the resulting system. This occurs when the source admittance is equal to an optimum admittance Z_{OPT}.

The input signal source of the LNA is usually a band-select filter or an antenna. A band-select filter is typically designed and characterized with a standard termination of 50 Ω. If the load impedance seen by the filter deviates from 50Ω, then the filter may exhibit performance degradations such as loss and ripple. In the absence of a filter, the antenna directly provides the

incoming signal to the LNA, and it is also designed for a certain real load impedance, typically equal to 50 Ω. Thus, in order to achieve input matching, the LNA must have an input impedance of 50 Ω (when $\Gamma = 0$). Note that in this case, power matching is identical to impedance matching. Poor matching at the receiver input causes reflections, loss and possibly voltage attenuation.

In similar fashion, the output impedance matching should also be achieved when the LNA is considered a standalone circuit. Usually to facilitate measurements, output impedance is also set to 50 Ω. Otherwise, the input impedance of the LNA, should match that of the mixer.

3.1.3 GAIN

The gain of the LNA must be large enough to minimize the noise contribution of subsequent stages (Eq. 3.1), specifically the downconversion mixer(s). The choice of high gain though, leads to a compromise between the noise figure and the linearity of the receiver as higher gain makes the nonlinearity of the subsequent stages more pronounced. In modern RF design, the LNA directly drives the downconversion mixer with no impedance matching between the two. Thus it is more meaningful and simpler to perform the chain calculations in terms of voltage gain, rather than power gain, of the LNA. However, in the case of a 50 Ω input and output impedance matched LNA, voltage and power gain are the same. Several types of power gain can be found in literature and are commonly used in LNA design. *Transducer power gain*, G_T, is the ratio of the power delivered to the load, to the power available from the source. *Operating power gain*, G_P, is the ratio of the power delivered to the load, to the power absorbed at the input. *Available power gain*, G_{AV}, is the ratio of the available output power to the available power of the source. The simplified expressions for all the power gain expressions in terms of S-parameters can be found in equations 3.3 through 3.5. In ,most cases the LNA power gain is represented by the transducer power gain, and it is equal to S_{21}^2.

$$G_T = |S_{21}^2| \quad (3.3) \qquad G_P = \frac{1}{1-|S_{11}|^2}|S_{21}|^2 \quad (3.4) \qquad G_{AV} = |S_{21}|^2 \frac{1}{1-|S_{22}|^2} \quad (3.5)$$

3.1.4 STABILITY AND REVERSE ISOLATION

Low noise amplifiers may become unstable due to ground and supply parasitic inductances from the packaging. Feedback paths from the output to the input may also lead to instability issues for certain combinations of input and output impedances. The “Stern stability factor”, K, is often used to describe the stability of amplifier circuits. It is defined as :

$$K = \frac{1+|\Delta|^2-|S_{11}|^2-|S_{22}|^2}{2|S_{21}||S_{12}|} \quad (3.6)$$

The parameter Δ is the determinant of the S-parameters matrix and equals $\Delta = S_{11}S_{22} - S_{12}S_{21}$. When K > 1 and Δ < 1 the circuit is unconditionally stable, i.e. it does not oscillate with any combination of source and load impedances. K should remain greater than one not only at the LNA operation frequency but at all frequencies. Equation (3.6) implies that as coupling (S_{12}) decreases or reverse isolation ($-S_{12}$) increases, stability improves. Reverse isolation is also important for the forward gain of the LNA. Often a low inverse isolation reduces the signal efficiency and thus LNA gain.

3.1.5 LINEARITY

In most applications, the LNA does not limit the linearity of the receiver. Owing to the cumulative gain through the receive chain, the latter stages, e.g. the baseband amplifiers or filters, tend to limit the overall input *IP3* or *P1db*. We therefore design and optimize LNAs with little concern for their

linearity. An exception to the above rule arises in "full-duplex" systems, i.e. applications that transmit and receive simultaneously. The linearity of the LNA also becomes critical in wideband receivers that may sense a large number of strong interferes.

3.1.6 POWER DISSIPATION

The need for circuits consuming less power becomes more intense with technology scaling. However, as the supply voltage is reduced, the available voltage headroom may become too small to design circuits with sufficient signal integrity at reasonable power consumption. The LNA consumes only a small fraction of the overall receiver power, however, when the power is of primary interest, e.g. in portable devices, power dissipation should be taken into strong consideration during the design process. For LNA design, power dissipation has to be considered along with figures of merit, such as noise, linearity, gain, and trade-offs among them should be handled efficiency by circuit designers.

3.2 LNA TOPOLOGIES

There is a wide range of different low noise amplifier topologies that fulfill the principal requirements that have been analyzed above. Each topology has its strengths and its weaknesses, providing the analog designer with different solutions according to the specifications of the application. This section will present the basic topologies that are commonly used, and specify the trade-offs in each circuit.

3.2.1 COMMON SOURCE LNA

We begin the low noise amplifier topology analysis, with the one transistor common source topology, with resistive load shown in figure 23(a). The capacitance C_F represents the gate-drain overlap capacitance. At very low frequencies the resistance R_D is much smaller than the impedances C_F and C_{gs}, and the input impedance is roughly equal to $[(C_{gs}+C_F)\,j\omega]^{-1}$. At very high frequencies however, the capacitance C_F shorts the gate and drain terminals of the transistor M_1, yielding an input resistance equal to $R_D \| (1/g_m)$. More generally the real and imaginary parts of the input impedance are, respectively equal to :

$$\Re\{Y_{\text{in}}\} = R_D C_F \omega^2 \frac{C_F + g_m R_D (C_L + C_F)}{R_D^2 (C_L + C_F)^2 \omega^2 + 1} \quad (3.7)$$

$$\Im\{Y_{\text{in}}\} = C_F \omega \frac{R_D^2 C_L (C_L + C_F) \omega^2 + 1 + g_m R_D}{R_D^2 (C_L + C_F)^2 \omega^2 + 1} \quad (3.8)$$

Input matching in high frequencies cannot be achieved because the input impedance is purely capacitive. In order to overcome this, we have to employ a simple resistive termination at the input as shown in figure 23(b). In this realization of the common source LNA, the transistor M_1 and the resistor R_D, provide the required noise figure and gain, the resistor R_P is placed in parallel with the input to provide $\Re\{Z_{\text{in}}\} = 50\,\Omega$, and an inductor is interposed between R_S and the input to cancel out the capacitive effects and provide $\Im\{Z_{\text{in}}\} = 0$. The input impedance is equal to $Z_{\text{in}} = \frac{R_S}{1 + j\omega/\omega_p}$, where ω_P is equal to $\omega_P = \frac{1}{R_S(C_{gs} + M C_{gd})}$.The noise contribution of the input resistors is calculated as $S_{resin} = 4kT\,(R_S \| R_P)$. The parameter M accounts for the *Miller effect* and is equal to

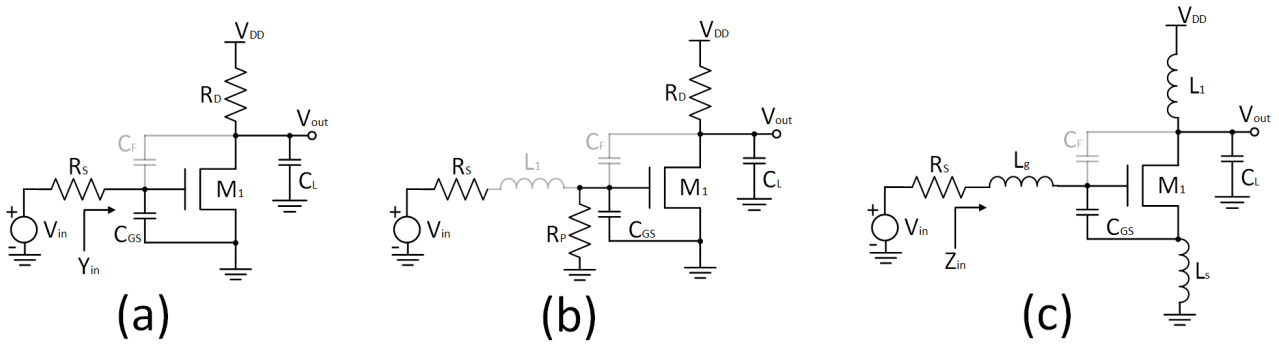

Figure 23: Common source LNA topologies with (a) resistive load, (b) resistive load and matching network, (c) inductive load and matching network

$1+g_m R_D$.

For R_P roughly equal to R_S the noise figure of the circuit is substantially high. In addition, the resistor R_S, causes a high DC voltage drop. The voltage of the common source transistor should be high enough to ensure operation in the saturation region, that is, the drain-source voltage should be at least $V_{DS,sat} = V_{GS} - V_T$, where V_T is the threshold voltage of the transistor. Therefore, R_L is limited to : $R_L < \frac{V_{DD} - V_{DS,sat}}{I_D}$.

The key point of the foregoing study is that the LNA must provide a 50 Ω input resistance without the thermal noise of a physical 50 Ω resistor, and avoid the resistive load that limits the voltage gain:

$$|A_V| = g_m R_D \approx \frac{2 I_D}{V_{GS} - V_T} \cdot \frac{V_{RD}}{I_D} = \frac{2 V_{RD}}{V_{GS} - V_T} \quad (3.9)$$

where V_{RD} denotes the DC voltage drop across the resistor R_D.

In order to circumvent the trade-off expressed by equation (3.9) and also operate in higher frequencies, the common source stage can incorporate an inductive load. Illustrated in figure 23(b), such a topology operates with low supply voltages, since the inductor sustains a significantly smaller DC voltage drop than the resistor does (for an ideal inductor the voltage drop is 0). Moreover, the inductor L_1, resonates with the total capacitance at the output node, affording much higher operation frequency than does the resistive loaded counterpart.

For the input matching requirement the input inductor L_g is placed in order to cancel out the gate-drain capacitance C_F, and the inductor L_S is placed at the source of the transistor to resonate with C_{gs}, in order to cancel out the gate-source capacitance figure 23(c). The circuit of figure 23(c)

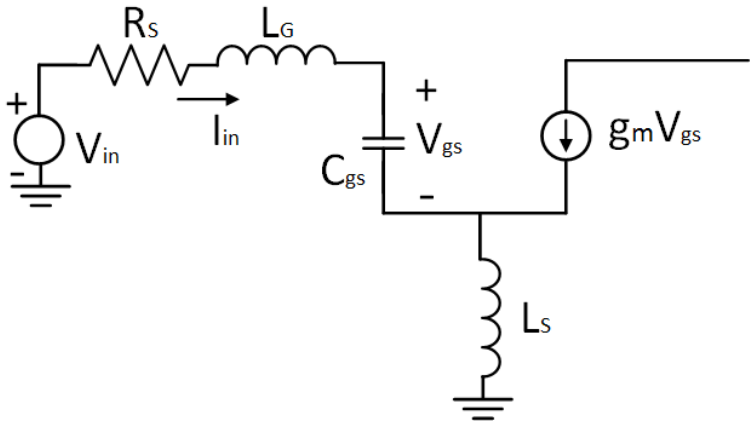

Figure 24: Small signal equivalent of figure 23(c)

is often called common source LNA with inductive degeneration.

In order to calculate the input impedance, we use the small-signal equivalent circuit of figure 24. Using Kirchhoff's voltage law in the circuit we have :

$$V_{\text{in}} = I_{\text{in}} L_G \cdot s + I_{\text{in}} \frac{1}{C_{gs} \cdot s} + (I_{\text{in}} + g_m V_{gs}) L_S \cdot s \quad (3.10)$$

Dividing the right and left parts of equation (3.10) by the input current I_{in}, we get the input impedance Z_{in} :

$$Z_{\text{in}}(s) = \frac{V_{\text{in}}}{I_{\text{in}}} = L_G \cdot s + \frac{1}{C_{gs} \cdot s} + (1 + \frac{g_m V_{gs}}{I_{\text{in}}}) L_S \cdot s \quad (3.11)$$

The factor V_{gs}/I_{in} is equal to $1/C_{gs} \cdot s$, replacing this in equation (3.11) we get :

$$Z_{\text{in}}(s) = \frac{V_{\text{in}}}{I_{\text{in}}} = L_G \cdot s + \frac{1}{C_{gs} \cdot s} + L_S \cdot s + \frac{g_m L_S \cdot s}{C_{gs} \cdot s} \quad (3.12)$$

$$Z_{\text{in}}(s) = \frac{V_{\text{in}}}{I_{\text{in}}} = L_G \cdot s + \frac{1}{C_{gs} \cdot s} + L_S \cdot s + \omega_T L_S \quad (3.13)$$

In equation (3.13) the parameter ω_T is the transit frequency of the transistor, i.e. the frequency of the MOST at which the small-signal current gain drops to unity while the source and drain terminals are held at ground.

In order to have perfect input matching, at the operating frequency ω_0 the imaginary part of (3.13) should be equal to zero, while the real part should be equal to the input impedance R_S. In equation (3.13) the only purely ohmic part is the $\omega_T L_S$, thus we must chose an inductor such as $L_S = R_S/\omega_T$. If the value of the inductance is impractical, then we can artificially reduce the ω_T, by inserting a capacitor in shunt with C_{gs}. From the imaginary part we get :

$$\Im\{Z_{\text{in}}(s)\} = 0 \Leftrightarrow (L_G + L_S) \cdot s + \frac{1}{C_{gs} \cdot s} = 0 \quad (3.14)$$

Substituting s with $j\omega_0$ we get:

$$(L_G + L_S) \cdot j\omega_0 + \frac{1}{C_{gs} \cdot j\omega_0} = 0 \quad (3.15)$$

It is important to introduce a quantity for the LNA, namely the effective quality factor of the amplifier input circuit, Q_{in}. The effective quality factor is defined as :

$$Q_{\text{in}} = \frac{V_{gs}}{V_{\text{in}}} = \frac{1}{\omega_0 C_{gs} R_S} \quad (3.16)$$

The quality factor is a convenient way of representing the relationship between the reactive part of the input network and the combination of the signal source resistance and the parasitic gate resistance. A high Q_{in} is beneficial for reducing channel current noise.

Continuing the study of the common source LNA with inductive degeneration, we now calculate the noise figure of the circuit, excluding the parasitic noise sources of the inductors, and capacitors for simplicity. The noise of the MOST M_1, is represented by I_{n1} in figure 25. For now, we assume the output of interest is the current I_{out} :

$$I_{out} = g_m V_{gs} + I_{n1} \quad (3.17)$$

Also, since L_S sustains a voltage of $L_S \cdot s(I_{out} + V_{gs} C_{gs} \cdot s)$, a KVL around the input loop yields

$$V_{\text{in}} = (R_S + L_G \cdot s) V_{gs} C_{gs} \cdot s + V_{gs} + L_S \cdot s (I_{out} + V_{gs} C_{gs} \cdot s) \quad (3.18)$$

Substituting for V_{gs} from (3.17) gives :

$$V_{\text{in}} = I_{out} L_S \cdot s + \frac{(L_S + L_G) C_{gs} \cdot s^2 + 1 + R_S C_{gs} \cdot s}{g_m} (I_{out} - I_{n1}) \quad (3.19)$$

The input network is designed to resonate at the frequency of interest, ω_0 . That is

$(L_G+L_S)C_{gs}=\omega_0^{-2}$. We therefore, obtain :

$$V_{\text{in}}=I_{out}\left(jL_S\omega_0+\frac{jR_SC_{gs}\omega_0}{g_m}\right)-I_{n1}\frac{jR_SC_{gs}\omega_0}{g_m} \quad (3.20)$$

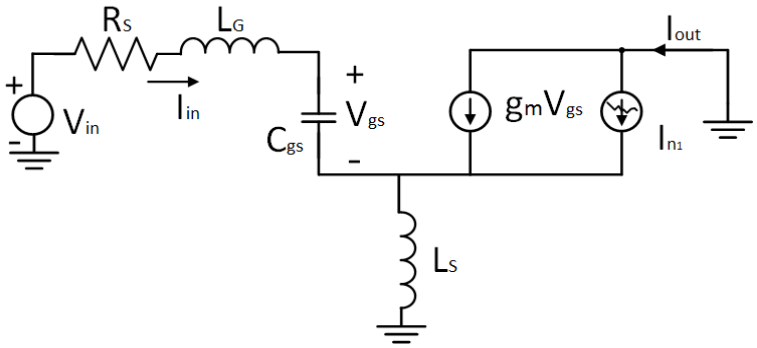

Figure 25: Equivalent circuit for NF calculation

The coefficient I_{out} represents the transconductance gain of the circuit including R_S :

$$\left|\frac{I_{out}}{V_{\text{in}}}\right|=\frac{1}{\omega_0\left(L_S+\frac{R_SC_{gs}}{g_m}\right)} \quad (3.21)$$

Now recall from equation (3.13) that, for input matching, $g_mL_S/C_{gs}=R_S$. Since $g_m/C_{gs}=\omega_T$:

$$\left|\frac{I_{out}}{V_{\text{in}}}\right|=\frac{\omega_T}{2\omega_0}\cdot\frac{1}{R_S} \quad (3.22)$$

Interestingly, the transconductance of the circuit remains independent of L_S, L_G and g_m, so long as the input in matched.

