TECHNICAL UNIVERSITY OF CRETE

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

THESIS PROJECT

« Design flow and Implementation of a DSB (Double Side Band) mixer using Gilbert double balanced mixer architecture, for a fast-hopping frequency synthesizer for MB-OFDM UWB (Ultra Wideband) in IBM 0.18um CMOS technology. »

Select Committee:
Apostolos Samelis (Supervisor)
Matthias Bucher
Alexandros Potamianos

Andreou Charalambos
Chania, July 2006

PREFACE

During the last years, many people have predicted there soon would be little or no need for analog circuits because the world would rely on digital circuits. Nevertheless, although many applications have indeed replaced many analog circuits with their digital counterparts, the need for good analog circuit design remains strong. In addition, new applications continue to appear in which speed and power consumption requirements often need the use of high-speed analog front ends. Furthermore as integrated circuits become larger due to the system integration, it is much more likely that at least some portion of a modern integrated circuit will include analog circuitry required to interface to the real world. Usually this analog circuitry, despite the fact that it constitutes only a small portion of the total chip are, may be the limiting factor on overall system performance and the most difficult part of the Integrated Circuit design.

The revolutionary advances in solid-state devices and integrated circuits (ICs) have brought the world of wireless communications into a completely new era. The once bulky and luxury mobile phones for example, are now handy and common tools for every individual from everywhere. Here new technologies act as the driving force which has been pushing the personal wireless communications market into today's boom. In return, the market motivates the technology by putting even aggressive demands on the performance, cost, form-factor and power consumption of the mobile receiver/transceiver.

Analog RF circuits must process analog signals with a wide dynamic range at high frequencies. The trade off's involved in the design of such circuits can be summarized in the "analog design octagon" shown in figure 1, where almost any two of the six parameters trade with each other to some extent.

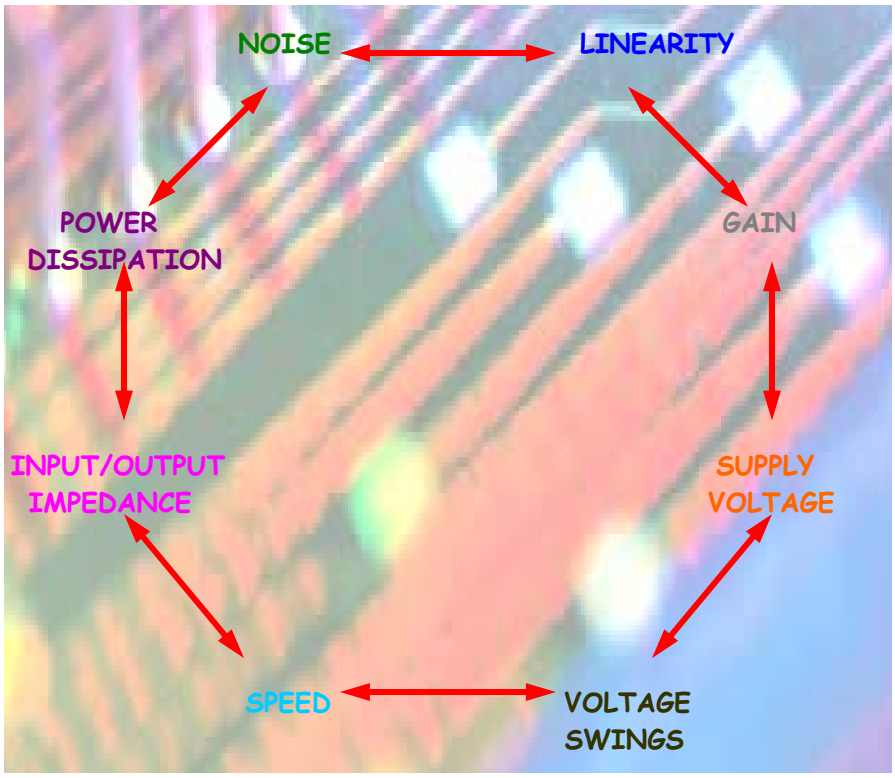
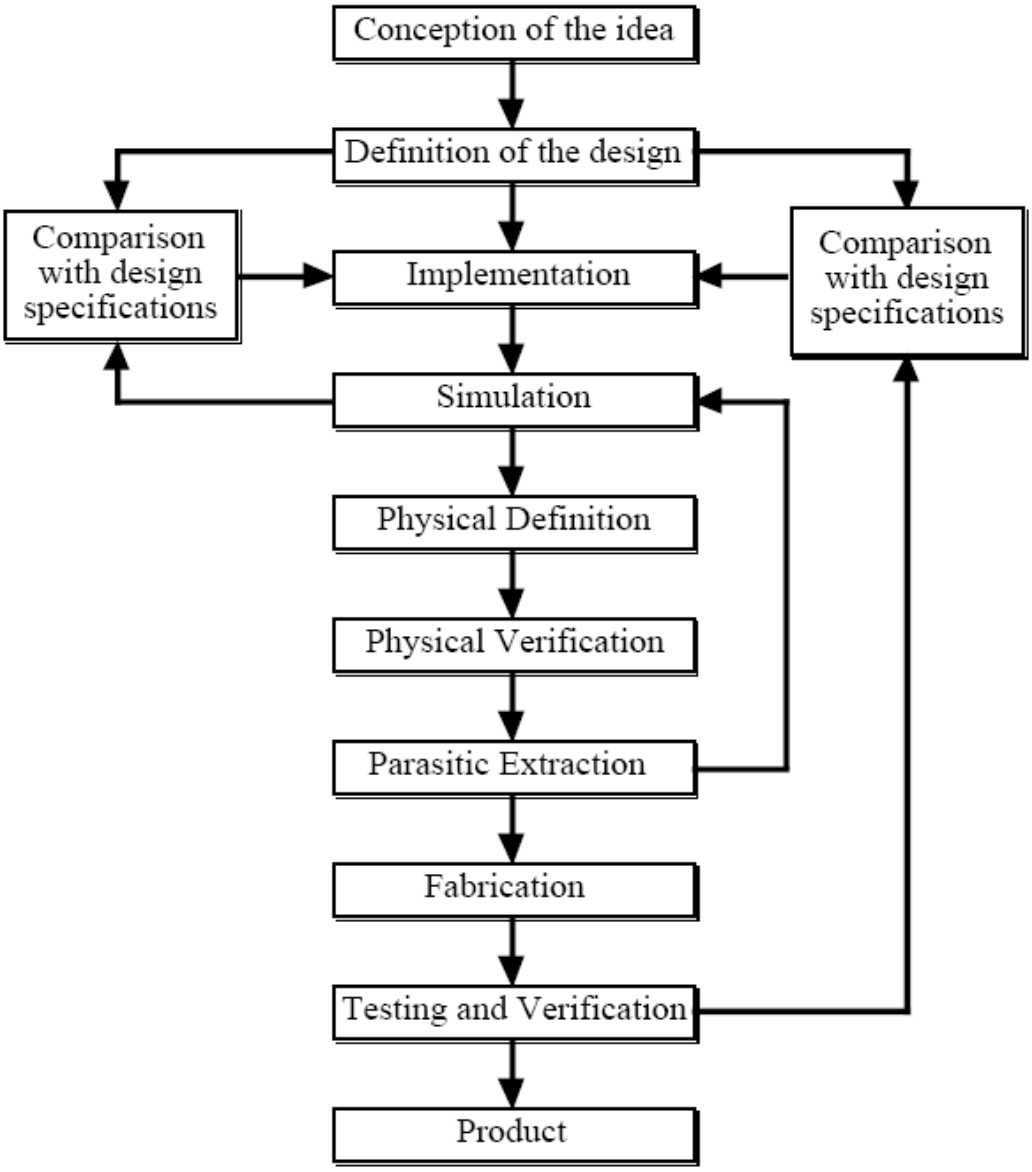


Figure 1: Analog Design Octagon

In this diploma thesis we present, the Design flow and Implementation of a DSB (Double Side Band) mixer using Gilbert double balanced mixer architecture, for a fast-hopping frequency synthesizer for MB-OFDM UWB (Ultra Wideband) using Cadence and Agilent ADS in IBM 0.18um CMOS technology.

Unlike conventional wireless systems, which use narrowband modulated carrier waves to transmit information, Ultra-Wideband transmits over a wide swath of radio spectrum, using a series of very narrow and low-power pulses. The combination of broader spectrum, lower power and pulsed data means that Ultra-Wideband causes significantly less interference than conventional narrowband radio solutions while safely coexisting with other wireless technologies. Modern UWB systems use special modulation techniques, such as Orthogonal Frequency Division Multiplexing (OFDM), to occupy these extremely wide bandwidths. In addition, the use of multiple bands in combination with OFDM modulation can provide significant advantages to traditional UWB systems.

THE ANALOG IC DESIGN PROCESS



CONTENTS

NOMENCLATURE

BW Bandwidth

dB Decibels

dBm Decibels with respect to 1mW

C Capacity

CDF Component Description Format

DBM Double Balanced Mixer

DRC Design Rule Checking

DSB NF Double-Sideband Noise Figure

F Noise Factor

gm Gate Transconductance of a MOSFET

IC Integrated Circuit

IF Intermediate Frequency

IM Intermodulation

IM3 Third-Order Intermodulation Products

IMR Intermodulation Ratio

IP3 Third-Order Intercept Point

IIP3 Input-IP3

LNA Lower Noise Amplifier

LO Local Oscillator

LPE Layout Parasitic Extraction

LVS Layout Versus Schematic)

NMOS N-channel MOSFET

NF Noise Figure

PMOS P-channel MOSFET

Pn Noise Power

PRE Parasitic Resistance Extraction

Ps Signal Power

PSS Periodic Steady State

R Resistance

RF Radio Frequency

RFIC Radio Frequency Integrated Circuits

SBM Single Balanced Mixer

SNR Signal-to-Noise Ratio

SSB NF Single-Sideband Noise Figure

UWB Ultra Wide Band

VRF Voltage Amplitude of RF signal

VLO Voltage Amplitude of LO signal

VoIP Voice Over IP

CHAPTER 1

1.1 UWB Overview

UWB is considered to be an exciting and breakthrough technology and wireless radios are capable of carrying extremely high data rates for up to 250 feet with little transmit power. UWB technology is defined as any wireless transmission scheme that occupies more than 500 MHz of absolute bandwidth.

It's believed that the current emphasis on low power, low interference and low regulation makes the use of UWB an attractive option for current and future wireless applications. Ideal targets for UWB systems are low power, low cost, high data rates, precise positioning capability and extremely low interference. Although UWB systems are years away from being widespread, the technology is changing the wireless industry today. UWB technology is different from conventional narrowband wireless transmission technology – instead of broadcasting on separate frequencies, UWB spreads signals across a very wide range of frequencies. The typical sinusoidal radio wave is replaced by trains of pulses at hundreds of millions of pulses per second. The wide bandwidth and very low power makes UWB transmissions appear as background noise.

The capacity of a communication channel, C, with a bandwidth of BW and additive white noise, in accordance to Shannon theory, is:

$$C = BW \log_2(1 + S/N),$$

where S and N are signal power and noise power respectively.

This formula gives an upper limit for the data rate. Also we can derive that, the higher the bandwidth, the higher the maximum achievable data rate.

The key benefits of UWB can be summarized as:

I. High data rates

The high data rates are perhaps the most compelling aspect from a user's point of view and also from a commercial manufacturer's position. Higher data rates can enable new applications and devices that would not have been possible up until now. Speeds of over hundreds Mbps have been demonstrated, and the potential for higher speeds over short distances is there due to the extremely large bandwidth occupied by UWB.

II. Low equipment cost

The ability to directly modulate a pulse onto an antenna is perhaps as simple a transmitter as can be made, leading many manufacturers to get excited by the possibilities for extremely cheap transceivers. This is possible by eliminating many of the components required for conventional sinusoidal transmitters and receivers.

III. Multipath immunity

The narrow pulses used by UWB, which also give the extremely wide bandwidth, if separated out provide a fine resolution of reflected pulses at the receiver. This is important in any wireless communication, as pulses (or sinusoids) interfering with each other are the major obstacle to error-free communication.

IV. Ranging and communication at the same time

The use of both precise ranging and high speed data communication in the same wireless device presents intriguing possibilities for new devices and applications.

1.2 Mixer Overview

Mixers are found in virtually all wireless communication systems. They are frequency translating devices that convert input signals from one frequency to another by mixing these signals with another signal of known frequency. One reason frequency translation is a necessary process in wireless transmission is that information signals such as human speech or digital data are usually low frequency signals and are not suitable for a wireless channel. Another is that wireless channels are common channels that are shared by many signals and these signals must be separated into different frequency bins so that electronic circuits (which contain frequency selective components) can keep them from destructively interfering with each other. Among many other properties, frequency is one that is most easily exploited in signal identification.

Mixers can be classified into two broad categories: passive or active. The most commonly available and used are passive diode mixers since they are easier to design and more thoroughly understood. Active mixers, on the other hand, involve transistors and the most popular ones are built from the basic Gilbert cell structure. Among these types, the Gilbert cell structure has the most desirable characteristics in terms of isolation and

harmonic suppression due to its balanced structure. Most down converting mixers are three-port devices, as shown in Figure 1.1. They take two input signals: the RF and the LO (local oscillator) signals. The output is a mixing product of these two inputs and is an intermediate frequency (IF) signal.

Mixers perform the mixing operation by multiplying the two input signals. The output, IF, is the product of the two signals RF and LO, and it contains the sum and difference of the two input frequencies. In receivers, the lower frequency component is usually the desired one and can be obtained by lowpass filtering the mixer output signal.

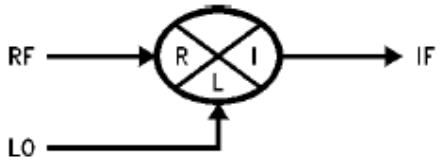


Figure1.1: Three port mixer

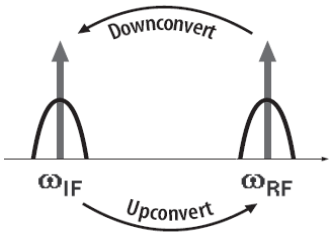
1.3 Downconversion Mixer Overview

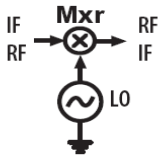
Radio communication requires that we shift a baseband information signal to a frequency or frequencies suitable for electromagnetic propagation to the desired destination. At the destination, reverse this process, shifting the received radiofrequency (RF) signal back to baseband to allow the recovery of the information it contains.

$$Y(\omega) = \frac{A}{2}[X(\omega - \omega_0) + X(\omega + \omega_0)]$$

Upconverted Component

Downconverted Component





$$\omega_{LO} = \omega_{RF} - \omega_{IF}$$

Figure 1.2:Mixer upconvert and downconvert operation.

- In downconversion mixers the output stage is at IF as shown in Figure 1.3.
- In upconversion mixers the output stage is at RF.

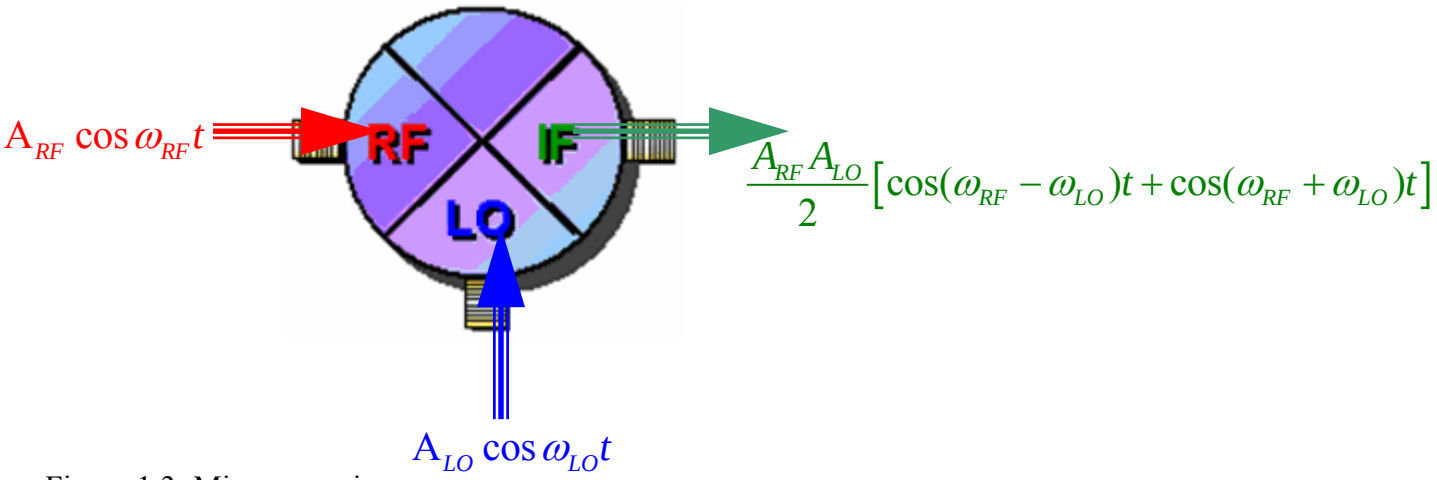


Figure 1.3: Mixer overview.