Setting V_{in} to zero at equation (3.20), we compute the output noise due to the transistor M_1 :

$$\left|I_{n,out}\right|_{M_1}=\frac{|I_{n1}|}{2} \quad (3.23)$$

and hence,

$$S_{M_1}=\left|\overline{I_{n,out}^2}\right|_{M_1}=kT\gamma g_m \quad (3.24)$$

Dividing the output noise current by the transconductance of the circuit and by $4kTR_S$ and adding a unity to the result, we arrive at the noise figure of the circuit :

$$NF=1+g_mR_S\gamma\left(\frac{\omega_0}{\omega_T}\right)^2 \quad (3.25)$$

It is important to bear in mind that this result holds only at the input resonance frequency and if the input is matched. Interestingly, for a fixed operating frequency, the noise factor and thus the noise figure reduce with technology scaling, since the transit frequency increases.

In the foregoing analysis, the induced-gate noise has not been taken into account in noise calculations, for simplicity reasons. However, as the operating frequency increases, induced-gate noise becomes a remarkable part of the total noise of the input device.

The voltage gain A_V of the topology is equal to the product of the transconductance gain and the output (load) resistance R_L. Using equation (3.22), knowing that $I_{out}=V_{out}/R_L$, we have :

$$A_V=\frac{V_{out}}{V_{\text{in}}}=\frac{\omega_T}{2\omega_0}\cdot\frac{R_L}{R_S} \quad (3.26)$$

In equation (3.26), we can see that the voltage gain increases with the transit frequency ω_T, thus, for a specific operating frequency moving towards deep submicron technologies, improves the

performance of the low noise amplifier. Accordingly, the power gain A_P, of the circuit is given by equation (3.27) :

$$A_P = \frac{V_{out}^2 / R_L}{V_{in}^2 / R_S} = \left(\frac{\omega_T}{2\omega_0} \cdot \frac{R_L}{R_S} \right)^2 \frac{R_L}{R_S} = \left(\frac{\omega_T}{\omega_0} \right)^2 \frac{R_L}{4 R_S} = Q_{in}^2 g_m^2 R_L R_S \quad (3.27)$$

As mentioned before, a high quality factor Q_{in} leads to the reduction of channel current noise. However, in a design where the gate induced noise is not taken into account, one may end up with a large Q_{in} , and a noise that is dominated by the gate induced noise and not the channel thermal noise. To resolve the problem, the additional capacitance that can be inserted in shunt to the intrinsic gate capacitance C_{gs}, apart from artificially reducing the ω_T as mentioned before, it also decouples Q_{in} from C_{gs}, allowing for an adjustable reduction of Q_{in} for any given C_{gs}. This is crucial since induced gate noise increases with the square of C_{gs} as seen from equation (2.51). This method achieves noise reduction, without deteriorating power consumption.

3.2.2 CASCODE COMMON SOURCE LNA

The cascode common topology, is another variation of the common source low noise amplifier that has been analyzed in the previous section. This realization, introduces a second device in the circuit, that operates as a current source, as seen in figure 26. The analysis for the common source transistor is the same as in section 3.2.1. Additionally, the transistor M_2, has a minor influence on the noise behavior of the LNA, and its contribution to the total noise is disregarded in the analysis, thus it is safe to assume that the noise figure of the cascode LNA topology is also expressed by equation (3.25).

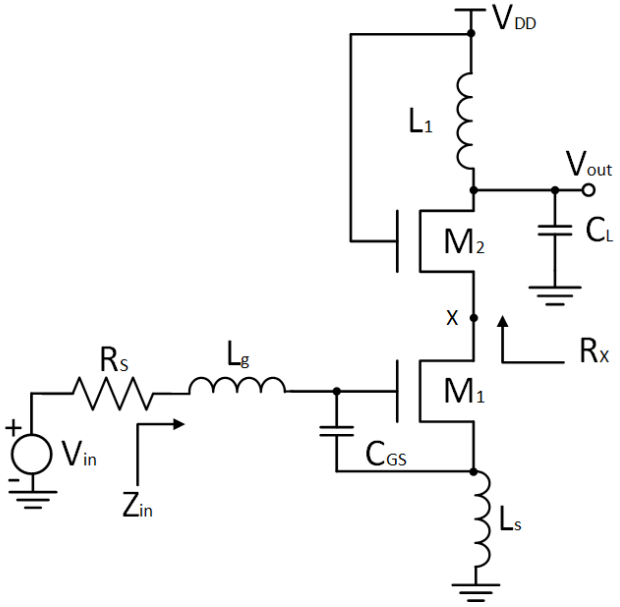

Figure 26: Cascode common source LNA with inductive degeneration

As mentioned previously the common source topology suffers from the added Miller capacitance. For this reason, the common source topology strongly limits the frequency response and gives rise to a very poor reverse isolation The advantage of the cascode topology over the

common source of figure 23(c), is that it significantly decreases the Miller effect, since the Miller capacitance is now decoupled from the gain of the circuit. Decreased Miller effect leads to higher operating frequencies, and higher reverse isolation, which in turn leads to increased stability.
The disadvantage of this topology though, is that the second device, introduces non-linearities into the design, deteriorating the 1-dB compression point. This means that the amplifier should operate for lower input power values in order to output a linear gain as seen in figure 10.

3.2.3 COMMON GATE LNA

The low input impedance of the common gate topology, makes it attractive for low noise amplifier design. Since a resistively loaded stage suffers from the same gain-headroom trade-offs as its common source counterpart, we consider only a common gate circuit with inductive loading. Here, L_1 resonates with the total capacitance at the output node, and R_1 models the losses of the inductor. If channel length modulation[1] is neglected, then $R_{in} = 1/g_m$. Thus, the dimensions and bias current of the common gate transistor is selected as to yield $g_m = 1/R_S = (50\,\Omega)^{-1}$. The voltage gain from X to the output node at the output resonance frequency is then equal to :

$$\frac{V_{out}}{V_X} = g_m R_1 = \frac{R_1}{R_S} \quad (3.28),$$

and hence, $V_{out}/V_{in} = R_1/(2R_S)$.

Let us now determine the noise figure of the circuit under the condition $g_m = 1/R_S$, and at the resonance frequency. Modeling the thermal noise of the transistor M_1 as a voltage source in series with its gate with power spectral density $\overline{V_{n1}^2} = 4kT\gamma/g_m$, and multiplying it by the gain from the gate of M_1 to the output, we have :

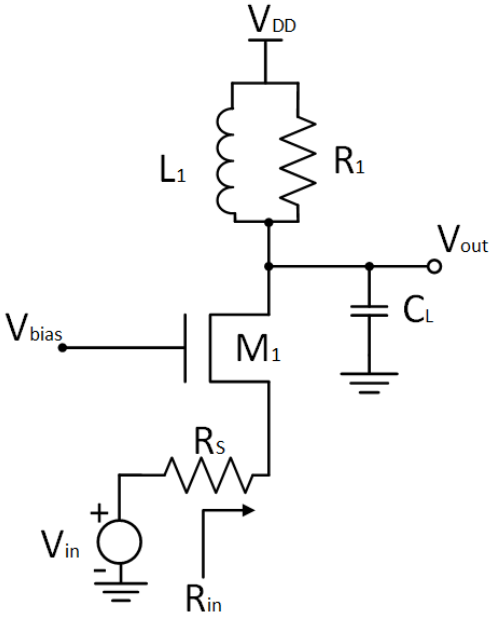

Figure 27: Common gate LNA topology

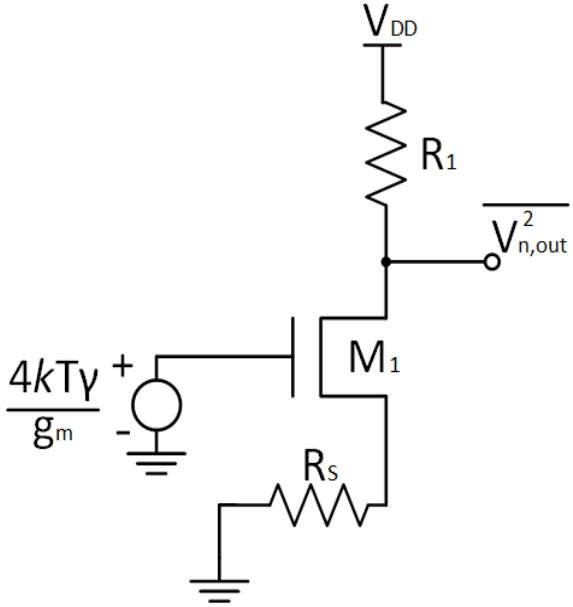

Figure 28: Effect of noise of M1

1 Channel length modulation (CLM) is one of the several second order short-channel effects. CLM is a shortening of the length of the inverted channel region with increase in drain bias for large drain biases. The result of CLM is an increase in current with drain bias and a reduction of output resistance. CLM is important because it decides the MOSFET output resistance.

$$\overline{V_{n,out}^2}\Big|_{M_1} = \frac{4kT\gamma}{g_m}\left(\frac{R_1}{R_S + 1/g_m}\right)^2 = kT\gamma\frac{R_1^2}{R_S} \quad (3.29)$$

The output noise of R_1 is simply equal to $4kT\gamma R_1$. To obtain the noise figure, we divide the output noise due to the transistor M_1 and the resistance R_1, by the gain and $4kT\gamma R_S$ and add unity to the result :

$$NF = 1 + \frac{\gamma}{g_M R_S} + \frac{R_S}{R_1}\left(1 + \frac{1}{g_m R_S}\right)^2 = 1 + \gamma + 4\frac{R_S}{R_1} \quad (3.30)$$

Even if $4R_S/R_1 \ll 1+\gamma$, the noise figure still reaches 3 dB (assuming $\gamma=1$ for simplicity), a price paid for the condition $g_m = 1/R_S$. However, as shown in chapter 2, the value for γ reaches higher values than that of unity in deep submicron technologies. Hence, the high levels of noise figure renders the common gate topology inappropriate for realizing LNAs with very low noise levels. A lower noise figure can be achieved if gm is increased, however this would also produce a lower input resistance. It can be seen that in this topology there is a trade-off between the input matching and the noise figure performance.

The previous analysis, has not taken into consideration the effect of channel length modulation. In deep submicron technologies, CLM significantly impacts the behavior of the common gate stage. As shown in figure 29, the resistance r_o raises the input impedance. Since the drain-source current of M_1 (without r_o) is equal to $-g_m V_X$, the current flowing through r_o is equal to $I_X - g_m V_X$, yielding a voltage drop of $r_o(I_X - g_m V_X)$ across it. I_X also flows through the output tank, producing a voltage drop of $I_X R_1$ at the resonance frequency. Adding this voltage to the drop across r_O and equating the result to V_X, we obtain :

$$V_X = r_O(I_X - g_m V_X) + I_X R_1 \quad (3.31)$$

That is,

$$\frac{V_X}{I_X} = \frac{R_1 + r_O}{1 + g_m r_O} \quad (3.32)$$

If the intrinsic gain, $g_m r_O$, is much greater than unity, then $V_X/I_X \approx 1/g_m + R_1/(g_m r_O)$. However, in todays technology, $g_m r_O$ hardly exceeds 10, and the term $R_1/(g_m r_O)$ may become comparable with

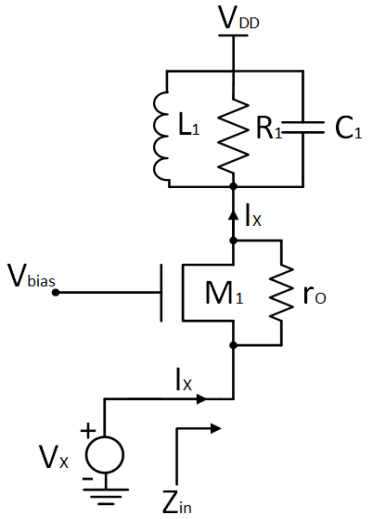

Figure 29: Input impedance of CG topology in presence of CLM resistance r_O

or even exceed the first term $1/g_m$, yielding an input resistance substantially higher than 50 Ω.

With the strong effect of R_1 on the input resistance, we have to equate the actual input resistance to R_S to guarantee input matching :

$$R_S = \frac{R_1 + r_O}{1 + g_m r_O} \quad (3.33)$$

The voltage drop of the source resistance R_S is :

$$V_{R_s} = V_{\text{in}} - V_X \Leftrightarrow I_X R_S = V_{\text{in}} - V_X \quad (3.34)$$

The voltage at the output is equal to the product of the current I_X and the output resistance R_1. Knowing this, we can calculate the voltage gain using the equation (3.34) :

$$\frac{V_{out}}{V_{\text{in}}} = \frac{I_X R_1}{I_X R_S + V_X} = \frac{R_1}{R_S + V_X / I_X} \quad (3.35)$$

Using equation (3.32) :

$$\frac{V_{out}}{V_{\text{in}}} = \frac{R_1}{R_S + \dfrac{R_1 + r_O}{1 + g_m r_O}} = \frac{g_m r_O + 1}{g_m R_S r_O + R_S + R_1 + r_O} R_1 \quad (3.36)$$

Which, from equation (3.33) reduces to :

$$\frac{V_{out}}{V_{\text{in}}} = \frac{g_m r_O + 1}{2\left(1 + \dfrac{r_O}{R_1}\right)} \quad (3.37)$$

If r_O and R_1 are comparable, then the voltage gain is on the order of $g_m r_O/4$, i.e. a very low value. In summary, the input impedance of the common gate stage is too low if CLM is neglected and too high if it is not. In order to alleviate this issue a common technique is to increase the length of the transistor, thus reducing the effect of CLM and raising the achievable $g_m r_O$. Since the device width must also increase proportionally so as to retain the transconductance value, the gate-source capacitance of the transistor rises considerably, degrading the input match and lowering the maximum operating frequency.

3.2.4 CASCODE COMMON GATE LNA

An alternative approach to lowering the input impedance is to incorporate a cascode device as shown in figure 30. Here, the resistance seen when looking into the source of M_2 is given by :

$$R_X = \frac{R_1 + r_{O2}}{1 + g_{m2} r_{O2}} \quad (3.38)$$

The load resistance is now transformed to a lower value by M_1, again according to (3.32) :

$$R_{\text{in}} = \left(\frac{R_1 + r_{O1}}{1 + g_{m2} r_{O2}} + r_{O1} \right) \div \left(1 + g_{m1} r_{O1}\right) \quad (3.39)$$

If the intrinsic gain, $g_m r_O$, is much greater than unity, then

$$R_{\text{in}} = \frac{1}{g_{m1}} + \frac{R_1}{g_{m1} r_{O1} g_{m2} r_{O2}} + \frac{1}{g_{m1} r_{O1} g_{m2}} \quad (3.40)$$

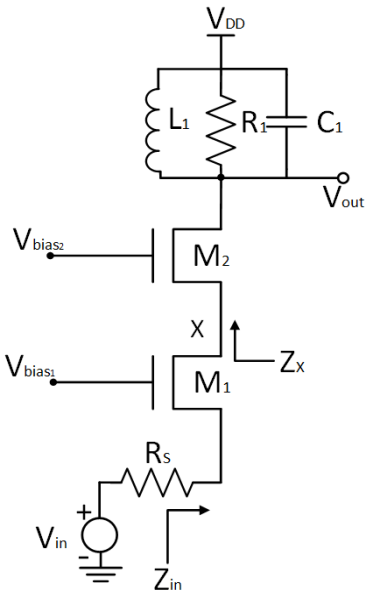

Figure 30: Cascode common gate low noise amplifier

Since the output resistance R_1 is divided by the product of two intrinsic gains, its effect remains negligible. Similarly, the third term in equation (3.40), is much less than the first if g_{m1} and g_{m2} are roughly equal. Thus $R_{in} \approx 1/g_{m1}$.

The addition of the cascode device entails two issues : the noise contribution of M_2 and the voltage headroom limitation due to stacking two transistors. The drain voltage of M_2 begins at V_{DD} and can swing below its gate voltage by as much as V_{T2} (threshold voltage of M_2) while keeping M_2 in saturation, thus we can simply choose V_{bias2} to be equal to V_{DD}. Now the voltage at the node X is equal to $V_X=V_{DD}-V_{GS2}$, allowing a maximum value of $V_{DD}-V_{GS2}+V_{T1}$ for V_{bias1} if M_1 must remain saturated. Consequently, the source voltage of M_1 cannot exceed $V_{DD}-V_{GS2}-(V_{GS1}-V_{T1})$. We say that the two devices consume a headroom of one V_{GS} plus one overdrive $(V_{GS1}-V_{T1})$.

One of the strengths of the cascode common gate topology is its wide bandwidth. This is because the matching is realized by the proper bias of the transistor M1, and not with a matching network that limits the bandwidth, like the ones of the common source topologies. Another strong point, is the high reverse isolation, since the second device offers a high output isolation. Lastly, this design is highly linear, which comes with the trade-offs of low gain values. The disadvantage of the topology is that it cannot achieve good noise performance and input matching at the same time.

CHAPTER 4

FIGURE OF MERIT

The purpose of this chapter is firstly to answer the question what is considered an optimum and later on propose a figure of merit that provides a guideline to initiate the procedure of designing an optimum low noise amplifier. There is a wide range of proposed figures of merit for individual devices such as the transit frequency f_t and the maximum frequency of oscillation (f_{max}). The transit frequency is defined as the frequency at which the extrapolated small-signal current gain of the transistor in common source configuration falls to unity, while the maximum frequency of oscillation, is the frequency at which the extrapolated unilateral power gain becomes unity under the condition of matched source and load impedances. Though f_t and f_{max} are useful for defining the limits on the speed of operation of a device, they do not provide any indication about its power efficiency.

Another common figure of merit is the transconductance efficiency G_m/I_D, which is the ratio of the transconductance over the drain current. This FoM provides enough knowledge about the trade off between gain and current consumption, yet it lacks information about the RF characteristics like the f_t and f_{max}. The FoM that is going to be analyzed in this section incorporates the current efficiency and RF performance metrics, and ultimately can be used to locate the transistor's optimum bias point.

4.1 DEFINITION OF THE FOM

The proposed FoM for LNA design is the $G_m f_t / I_D$. It is a simultaneous representation of the current efficiency and the unity gain frequency, that provides enough information about the operating point of a transistor both in the current-efficiency and the RF point of view. In order to analyze the behavior of the FoM we use normalized parameters of the EKV[1] model [17]. In saturation the normalized drain current i_d is almost equal to the *forward* current i_f and defined as :

$$i_d = i_f = \frac{I_D}{I_{spec}} \quad (4.1)$$

It can be expressed in terms of the normalized inversion charge density at the source, q_s as :

$$i_d = q_s + q_s^2 \quad (4.2)$$

where the normalized inversion charge is defined as :

1 The EKV model is an industry standard mathematical charge-based model of MOSFETs which is intended for circuit simulation and analog circuit design.

$$q_s = \frac{Q_s}{Q_{spec}} \quad (4.3)$$

q_s is in turn related to pinch-off (and hence gate) voltage, via the voltage-charge relation $v_p - v_s = 2q_s + \ln(q_s)$, where v_p is the pinch-off voltage and is defined as $v_p \approx (V_g - V_{T0})/n$, with n being the slope factor. The source transconductance is defined as :

$$g_{ms} = \frac{G_{ms}}{G_{spec}} \quad (4.4)$$

The normalizing quantities are defined as following :

$$Q_{spec} = -2\,n\,C'_{ox}\,U_T \quad (4.5)$$

$$I_{spec} = 2\,n\,\mu\,C'_{ox}\,U_T^2\,\frac{W}{L} \quad (4.6)$$

$$G_{spec} = \frac{I_{spec}}{U_T} \quad (4.7)$$

In these equations the parameter n represents the slope factor, μ the carrier mobility and U_T is the thermal voltage. The transit frequency ft is given as the ratio of the normalized transconductance to the normalized total gate capacitance c_G :

$$f_t = \frac{g_m}{c_G} \quad (4.8)$$

The total gate capacitance c_G , is the sum of the capacitances c_{gs} and c_{gb}, along with the overlap capacitance c_{ov}. c_{gs} and c_{gb} are equal to :

$$c_{gs} = \frac{q_s}{3}\frac{2q_s + 3}{(q_s + 1)^2} \quad (4.9) \qquad c_{gb} = \frac{n}{n-1}\frac{q_s^3 + 3q_s + 3}{3(q_s + 1)^2} \quad (4.10)$$

And finally the proposed figure of merit is defined as :

$$FoM = \frac{g_m f_t}{i_d} \quad (4.11)$$

4.2 THE VELOCITY SATURATION MODEL

The expressions in the previous section have been extrapolated without accounting for the *velocity saturation* effect, thus the experimental results are differ from the theoretical as seen in [2]. Velocity saturation is an important short-channel effect that causes the degradation of the drain current and eventually the source transconductance.

In [2] the authors derive that the normalized drain current accounting the velocity saturation effect is given by the expression :

$$i_{dsat} = \frac{4(q_s + q_s^2)}{2 + \lambda_c + \sqrt{4(1+\lambda_c) + \lambda_c^2(1+2q_s)^2}} \quad (4.11)$$

where λ_c is the parameter that accounts for the velocity saturation effect and is defined as :

$$\lambda_c = \frac{2\,U_T}{E_c L} \quad (4.12)$$

where E_c is the critical longitudinal field and L the transistor's channel length. Additionally the normalized charge at the drain with the velocity saturation effect is given by :

$$q_{dsat} = \frac{2\lambda_c(q_s + q_s^2)}{2 + \lambda_c + \sqrt{4(1+\lambda_c) + \lambda_c^2(1+2q_s)^2}} \quad (4.13)$$

Inverting (4.13) to write q_s as a function of velocity saturated drain current i_d results in :

$$q_s = \frac{\sqrt{i_d^2\lambda_c^2 + 2i_d\lambda_c + 4i_d + 1}}{2} - \frac{1}{2} \quad (4.14)$$

Finally the source transconductance with the velocity saturation effect is written as :

$$g_{ms} = \frac{2q_s}{\sqrt{4(1+\lambda_c) + \lambda_c^2(1+2q_s)^2}} \quad (4.15)$$

It is important to introduce the channel inversion coefficient *IC*. The parameter *IC* provides knowledge about the level of inversion, and it is defined as :

$$IC = \frac{I_D}{I_{spec}} \quad (4.16)$$

Equating this expression with the expression (4.1) it can be seen that the inversion coefficient is the same as the normalized drain current when the transistor operates in the saturation region. The inversion level is divided into three regions, the weak, moderate and strong inversion regions. The next figures illustrate the behavior of the common figures of merit in the different inversion regions. It can be seen that the weak inversion applies for inversion coefficient values less than 0.1, for the moderate inversion it applies that $0.1 < IC < 10$, and finally the strong inversion is found for values larger than 10.

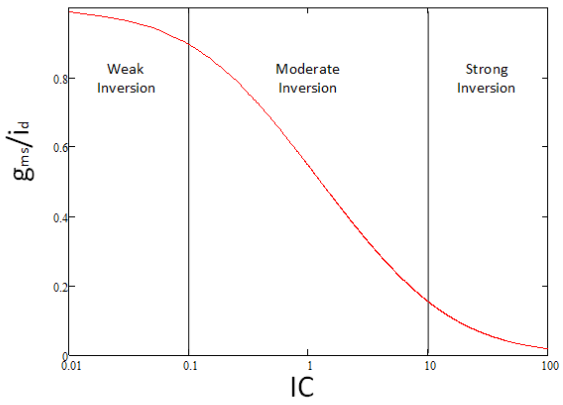

Figure 31: g_{ms}/i_d as a function of the inversion coefficient IC

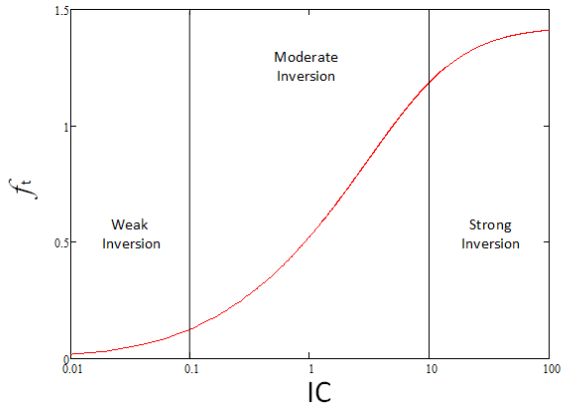

Figure 32: f_t as a function of the inversion coefficient IC

4.3 THE OPTIMUM OPERATING POINT

Figure 31 shows that as the inversion level goes towards the strong inversion the current efficiency drops, while as figure 32 suggests the maximum unit gain frequency rises. These two statements are contradicting, and thus the optimum operating point has to lie in the moderate inversion. This case is supported by [2] and [18].

Additionally the maximum value of the suggested FoM, not only lies in the moderate inversion, but for inversion coefficient equal to $1/\lambda_c$. This is a really convenient observation, since the optimum operating point of the transistor can easily be calculated, and it is a starting point for the design of the RF circuits. Figure 34 shows that the normalized values of the proposed *FoM* of (4.11) as a function of the inversion coefficient *IC*, and as it can be seen the peak of the graph lies in the aforementioned *IC* value. Since λ_c would typically lie between 0.5 and 1 for short channel

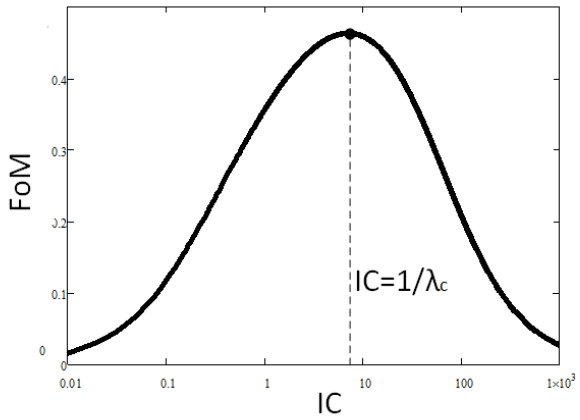

Figure 34: FoM as a function of the inversion coefficient IC

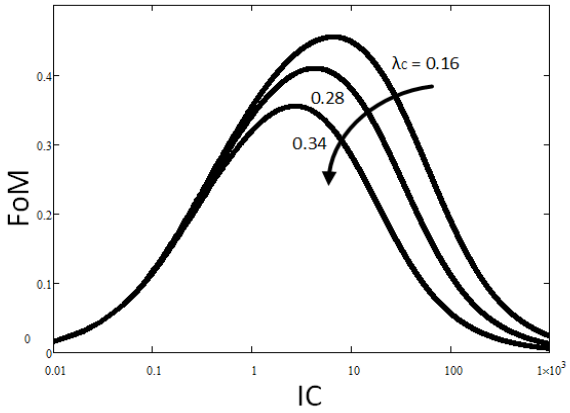

Figure 33: FoM vs IC plotted for different values of λ_C

devices ($1/\lambda_c$ between 2 and 1), the optimum bias point would always lie in moderate inversion region. This reinforces the importance of moderate inversion as an operating region of choice for RF applications where low power operation without a significant degradation of noise and linearity is critical.

The importance of the velocity saturation model is going to be presented in the following figures. The graph of figure 36 clearly shows that if we ignore the effects of the velocity saturation, we overestimate the performance of the design regarding the current efficiency g_{ms}/i_d. Consequently, this case leads to a poor modeling of the figure of merit and we cannot take advantage of the previous analysis, since there is no maximum point in the response of the proposed figure of merit. The result of this faulty approximation can be seen in figure 35, where the analytical plot of the figure of merit shows different behavior than that of the measurements that can been made by the authors in [2].

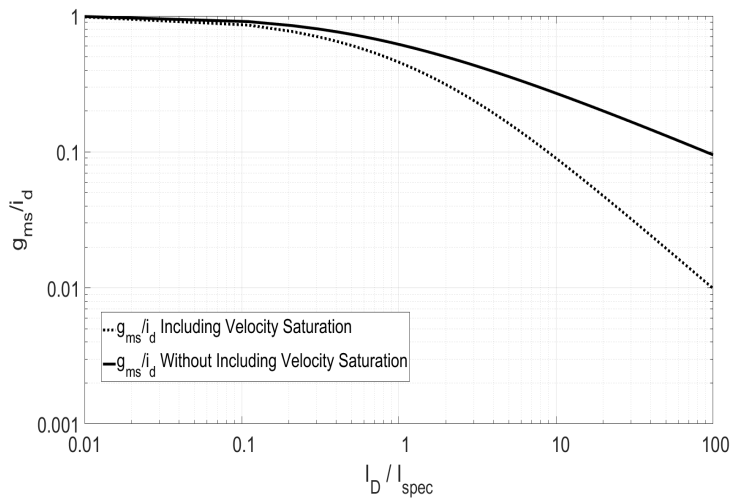

Figure 36: g_{ms}/i_d versus inversion coefficient with and without including velocity saturation

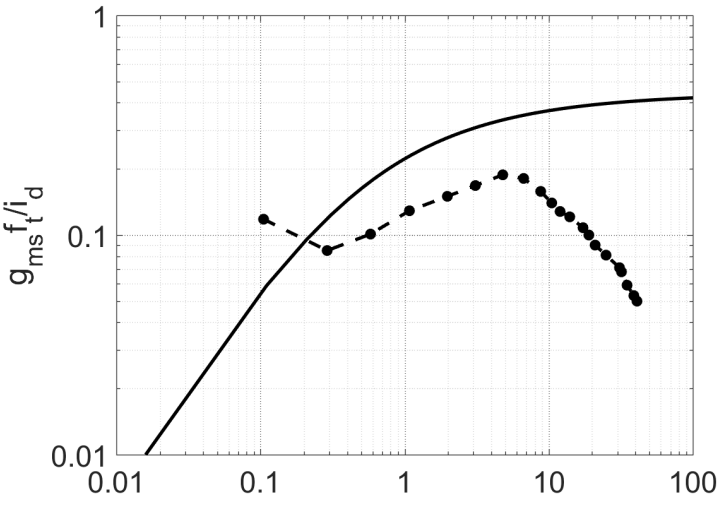

Figure 35: $g_{ms}f_t/i_d$ versus inversion coefficient without including velocity saturation. The dotted line corresponds to the measurements made by [2]

CHAPTER 5

OPTIMIZATION ALGORITHM

Chapter 3 presents the most common LNA topologies, and the basic equations that define the operation of each case. In order to simplify the analysis though, these equations are derived by assuming that the components are ideal, thus they cannot be applied to a real design efficiently. The results of the design would not be the expected ones, and in some occasions not even close to the theoretical calculations. Hence, in the interest of designing a LNA, a method of optimization has to be introduced. The previous chapter answers the question of how to define an optimized operation. The proposed figure of merit (*FoM*) of chapter 4 provides some insight at the operation of the design, and aid the designer with a tool to compare different designs. The next step in the design optimization is to propose an equations model that takes advantage of the aforementioned figure of merit, and try to drive the design into the peak value of the FoM. The last step in the optimization sequence, applies to the non ideal nature of the components. The real behavior of the components is not easy to be modeled, especially that of the inductors, thus a more solid approach is the use of an iterative method to fine tune the design. All these steps are going to be analyzed in this chapter.

5.1 COMMON SOURCE LNA DESIGN METHODOLOGY

Figure 37 shows a variation of the common source with inductive degeneration topology. The optimization algorithm of this section aims to produce a design that applies to this topology. As mentioned previously the external capacitor that appears in this design, decouples the quality factor of the input matching network from the gate to source capacitance, and provides adjustability when an inductor value is not realizable.