In a design of a mixer for a wireless system we must pay attention to the following design issues:

- Noise figure which has influence on receiver sensitivity
- Linearity which has influence the receiver blocking performance
- Conversion gain which influence the noise impact of the following stages of the mixer.
- Power match which means that we want the maximum voltage gain rather than power match.
- Isolation, so we want the minimum possible interaction between the RF, IF and LO ports.
- Power. We want low power dissipation due to the fact that in most cases our designs are for wireless portable devise, where power dissipation is more than important.

I. Ideal Mixer Respond

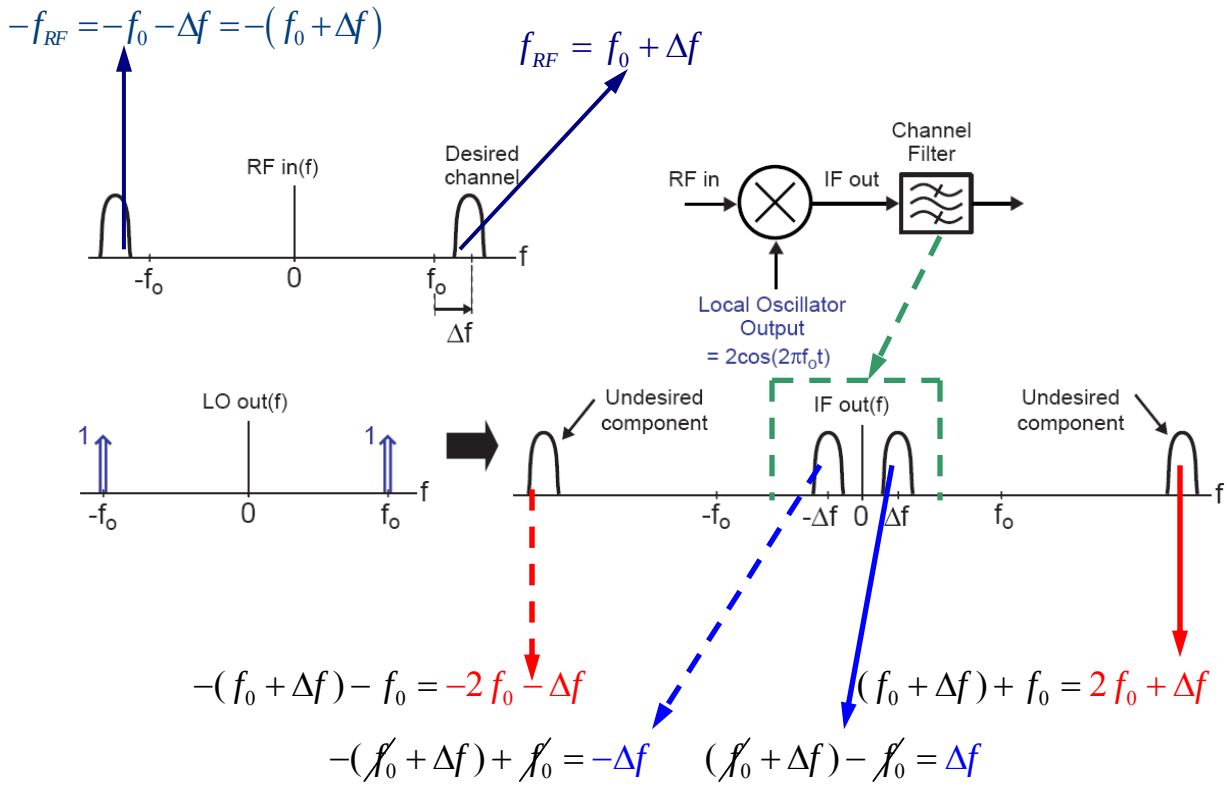


Figure 1.4: Ideal mixer respond

The RF spectrum is converted to a lower IF center frequency. We can use a filter at the IF Output to remove undesired high frequency components. The "Channel filter" of figure 2.5 must have a center frequency at «Δf» and a bandwidth of «2Δf» , ensuring that

$$(f_0+\Delta f)+f_0-\Delta f > (f_0+\Delta f)-f_0+\Delta f \Rightarrow 2f_0 > 2\Delta f \Rightarrow f_0 > \Delta f$$

If the IF frequency is nonzero, we call the receiver as heterodyne of low IF receiver.

If the IF frequency is zero, then we call the receiver as homodyne receiver.

II. Problem of Image Interference (Aliasing)

In heterodyne receivers, where IF frequency is nonzero, we have an image band for a giver desired channel band. While each wireless standard imposes constraints upon the signal emissions by its own users, it may have no control over the signals in other bands. The image power can therefore be much higher than that of the desired signal, requiring

proper “image rejection”. As a result, the frequency content in image band will combine with that of the desired channel at the IF output, giving the image interference which cannot be removed through filtering at the IF output.

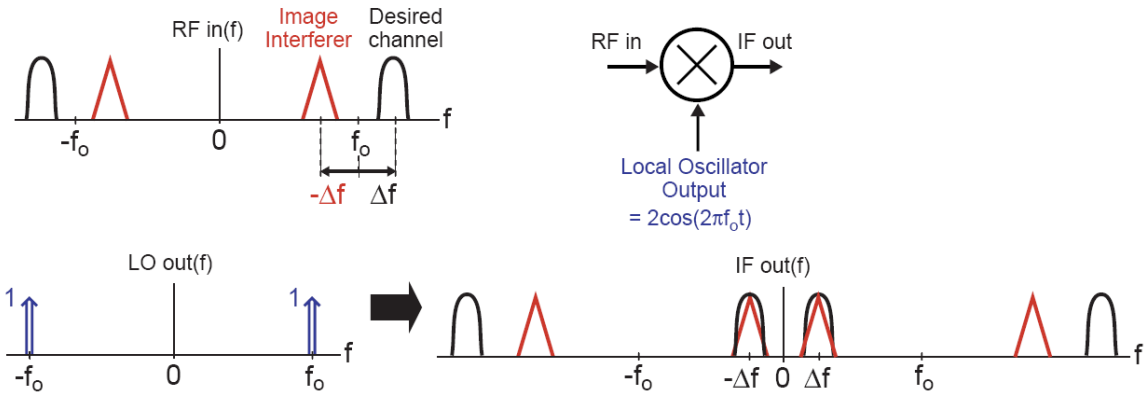


Figure 1.5: Problem of image interferers in mixers heterodyne receivers.

III. Solutions to the Problem of Image Interference

A) Image-Reject Filter

A common approach to suppressing the image is through the use of an “image-reject filter”, placed before the mixer. Therefore we prevent the image content from aliasing into the desired channel. For the success of this approach we need to have high IF frequency due to the fact that low IF frequencies require high filter Q. The drawback is that filter Q cannot be arbitrarily large. Further more filter bandwidth must be large enough to pass all channels.

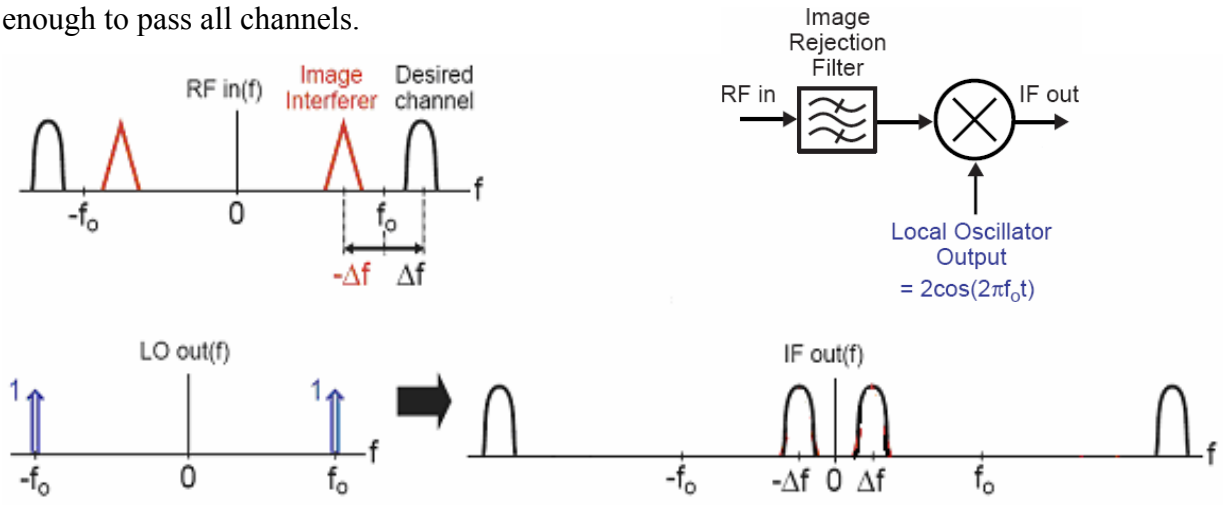


Figure1.7: Image reject filter.

B) Image Reject Mixer

Another solution for the removal of Image Interferers is the Image Reject Mixer. Using this approach we can cancel out the image using an image reject mixer, as shown in figure 1.7, rather than filtering it out. This method allows low IF frequencies to be used, thus removing the need of high Q filter. On the other hand we have some drawbacks. The level of image rejection is determined by mismatch in gain and phase of the top and bottom paths. Furthermore, practical architectures can do image rejection up to 40-50 dB.

Graphical explanation of Image Reject Mixer in Frequency domain

As a **1st step** we can multiply all the instances of the RF (4 instances) signal with all the instances of the LO (2 instances) signal. For now we're assuming that the RF signal is purely real. The frequencies of the mixed IF signal are calculated as before. The amplitude of the signal, for the first path is always "1". For the second path we have "j" for the multiplication with "$-f_1$" and "-j" for the multiplication with "f_1".

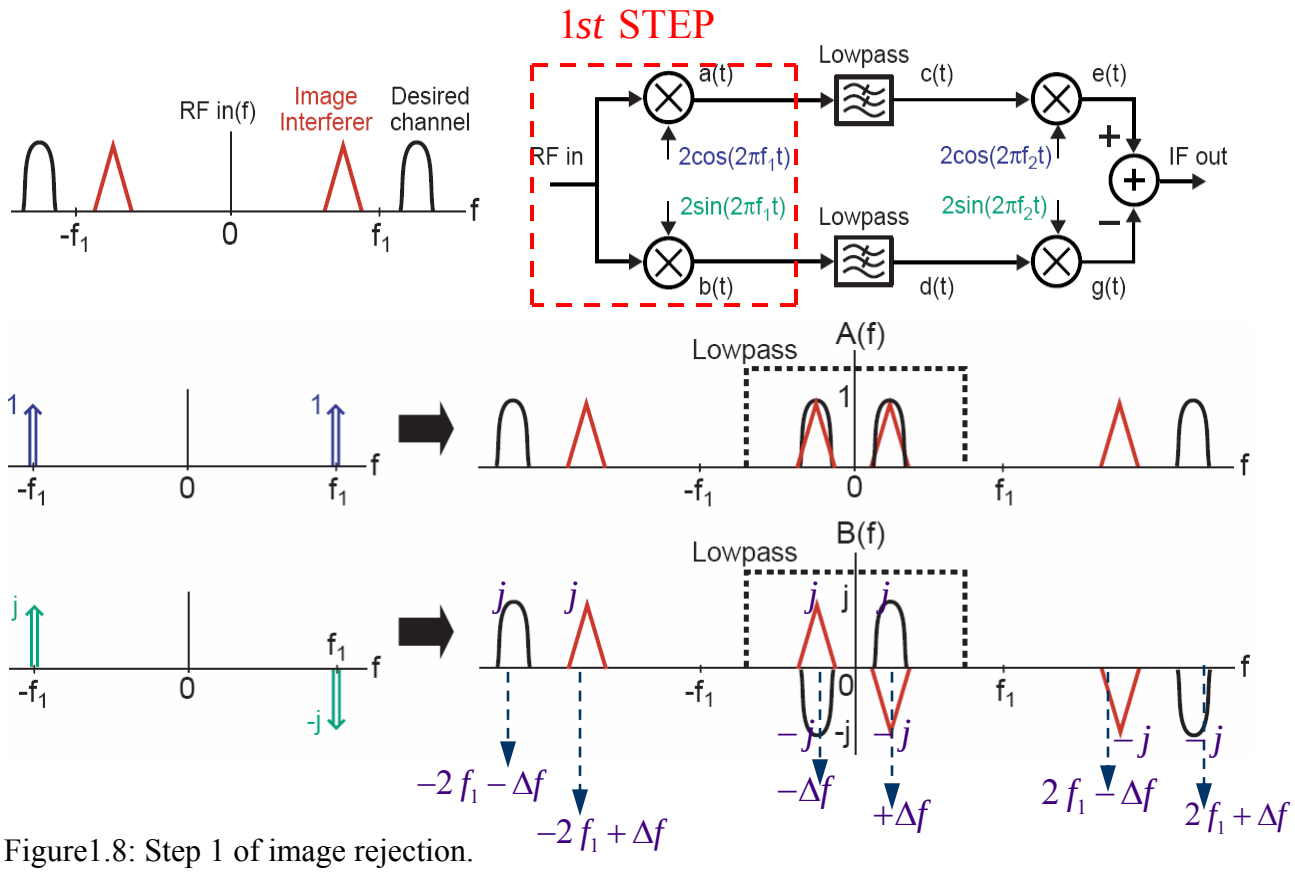


Figure1.8: Step 1 of image rejection.

As a **2nd step** we can use the Lowpass filters of figure 1.9 to reject the high frequency components and leave only the components at the frequencies "$-\Delta f$" and "Δf" to pass through the filter.

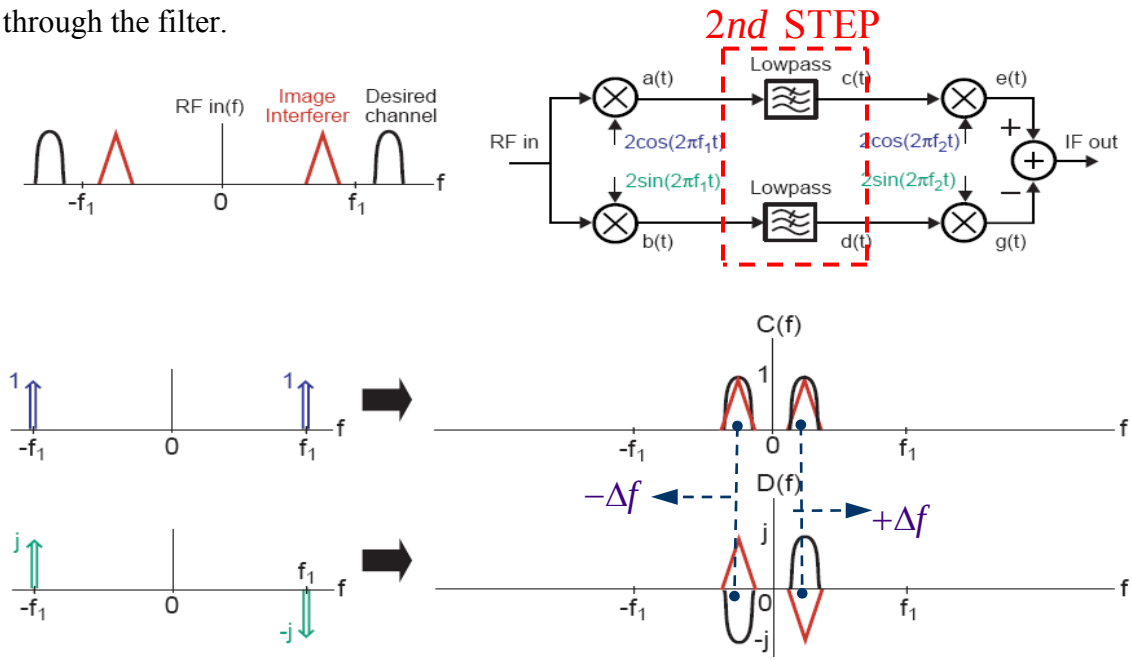


Figure1.9: Step 2 of image rejection.