In the output node of the circuit, there is an L-C tank, that acts as a resonator. The purpose of the resonator is to tune the amplifier to the frequency of interest. The output inductance, resonates with the transistor's capacitance, the output capacitance and the capacitance of the next stage. When the gain of the amplifier cannot be tuned to the center frequency due to inductor limitations, e.g. the process design kit does not provide the necessary inductor values, an additional capacitance, C_{reson} is used to provide flexibility. Finally, at the gate of the common source transistor, an L-matching network is realized to provide the appropriate input matching. In figure 23(c), the L-matching network does not appear in the design, but the analysis shows that an input match condition can be achieved. The advantage of this matching network though, is that it provides flexibility as the components' values can change accordingly, and still provide adequate input match. Additionally, as

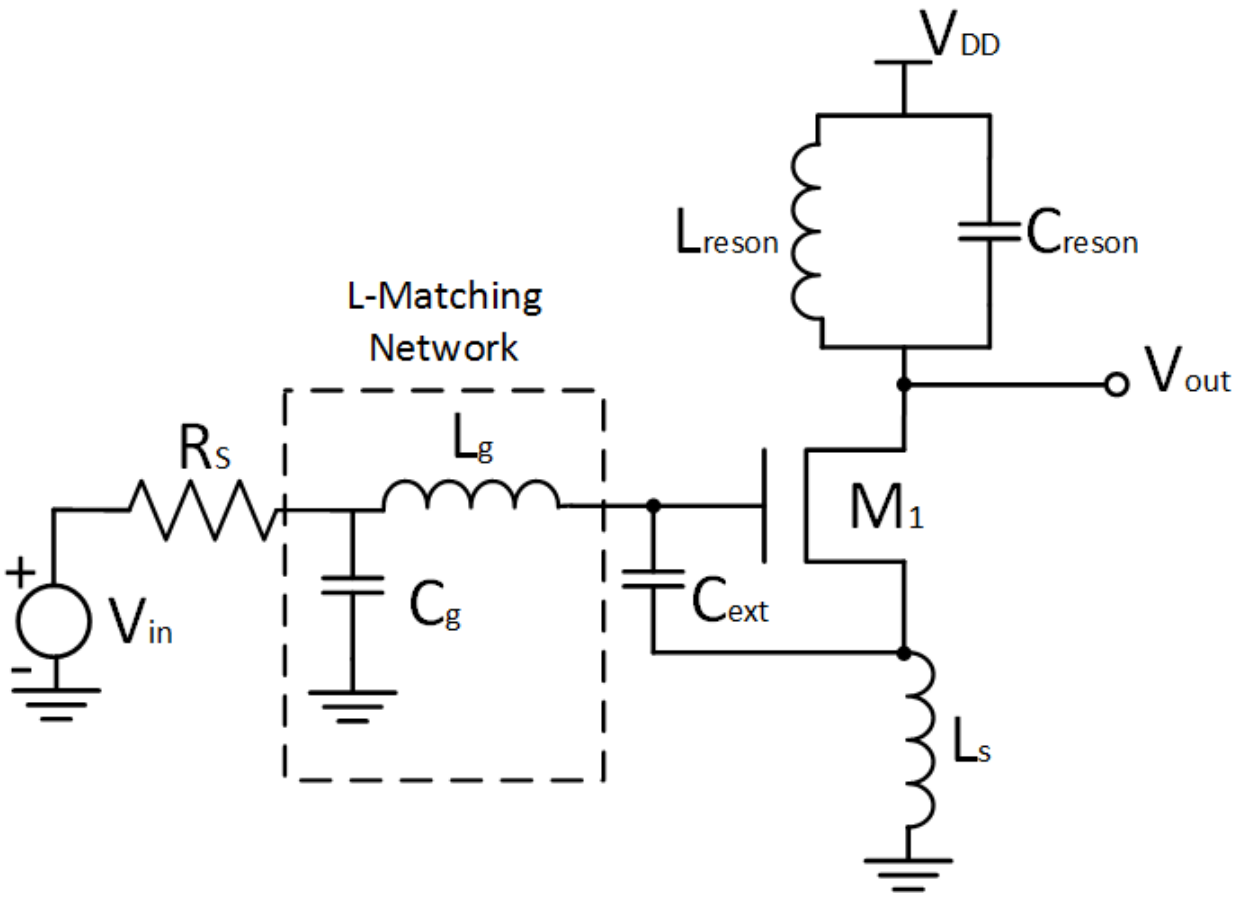

Figure 37: The selected common source topology

discussed in chapter 2, this kind of network provides a narrowband match, which ultimately, helps to design a narrowband amplifier.

The figure of merit analysis of the previous section provides a starting point for the design and optimization of the common source LNA of figure 34. According to the plot of figure 34, the optimum operating point lies in the moderate inversion and more specifically for inversion coefficient equal to $1/\lambda_c$. When the transistor operates in saturation we can use the expression (4.16) and equate it to $1/\lambda_c$. Afterwards, using the definition of the specific current I_{spec} from (4.6), and defining the desired channel current I_D, the optimum transistor width can be calculated :

$$IC = 1/\lambda_c \Leftrightarrow \frac{I_D}{I_{spec}} = 1/\lambda_c \Leftrightarrow I_{spec} = \lambda_c I_D \quad (4.17)$$

The parameter λ_c is given from the expression (4.12) and it depends on the technology process, thus its value is known a priori. Having calculated the specific current we get :

$$I_{spec} = 2\,n\,U_T^2\,\mu C_{ox}'\frac{W}{L} \Rightarrow W_{opt} = \frac{I_{spec} L}{2\,n\,U_T^2\,\mu C_{ox}'} \quad (4.18)$$

The analysis of the circuit in chapter 3, shows that the value of the degeneration inductance

can be calculated from $L_s = R_s/\omega_T$ (4.19), where in this case due to the external capacitor the transit frequency is equal to :

$$\omega_T = \frac{g_m}{2\pi(C_G + C_{ext})} \quad (4.20)$$

Using the expressions (4.14), (4.15), (4.9) and (4.10) the values of the normalized parameters q_s, g_{ms} c_{gs} and c_{gb} can be calculated. The transconductance gain g_m is simply equal to g_{ms}/n, where n is the slope factor, and in small channel devices can be equal to 1.4. The capacitance C_G can be calculated using the normalized values produced from (4.9), (4.10) the overlap capacitance and the optimized transistor width that has been calculated from (4.17) :

$$C_G = C'_{ox} L W_{opt} c_{gs} + C'_{ox} L W_{opt} c_{gb} + 2 \cdot C_{ov} \quad (4.21)$$

The value for the overlap capacitance is known from the technology process. Additionally the expression (4.21) adds the overlap two times, one for the gate-source and one for the gate-drain overlap capacitances, which are usually equal.

Knowing these values, the inductance L_s can be calculated from (4.19) by choosing a source inductance and external capacitor pair. The preferred technique is to chose the smallest available value for the inductor, and then pick the corresponding value for the external capacitor. This way, since the degeneration inductor has the smallest available value, the parasitic resistance that accompanies a non-ideal component, will also have the smallest available value, subsequently, adding less noise to the signal (noise rises with higher resistance values) and have less power dissipation.

The gate inductance is calculated from (3.15), but instead of the parameter C_{gs}, we have to use the C_{tot} to account for the external capacitor ($C_{tot}=C_{gs}+C_{ext}$) :

$$L_g = \frac{1}{\omega_0^2 C_{tot}} - L_s \quad (4.22)$$

The value for the capacitance C_g that completes the L-matching network is being calculated using the expression (2.18), since it is a shunt element.

Finally, the L-C tank has to provide the maximum gain at the center frequency. In order to do so, the values of the L-C pair have to be chosen according to the expression :

$$f_0 = \frac{1}{2\pi\sqrt{L_{reson} C_{reson}}} \quad (4.23)$$

This expression does not account for the parasitic components of the inductor and the capacitor though, and the more complete expression is :

$$f_0 = \frac{1}{2\pi\sqrt{L_{reson} C_{reson} R_{parasitic}}} \quad (4.24)$$

The problem is that there is not an easy method to calculate the parasitic resistance of the tank, and since the main parasitics contributor is the inductor, we chose the minimum available value, just like in the case of the degeneration inductance. This design procedure is called current specified technique, since the starting point of the methodology, is the definition of the drain current. A counterpart to this, is the inductance specified technique, where the starting point is the definition of the degeneration inductor, and the design parameters are extracted by following the same procedure in reverse. Figure 38 sums up the steps of the design methodology with a simple flowchart. In the case of the cascode common source LNA, the design procedure is the same as analyzed above. The

additional device that is used can have the same dimensions as the common source one, and the gate voltage can be equal to the supply voltage.

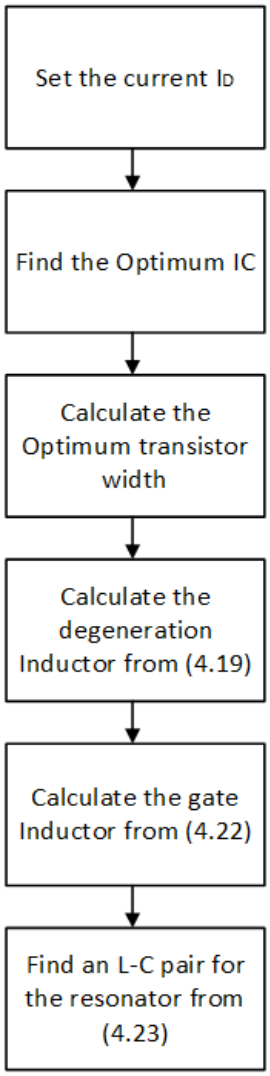

Figure 38: Design methodology of the common source LNA

5.2 OPTIMIZATION USING GENETIC ALGORITHM

The procedure of designing a common source LNA that has been introduced in the previous section, provides an approximation of the optimum design, due to the fact that the analysis does not account for the parasitic components of the non-ideal elements. Figure 39 shows the equivalent circuit of the spiral inductors of the *TSMC90RF* process design kit (PDK). It clearly illustrates the complexity of the non ideal inductor, and the numerous parasitic components. The modeling of these parasitics is not an easy procedure, and even if a model has been extracted, it will only apply to a certain process design kit, thus designers that use different PDKs cannot take advantage of the methodology. The final step of the LNA optimization is the appropriate choice of the design components using a genetic algorithm (*GA*).

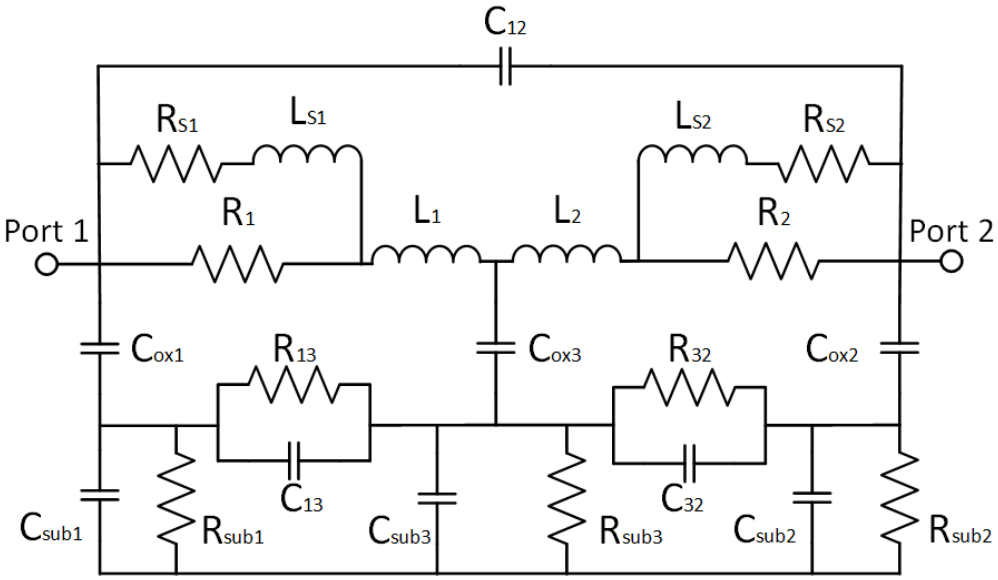

Figure 39: Equivalent circuit of the inductor in the TSMC90RF library

5.2.1 GENETIC ALGORITHMS

A genetic algorithm is a method for solving optimization problems, that is based on natural selection, the process that drives biological evolution, and belongs to the larger class of *evolutionary algorithms*. The genetic algorithm can be applied to solve a variety of optimization problems that are not suited for the standard optimization algorithms, including problems in which the objective function is discontinuous, non-differentiable, stochastic, or highly non-linear.

In a genetic algorithm, each unknown circuit design variable is called a *gene*, and the collection of the design variables-genes a *chromosome*. A *population* is the collection of all chromosomes. The idea of the genetic algorithm, is to *evolve* the population in a way that in each *generation*, the individuals[1] produce better results. The algorithm creates the population in each generation by using random individuals from the current population to be parents and produce the children for the next generation.

1 *Individual* and *chromosome* hold the same meaning and will be used interchangeably in this thesis.

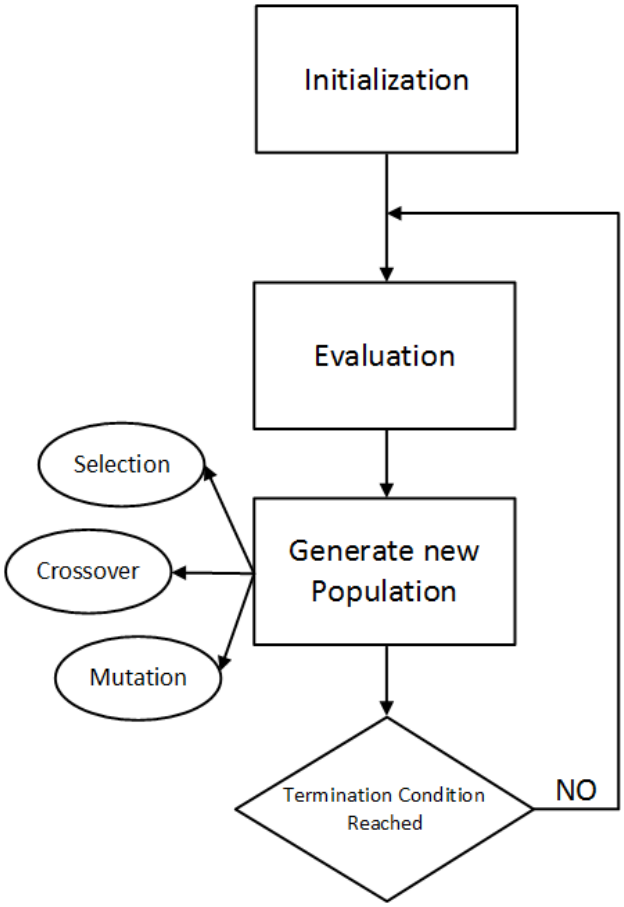

Figure 40: The flowchart of a typical genetic algorithm

The genetic algorithm usually begins its operation by creating a number of chromosomes (population) through randomly selecting values for each gene. This population can be any desired size, from only a few individuals to thousands.

Each chromosome of the population is then evaluated using an evaluation tool, and we calculate a *fitness* for each individual. The fitness value is calculated by how well it fits with our desired requirements. These requirements could be the gain of the design, the power consumption, or most commonly, a combination of two or more requirements.

In each generation the desired outcome is to improve the population's overall fitness. In order to do so, the genetic algorithm uses three types of operations to create the new generation from the current population :

- **Selection** : The selection operation is discarding the bad designs and only keeps the best individuals in the population, according to their fitness value. There are a few different selection methods but the basic idea is the same, make it more likely that fitter individuals will be selected for the next generation. For that, the algorithm selects randomly a number of chromosomes (*tournament*), and from these chromosomes the best passes to the next generation.
- **Crossover** : During the crossover operation, the algorithm creates new individuals by combining aspects from two (or more) of the selected individuals in the tournament. The new individual will consist of genes that are acquired from its two parents. The hope is that

by combining certain traits from two (or more) individuals, the algorithm creates fitter offsprings which inherit the best traits from each of their parents.

- **Mutation** : The mutation function is used to add randomness into the populations' genetics, otherwise every combination of solutions, will be a product of the initial population. Using a small amount of randomness enlarges the search space for the optimized solution. Mutation typically works by making very small changes at random to an individuals genome.