The **3rd step** is the multiplication of the signals of each path, from step 2 with $2\cos(2\pi f_2 t)$ and $2\cos(2\pi f_2 t)$ respectively, as show at figure 1.10

The amplitude of the instances after the multiplication can be calculated as:

A $\Rightarrow$ $1*1=1$

B $\Rightarrow$ $\left.\begin{matrix}1*1=1\\1*1=1\end{matrix}\right\}=1+1=2$

C $\Rightarrow$ $1*1=1$

D $\Rightarrow$ $(j)*(-j)=-j^2=1$

E $\Rightarrow$ $\left.\begin{matrix}(-j)*(j)=-j^2=1\\(j)*(-j)=-j^2=1\end{matrix}\right\}=2$

F $\Rightarrow$ $(-j)*(j)=-j^2=1$

G $\Rightarrow$ $j*j=j^2=-1$

$$\text{H} \quad \Rightarrow \quad \left.\begin{matrix}(-j)*(-j)=j^2=-1\\(j)*(j)=j^2=-1\end{matrix}\right\}=-1-1=-2$$

$$\text{I} \quad \Rightarrow \quad (-j)*(-j)=j^2=-1$$

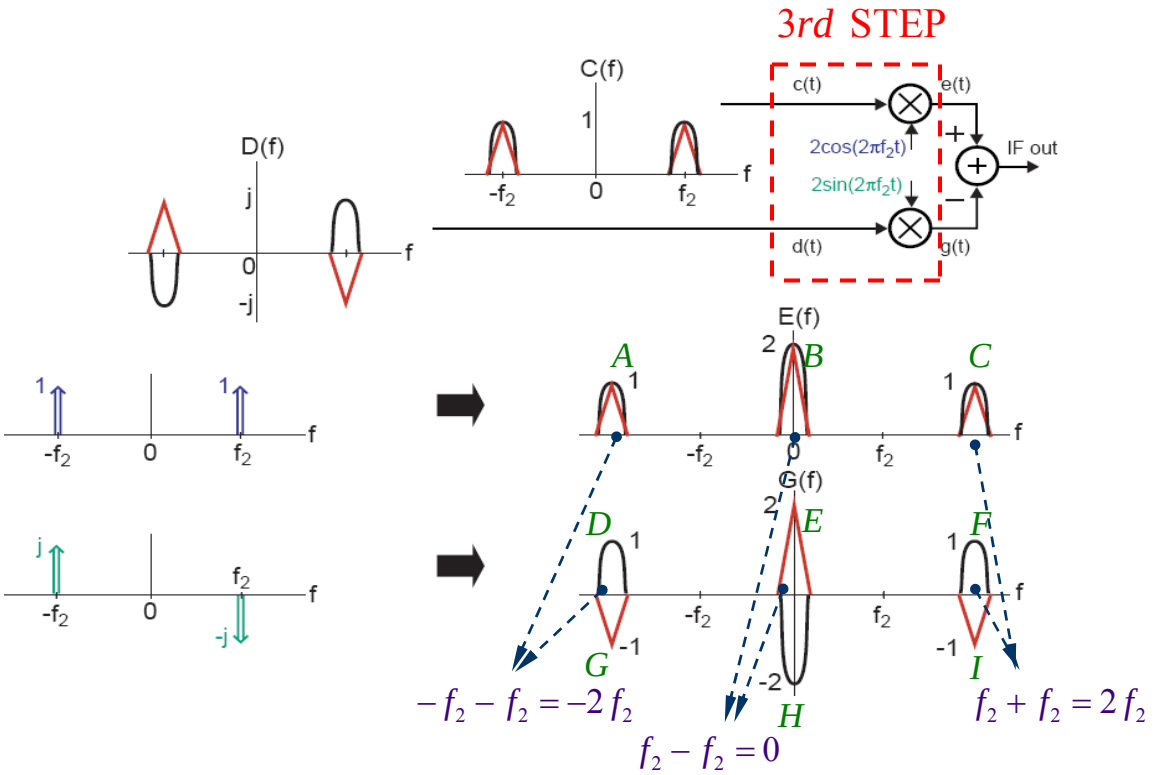


Figure1.10: Step 3 of image rejection.

In **4th step** we add the signals (components) of each pathway in order to cancel the undesired interferers, as shown in figure1.11. After that step we have the pure signal without any interferers in the specific band. The two interferers that remain at the frequencies $2f_2$ and $-2f_2$ can be filtered out using a Baseband Filter.

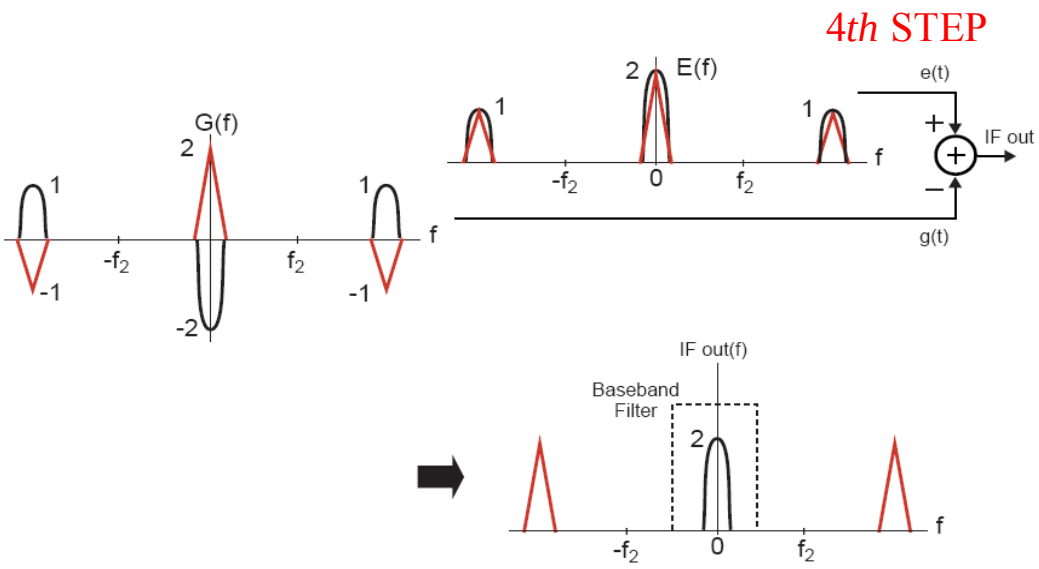


Figure1.11: Step 4. Adding of the signals (components) of each pathway for the cancellation of the undesired interferers.

After the filtering we can take the pure signal without any interferers as shown in figure1.12, where we can also see the whole implementation of the image reject mixer.

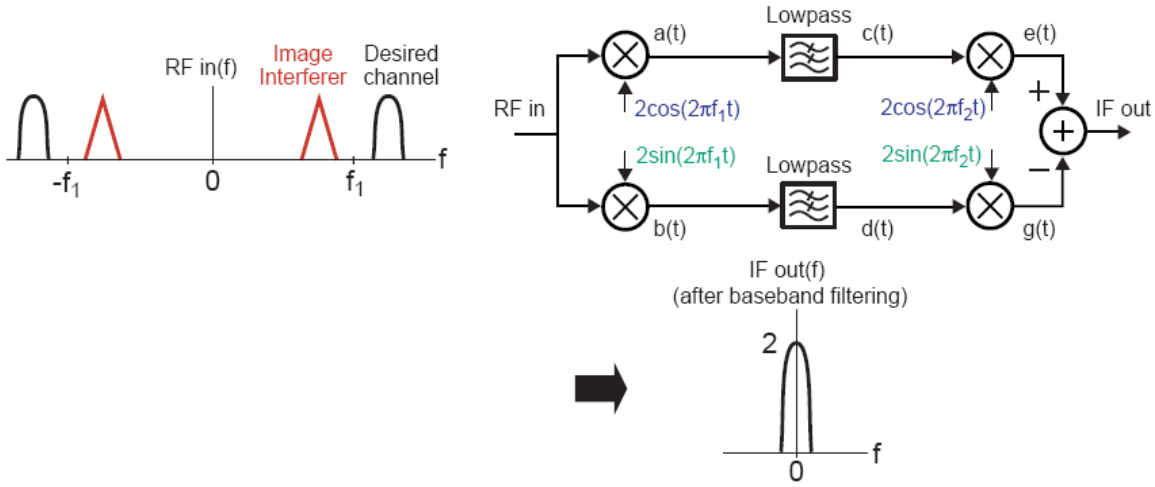


Figure1.12: Final implementation of image reject mixer.

The modification of the above method as shown figure 1.13 can be used when RF is purely imaginary. So at the 3rd Step of the above implementation we can interchange $2\cos(2\pi f_2 t)$ and $2\sin(2\pi f_2 t)$ with each other. Therefore, "sine" and "cosine" demodulators are switched in the second half of the image rejection mixer. Furthermore the two paths are now added rather than subtracted. If we don't do this "trick", then

following again the above method (4 steps) which was used for the cancellation of the interferers for purely real RF, we can derive that the above architecture cancels both desired and image signals which is destructive. This architecture works perfectly for imaginary RF, but on the other hand it cancels out the desired channel for purely real RF.

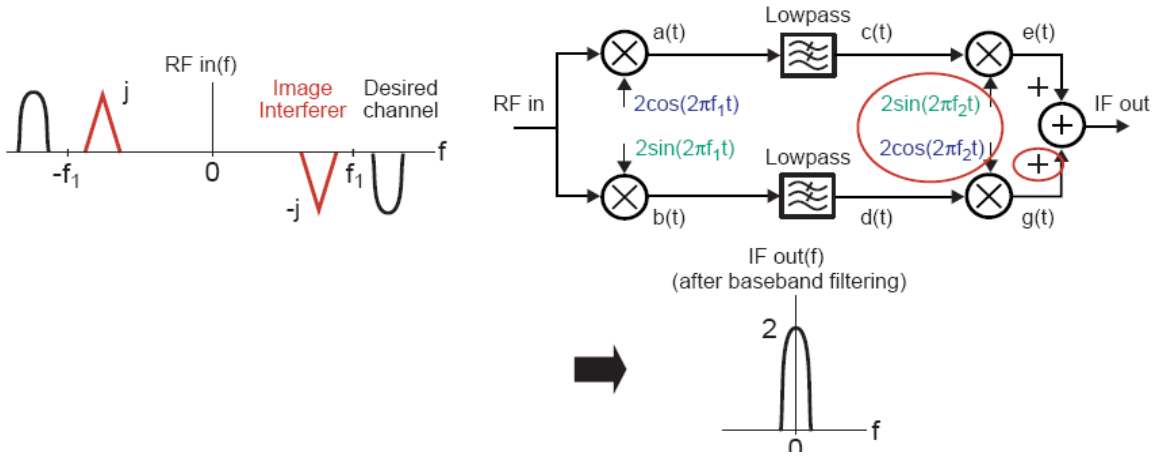


Figure1.13: Modification of the final implementation of image reject mixer.

In figure 1.14 we can see the Overall Mixer Architecture that is capable to cancel all the interferers, and in addition to the above architectures, both real and imaginary parts of RF input pass through. The success of this architecture lies on the combination of both of the above architectures, the one that works for real RF and the one that works for imaginary RF. Therefore is like two different mixers that they share the common parts of the architecture (1^{st}&2^{nd} steps) and have their own parts where is needed (3^{rd}&4^{th} steps).

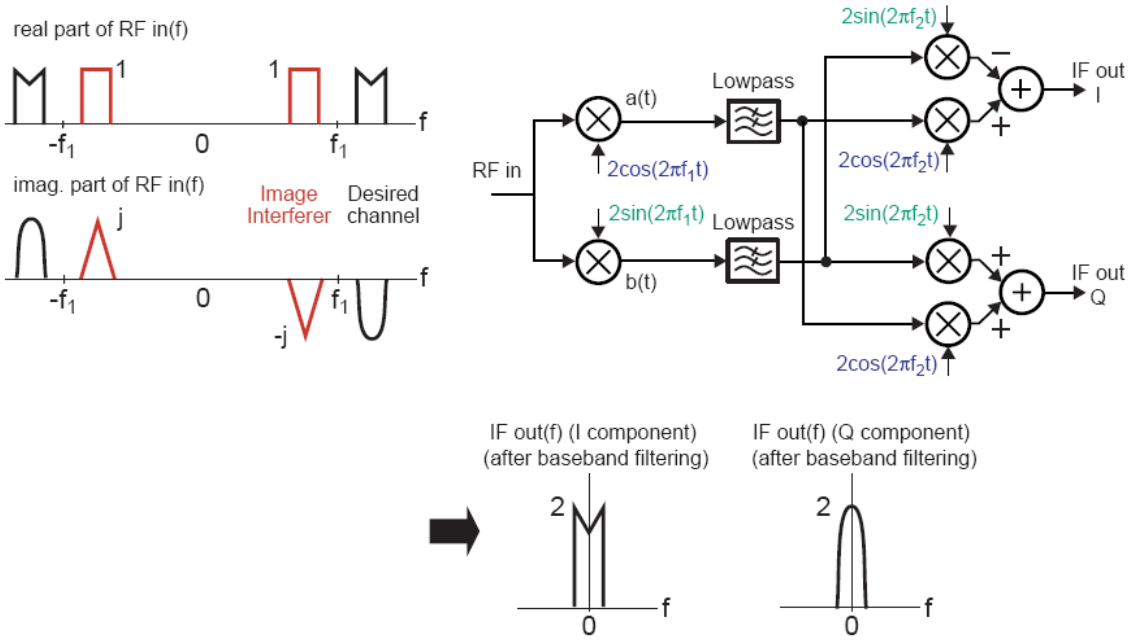


Figure1.14: Overall Mixer Architecture

1.3.1 Simple Switch Mixer

Mixers perform frequency translation by multiplying two signals (and possibly their harmonics). Downconversion mixers employed in the receive path have two distinctly different inputs, called the RF port and the LO port. The RF port senses the signal to be downconverted and the LO port senses the periodic waveform generated by the local oscillator. This operation can also be viewed as multiplication of the RF signal by a rectangular waveform as we can see Fig.1.15. The circuit is a linear, time-variant system with respect to the RF port and a nonlinear, time-variant system with respect to the LO port. It is desirable that the switch turn on and off as abruptly as possible.

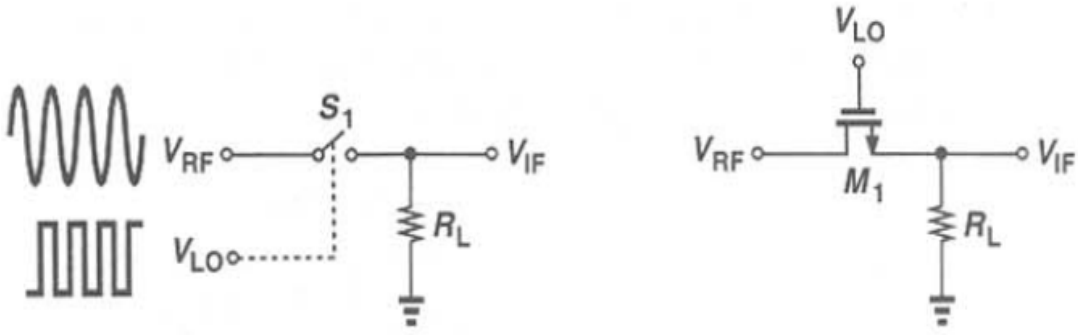


Figure 1.15: (a) Simple switch used as mixer, (b) implementation of switch using an NMOS device.