After the new population has been generated, the algorithm repeats the same steps again until a termination condition is reached. There are a few reasons why the algorithm should stop searching a solution. The most likely reason is that the algorithm has found a solution which meets the predefined minimum criteria. Other reasons for terminating could be the total number of generations or the fact that the population has not produced a better result for a set number of consecutive generations.

5.2.2 THE EVALUATION TOOL – GENETIC ALGORITHM INTERFACE

To establish an automated optimization procedure, the selected evaluation tool is the circuits simulator *Spectre RF* ™ of the *Cadence Virtuoso*® analog design environment. The simulator is invoked through the use of scripts that are generated from the genetic algorithm, and contain the design variables-genes of each individual. The scripts were written in OCEAN (Open Command Environment for Analysis) that is based on the Cadence® SKILL programming language, that allows users to specify a list of simulations in one or more designs and instruct the simulator to perform them.

The Virtuoso environment is constantly checking if the genetic algorithm script has sent a new individual for evaluation through the OCEAN script. In such case, it extracts the design variables and runs the appropriate simulations. Afterwards, the results of the simulations, are saved

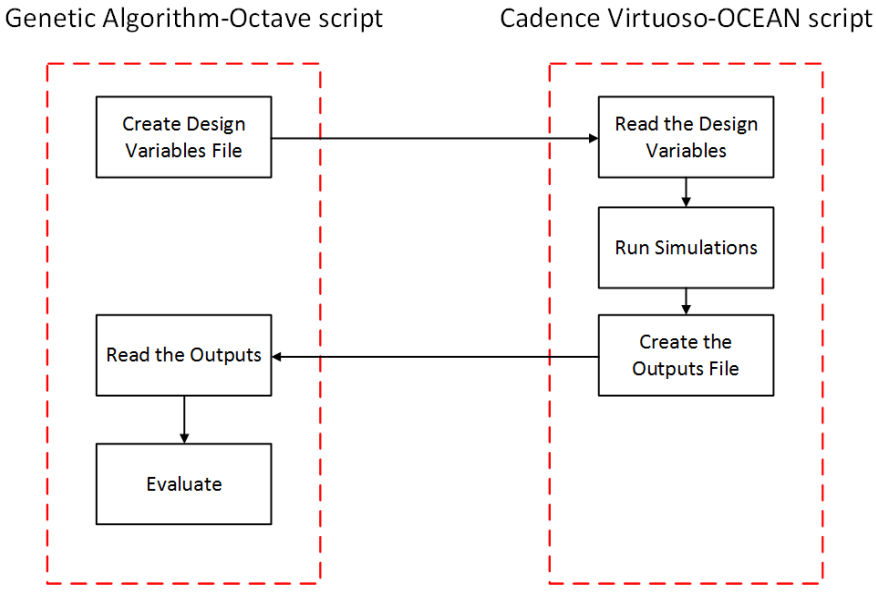

Figure 41: Evaluation Tool - Genetic Algorithm interface

into a file, also with the use of the OCEAN script. This file is then opened by the genetic algorithm and the results that are saved, are used for the evaluation of the individual through the fitness function.

The genetic algorithm has been implemented in *octave*, a MATLAB® open source equivalent and is integrated to the OCEAN environment through OCEAN text scripts. The octave script creates a file containing the design variables, then the OCEAN script uses this file to extract the variables. This back and fourth procedure is illustrated in figure 41. This procedure is repeated for every individual.

5.2.3 THE FITNESS FUNCTION

The performance of each design is reflected by the fitness value of each chromosome. The fitness function is one of the key features of the genetic algorithm. The accuracy of a genetic algorithm directly depends on the definition of the fitness function. The fitness function has an additive philosophy, thus the goal of the genetic algorithm is either to maximize the value of the fitness function or minimize it.

The best approach in choosing the fitness function, is to define one that has achievable upper or lower bounds. This way, if the genetic algorithm ever reaches these values the optimum result has been achieved. In many applications though it is not easy to know the bounds beforehand and the definition of the fitness function might become more tricky. If it incorporates many variables the search space increases but there is a compromise in computational efficiency and time consumption. The general definition of the fitness function is :

$$fitness = \sum_{i=1}^{N} w_i \cdot h_i \quad (4.25)$$

where h_i are the performance metrics (e.g. consumption, gain, noise figure or matching in the case of the LNA), and w_i are the weight factors. The weight factors are optional, but using them helps the algorithm to provide a result that favors some performance metrics against some others. This approach is usually helpful when trade-offs take place in the design and some results are preferred over the others.

5.2.4 LIMITATIONS

In the case of a fitness function that the bounds cannot be known beforehand, the problem of the local optimum emerges. The algorithm might improve the design, but it is not easy to be aware if the result is actually the optimum one or there is still room for improvement (figure 42). Thus, the algorithm often is not able to guarantee that the output is the global optimum to the solution. Additionally, for complex problems it is usually unreasonable expectation to find a global optimum.

Another limitation of the genetic algorithm is that it is time consuming. Since it is a randomized algorithm, the convergence time is not known beforehand. Additionally, for every individual there has to be a simulation that is also time consuming. For a realistic example of a population of 30 individuals, simulation time 0.5 seconds and 150 generations the total time for convergence is :

$$time = 30\ individuals \times 0.5 \frac{seconds}{individual} \times 150\ generations = 2250\ seconds = 37.5\ minutes$$

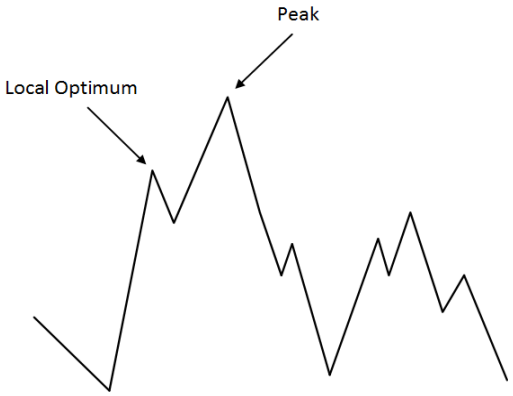

Figure 42: Illustration of the local and global optimum

5.3 COMMON SOURCE LNA – A CASE STUDY

In this section, we will use the two methodologies that have been analyzed previously and try to design an optimum common source LNA with inductive degeneration like the one presented in figure 37 (page 43), with a center frequency of 5GHz.

5.3.1 FIRST APPROXIMATION USING THE DESIGN MODEL

As mentioned previously, the technique that has been analyzed is called current specified technique, thus the first parameter that we have to define is the drain current, where in this case we choose a current of 10 mA. Afterwards, using the figure of merit curve of figure 34, we chose the optimum inversion region, that obviously is in the moderate inversion. We find that the optimum inversion coefficient is around 6. Then from equation (4.18) we get the optimum transistor width, which in this case is 190 um.

As mentioned previously, the way the device is realized in the layout affects the noise performance of the design. In order to minimize the addition of gate noise, the transistor can be laid out using multiple fingers. This means that, a large transistor can be broken down into multiple

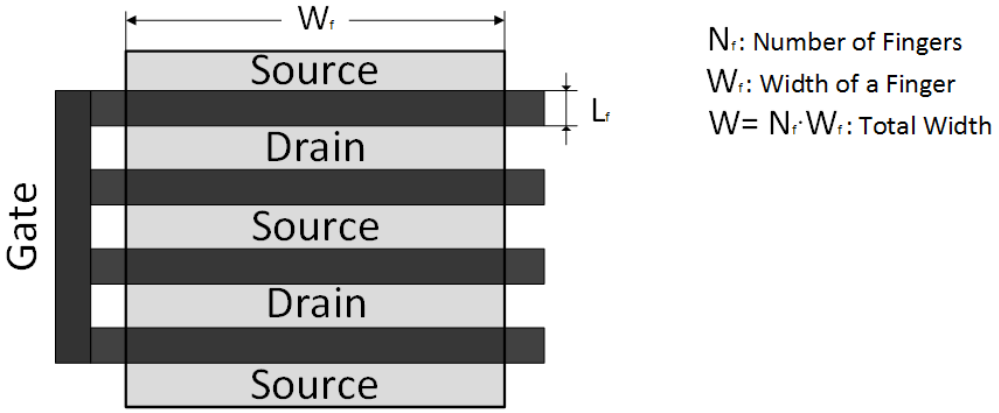

Figure 43: The layout of a multi-fingered device

smaller ones that each of them share the same source-drain area. Each device is also accompanied by its gate terminal, and ultimately all of the gates connect together in order to share the same input signal and operate as a single transistor.

Figure 43 presents a transistor that has been realized using multiple fingers. What this offers is that the gate terminal surface has been enlarged, so the resistance of the gate is reduced. As a reminder, the resistance of a transmission line is inversely proportional to the total area of the line. Since the gate resistance can be reduced with this layout technique, the noise added to the circuit is also reduced. The downside of this technique though, is that multiple fingered device show larger gate capacitance values, thus reducing the maximum unit gain frequency f_t. Additionally, the threshold voltage of the device also increases, which means that the available voltage headroom decreases, and it has to be taken into account, especially when designing circuits that incorporate a cascode device. Knowing these cases, it can easily be seen that there is a trade off between noise and speed. This trade off is going to be optimized using the genetic algorithm.

The next step in the design procedure, is to calculate the values of the inductors and the capacitors. Firstly, we choose the smallest available inductor to maximize the value of the unity gain frequency according to (4.19). In TSMC90RF process design kit that is used in this thesis, the smallest available inductor is 0.22 nH. Later using equation (4.22) we can calculate the gate inductor, and in this case we chose an inductor with value 2.63 nH. From there we deduce that the external capacitance is going to be 120 fF, and the gate capacitance that completes the matching network is 426 fF. Finally, the last two components that compose the L-C tank have to resonate at the center frequency of 5 GHz, thus according to equation (4.23), we chose the pair of values to be 0.22 nH and 4.60 pF.

5.3.2 OPTIMIZATION USING THE GENETIC ALGORITHM

After the design variables have been calculated, we have an approximation of the LNA. The values then, are imported into the genetic algorithm as an individual. The rest chromosomes are generated randomly as in the classic genetic algorithm approaches. The design variables that define the chromosome are the turns and the inner radius of the three inductors, along side with the width and number of fingers of the transistor's width, thus 8 genes. Bear in mind that the optimizing algorithm

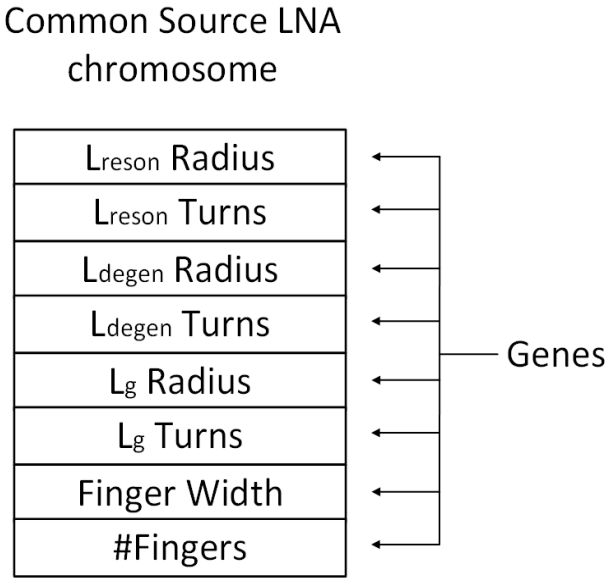

Figure 44: The chromosome of the common source LNA

does not change the values for the capacitors, which means that the main purpose of this method is to find the best available inductors and width-number of fingers pair that produces the optimum outcome. The parasitics of the capacitors do not affect the design nearly as much as the parasitic of the inductors so there is no point in adding them in the optimization sequence, thus decreasing the convergence time.

The population size is chosen to be equal to 20. The parameters of the initial design are set to be the first chromosome in the population. The mutation probability is set to 15% in order to have frequent mutations, thus wide search space. For the calculation of the fitness function, we use seven parameters :

$$fitness = h_{center} + 0.15 \cdot h_{s21} + 0.3 \cdot h_{NF} + h_{s11} + 0.25 \cdot h_{s11,mag} + 0.10 \cdot h_{s22} + h_{ID} \quad (4.26)$$

In this realization, the genetic algorithm tries to minimize the value of the fitness, thus better designs have lower fitness values. The expression (4.26) also shows the selected weights for each function. Bear in mind that the method to calculate the fitness function is flexible, meaning that we can change either the weights of (4.26) or the individual functions themselves, and apply the same algorithm to our specifications. The next table shows how the functions in (4.26) are calculated and the though process behind each definition :

Function	Definition	Explanation
h_{center}	$\lvert f_{target} - f_{S_{21}} \rvert^2$	The parameter $f_{S_{21}}$ shows the tuning frequency of the LNA. Thus the h_{center} function is the square distance from 5 GHz (f_{target}). The square is used in order to expand the output range and small differences from the center frequency lead to higher values of the h_{center}.
h_{s21}	$\frac{1}{\lvert S_{21@5GHz} \rvert}$	This function is getting a lower value when the gain at the center frequency of 5 GHz is getting higher.
h_{NF}	NF^2	This function is simply the square of the noise figure that has been extracted from the simulations.
h_{s11}	$\lvert f_{target} - f_{S_{11}} \rvert^2$	Just like h_{center}, this is the square distance of the minimum value of the S_{11} parameter from 5 GHz.
$h_{s11,mag}$	$\frac{1}{\lvert S_{11@5GHz} \rvert}$	This function is getting a lower value when the magnitude of the S_{11} parameter at the center frequency of 5 GHz is getting higher in absolute values.
h_{s22}	$0.35 \cdot \Re\{Z_{22}\} + 0.65 \cdot \Im\{Z_{22}\}$	The function h_{s22} is calculated using the output impedance real and imaginary values. The weights are different because the imaginary part of the Z_{22} parameter is responsible for where the output is tuned, thus it is more important, than the real value that translates to magnitude.
h_{ID}	$\left\lvert 1 - \frac{I_D}{I_{target}} \right\rvert^2$	The h_{ID} function shows how different is the current of a design from the desired current consumption. Another useful way to implement this function is to equate the h_{ID} to the simulated current I_D. This way the fitness gets better values when the design draws less current, thus lowering the consumption.

Table 1: Definition of the functions that comprise the fitness function and their explanation

5.3.3 OPTIMIZATION RESULTS

The next figures show the performance of the 5 GHz low noise amplifier after the genetic algorithm has fine tuned the initial design that has been generated using the equation methodology. As it can be seen the amplifier is tuned in the desired 5 GHz frequency, the input is matched and the noise figure is kept at low values. Thus, we can assume that the combination of the two methodologies managed to produce satisfying results for the common source low noise amplifier with inductive degeneration.