The signal amplified by the LNA is applied to the RF port of the mixer. Thus, this port must exhibit sufficient low noise and high linearity, because nearby interferers are amplified by the LNA and hence can produce stronger intermodulation products. In addition, as the RF input signal varies, the gate-source overdrive voltage of M_1 transistor and hence its on-resistance change, introducing nonlinearity in the voltage division between M_1 and R_L.

1.3.2 Passive and Active Mixers

Passive mixers do not provide any gain and an example of a passive mixer is shown in Fig. 1.15(b). In contrast, active mixers generally provide gain and an example of active mixer is shown in Fig. 1.16. In active mixers, the RF input varies the drain current of M_1, and M_2 and M_3 function as a switching pair driven by the LO. So, the drain current of M_1 is in essence multiplied by a square wave as it is routed to R1 and R2 alternately.

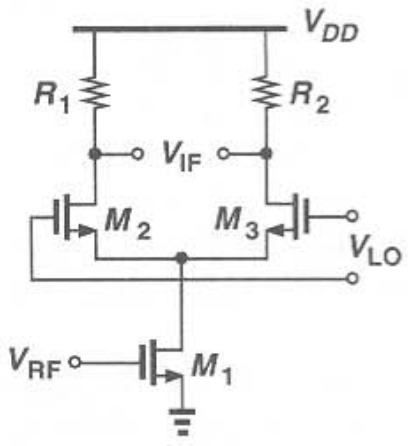


Figure 1.16: Active Mixer

In addition of their gain, active mixers reduce the noise contributed by subsequent stages and are widely used in RF systems. On the other hand, passive mixers usually achieve a higher linearity and speed and find application in microwave and base station circuits.

PASSIVE MIXER

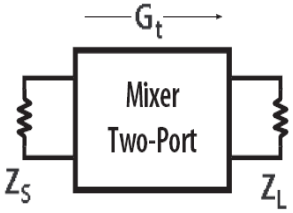
- No static current dissipation
- No gain greater than 2/p: high NF

- High linearity
- Large LO drive needed.

ACTIVE MIXER (Gilbert cell)

- Static current dissipation
- High conversion gain and low NF
- More noise sources (flicker noise)
- Linearity mainly limited by RF pair
- Clear design trade-off between gain, noise, linearity and power consumption

1.3.3 Conversion Gain



$$\text{Conversion Gain } (G_t) = \frac{\text{Power into Load, } Z_L}{\text{Available Power from Source, } Z_S}$$

Figure 1.17: Conversion gain.

The "voltage conversion gain" of a mixer is defined as the ratio of the rms voltage of the IF signal to the rms voltage of the RF signal, with these two signals centered around two different frequencies. The voltage conversion gain can be measured by applying a sinusoid at ω_{RF} and examining the amplitude of the downconverted component at ω_{IF}.

The "power conversion gain" of a mixer is the IF power delivered to the load divided by the available RF power from the source. If the input impedance and the load impedance of the mixer are both equal to the source impedance, then the voltage conversion gain and power conversion gain of the mixer are equal when expressed in db.

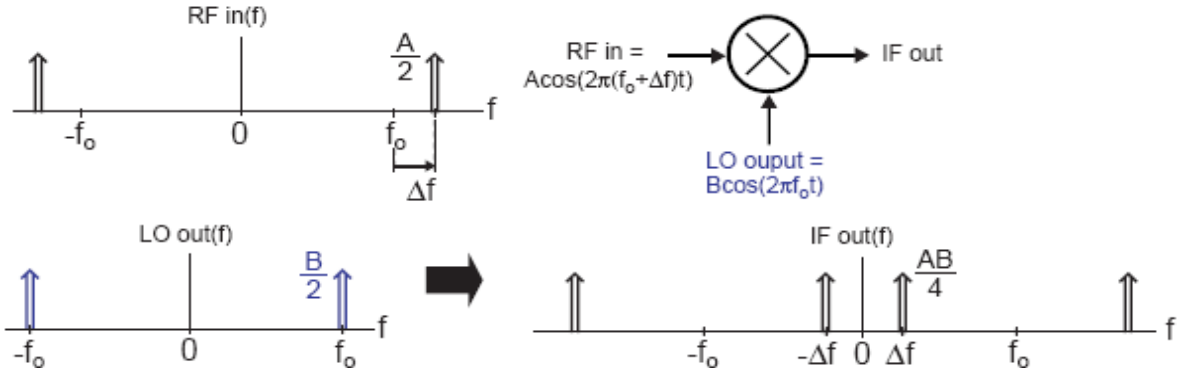


Figure 1.18: Voltage conversion gain.

For an ideal mixer with → RF input = Asin(2π(fo+ Δf)t)

and sine wave → LO signal = Bcos(2πfot)

$$IFout(t) = \frac{AB}{2}[\cos(2\pi(\Delta f)t) + \cos(2\pi(2f_0 + \Delta f)t)]$$

$$\Rightarrow Voltage\_Conversion\_Gain = \frac{AB/4}{A/2} = \frac{AB/2}{A} = \frac{B}{2}$$

The main benefit of high voltage gain is that the noise of later stages will have less of an impact. On the other hand, high voltage gain may be accompanied by higher noise figure than could be achieved with lower voltage gain. Furthermore high voltage gain may accompanied by nonlinearities that limit interference rejection.

1.3.4.0 Noise

Noise in electrical systems is defined as random fluctuations in voltage and current. It can be generated internally by components employed in the system or externally by electrical radiation from other systems or induced mechanical vibrations. Transistors exhibit flicker noise, which is caused by a change in conductance caused by a relatively slow process (e.g. the exchange of charge with surface traps or metallic impurities through tunneling), and shot noise, which is due to random one-way crossings of some barrier by discrete quantities of charge.

1.3.4.1 Noise in mixers

Noise in mixers is defined as any random interference that is unrelated to the signal of interest. The different types of noise in all circuits are thermal noise from resistors and channel resistance of MOSFETs. Shot Noise is associated with the transfer of charge across an energy barrier. Flicker Noise arising from random trapping of charge at the oxide-silicon interface of MOSFETs. In addition, mixers also suffer noise from the thermal noise generated by the output resistance of the LO and noise contributed by the switching pairs.

The strength of a signal relative to noise is often quantified by the signal-to-noise ratio (S/N). It is defined as the ratio of the signal power (Ps) to the noise power (Pn). The noise

factor (F) measures the degradation in (S/N) due to the noise added by a circuit. The noise factor is often described in decibels, in which case it is called the noise figure "NF". The noise figure is difficult to be calculated in mixers, due to the time variance and frequency translation. The noise components of interest lie in the RF range before downconversion and in the IF range after downconversion. The translation of the RF noise components by the switching stage prohibits the direct use of small-signal ac and noise analysis, mandating simulations in the time domain. Another difficult is that the noise contributed by the switching stage exhibits time-varying statistics.

With the help of the single-balanced mixer of figure 1.19 we can make an analysis of the noise in this topology. In the RF part, we have the thermal noise due to the base resistance of Q1 and the emitter resistor, R_E and the collector shot noise of Q1. In the IF path we have thermal noise from resistors R_{C1} and R_{C2}.

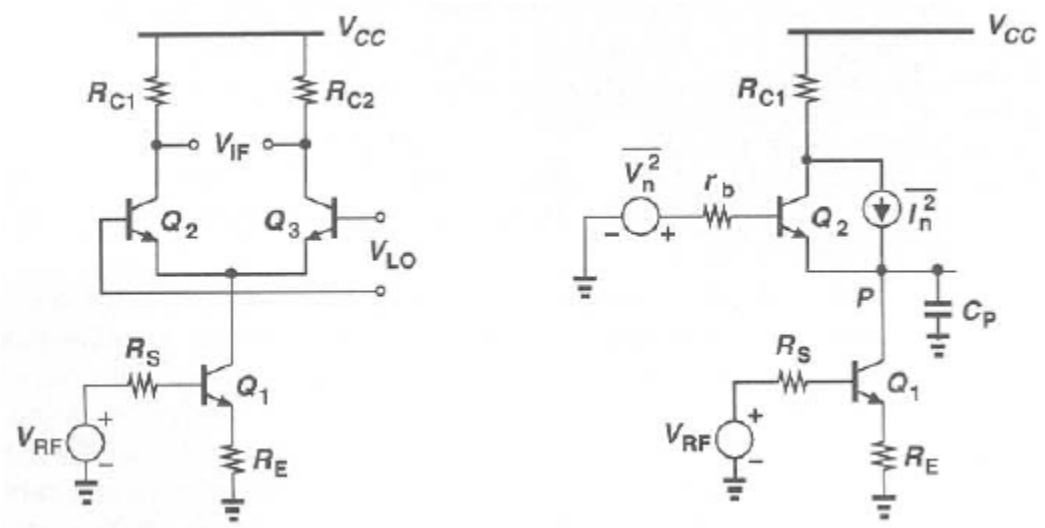


Figure 1.19: Single-balanced mixer and on the right side, is the noise contribution of Q2 when Q3 is off.

The noise contributed by Q2 and Q3 is more important. If we assume that Q2 and Q3 switch instantaneously, then as shown in figure 2. each device is on for approximately half of the LO period, pumping noise to the output because the parasitic capacitance at node P, Cp, provides a finite impedance to ground. Arising from the base-emmiter junction of Q2 and Q3 and collector-base and collector-substrate junctions of Q1, this capacitance can be substantial if the transistors are large structures so as to minimize their

base resistance. Thus, the RF noise due to the base resistance and collector current of Q2 is translated to IF by the switching action of this transistor. In a most realistic case, where the LO signal is not a square wave and Q2 and Q3 are simultaneously on for a part of the period, the two transistors amplify the thermal noise of their base resistance and inject their collector shot noise to the output. As a conclude of the above observations, we can derive that the contribution of thermal and shot noise of Q2 and Q3 can be minimized by:

- Using large LO swings.
- Lowering Cp, which means smaller sizes for Q1-Q3 and therefore higher base resistance noise.
- Reducing the base resistance of Q2 and Q3, which leads to higher Cp.
- Decreasing the collector currents of Q2 and Q3. Since Q2 and Q3 appear in the signal current path, their shot noise current, $\overline{I}_n^2 = 2qI_C$, directly corrupts the signal and is lowering if IC decreases. By contrast, the effect of shot noise of Q1 can be considered as an input referred voltage, $\overline{V}_n^2 = 2kT / g_m = 2kT \frac{V_T}{I_C}$,and is minimized if Ic increases. In other words, as the collector current of Q1 is reduced, this transistor contributes more noise while Q2 and Q3 contributes less.

The conclusion of the above analysis is that we have interesting and important trade-offs to deal with, which require a careful choice of device size and bias currents so as to minimize the overall noise figure.

1.3.4.2 Noise Figures

Noise figure (NF) measures how much the signal to noise ratio (SNR) of a signal degrades because of the added noise as it passes through the mixer. The NF of the mixer is defined as the total SNR at the RF frequency divided by the SNR at the IF frequency.

$$Noise\_Figure = 10\log\left(\frac{SNR_{in}}{SNR_{out}}\right)$$

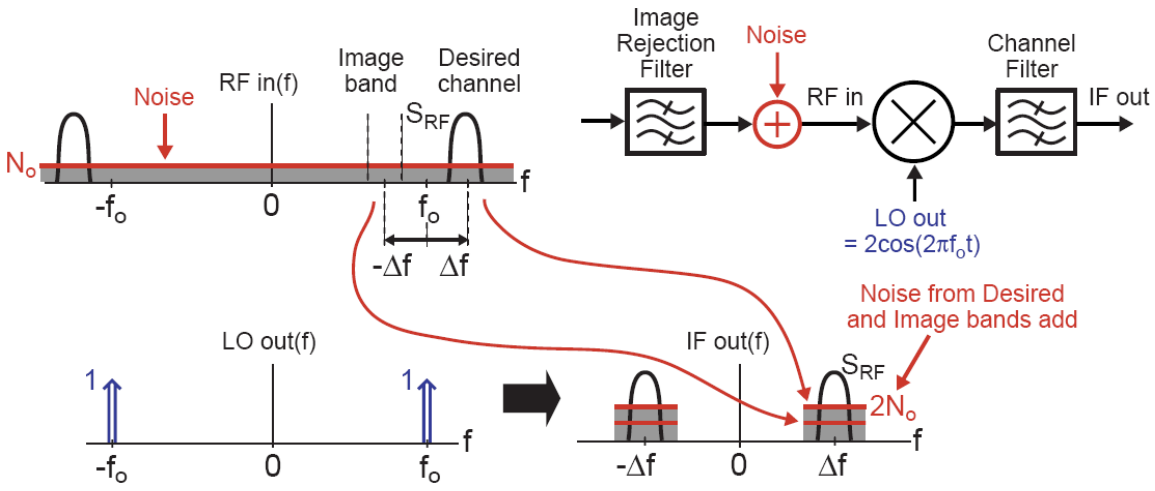
Because many signals have a very wide dynamic range, SNRs are usually expressed in terms of the logarithmic decibel scale. Signal to Noise Ration (SNR) is the ratio, expressed in dB, of the rms value of the input signal at the output to the rms value of the

sum of all other spectral components below one-half the sampling frequency, not including harmonics or dc.

In decibels, the SNR is 20 times the base-10 logarithm of the amplitude ratio, or 10 times the logarithm of the power ratio:

$$SNR(dB) = 10\log_{10}\left(\frac{P_{signal}}{P_{noise}}\right) = 20\log_{10}\left(\frac{A_{signal}}{A_{noise}}\right)$$

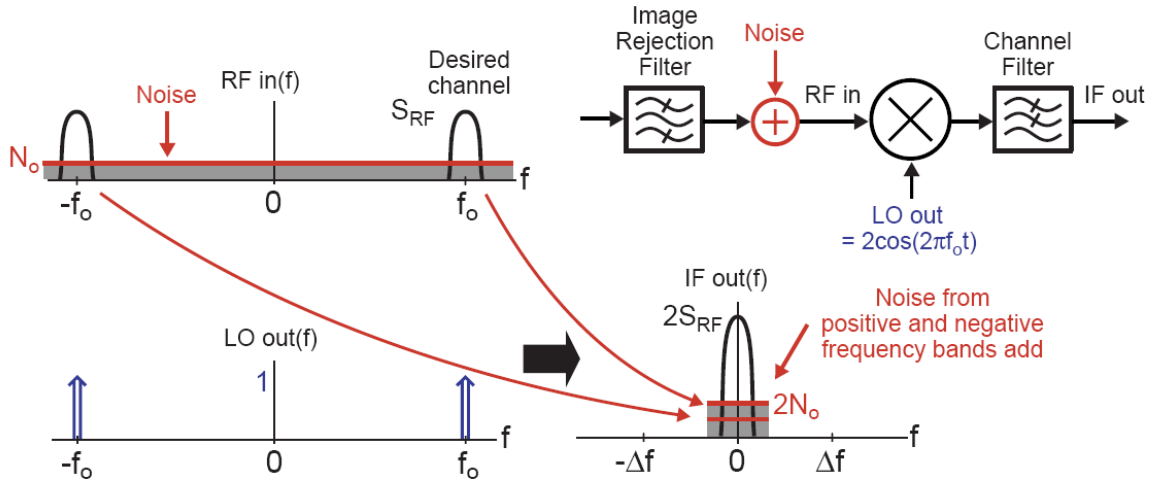
If we consider a noiseless mixer with unity gain as shown in Fig.1.20, the spectrum sensed by the RF port consists of a signal component and the noise component in both the signal band and the image band. Through the downconversion, the signal, the noise in the signal band, and the noise in the image band are translated to ω_{IF}. As a result, the output SNR is half the input SNR if the input frequency response of the mixer is the same for the signal band and the image band. This is the single-sideband noise figure (SSB NF) of the mixer, due to the fact that the desired signal spectrum resides on only one side of the LO frequency, a common case in heterodyne systems. The noise from both image and desired bands will combine in the desired channel at IF output and cannot be removed with a channel filter.



$$SNR_{out} = SNR_{in}/2 \Rightarrow NF = 10\log_{10}(1/2) \Rightarrow NF = 3dB$$

Figure 1.20: RF and image noise into the IF band for a SSB mixer.

Considering the homodyne downconversion that is shown in Fig.1.21 by means of a single noiseless mixer we can derive that the input and output SNR are equal, giving a noise figure of 0 db. This is the "double-sideband" noise figure (DSB NF) because the input signal spectrum resides on both sides of ω_{LO}. Noise from positive and negative frequencies combine, but the signals do as well. If LO has harmonics, more noise is produced at output unless suppressed by mixer gain at higher frequencies. Typical noise figure meters measure the DSB NF and predict the SSB value by simply adding 3 dB.



$$SNR_{out} = SNR_{in} \Rightarrow NF = 10\log_{10}(1) \Rightarrow NF = 0dB$$

Figure 1.21: Double sideband case

Noise Figure of Cascaded Systems

In a cascaded system such as the one shown in figure 1.22, the total noise is the combined noise contribution of each stage divided by the total available gain between the stages. It is dependent on the gain of subsequent stages because noise becomes less important once the signal has been amplified. Therefore, the system noise figure (NF) is dominated by the noise performance of the first couple of stages of the system. The total noise factor is the sum of these individual contributions.

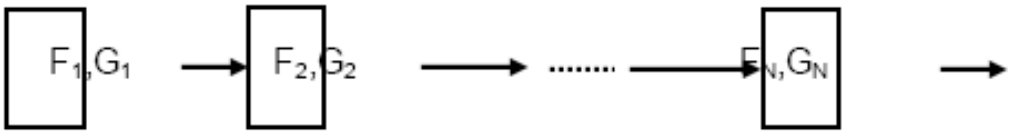


Figure 1.22: Noise figure computation for cascaded systems

Noise factor of the system

$$F = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G1G2} + \ldots + \frac{F_N - 1}{\Pi_{n-1}^{N-1} G_n}$$

Noise Figure (NF) = 10log (F)

Noise figure, for a mixer, considers only the noise associated with the IF ($f_{LO} - f_{RF}$) frequency, according to the IEEE's NF definition for mixers, it assumes that there is no noise contribution at the image frequency (2 $f_{LO} - f_{RF}$) due to mixing.

1.3.5 Port to Port Isolation

The LO-RF feedthrough results in LO leakage to the LNA and eventually the antenna, whereas the RF-LO feedthrough allows strong interferers in the RF path to interact with the local oscillator driving the mixer. The LO-IF feedthrough is important because if substantial LO signal exist at the IF output, then the following stage may be desensitized. Thus, the isolation between each two ports of a mixer is critical. Furthermore, the RF-IF isolation determines what fraction of the signal in the RF path directly appears in the IF, a critical issue with respect to the even-order distortion problem in homodyne receivers.

1.3.6 Single-Balanced and Double-Balanced Mixers

The issue of balance is very important in mixers. A balanced signal is defined to have a zero DC component. Mixers have two signals of concern, LO and RF signals. Thus, unbalanced RF input causes LO feedthrough and unbalanced LO signal causes RF feedthrough.

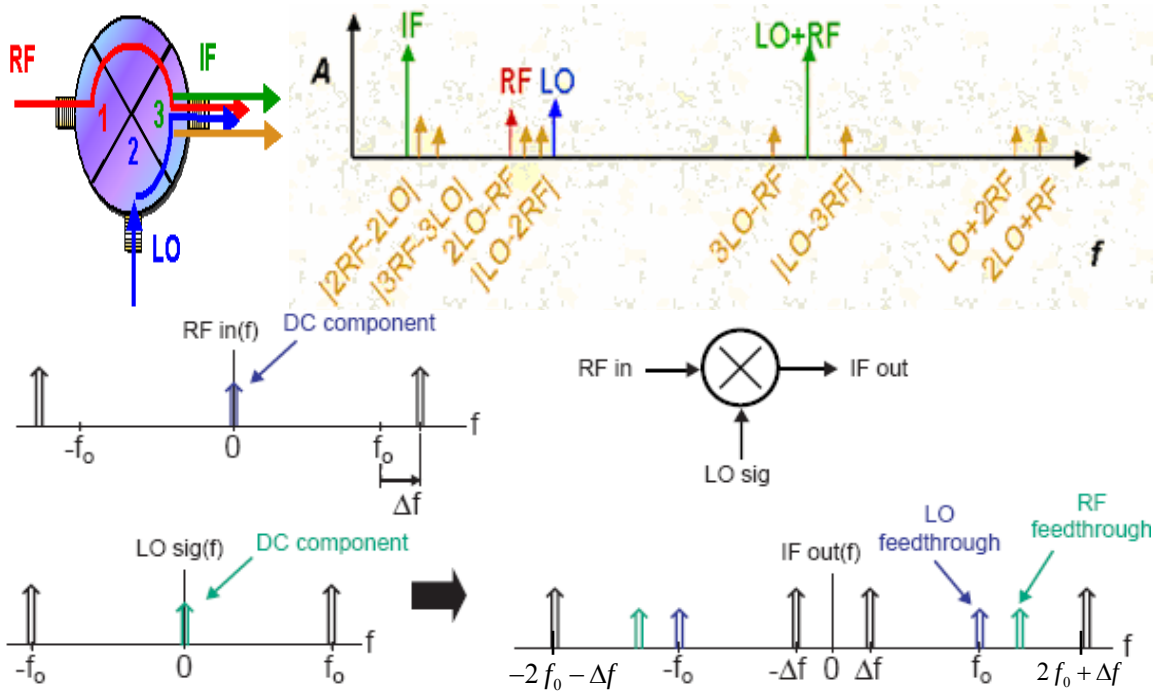


Figure 1.23: Mixing with the DC component.

In order to cancel the DC component of Fig.1.23 we can use the architecture of Fig.1.24 where, we combine two mixer paths with LO signal 180 degrees out of phase between the paths.

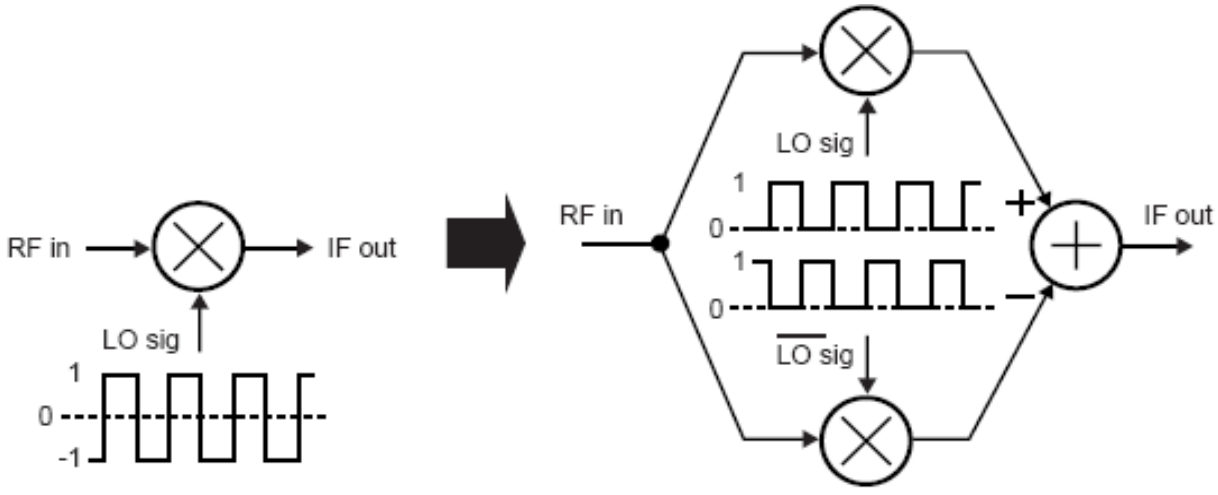


Figure 1.24: Architecture for cancellation of DC component

If a mixer accommodates a differential LO signal but a single-ended RF signal, it is called “singled-balanced”, as shown in Fig.1.25. If a mixer operates with both differentials LO and RF inputs, then it is called “double balanced”, the active version of which assumes the form of a Gilbert cell as shown in Fig.1.27.

In both mixer circuits, single or double balanced, the output can be sensed as either a differential or a single-ended signal. Differential output provides higher conversion gain as well as much more immunity to feedthrough of the RF signal to the IF output.

1.3.6.1 Single-Balanced Mixers

The single-balanced mixer works by converting RF input voltage to a current and then switching current between each side of differential pair. The LO balance is achieved using the technique of Fig.1.24. In order to fully turn on and off the differential pair, LO swing should be no larger than needed. Furthermore, square wave is best to minimize the noise produced by M_1 and M_2. The transconductor is needed to have high linearity.

The single-balanced configuration exhibits less input-referred noise for a given power dissipation than the double-balanced counterpart. However, it is more susceptible to noise in the LO signal. VLO is a large symmetrical wave switching On/Off the coupled M_1 and M_2 (~1Vpp), several harmonics of LO contribute to mixing. Differential output is preferred for higher gain and more immunity to RF→IF feedthrough (strong interferers undergo intermodulation). The main drawback of the single-balanced mixer is the high LO→IF feedthrough as shown in figure 2.. M_1 and M_2 operate as differential pair, thus amplifying the LO signal.

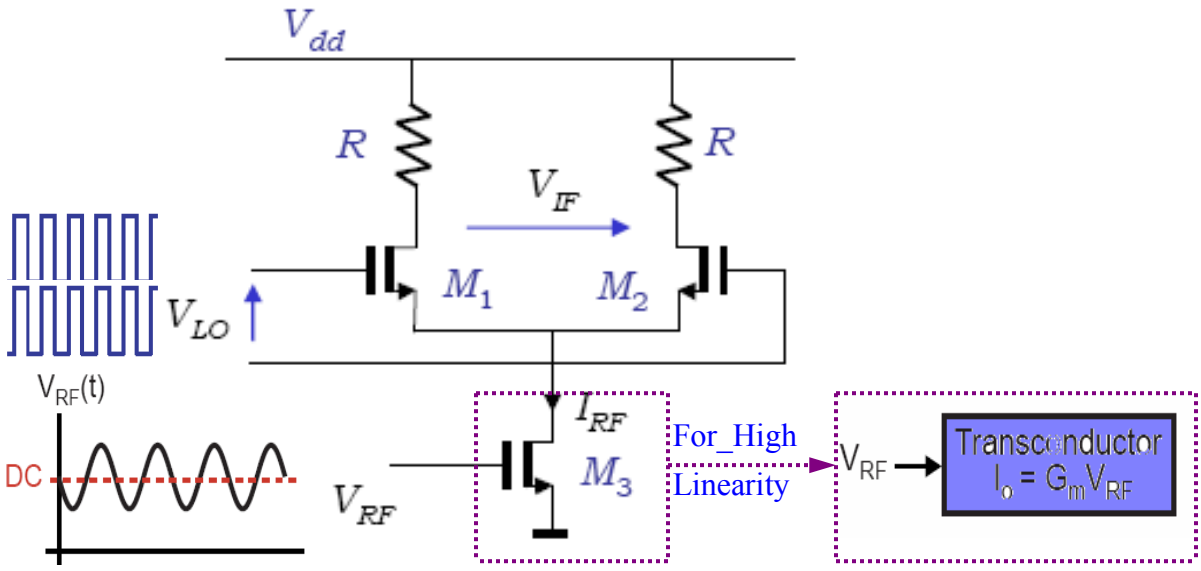


Figure 1.25: Single-balanced mixer. The transconductor converts the RF input voltage to a current, then switching current between each side of the differential pair.

LO Feedthrough in Single Balanced Mixers

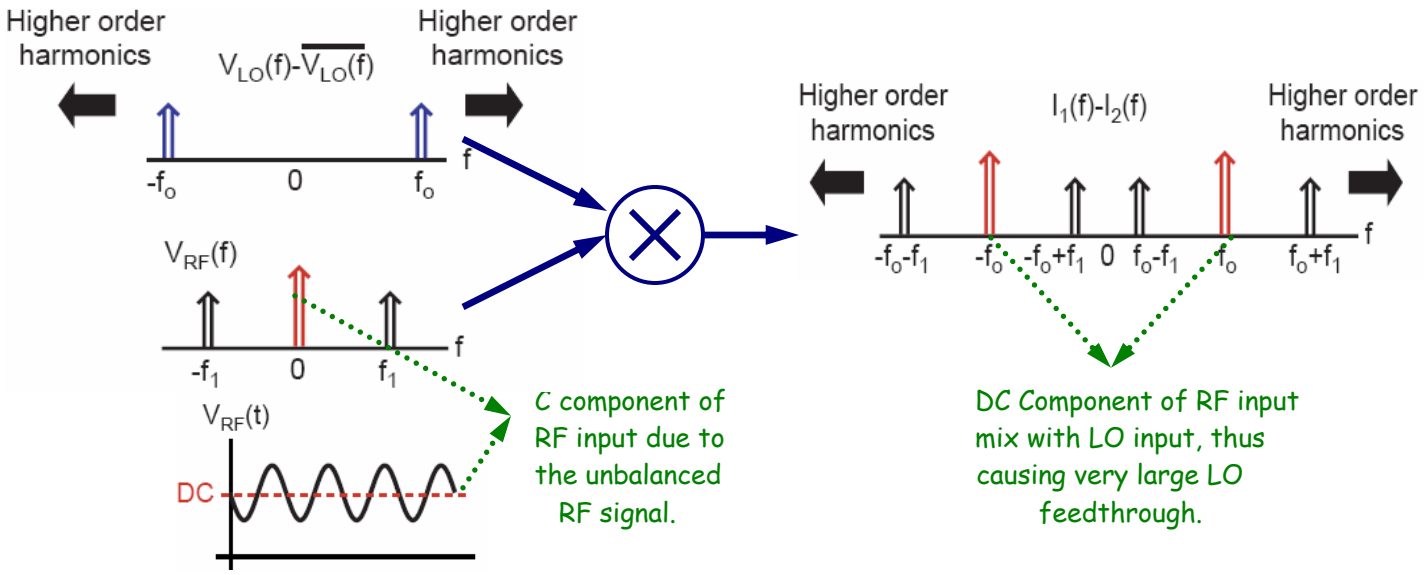


Figure 1.26: LO Feedthrough in Single Balanced Mixers

The DC component of RF input as shown in Figure1.26. causes very large LO feedthrough and can be removed by filtering or by achieving a zero DC value for RF input. But, practical DC component of RF input cannot be zero (balanced RF input)

because the transistors need bias current. The solution to this drawback is the DSB (double side band) mixer.

1.3.6.2 Double-Balanced Mixers

The double-balanced mixer generates less even-order distortion (~fully differential), thus relaxing the half-IF issue in heterodyne receivers and lowering the beat components in homodyne. Nevertheless, since the signal processed by LNA is usually single ended, one of the input terminals of the double-balanced mixer is simply connected to a bias voltage, so it degrades symmetry and device mismatch as well. M_3 and M_4, and, M_5 and M_6 add the amplified LO signal with opposite phases, thereby providing a first-order cancellation of the LO→IF feedthrough. The substraction at the output is achived by cross-coupling the output current of each stage.

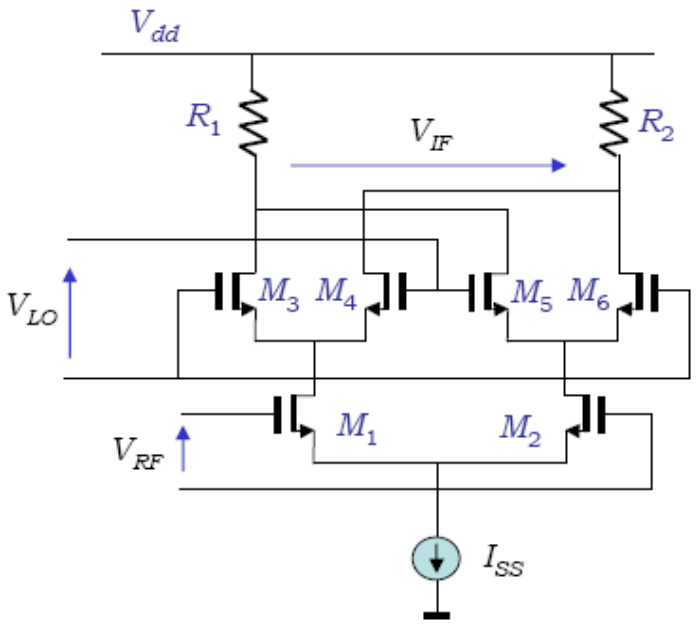


Figure 1.27: Double-balanced mixer implementation.