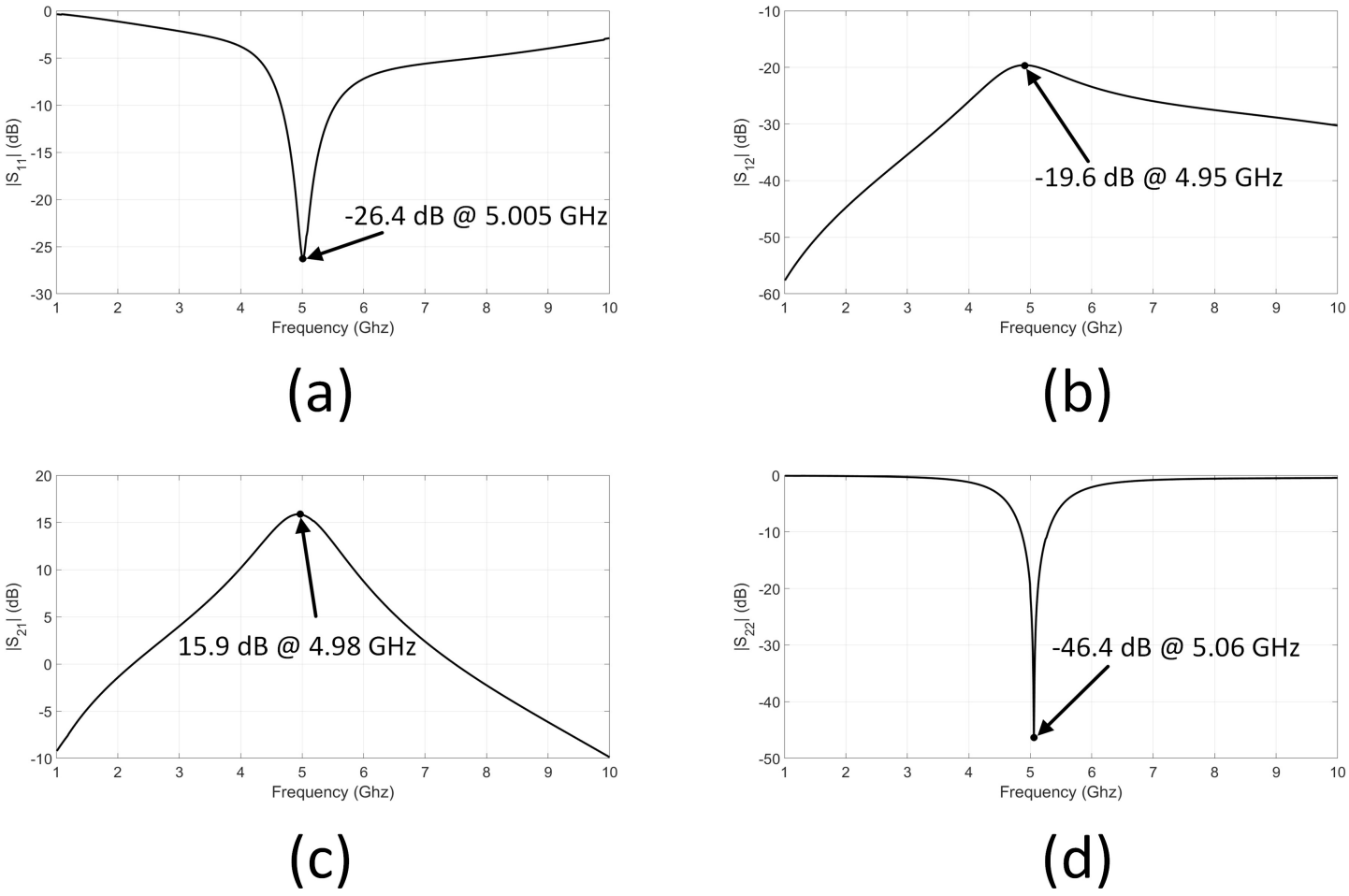

Figure 45: The S-parameter plots of the common source LNA simulation : (a) S_{11}, (b) S_{12}, (c) S_{21}, (d) S_{22}

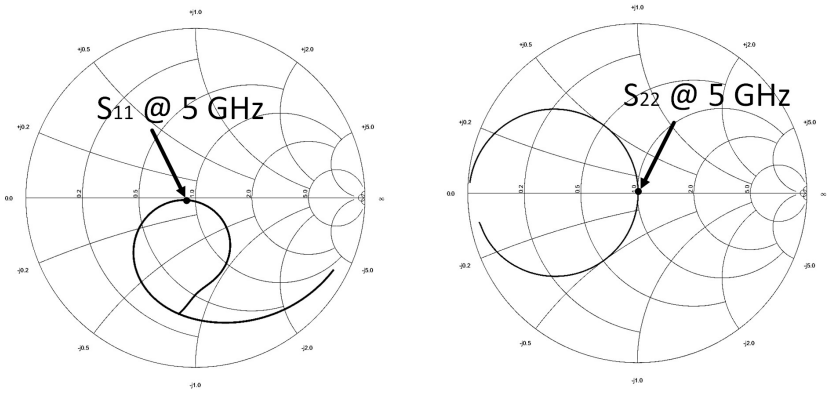

Figure 46: The S_{11} and S_{22} parameters of the design presented on the Smith Chart

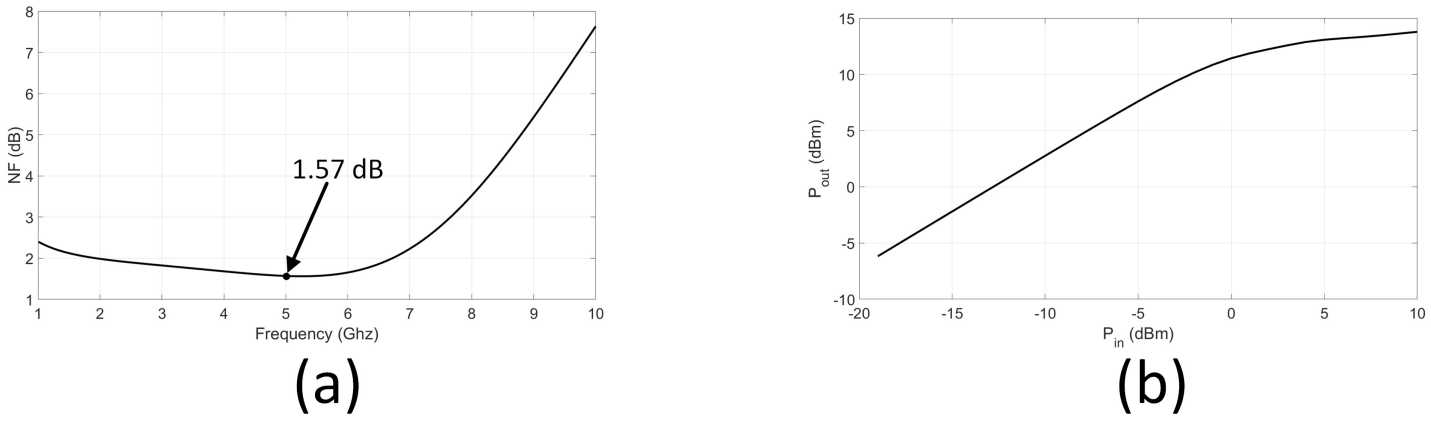

Figure 47: (a) NF performance of the LNA versus the frequency, (b) The P_{in} versus P_{out} gain plot

The most striking result of these plots is the noise performance of the LNA that is seen in figure 47(a). The 1.57 dB value is a sufficient result for the majority of applications. Additionally, another note worthy result is the high gain value of 15.9 dB as seen in figure 45(c).

The S-parameters of figure 45 are obtained assuming that the design is linear, which is not the case. For this reason, we often use the large signal analysis that takes into account the non-linearities of the design. The result of this analysis is the plot of figure 47(b). It presents the power gain of the amplifier for different levels of the input power. This analysis also give us the metrics for the linearity performance of the LNA. The 1dB compression point at an input power level of -2.22 dB.

An alternative view of the system's input and output matching can be seen in figure 46. The smith charts of the S_{11} and S_{22} parameters show that the circuit fulfills the requirements for input and output matching. In the 5 GHz frequency point on the smith chart both responses are close to the center, that translates to an impedance of 50 Ω.

Figure 48, shows the stability factor that has been introduced in chapter 3. Recall that a design is considered stable if the K_f value is over unity. The simulations that have been made range from 1 to 10 GHz. In this search space the stability factor is always larger than unity, thus the design is unconditionally stable.

Finally, the consumption of the LNA is 13.8 mW. The supply voltage is the maximum allowed by the process at 1.2 V and the current consumption is 11.5 mA.

Altogether, the methodology seems to have produced a satisfying design. The question that surfaces now is if it is the optimum design.

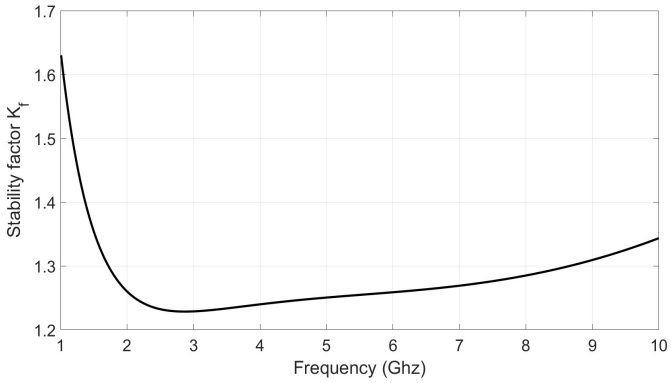

Figure 48: Stability factor K_f versus frequency

If we recall the analysis of the figure of merit, the optimum operating region is the moderate inversion. The design of the LNA begins with the notion that we have to design in this moderate inversion, but what happens in different regions? To answer that, we have designed the LNA for different values of the inversion coefficient, and we use the figure of merit of (4.11) to deduce where is the optimum inversion region. The result of this procedure is illustrated in figure 49. The

shape of the curve resembles the one of figure 34, thus the theoretical analysis is verified by the simulated results. Consequently, the optimum operating region is indeed in the moderate inversion, and for inversion coefficient around 6.2. Hence, we can safely assume that our methodology has produced an optimum design according to the figure of merit $g_m f_t / I_D$.

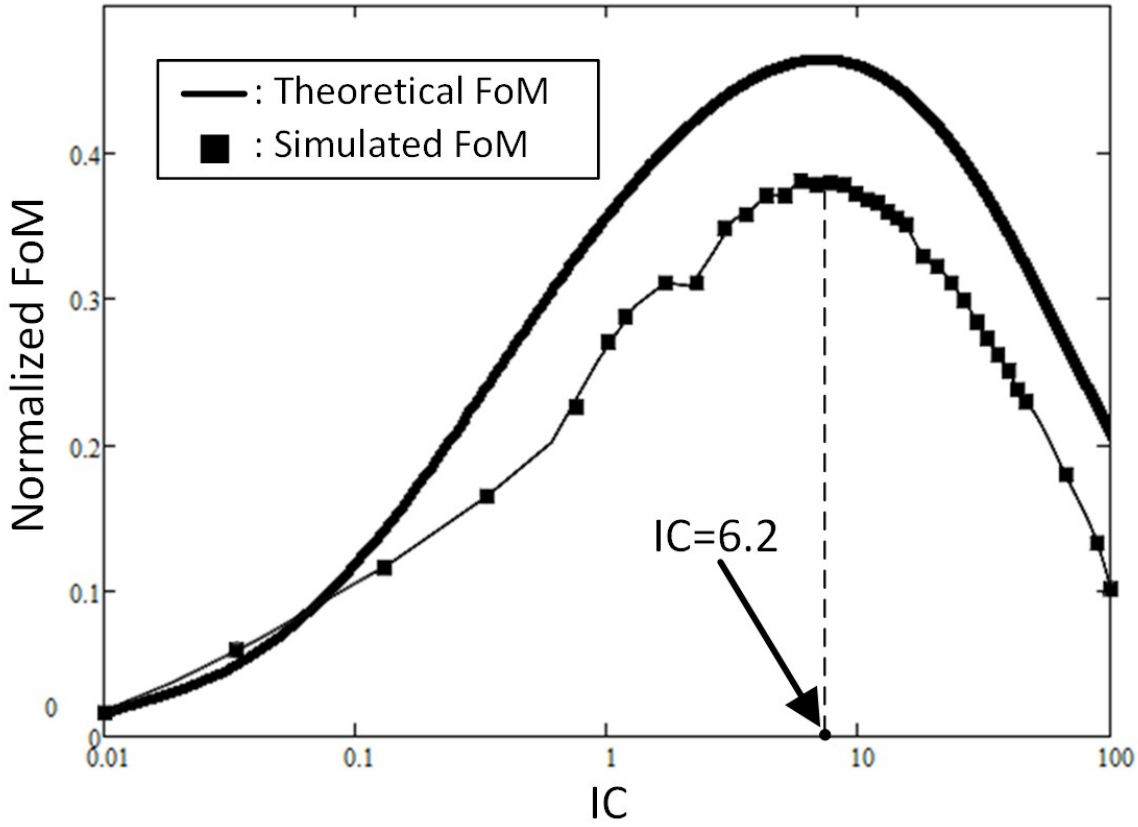

Figure 49: Normalized Figure of Merit $g_m f_t / I_D$ according to the analytical model and the simulated results

In order to compare our LNA with similar works, the next table presents the results of our common source LNA with inductive degeneration design with other implementations of the same topology that have been found in literature. The table clearly shows that our design methodology has proven its capability to output satisfactory results, with exceptional noise performance, adequate gain and good input and output matching.

Process	S11	S21	S22	NF	Supply Voltage	Frequency
[35] 180nm CMOS	-10 dB	9.3 dB	-12 dB	2.8 dB	1.5 V	5 GHz
[36] 130nm CMOS	-18.6 dB	28 dB	-	2.46 dB	1.2 V	2.4 GHz
[37] 65nm CMOS	-13 dB	25.6 dB	-	2.50 dB	1.2/1.8 V	1.7 GHz
This Work 90 nm CMOS	-26.40	15.9 dB	-46.6 dB	1.57 dB	1.2 V	5 GHz

Table 2: Comparison between different designs of the common source LNA with inductive degeneration

Table 4 presents a summary of the low noise amplifier output parameters, on a device level (like the drain source conductance g_{ds}, the equivalent noise resistance R_n or the overdrive voltage V_{GS}-V_{TH}) and on a system level (like the noise figure NF and unity gain frequency f_t). On the other hand table 3 shows the design variables after the optimization sequence.

L_{reson} : 0.23 nH	Inductor Width : 15 um
	Inner Radius : 27 um
	#Turns : 0.5
L_S : 0.22 nH	Inductor Width : 15 um
	Inner Radius : 17 um
	#Turns : 0.75
L_g : 2.7 nH	Inductor Width : 3 um
	Inner Radius : 79 um
	#Turns : 2.5
C_{reson}	4.16 pF
C_g	0.42 pF
C_{ext}	0.12 pF
Finger Width	4.88 um
# Fingers	43
Length	100 nm
V_{DD}	1.2 V

Table 3 : Design Parameters

Frequency (GHz)	5
I_D (mA)	11.5
IC	6.2
V_{GS}-V_{TH} (V)	0.15
g_m (mS)	177
g_{ds} (mS)	13.57
f_t (GHz)	79
NF (dB)	1.57
NF_{min} (dB)	1.28
C_{gg} (fF)	230
R_n (Ω)	8.8
S_{id} ($pA/\sqrt{(Hz)}$)	14.2
P_{DC} (mW)	13.8

Table 4 : Optimized LNA Outputs

5.4 DESIGN OF DIFFERENT TOPOLOGIES

In section 5.3 we used the design methodology and the genetic algorithm to produce an optimum design for the common source LNA with inductive degeneration of figure 37. We can also use the procedure to design different LNA topologies such as the cascode common source LNA of figure 26 (page 33), and the common gate topologies of figures 26 and 30. Additionally, in order to show the adaptability of the methodology, we apply it in variations of those topologies, that show different strengths in sizes like linearity and bandwidth. The following figure presents the circuits that are going to be studied, where CS_1, is the LNA of the previous case study.

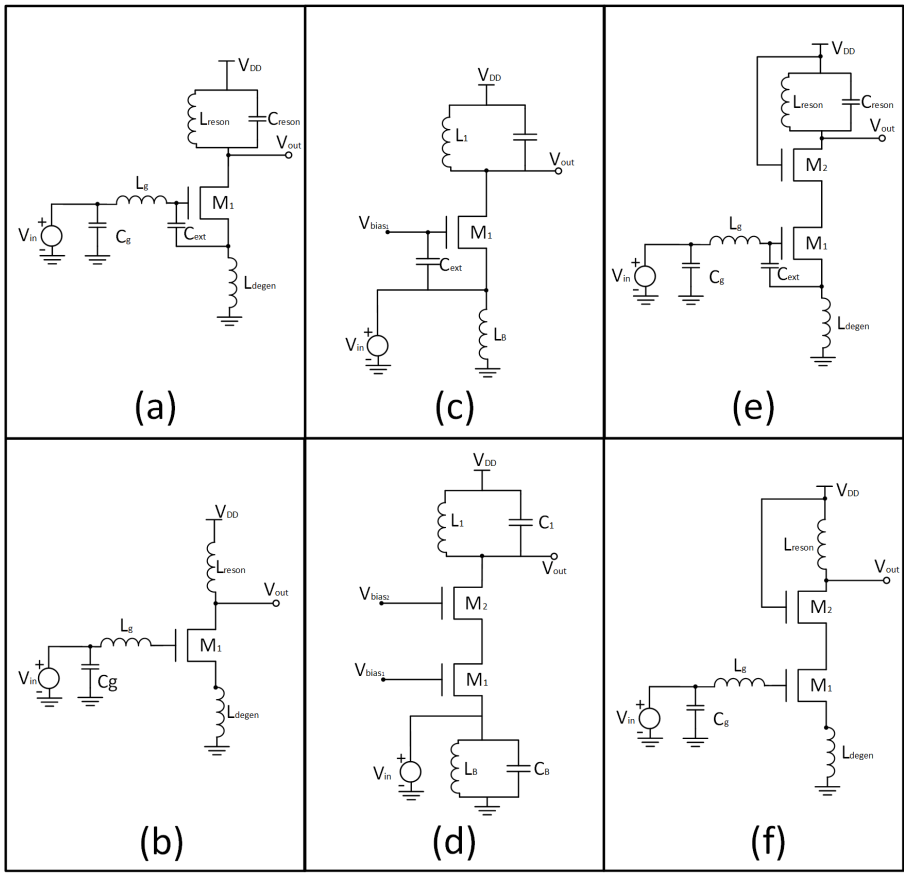

Figure 50: LNA topologies and variations : (a) CS_1, (b) CS2, (c) CG_1, (d) CG_2, (e) CCS_1, (f) CCS_2

The circuits of the first row in figure 50, are used for narrowband applications while the ones of the second row are used for wideband applications. This is a result of the input matching network, and the resonator design. The narrowband circuits incorporate, narrowband matching networks as they have been analyzed previously. Additionally, in the wideband designs, the fact that the resonator does not make use of an extra capacitor for tuning, leads to the use of inductance with larger values which in turn leads to broadened bandwidth.

Due to the design philosophy of the common gate circuits, we cannot make an initial approximation of the circuit. Recall that the losses due to the resonator and the input resistance has to be some orders of magnitude greater than the input impedance. This means that we have to know the parasitic losses beforehand which is not the case. Additionally, the fact that we have to incorporate the inductor losses into the design, means that we might have to change the inductance value from the initial guess, thus the value for the resonator capacitor has to change accordingly. For those reasons, the common gate circuits CG_1 and CG_2, cannot be optimized with fixed capacitor values, but they must be incorporated in the optimization sequence.

The following table presents the results after the optimization of the circuits in figure 50. As expected the circuits CS_1, CG_1 and CCS_1, show lower bandwidth values than the rest of them supporting the case that they are used for narrowband applications. The next note worthy result is that the topologies that incorporate a cascode device show larger values of reverse isolation.

Another observation regarding the common gate topologies is that as analyzed in chapter 3, good matching and low noise figure values cannot be achieved simultaneously, thus the optimizer tries to find a balance between those parameters. The poor matching also leads to low gain values

but in favor of very high linearity results. Finally, the cascode common source topologies show poor results in the linearity metrics due to the use of a second device that is highly non linear.

Topology Variation	S_{11} (dB)	S_{12} (dB)	S_{21} (dB)	S_{22} (dB)	NF (dB)	1dB Compression (dBm)	BW (GHz)	I_D (mA)
CS_1	-26.40	-19.61	15.90	-46.60	1.57	-2.22	1.14	11.50
CS_2	-15.12	-18.30	14.11	-15.57	2.30	-3.90	2.77	25.96
CG_1	-5.20	-16.30	4.63	-10.50	1.97	>10	3.36	29.12
CG_2	-3.00	-33.45	3.17	-2.69	2.90	9.26	3.22	9.77
CCS_1	-17.11	-40.89	13.40	-5.10	1.90	-9.70	0.90	18.22
CCS_2	-24.40	-35.12	18.21	-2.05	1.73	-10	3.10	11.01

Table 5: Comparison in the performance of the six LNA topologies of figure 50*. The different colors show the best performing design in each category, where green is the best and red the worst result*

CHAPTER 6

CONCLUSIONS

The purpose of this thesis was to present an analysis of the most common low noise amplifier topologies and ultimately suggest a methodology that helps the RF designer design an optimized low noise amplifier. The work was mostly focused on the common source amplifier with inductive degeneration, which showed the best results overall in the simulations. We started the analysis of the LNA using the notion that in sub micron technologies the optimum inversion region is the moderate inversion, where the plots of the proposed figure of merit backs this hypothesis up. Afterwards we came up with an equations model that incorporates the compact EKV model for MOSFETs. This led to designing an approximation of the low noise amplifier, but the final circuit could not be designed only using this result, since the equations model do not take into account the non ideal components of a real design. Thus, in pursuance of an optimum LNA we created a genetic algorithm to optimize the behavior of the design. Finally, we designed a number of circuits in different operating regions in order to show that the result of the optimizer is indeed the optimum, according to the proposed figure of merit.

The LNA case study of chapter 5, showcases the performance of the optimization methodology, proving that the optimum operating region is indeed in the moderate inversion region and for inversion coefficient of $1/\lambda_c$.

One of the key points of this thesis is the use of both an analytical model, and an iterative method to find the optimum result. The genetic algorithm uses the result of the analytical model as an individual in the population thus gaining a strong starting point. In average, the algorithm converged to an output in 122 generations, for the common source LNA. In a case where there is no initial near optimum chromosome, thus all the individuals are randomly generated, the algorithm converged in average after 319 generations. This shows a decrease of 61.7% in computing time, which is a remarkable result.

As mentioned previously, the common gate topologies incorporate the capacitors in the optimization sequence. Each capacitor is composed from two parameters the width and length. Thus, the final chromosome consists of more genes than the ones of the common source, leading to increase in convergence time. In addition, since there cannot be an initial approximation using the model equation, all the individuals of the population are generated randomly. These facts lead to a convergence time of 463 generations in average.

In summary, we have shown a method to take advantage of the combination of equations model and a genetic optimizer in order to help the RF designer find a balance of the trade offs and design an optimum low noise amplifier.

6.1 FUTURE WORK

The tool can be extended to provide the capability to optimize the LNA in the layout design sequence. During the layout procedure, the designer finds himself opposing new challenges like the die area, and transmission line parasitics. Thus, expanding the genetic algorithm can optimize not only the operation of the LNA in the post-layout simulations, but also minimize the required chip area, by finding the optimum position and spacing for each component.

Additionally, the work can be extended by showcasing the variability of the low noise amplifier design using Monte Carlo simulations. This analysis will present the behavior of the design by adding extra degrees of freedom like the temperature of operation, the variations of the supply voltage and device mismatch. The Monte Carlo analysis, will produce the statistical behavior that describes the operation of the LNA for variations in design parameters.

One of the most interesting research opportunities for future work is to apply the procedure to an evaluating system-simulator that incorporates the latest industry model standards like the EKV3.0 or BSIM6.0. These models use a charge based approach on the modeling of MOSFETs, just like the analysis of this work. Using a simulator with these models would be more beneficial, since the results should be closer to the theoretical analysis.

The present work has a clear potential for introducing further electronic design automation (EDA) methods in the demanding field of RF front-end design. The technique of modeling the low noise amplifier and then optimize it with a genetic algorithm can be extended to other RF and analog building blocks of the transceiver topology, like RF mixers, RF synthesizers, or transconductance amplifiers. Of course for each of these building blocks the suitable objective functions need to be established. The use of a different PDK, can also be examined, where we could study the performance of the methodology in newer technology nodes like the 65 nm and 45 nm.

REFERENCES

[1] B. Razavi, "*RF Microelectronics*" Upper Saddle River, NJ : Prentice Hall, 1998

[2] A. Mangla, C. C. Enz , J.-M. Sallese "*Figure-of-Merit for Optimizing the Current-Efficiency of Low-Power RF Circuits*", in Mixed Design of Integrated Circuits and Systems, MIXDES 2011 18 th International Conference on Jun. 2011

[3] http://ekv.epfl.ch/files/content/sites/ekv/files/pdf/ekv_v262.pdf

[4] A. Antonopoulos, M. Bucher, K.Papathanasiou, N. Mavredakis, N.Makris, R. K. Sharma, P. Sakalas, and M. Schroter, "*CMOS Small-Signal and Thermal Noise Modeling at High Frequencies*", in IEEE Transactions on Electron Devices, Vol. 60 No. 11 nov. 2013 , pp. 3726-3733

[5] R.F. Martegani "*An All-Inversion-Region g_m/I_D Based Design Methodology for Radiofrequency Blocks in CMOS Nanometer Technology*", Phd Thesis, University of Seville, 2011, pp. 35-39

[6] V. Papageorgiou, S. Vlassis "*CMOS LNA Optimization Techniques: Comparative Study*", in Electronics, Circuits and Systems (ICECS) 2010, pp. 82-84

[7] V. H. V. Baroncini, Oscar da Costa Gouveia-Filho *"Design of RF CMOS Low Noise Amplifiers Using a Current Based MOSFET Model"*, IEEE Integrated Circuits and Systems Design 2004, pp. 81-87

[8] Geert D. J. Smit, Andries J. Scholten, Ralf M. T. Pijper, Luuk F. Tiemeijer, Ramses van der Toorn, Dirk B. M. Klaassen, "*RF-Noise Modeling in Advanced CMOS Technologies*", in IEEE Transactions on Electron Devices, 2013

[9] D. Leblebici, Y. Leblebici "*Fundamentals of High-Frequency CMOS Analog Integrated Circuits*", Cambridge University Press, 2009

[10] http://rfic.eecs.berkeley.edu/~niknejad/ee142_fa05lects/pdf/lect14.pdf

[11] L. Belostotski, J. W. Haslett, "*Noise Figure Optimization of Inductively Degenerated CMOS LNAs With Integrated Gate Inductors*", in IEEE Transactions on Circuits and Systems-I: Regular Papers , Vol. 53, No. 7, July 2006, pp. 1409-1422

[12] http://electronicdesign.com/what-s-difference-between/what-s-difference-between-third-order-intercept-and-1-db-compression-point

[13] Pieter L. D. Abrie, "*Design of RF and Microwave Amplifiers and Oscillators*", Second edition, Artech House, 2009

[14] David M. Pozar, "*Microwave Engineering*", Third edition, Wiley, 2005

[15] John B. Hagen, "*Radio-frequency Electronics*", Second edition, Cambridge University Press, 2009

[16] http://www.infineon.com/dgdl/Infineon-AN418_BGA524N6-AN-v01_00-EN.pdf?fileId=5546d462518ffd8501520ccc5b9e70c6

[17] C. C. Enz and E. A. Vittoz, "*Charge-Based MOS Transistor Modeling : The EKV model for low-power and RF IC design*", John Wiley & Sons, Ltd, 2006

[18] T. Taris, H. Kraimia, JB Begueret, Y. Deval "*MOSFET Modeling for Ultra Low-Power RF Design*", in nano-tera.ch, Lauzanne, Switzerland, 2011

[19] Angelos Antonopoulos, "*Nanoscale RF CMOS Transceiver Design* ", PhD Thesis, Technical University of Crete, Feb. 2013

[20] George Konstantopoulos, "*Automated prior knowledge aware optimization for radio frequency integrated circuits* ", Master Thesis, Technical University of Crete, Mar. 2007

[21] P. Andreani, H. Sjöland, "*Noise Optimization of an Inductively Degenerated CMOS Low Noise Amplifier* ", in IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing, Vol. 48, No. 9, Sep. 2001 pp. 835-841

[22] M. Muhamad, M. Soin, H. Ramiah, N.M. Noh, C.W. Keat "*A 0.13µm Inductively Degenerated Cascode CMOS LNA at 2.14GHz* ", in IEEE Symposium on Industrial Electronics and Applications, Sep. 2011, Langkawi, Malaysia, pp. 120-123

[23] Kuo-Liang Yeh, Jyh-Chyurn Guo, "*The Impact of Layout-Dependent STI Stress and Effective Width on Low-Frequency Noise and High-Frequency Performance in Nanoscale nMOSFETs* ", in IEEE Transactions on Electron Devices, Vol. 57, No. 11, Nov. 2010, pp. 3092-3100

[24] www.theprojectspot.com/tutorial-post/creating-a-genetic-algorithm-for-beginners/3

[25] M. B. Fallahpour, K. D. Hemmati, A. Pourmohammad, "*Optimization of a LNA Using Genetic Algorithm*", in Electrical and Electronic Engineering 2012, pp. 38-42

[26] Hao-Hui Chen, Ming-Huei Chen, Cheng-Yu Tsai, "*Optimization of Low Noise Amplifier Designs by Genetic Algorithms*", in Proceedings of the "2013 International Symposium on Electromagnetic Theory"

[27] mgh-courses.ece.gatech.edu/ece4430/ECE4430/Unit1/EKV_Model_Extraction/oceanref.pdf

[28] http://ngspice.sourceforge.net/external-documents/models/BSIMSOIv4.4_UsersManual.pdf

[29] R.L. Radin, G.L. Moreira, C. Galup-Montoro, M.C. Schneider "*A Simple Modeling of the Early Voltage of MOSFETs in Weak and Moderate Inversion* ", in International Symposium on Circuits and Systems – ISCAS 2008, pp. 1720-1723

[30] M. A. Chalkiadaki, C.C. Enz, "*RF Small-Signal and Noise Modeling Including Parameter Extraction of Nanoscale MOSFET From Weak to Strong Inversion* ", in IEEE Transactions on Microwave Theory and Techniques, Vol. 63, No. 7, Jul. 2015 pp. 2173-2184

[31] C.C. Enz, M. A. Chalkiadaki, A. Mangla "*Low-power Analog/RF Circuit Design based on the Inversion Coefficient* ", in Solid-State Circuits (ESSCIRC), European Conderence 2015, pp. 202-208

[32] G. Konstantopoulos, K. Papathanasiou, A. Samelis, "*Optimization of RF Circuits by Expert System monitored Genetic Computation*", in IEEE Circuits and Systems, ISCAS 2006, pp. 5247-5250

[33] A. Mangla, M. A. Chalkiadaki, F. Fadhuile, T. Taris, Y. Deval, C.C Enz *,"Design methodology for ultra low-power analog circuits using next generation BSIM6 compact model",* in Microelectronics Journal 44 (2013) pp.570-575

[34] I. Song, J. Jeon, H. S. Jhon, J. K. B. G. Park, J. D. Lee, "*A Simple Figure of Merit of RF MOSFET for Low-Noise Amplifier Design*", in IEEE Electron Device Letters, Vol. 29, No. 12, 2008, pp. 1380-1382

[35] M. T. Hsu, T. Y. Chih, G. R. Li, "*1.5V 5 GHz Low NoiseAmplifier With Source Degeneration*", in Proceedings of Asia-Pacific Microwave Conference 2006

[36] M. Rezvani, S. Ardalan, K. Raahemifar, "*High gain, low power, CMOS current reused LNA with noise optimization* ", in 26th IEEE Canadian Conference Of Electrical And Computer Engineering (CCECE), 2013

[37] P.-I. Mak, R. Martins, "*A 0.46mm2 4dB-NF unified receiver front-end for full-band mobile TV in 65nm CMOS"*, in IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), pp.172-174, Feb. 2011.

[39] http://www.ece.ucsb.edu/~long/ece145a/LNAdesign.pdf

APPENDIX

The octave code for the genetic algorithm and the OCEAN script for the genetic algorithm–Virtuoso Interface. This is the code for the common source with inductive degeneration LNA only. The cascode common source and the common gate ones follow the same principal but they use more genes in the individuals.