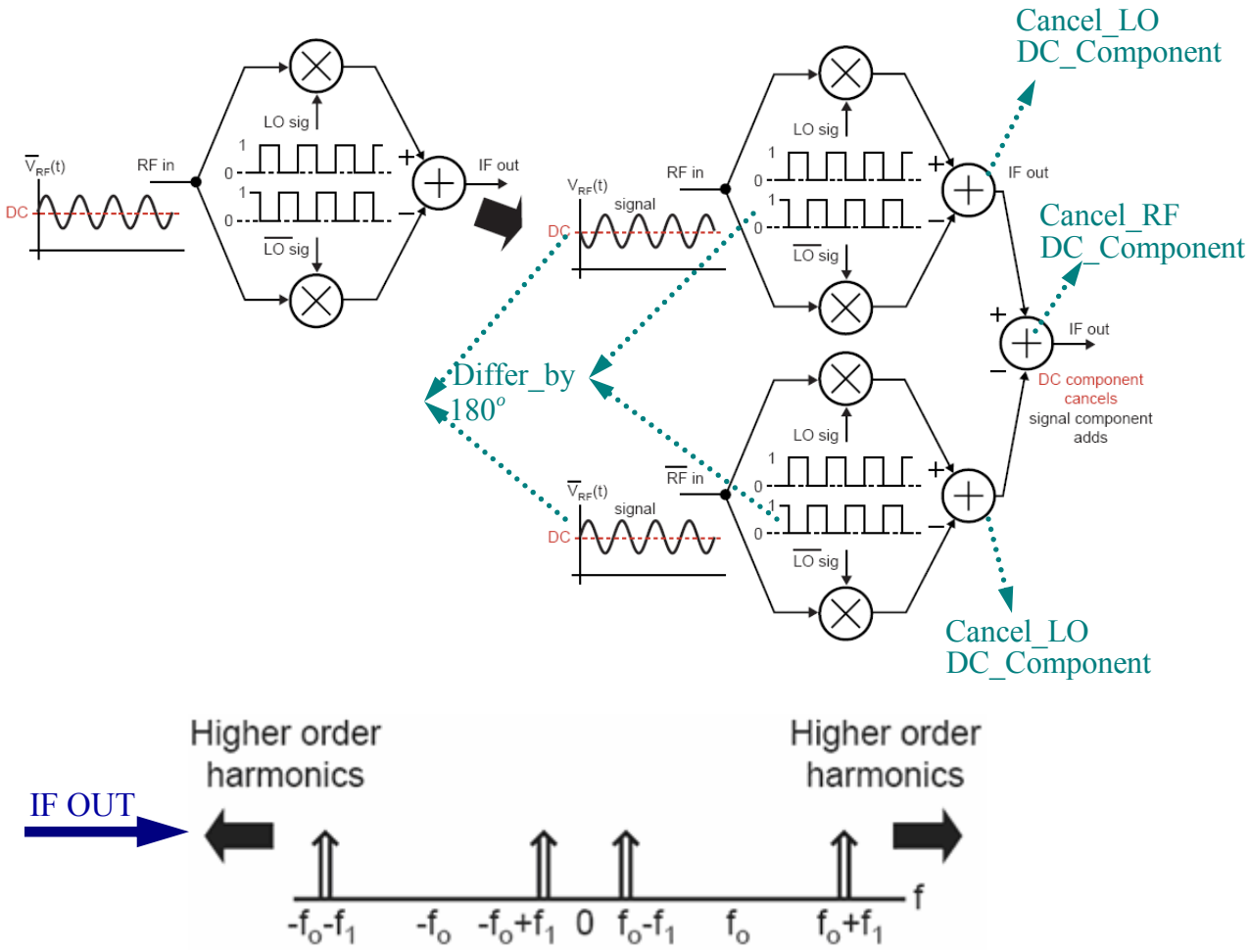


Figure 1.28: Double-balanced mixer implementation. Both LO and RF signals are now balanced (no DC components).

Advantages:

- Both LO and RF are balanced, providing both LO and RF Rejection at the IF output.
- All ports of the mixer are inherently isolated from each other.
- Increased linearity compared to singly balanced.
- Improved suppression of spurious products (all even order products of the LO and/or the RF are suppressed).
- High intercept points.
- Less susceptible to supply voltage noise due to differential topography.

Disadvantages:

- Require a higher LO drive level.

- Require two circuits (although mixer will usually be connected to differential amplifiers).
- Ports highly sensitive to reactive terminations.

1.3.7 Mixer Spurious Response

Generally, a mixer generates cross-products of the RF and LO signals and their harmonics. The frequency of the resulting components can be expressed as $|m\omega_{RF} \pm n\omega_{LO}|$, where m and n are integers. A difficult, but important task in a receiver design is to ensure that, except $|\omega_{LO} - \omega_{RF}|$ (downconversion product), such components do not fall in the IF band. Due to nonlinearities in the RF path, it is possible that harmonics of the interferers beat with harmonics of the LO, corrupting the downconversion signal.

1.3.8 CMOS Mixers

WHY CMOS?

CMOS is nearly ideal for mixed-signal designs:

- Dense digital logic
- High-performance analog

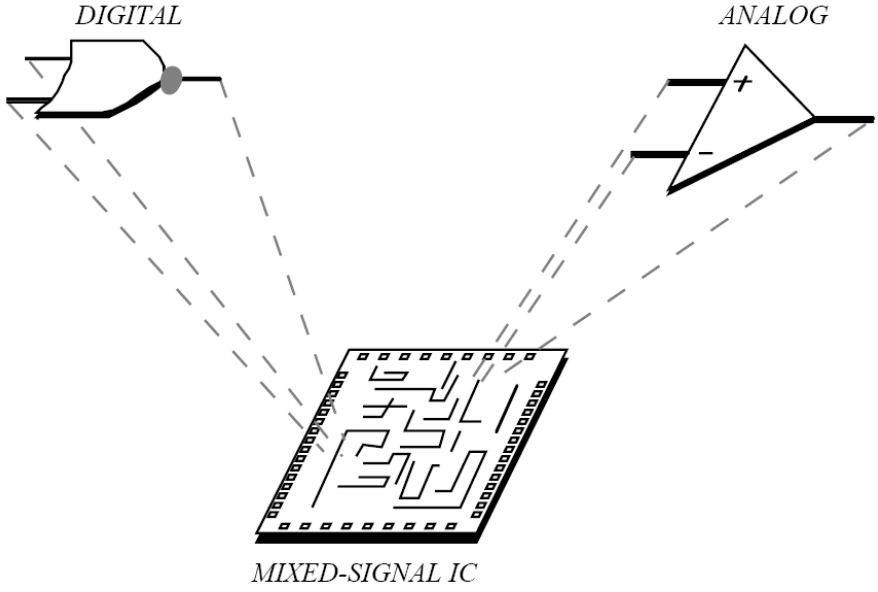


Figure 1.29: CMOS-Mixed Signal.

Active CMOS mixers can be implemented as both, single-balanced and double-balanced. The trade-offs among noise, nonlinearity, and power dissipation, especially in the RF voltage-to-current converter, typically lead to a compromise in the design.

Considering the single-balanced mixer of Fig.1.30, the LO voltage swing in this circuit directly affects the conversion gain and the noise. This happens because the MOS switching pair, $M_1 - M_2$, typically requires great swings to experience complete switching. If M_1 and M_2 are simultaneously on for a significant part of the period, then some of the RF current generated by M_3 is "wasted" as a common-mode signal during this time. This fact lowers the conversion gain and increases the noise contributed by M_1 and M_2.

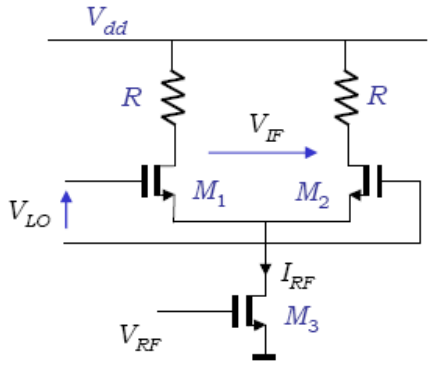


Figure 1.30: CMOS active mixer

For a given LO drive, the switching M_1 and M_2 can be made more abrupt by increasing their width or decreasing their drain current. Increasing their width leads to the increase of the capacitances seen by their common node, thus shunting the RF current. Decreasing their drain current leads to the decrease of the transconductance of M_1 and M_2, thus raising the impedance seen looking into the source terminals of these devices and hence allowing more RF current to flow through the capacitance at the common node of M_1 and M_2. From the above analysis, the consequence is that, the total bias current and the width of M_1 and M_2 must be chosen with the available LO swing in mind.

The RF port of the mixer can be linearized by increasing the gate-source overdrive voltage of the input transistor. However, for a given bias current, a higher overdrive leads to a lower transconductance, thus increasing the noise figure and decreasing the conversion gain. Thus the trade-off between the linearity and the gain of the mixer in low-voltage designs is a difficult problem.

1.3.9 Linearity

Linearity describes the region of operation where the output signal varies proportionally to the input signal. There are several ways to measure the linearity of a circuit like, 1dB Compression Point and Third Order Intercept Point (IP3).

Firstly we must explain the meaning of linearity. A system is linear if its output can be expressed as a linear combination of responses to individual inputs.

For inputs $x_1(t), x_2(t)$

When, $x_1(t) \to y_1(t), x_2(t) \to y_2(t)$,

then, $ax_1(t) + bx_2(t) \to ay_1(t) + yb_2(t)$

Any system that does not satisfy this condition is nonlinear. According to this definition, we consider a system nonlinear if it has nonzero initial conditions or finite "offsets".

An effect of nonlinearity are harmonics. If a sinusoid is applied to a nonlinear system, the output generally exhibits frequency components that are integer multiples of the input frequency. If we have an input x(t)=Acosωt, to the system $y(t) = a_1 x(t) + a_2 x^2(t)t + a_3 x^3(t)$ then:

$$y(t) = a_1 A\cos\omega t + a_2 A^2 \cos^2 \omega t + a_3 A^3 \cos^3 \omega t$$

$$= a_1 A\cos\omega t + \frac{a_2 A^2}{2}(1+\cos 2\omega t) + \frac{a_3 A^3}{4}(3\cos\omega t + \cos 3\omega t)$$

$$= \frac{a_2 A^2}{2} + \left(a_1 A + \frac{3a_3 A^3}{4}\right)\cos\omega t + \frac{a_2 A^2}{2}\cos 2\omega t + \frac{a_3 A^3}{4}\cos 3\omega t$$

The term with the input frequency is called the "fundamental" and the higher-order terms the "harmonics".

1.3.9.1 1-dB Compression Point

In a real device, the input and the output power is linear only to a certain point. The 1dB Compression Point measures this point where the output power level is 1dB less than it would have been if it were an ideally linear device (is the input signal level that causes the small-signal gain to drop by 1-dB).

Like other non-resistive networks, a mixer is amplitude-nonlinear above a certain input level resulting in a gain compression characteristic as shown in Figure 1.31.

Above this point the If fails to track the RF input power level – normally a 1dB rise in RF power will result in a 1dB rise in the IF power level. The 1dB compression point is measured by plotting incident RF power against IF power as shown in the figure below.

The 1dB compression point gives rise to the dynamic range of the mixer, which is the difference between the 1dB compression point and the minimum discernable signal MDS (dependant on the noise floor of the device).

Most mixers have the 1dB compression point specified at the input i.e. the single-tone input signal level at which the output of the mixer has fallen 1dB below the expected output level. For typical double balanced mixers this figure is ~ 6dB below the LO power level. (So performance can be improved by overdriving the LO port).

To calculate the 1-dB compression point we use the formula from the expression of harmonics:

$$20\log\left|a_1+\frac{3}{4}a_3A_{1-dB}^2\right|=20\log|a_1|-1dB$$

$$\Rightarrow A_{1-dB}=\sqrt{0.145\left|\frac{a_1}{a_3}\right|}$$

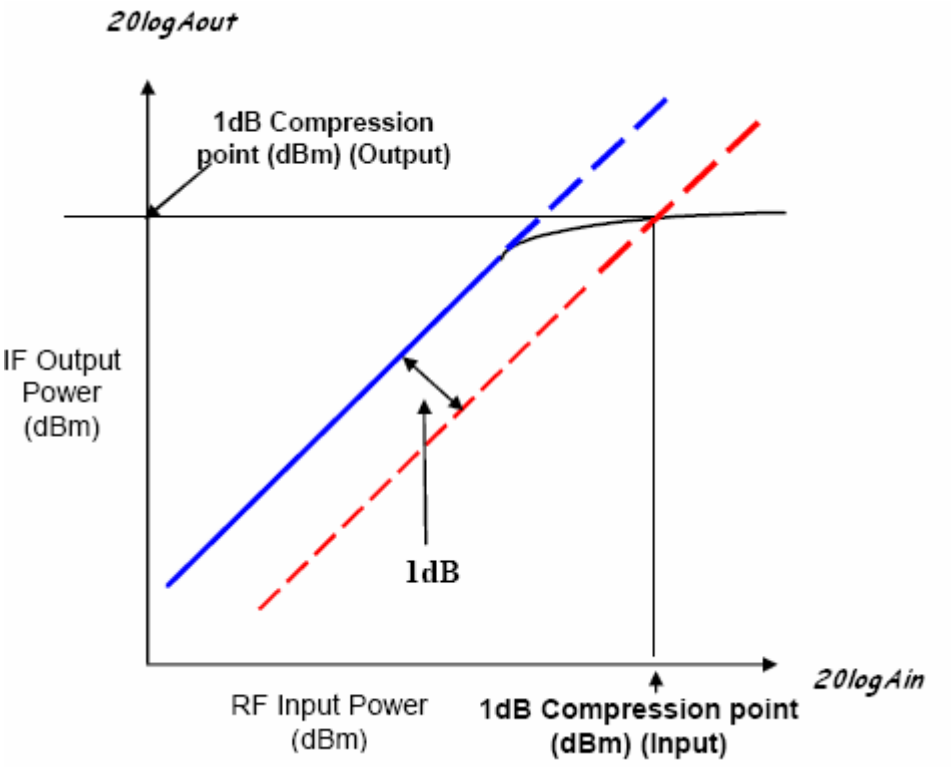


Figure1.31: Typical gain compression characteristic for a non-linear Mixer, showing the measurement of the 1dB compression point. If plotted on log-log scale as a function of input level, the output level falls below its ideal value by 1-dB at the 1-dB Compression Point.

1.3.9.2 Intermodulation

While harmonic distortion is often used to describe nonlinearities of analog circuits, certain cases require certain measures of non-linear behavior. When two signals with different frequencies are applied to a non-linear system, the output in general exhibits some components that are not harmonics of the input frequencies. This phenomenon is called Intermodulation (IM) and arises from "mixing" (multiplication) of two signals when their sum is raised to a power greater than unity.