1. geneticAlgorithm.m

```
% Genetic Algorithm to Optimize the operation of an Inductive degenerated LNA
more off;

% First we must create the first random population. The population consists of
% "populationNum" Chromosomes and each Chromosome consists of "number_of_genes"
% genes.
populationNum = 20;
number_of_genes = 9;
population = zeros(number_of_genes,1,populationNum);

% The two genes are the turns and the inner radius of the inductor
turns = 0.5:0.25:5.5; % 21 available turns values
radius = 15:90; % 76 available radius values

% Generate a random Turn value for the initialization
xmin = 1;
xmaxT = 21;
xmaxR = 76;

xmin_bias = 0.2;
xmax_bias = 0.6;

% Place the values that have been extracted from the model analysis
population(:,:,1) = [2.25; 58; 0.5; 31; 4.25; 45; 3.9871; 57; 0.2586];

createParametersFile(population(:,:,1), 1, 0);
spectreOutputs = readSpectreOutputs();
fitness(1) = fitFunc(spectreOutputs(2:length(spectreOutputs)),2);

for i=2:populationNum
    LresonT = turns(round(xmin+rand*(xmaxT-xmin)));
    LresonR = radius(round(xmin+rand*(xmaxR-xmin)));
    LdegenT = turns(round(xmin+rand*(xmaxT-xmin)));
    LdegenR = radius(round(xmin+rand*(xmaxR-xmin)));
    LgT = turns(round(xmin+rand*(xmaxT-xmin)));
    LgR = radius(round(xmin+rand*(xmaxR-xmin)));
    width_per_finger = rand()*4 + 1; % 1 <= Width per Finger <= 5 with 2 decimal precision
    num_of_fingers = round(rand()*63) + 1; % 1 <= Fingers <= 64
    Vbias = xmin_bias+rand*(xmax_bias-xmin_bias);

    population(:,:,i) = [LresonT ; LresonR; LdegenT; LdegenR; LgT; LgR; width_per_finger;
                         num_of_fingers; Vbias];
    % For every chromosome generated we have to create an ocean file with
    % the design variables
    % For this, the createParametersFile() function is used
    createParametersFile(population(:,:,i), i, 0);
    % Afterwards there has to be a simulation for each of the generated
    % chromosomes and calculate each fitness function
    % Run the ocean script through SpectreRF, then this script will evaluate
    % the results reading the SpectreRF output file

    % The function readSpectreOutputs() will return the parameters for
    % evaluation
    spectreOutputs = readSpectreOutputs();

    while (spectreOutputs(1) != i && i < populationNum)
```

```
            spectreOutputs = readSpectreOutputs();
    end

    % Using the fitFunc() function the algorithm extracts the evaluation
    % function for every chromosome and stores it in the vector fitness
    fitness(i) = fitFunc(spectreOutputs(2:length(spectreOutputs)),
                              population(7,:,i)*population(8,:,i));
end
% After the first evaluation of the random genes the genetic algorithm will
% start to take place and try to generate better fitness functions. The next
% generation of chromosomes are created through three distinct procedures :
% 1. The elit individuals will pass to the next generation
% 2. Some individuals will be generated through crossover
% 3. Some individuals will be mutated
% The genetic algorithm will stop when the fitness function is no longer getting
% smaller or the generation 20 is reached

% The propabilities of crossover and mutation
crossover_prop = 0.5;
mutation_prop = 0.15;
%

generation = 1;
newPopulation = zeros(number_of_genes,1,populationNum);
i = 1;
max_generation = 400;

while (generation < max_generation)
    printf("Creating the next generation\n");
    % The fittest individual will pass to the next generation

    [fittest_individual index_of_fittest] = getFittest(fitness);
    index_of_fittest
    newPopulation(:,:,1) = population(:,:,index_of_fittest);
    fittest(1) = fittest_individual;

    % Loop over the population size and create new individuals with
    % crossover
    % Afterwards run the mutation function to generate mutations in the new
    % population
    for (i = 2:populationNum)
        % Crossover function
            [indiv1 indiv1Id] = tournamentSelection(population, fitness, populationNum);
            [indiv2 indiv2Id] = tournamentSelection(population, fitness, populationNum);
            newIndiv = crossoverFunc(indiv1, indiv2, crossover_prop, number_of_genes);
            newPopulation(:,:,i) = newIndiv;

        % Mutation function
        newPopulation(:,:,i) = mutationFunc(newPopulation(:,:,i), turns, radius, mutation_prop);
        end

    % After the new population is generated it is time to evaluate the
    % chromosomes
    population = newPopulation;

    printf("Evaluating generation %d\n",generation);
    %
    for (i = 1:populationNum)
        % For every chromosome generated we have to create an ocean file with
        % the design variables
        % For this, the createParametersFile() function is used

        for j = 1:i
            if sum(population(:,:,i) == population(:,:,j)) == number_of_genes && j < i
                fitness(i) = fitness(j);
                break;
            else
                if generation == max_generation && i == populationNum-1
                    createParametersFile(population(:,:,i), i, 1);
                else
                    createParametersFile(population(:,:,i), i, 0);
                end
```

```
                % Afterwards there has to be a simulation for each of the generated
                % chromosomes and calculate each fitness function
                % Run the ocean script through SpectreRF, then this script will evaluate
                % the results reading the SpectreRF output file

                % The function readSpectreOutputs() will return the parameters for
                % evaluation
                spectreOutputs = readSpectreOutputs();

                while (spectreOutputs(1) != i && i <= populationNum)
                        spectreOutputs = readSpectreOutputs();
                end

                % Using the fitFunc() function the algorithm extracts the evaluation
                % function for every chromosome and stores it in the vector fitness
                fitness(i) = fitFunc(spectreOutputs(2:length(spectreOutputs)),
                                        population(7,:,i)*population(8,:,i));
            end
        end
    end
    generation = generation + 1;
end
% Just end the simulation loop
i = i + 1;
createParametersFile(population(:,:,1), i, 1);
```

2. createParametersFile.m

```
% A function that generates the ocean script containing the design parameters
% the parameter stop is 1 when the algorithm has to be terminated
function [] = createParametersFile(Chromosome, i, stop, filename)

% Open the file and for every parameter in the Chromosome matrix there has to be a coresponding
design varible.
% The parameter i is used to mark the number of each chromosome

fid = fopen(filename, "w");
fprintf(fid, "i = %d\n", i );
fprintf(fid, "Stop = %d\n", stop);
fprintf(fid, "desVar( \"Pin\" 0 )\n" );
fprintf(fid, "desVar( \"LresonT\" %s )\n", num2str(Chromosome(1)));
fprintf(fid, "desVar( \"LresonR\" %su )\n", num2str(Chromosome(2)));
fprintf(fid, "desVar( \"LdegenT\" %s )\n", num2str(Chromosome(3)));
fprintf(fid, "desVar( \"LdegenR\" %su )\n", num2str(Chromosome(4)));
fprintf(fid, "desVar( \"LgT\" %s )\n", num2str(Chromosome(5)));
fprintf(fid, "desVar( \"LgR\" %su )\n", num2str(Chromosome(6)));
fprintf(fid, "desVar( \"width_per_finger\" %su )\n", num2str(Chromosome(7)));
fprintf(fid, "desVar( \"num_of_fingers\" %s )\n", num2str(Chromosome(8)));
fprintf(fid, "desVar( \"Vbias\" %s )\n", num2str(Chromosome(9)));
fclose(fid);

end
```

3. crossoverFunc.m

```
% This is the crossover function that is used to generate new children. The
% inputs are the two chromosomes that are going to produce an offspring and the
% crossover propability
function chromosome = crossoverFunc(indiv1, indiv2, crossover_prop, number_of_genes)
    chromosome = zeros(number_of_genes,1);
        % Loop through genes
    for i = 1:number_of_genes
        % Crossover
        if (rand() < crossover_prop)
```

```
        chromosome(i,1) = indiv1(i,1);
        else
        chromosome(i,1) = indiv2(i,1);
        end
    end
end
```

4. fitFunc.m

```
% This function is used to calculate the fitness function of the LNA
% The parameters that are used to calculate the function are :
% 1. The absolute distance from the center frequency : parameters(1)
% 2. The imaginary Z11 value at 5GHz : parameters(2) + i*parameters(3)
% 3. The imaginary Z22 value at 5GHz : parameters(4) + i*parameters(5)
% 4. The value of the parameter S21 at 5GHz : parameters(6)
% 5. The value of the Noise Figure at 5Ghz : parameters(7)
% 6. The absolute distance of the logarithmic S11 from 5Ghz : parameters(8)
% 7. The bandwidth : parameters(9)
% 8. The magnitute of the logarithmic S11 at 5Ghz : parameters(9)
% 9. The DC current
% Each parameter has a weight that denotes how important is each factor

% The goal is to minimize the value of the fitness parameter

function fitness = fitFunc(parameters, W)

    % If the S21 parameter is below zero we have to get the absolute value
    % and add it to the fitness function in order to make it larger. As a
    % reminder if the S21 parameter is below zero the design is shit and
    % the fitness should be high.
    % On the contrary if the S21 is above 0 we use the 1/x function in order
    % to minimize the fitness for the desired higher values of S21
    if parameters(6) > 0
        f_s21 = 1/parameters(6);
    else
        f_s21 = abs(parameters(6)) + 1;
    end

    f_center = abs(5 - parameters(1))^2;
    f_s11 = 0.35*parameters(2) + 0.65*parameters(3);
    f_s22 = 0.35*parameters(4) + 0.65*parameters(5);
    f_nf = parameters(7)^2;
    f_s11_center = abs(5 - parameters(8))^2;
    f_bw = parameters(9);
    f_s11_mag = 1 / parameters(10);
    f_id = abs(1 - parameters(11)/0.01)^2;

    fitness = f_center + 0.15 * f_s21 + 0.3 * f_nf + f_s11_center + 0.25 * f_s11_mag + 0.10 * f_s22
              + f_id;
end
```

5. getFittest.m

```
% This function returns the minimum fitness and the chromosome index of a
% population.
% As a reminder better individuals have lower fitness values
function [fittest chromosome] = getFittest(fitness)
    [fittest chromosome] = min(fitness);
end
```

6. mutationFunc.m

```
% This is the mutation function that is used to generate mutations in the new
% population. The inputs are the chromosome that is going to be mutated, the
% turns and radius are the availabe mutations and the mutation_prop is the
% mutation propability
% If a mutation is going to take place, then the algorithm must randomly chose
% how many genes are going to be mutated with a certain distribution.
% Afterwards the genes that are going to be mutated are again chosen randomly
function chromosome = mutationFunc(oldChromosome, turns, radius, mutation_prop)

    chromosome = oldChromosome;
    number_of_genes = length(chromosome);

    if (rand() <= mutation_prop)
        % Create random mutation
        xmin = 1;
        xmaxT = 21;
        xmaxR = 76;

        xmin_bias = 0.2;
        xmax_bias = 0.6;

        % The distribution variable holds the mutation propabilities for
        % the numbers of genes affected
        % It follows an exponential 1/x distribution, thus more genes
        % are less common to be submited for mutations
        x = 1:number_of_genes;
        Distribution = 1./(2.^x);
        Distribution(number_of_genes) = Distribution(number_of_genes-1);

        % After the number of mutated genes is generated, the algorithm
        % choses the genes randomly. Before applying the mutation it has
        % to check what the genes are : turns, radius, width per fingers
        % or number of fingers

        % The two next vectors show the position of the turns, and radius
        % genes.
        turnPos = [1 3 5];
        radiusPos = [2 4 6];

        % the vector genePool holds the indices of the genes. When multiple
        % genes untergo mutation the respective indice is removed from the
        % vector so it is not selected again
        genePool = 1:number_of_genes;

        random = rand();
        % find how many genes are going to be mutated
        for i = 1:number_of_genes;
            if random > Distribution(i)
                break;
            end
        end

        num_of_genes = i;

        % Mutate the genes
        for i = 1:num_of_genes
            randomGene = round(rand()*(length(genePool)-1)) + 1;
            genePool(randomGene);
            chromosome(genePool(randomGene)) = returnMutated(turns, radius,
                        turnPos, radiusPos, genePool(randomGene), xmin, xmaxR,
                        xmaxT, xmin_bias, xmax_bias);
            % Remove that gene from the genePool
            genePool(randomGene) = [];
        end
    end


end

% Returns the mutated chromosome after checking the identity of the genes
```

```
function chromo = returnMutated(turns, radius, turnPos, radiusPos, randomGene, xmin, xmaxR, xmaxT,
xmin_bias, xmax_bias)
    if ( sum(randomGene == turnPos) == 1 ) % Mutation to inductor Turns
        chromo = turns(round(xmin+rand*(xmaxT-xmin)));
    elseif ( sum(randomGene == radiusPos) == 1 ) % Mutation to inductor Radius
        chromo = radius(round(xmin+rand*(xmaxR-xmin)));
    elseif ( randomGene == 7 ) % Mutation to the width per finger
        chromo = rand()*4 + 1;
    elseif (randomGene == 8) % Mutation to the number of fingers
        chromo = round(rand()*63) + 1;
    else
        chromo = xmin_bias+rand*(xmax_bias-xmin_bias);
    end
end
```

7. readSpectreOutputs.m

```
% This function will read the output of the SpectreRF simulator
function spectreOutputs = readSpectreOutputs(filename)
    % Read the output of the SpectreRF simulation
    spectreOutputs = dlmread("filename", '', 0, 0);
end
```

8. tournamentSelection.m

```
% Selects the best out of 3 individuals in order to take part in the crossover
% operation
function [fittestIndividual indexOfFittest] = tournamentSelection(population, fitness,
populationNum)
    % Create a tournament population
        tournament = zeros(1, 3);
        tournamentFitness = zeros(1, 3);

    [f fittestId] = getFittest(fitness);
    % For each place in the tournament get a random individual
        for (i = 1:3)
            randomId = round(rand()*(populationNum-1))+1;
        while(randomId == fittestId)
        randomId = round(rand()*(populationNum-1))+1;
        end
            tournament(i) = randomId;
            tournamentFitness(i) = fitness(randomId);
        end

        % Get the fittest from the tournament population
    [fittestIndividual chromosome] = getFittest(tournamentFitness);
    fittestIndividual = population(:,:,chromosome);
    indexOfFittest = tournament(chromosome);
end
```

9. commonSourceLNA.ocn

```
simulator( 'spectre )
design( "/home/rfic6/simulation/testing/spectre/schematic/netlist/netlist")
resultsDir( "/home/rfic6/simulation/testing/spectre/schematic" )
modelFile(
    '("/opt/CAD/Libraries/tsmc90_LP/tsmc90_LP/tsmcN90rf/../models/spectre/crn90lp_2d5_lk_v1d2.scs"
     "tt_mim")
    '("/opt/CAD/Libraries/tsmc90_LP/tsmc90_LP/tsmcN90rf/../models/spectre/crn90lp_2d5_lk_v1d2.scs"
```

```
   "tt_rfind")
    '("/opt/CAD/Libraries/tsmc90_LP/tsmc90_LP/tsmcN90rf/../models/spectre/crn90lp_2d5_lk_v1d2.scs"
      "tt_rfmos")
)
analysis('sp ?ports list("/PORT1" "/PORT0")  ?start "1e9"  ?stop "10e9"
              ?lin "1000"  ?donoise "yes"  ?oprobe "/PORT1"  ?iprobe "/PORT0"  )
analysis('dc ?saveOppoint t  )

out = outfile("/home/rfic6/prj/papajim/testing2/SpectreOutputs.dat", "w")

load("/home/rfic6/prj/papajim/testing2/parameters.ocn") ;;; Load the Design Parameters generated
                                                        ;;; from the GA
new = i

while( Stop == 0
       envOption(
              'analysisOrder  list("sp" "dc")
       )

       temp( 27 )
       run()
       fprintf(out "%d\t" i)
       fprintf(out "%f\t" xmax(db(sp(2 1 ?result "sp")) 1 )/1000000000)

       fprintf(out "%f\t" abs(value(real(sp(1 1 ?result "sp")) 5e9   )*50))
       fprintf(out "%f\t" abs(value(imag(sp(1 1 ?result "sp")) 5e9   )*50))

       fprintf(out "%f\t" abs(value(real(sp(2 2 ?result "sp")) 5e9   )*50))
       fprintf(out "%f\t" abs(value(imag(sp(2 2 ?result "sp")) 5e9   )*50))

       fprintf(out "%f\t" value(db(sp(2 1 ?result "sp")) 5e9 ))
       fprintf(out "%f\t" value(db10(getData("F" ?result "sp_noise")) 5e9))
       fprintf(out "%f\t" xmin(db(sp(1 1 ?result "sp")) 1 )/1000000000)


       x = bandwidth(sp(2 1 ?result "sp") 3 "band" )
       if(x
              fprintf(out "%f\t" x/1000000000)
              fprintf(out "5\t")
       )

       fprintf(out "%f\t" abs(value(db(sp(1 1 ?result "sp")) 5e9   )))

       fprintf(out "%f\t" pv("M0" "id" ?result "dcOpInfo"))
       close(out)

       ;;; Open the parameters.ocn file until the genetic algorithm generates
       ;;; a new one
       while(new == i
              load("/home/rfic6/prj/papajim/testing2/parameters.ocn") ;;; Load the Design
                                                                      ;;;Parameters generated from
                                                                      ;;;the GA
       )
       new = i
       out = outfile("/home/rfic6/prj/papajim/testing2/SpectreOutputs.dat", "w")

)
close(out)
```