For a better understand of the phenomenon, we are going to apply the same method as the one we did for the high order harmonics in linearity. The only different now, is that the input is: $x(t) = A_1 \cos \omega_1 t + A_2 \cos \omega_2 t$

Thus:

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

Expanding the left side of the above equation and discarding dc terms and harmonics, we can derive the following Intermodulation products:

$$\omega = \omega_1 \pm \omega_2 \rightarrow \alpha_2 A_1 A_2\cos(\omega_1+\omega_2)t + \alpha_2 A_1 A_2\cos(\omega_1-\omega_2)t$$

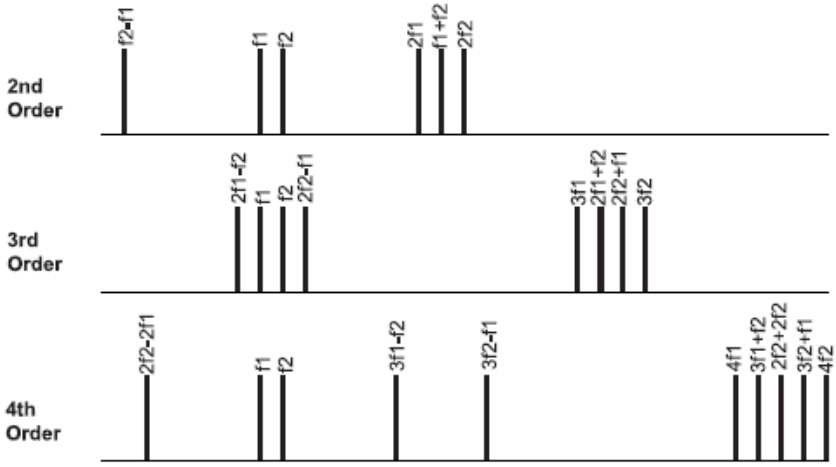
$$\omega = 2\omega_1 \pm \omega_2 \rightarrow = \frac{3a_3A_1^2A_2}{4}\cos(2\omega_1+\omega_2)t + \frac{3a_3A_1^2A_2}{4}\cos(2\omega_1-\omega_2)t$$

$$\omega = 2\omega_2 \pm \omega_1 \rightarrow = \frac{3a_3A_2^2A_1}{4}\cos(2\omega_2+\omega_1)t + \frac{3a_3A_2^2A_1}{4}\cos(2\omega_2-\omega_1)t$$

αnd these fundamental components:

$$\omega = \omega_1,\omega_2 \rightarrow \left(\alpha_1 A_1 + \frac{3}{4}\alpha_3 A_1^3 + \frac{3}{2}\alpha_3 A_1 A_2^2\right)\cos\omega_1 t + \left(\alpha_1 A_2 + \frac{3}{4}\alpha_3 A_2^3 + \frac{3}{2}\alpha_3 A_2 A_1^2\right)\cos\omega_2 t$$

The main interest is focused on the third-order IM products at $2\omega_1 \pm \omega_2$ and $2\omega_2 \pm \omega_1$ as shown in Figure2..



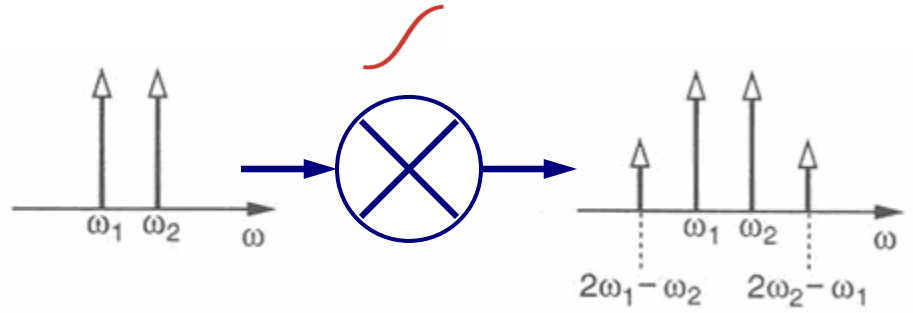


Figure1.32: (a) Distribution of harmonics and Intermodulation products
(b)Non-linear system focusing on the third-order IM products due to nonlinearity.

The key point here is that if the difference between ω_1 and ω_2 is small, then the components at $2\omega_1 - \omega_2$ and $2\omega_2 - \omega_1$ appear in the vicinity of ω_1 and ω_2, thus revealing nonlinearities.

As shown in figure 1.33 , if a weak signal is accompanied by two strong interferers experiences third-order nonlinearity, then one of the IM products falls in the band of interest, corrupting the desired component of the signal.

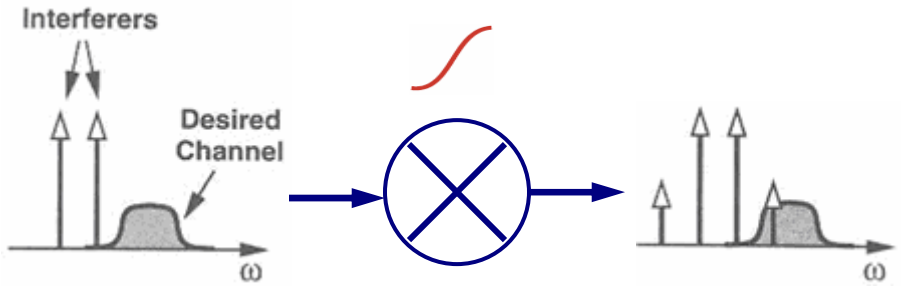


Figure1.33: Corruption of a signal due to Intermodulation between two intererers.

The corruption of signals due to third-order Intermodulation of two nearby interferers is so common and critical that a performance metric has been defined to characterize this behavior. This metric is called “third intercept point” (IP3), and is measured by a two tone test in which A is choosen to be sufficient small so that the higher order nonlinear terms are negligible and the gain is relatively constant and equal to a1.

Third Order Intercept Point (IP3)-IM3

This is a theoretical point at which the fundamental and third order response intercepts. This point is found when two signals that are very close in frequency are applied to the mixer. Third-order Intermodulation (IM3) appears at the output.

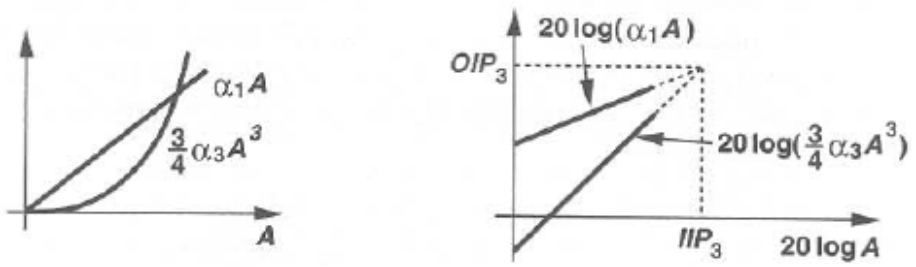
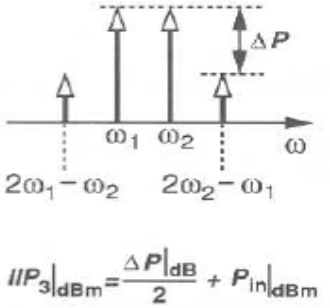


Figure1.34: Growth of the output components in an Intermodulation two tone test.

From figure 1.34 we can derive that as A increases, the fundamentals increase in proportion to A, whereas the third-order IM products increase in proportion to A^3. Plotted on a logarithmic scale, we can derive that the magnitude of the IM products grows at three times the rate at which the main components increase. The third-order intercept point is defined to be at the intersection of the two lines. The horizontal coordinate of this point is called the input IP3 (IIP3), and the vertical coordinate is called the output IP3 (OIP3) as shown in figure 1.35. If the magnitude of IM products is used as a measure of linearity, then the input amplitude with which the test is performed must be specified. The third intercept point on the other hand, is a unique quantity that by itself can serve as a means of comparing the linearity of different circuits. Thus IP3 is better than a simple IM measurement.



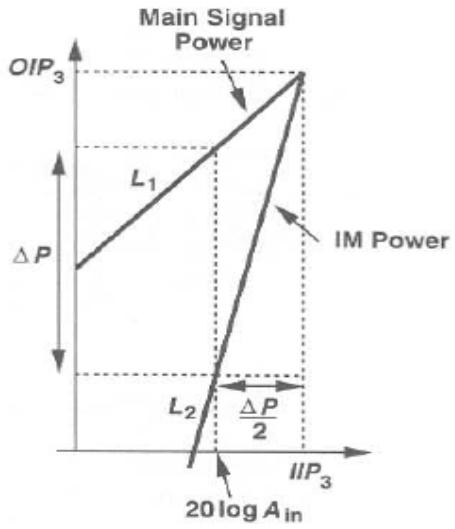


Figure1.35: Calculation of IP3 without extrapolation at the left side and graphical implementation of the calculation at the right side.

From figure 1.36 we can see that, if all signals are expressed in dBm, the input third intercept point is equal to half the difference between the magnitudes of the fundamentals and the IM3 products at the output plus the corresponding input level. The main point here is that IP3 can be measured with only one input level, obviating the need for extrapolation. The actual value of IP3, however, must still be obtained through accurate extrapolation to ensure that all nonlinearities and frequency-depended effects are taken into account.

Practically IP3 or IM3 is measured by applying two closely spaced input tones at frequencies F1 and F2. Third order products from the mixing of these tones with the LO (at frequency FLO) occur at frequencies given by: (2F1±F2) ±FLO and (2F2±F1) ±FLO. In the case of the mixer, the third order products of most interest are (2F1-F2)- FLO and (2F2-F1)-FLO as they fall in, or close to the IF band.

The IM3 performance is often summarized by giving the 3rd Order Intercept point (IP3 or IM3 Intercept) as shown in the compression characteristic of Figure 1.36, where the IM3, IM5 plots intersect with the extrapolated gain plot (blue dotted line). As a rule of thumb the IM3 intercept point is approximately 10dB above the 1dB compression point.

This figure of merit gives an indication of the mixer's signal handling capability. In particular it provides an indication of the levels of third order products a mixer is likely to produce under multi-tone excitation.

The IM3 and IM5 graphs will intercept the fundamental graph at the intercept point. (IM2 intercept point will be different and usually a lot higher).

For mixers the measurement is referred to the input ($IP_{3,in}$) and is given by:

$$IP_{n,in} = \frac{IMR}{(n-1)} + Input\_power(dBm),$$

Where IMR = Intermodulation ratio (The difference in dB between the desired output and spurious signal) and n = the IM order.

Typically, for double balanced mixers $IM_{3,\ in}$ is ~ 10-14dB greater than the single tone 1dB compression point and ~ 8dB greater than the LO power.

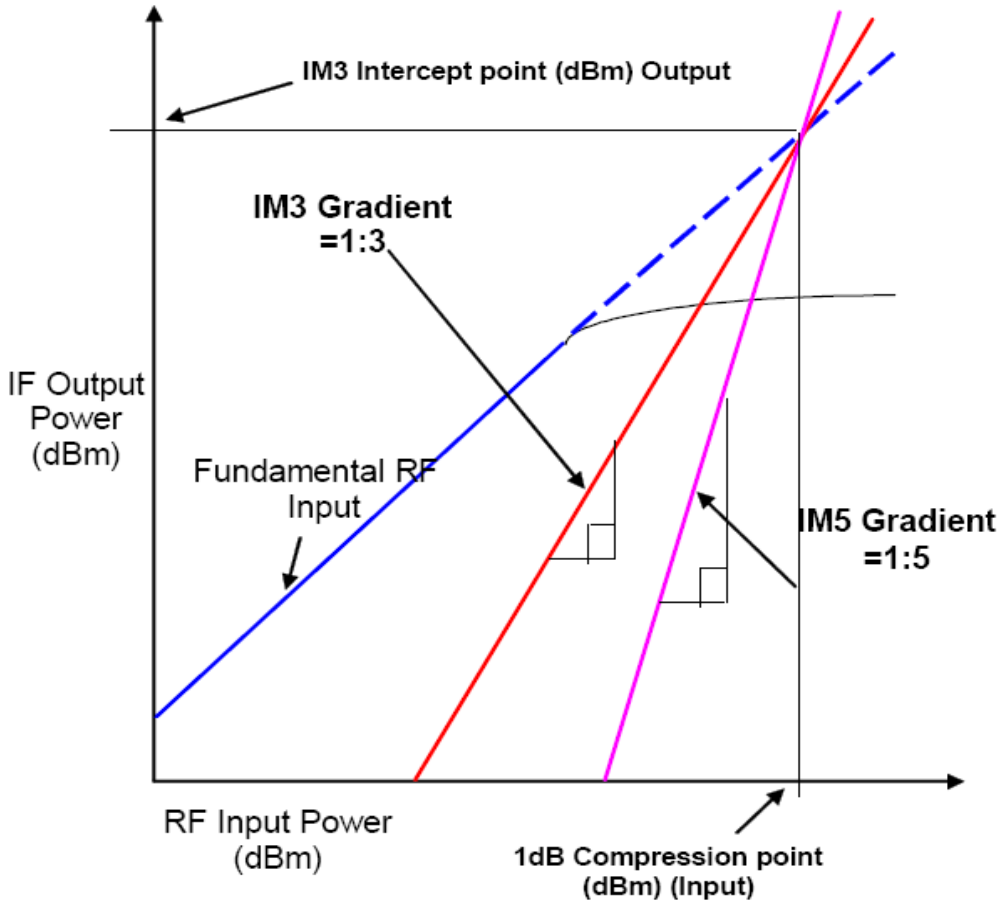


Figure 1.36 : IM3 gain compression characteristic

1.3.10 Source Degeneration – Transconductor Implementation

An important mixer requirement is linearity. There are several ways to increase linearity such as increasing the voltage supply or increasing the current. However, the most common and effective method to improve linearity is to use some type of source degeneration. Figure 1.37 shows the mixer with source degeneration resistors and with source degeneration inductors. Resistors are used when the size of the circuit needs to be minimized. Inductor degeneration is usually preferred because it has no thermal noise to degrade the noise figure, and it saves headroom because there is no voltage drop across it.

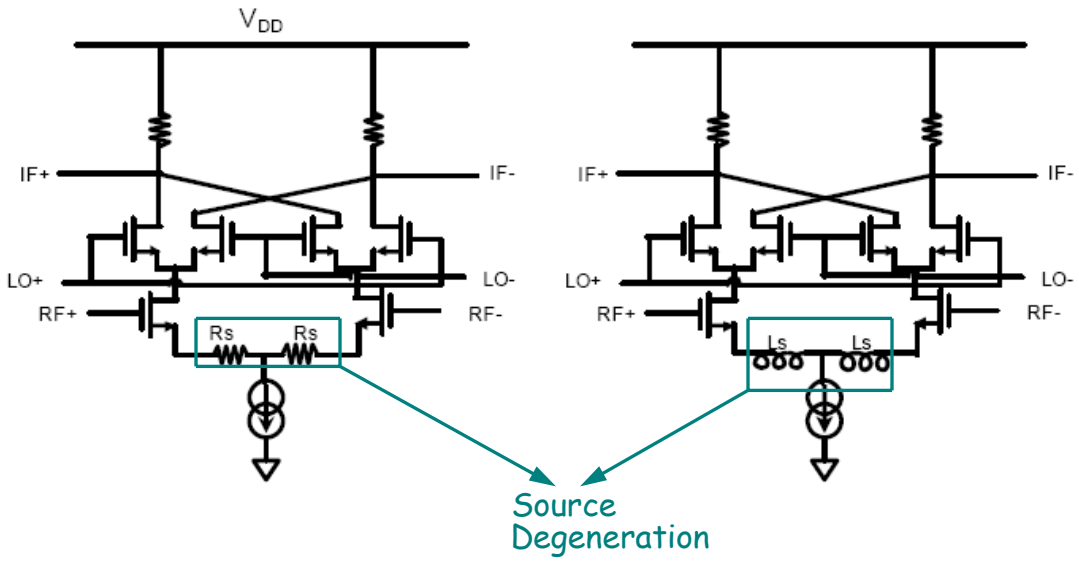


Figure 1.37: Implementation of a Gilbert cell mixer with source degeneration resistors (left) and inductors (right).

The implementation that is shown in figure 1.38 shows a transconductor with degeneration. Thus, we don't have DC voltage drop. Also we have increased impedance at high frequencies which helps to filter out the undesired high frequency components. Power match is usually not required for IC implementation due to proximity of LNA and Mixer.

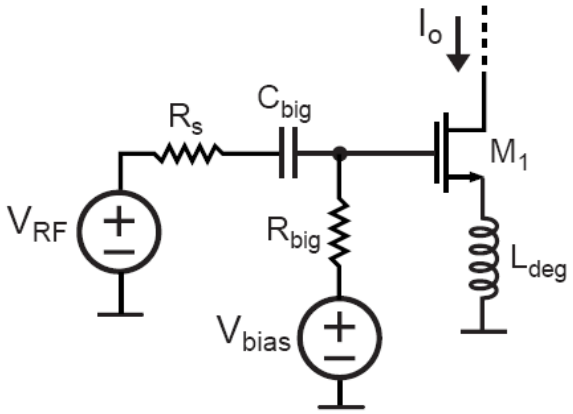


Figure 1.38: Transconductor Implementation.

1.4 Current Mirror

Current mirrors are fundamental building blocks in analog circuits. They can be used to provide a current source to the circuit and also act as an active load at the output.

We also know that the output resistance and capacitance and the voltage headroom of a current source trade with the magnitude of the output current. The main goal is the way that a MOSFET should be biased so as to operate as a stable current source.

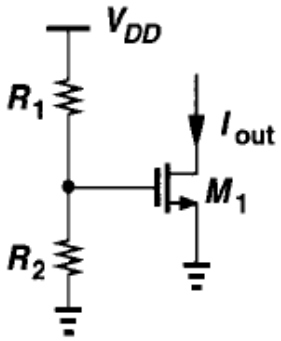


Figure 1.39: Definition of current by resistive divider.

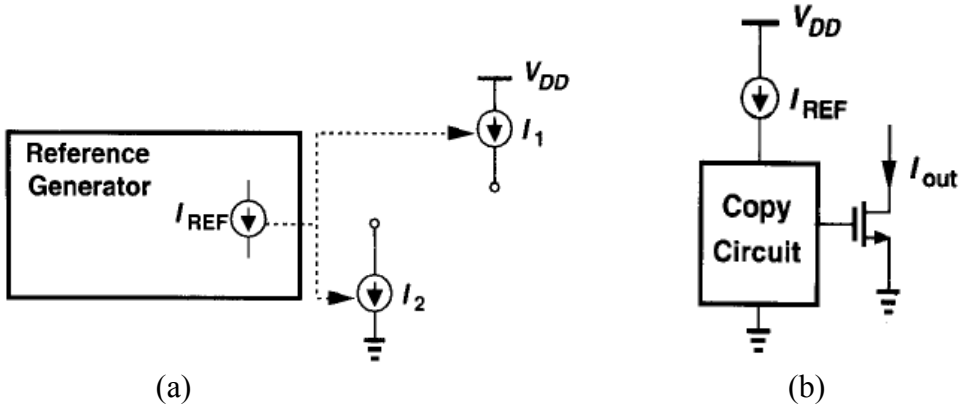
If we consider the simple resistive biasing shown in figure 1.39 and by assuming that M1 is in saturation, we can write:

$$I_{out} \approx \frac{1}{2}\mu_n C_{ox}\frac{W}{L}\left(\frac{R_2}{R_1+R_2}V_{DD}-V_{TH}\right)^2$$

As it can be revealed from the above formula, I_{out} is depended from the supply, process, and temperature. The overdrive voltage is a function of V_{DD} and V_{TH}, and both μ_n and V_{TH} exhibit temperature dependence. Therefore, I_{out} is poorly defined and the issue is even more severe as the device is biased with a smaller overdrive voltage (to consume less headroom). For those reasons we must find other methods of biasing analog integrated circuits.

The design of current sources in analog circuits is based on "copying" currents from a reference, with the assumption that one precisely-defined current source is already available, as it's shown in figure 1.40. It is used to generate a stable reference current,

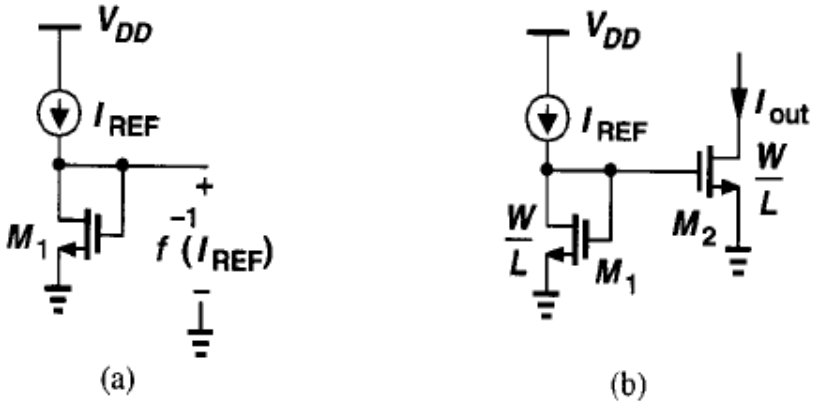
I_{REF} , which is then copied to many current sources in the system. For a Mosfet, if $I_D = f(V_{GS})$, where f(.) denotes the functionality of I_D versus V_{GS}, then $V_{GS} = f^{-1}(I_D)$.



(a) (b)

Figure 1.40: (a)Use of a reference to generate various currents. (b)Conceptual means of copying currents.

That is, if a transistor is biased at I_{REF}, then it produces $V_{GS} = f^{-1}(I_{REF})$ as shown in Figure 1.41(a). Thus if this voltage is applied to the gate and source terminals of a second Mosfet, the resulting current is $I_{out} = ff^{-1}(I_{REF}) = I_{REF}$ as shown in figure 1.41(b). From another point of view, if two MOS transistors have equal gate-source voltages and they operate in saturation, they carry equal currents (if λ=0).



(a) (b)

Figure 1.41: (a) Diode-connected device providing inverse function, (b) Basic current mirror.

The structure consisting of M1 and M2 in figure 1.41(b) is called "current mirror". In the general case, the devices need not be identical. Neglecting channel-length modulation, we can write

$$I_{REF} = \frac{1}{2}\mu_n C_{ox}\left(\frac{W}{L}\right)_1 (V_{GS} - V_{TH})^2$$

$$I_{out} = \frac{1}{2}\mu_n C_{ox}\left(\frac{W}{L}\right)_2 (V_{GS} - V_{TH})^2,$$

Obtaining

$$I_{out} = \frac{(W/L)_2}{(W/L)_1} I_{REF}$$

The key property of this topology is that it allows precise copying of the currents with no dependence on process and temperature. The ratio of I_{out} and I_{REF} is given by the ratio of device dimensions, a quantity that can be controlled with reasonable accuracy.

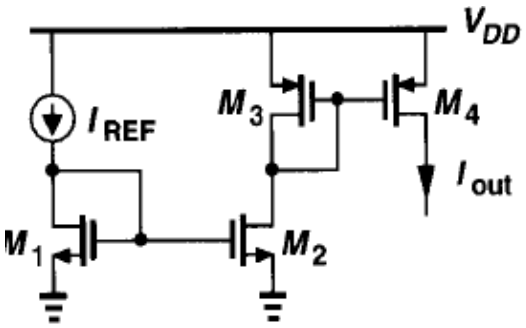


Figure 1.42: Biasing a differential pair using a current mirror.

From figure 1.42 and with the help of the above formulas we have

$$I_{D2} = I_{REF}\left[(W/L)_2/(W/L)_1\right]$$

$$I_{D3} = I_{D2}$$

$$I_{D4} = I_{D3}\left[(W/L)_4/(W/L)_3\right]$$

Thus, $I_{D4} = \alpha\beta I_{REF}$, where $a = (W/L)_2/(W/L)_1$ and $\beta = (W/L)_4/(W/L)_3$. Proper choice of α and β can establish large or small ratios between I_{D4} and I_{REF}. For

example if α=β=3, then we will have a magnification of 9. By the same way if α=β=0.3, then we will have generation of a small, well defined current.

1.5 Gilbert Mixer

Gilbert mixer is the preferred mixer implementation for most radio systems. The Gilbert cell topology is commonly used as a multiplier, where both inputs must remain linear with respect to the output. In implementing a Gilbert cell multiplier it is common to use pre-distortion to increase the linearity of the switching core (the LO input). A Gilbert cell can also realize a mixer, where only one input (the RF input) must remain linear with respect to the output (the IF output). When operating as a mixer pre-distortion is never used, in fact a very nonlinear stage is desired for the switching core (the LO input). The doubly balanced Gilbert cell mixer converts a differential input voltage to a differential current by an emitter coupled pair.

The output current of the long tailed pair feeds into the switching mixing core which commutates the signal current. This results in a conversion gain penalty of -3.9 dB. The Gilbert cell mixer typically compresses in the input transconductance stage due to the limited large signal handling capabilities of the differential pair. Emitter degeneration is often used to reduce the gain of the differential pair; this adversely effects the noise figure, resulting in only a modest increase in dynamic range. The noise figure is usually reduced by the front-end gain provided by the input emitter coupled pair. So there is a direct trade off between noise figure and compression point. The switching core can be viewed as a doubly balanced common gate amplifier with respect to the RF signal. If the applications desire minimum LO power, it is best to bias the LO core at the FET threshold. This is where the transconductance can be modulated over the widest range with the smallest amount of LO power.

CMOS Gilbert cell mixers can be implemented with either P-channel current source loads or with polysilicon resistor loads. The poor transconductance of P-channel devices results in physically large P-channel transistors. These devices have large shunt capacitances that attenuate high intermediate frequency (IF) signals resulting in reduced conversion gain. To alleviate this problem Gilbert up-converters typically use polysilicon load resistors. Resistive loads limit the current available to the mixer due to the voltage drop across the

resistors. This in turn limits the increase in IP3 possible from increased current consumption with a fixed supply voltage.

1.5.1 Gilbert Mixer Operation

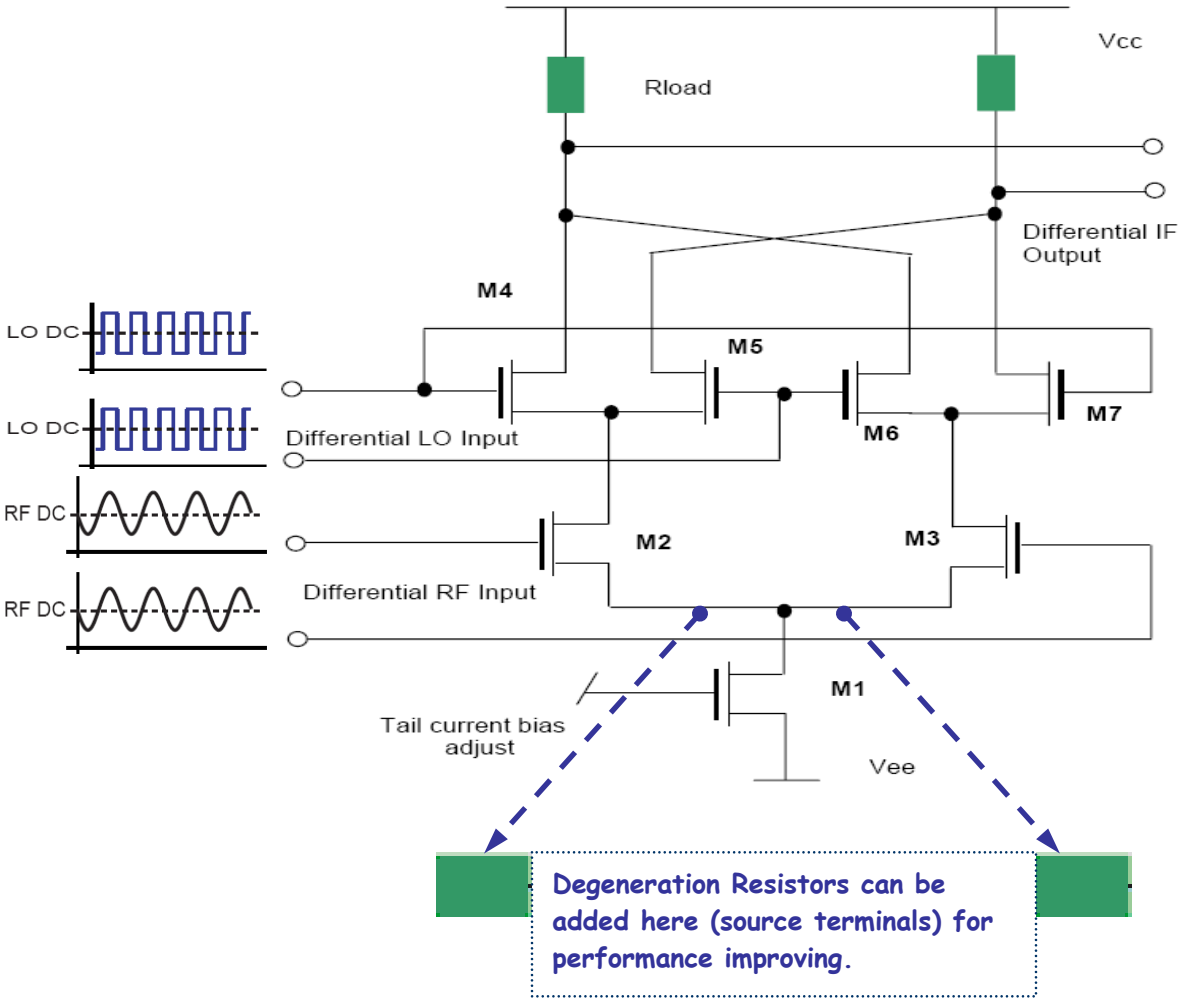


Figure 1.43: Basic circuit of the Gilbert Cell Double balanced mixer (DBM) with optional adding of source degeneration resistors.

With the help of Figure1.43 we will explain the basic operation of a simple Gilbert cell double balanced mixer. The RF signal is applied to the transistors M2 & M3 which perform a voltage to current conversion. For "correct" operation these devices should not be driven into saturation and therefore, signals considerably less than the 1dB

compression point should be used. Performance can be improved by adding degeneration resistors, on the source terminals of M2 & M3.

MOSFets M4 to M7 form a multiplication function, multiplying the linear RF signal current from M2 and M3 with the LO signal applied across M4 to M7 which provide the switching function.

M2 and M3 provide +/- RF current and M4 & M7 switch between them to provide the RF signal or the inverted RF signal to the left hand load. M5 & M7 switch between them for the right hand load.

The two load resistors form a current to voltage trans-formation giving differential output IF signals.

1.5.1.1 Gilbert Cell Design Guideline

All the transistors are operated in the saturation region. This region offers the largest gain and also makes the current less susceptible to the changing voltage across the transistors. The large signal current equation for a MOSFET in saturation is shown in equation the next equation.

$$I_{DS} = \frac{1}{2} U_n C_{ox} \frac{W}{L} (V_{GS} - V_T)^2 (1 + \lambda (V_{DS} - V_{DS,SAT}))$$

The next equation is a simplified version used to simplify calculations by neglecting the channel modulation.

$$I_{DS} = \frac{1}{2} U_n C_{ox} \frac{W}{L} (V_{GS} - V_T)^2$$

The transconductance of a device is depended on W/L and VGS as shown in the equation that follows.

$$g_m = u_n C_{ox} \frac{W}{L} (V_{GS} - V_T)$$

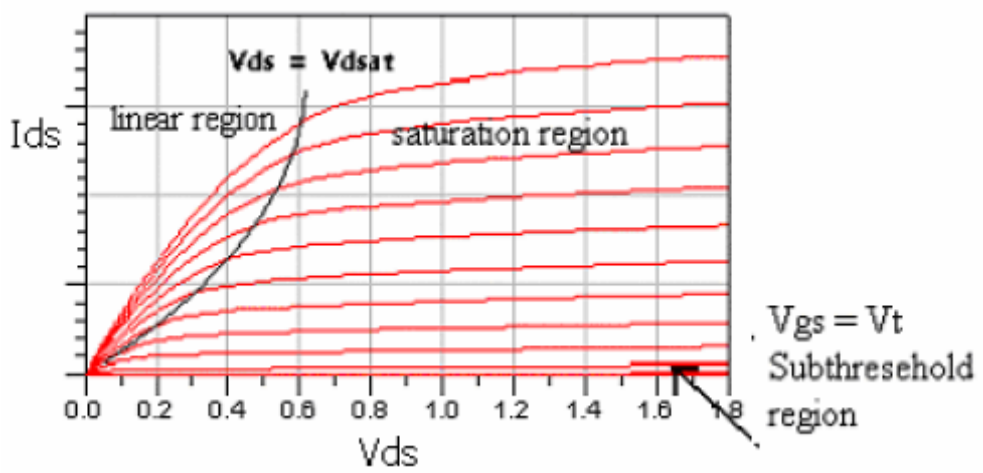


Figure 1.44: Ids Vs. Vds plot showing the different regions of operation.

The condition for saturation is $V_{DS} \geq V_{GS} - V_T$ and $V_{GS} > V_T$ as shown in Figure 1.44.

1.5.1.2 Gain Stage

The mixer must have high linearity to handle the power from the LNA. To help improve linearity, source degeneration resistors can be added just below the gain stage of the mixer. The transistors should be biased such that they have enough head room to swing without leaving the saturation region. The overdrive voltage (Vgs-Vt) should be around 200mV to 400mV. The gain of the mixer is proportional to *gm*. Therefore, higher overdrive voltage means higher gain. Increasing W while keeping L at minimum also increases the gain. Increasing current also increases the gain as shown by the equation that follows. Decreasing the source degeneration resistor will increase the gain but decreases linearity.

$gm = 2ID / (VGS - Vt)$

1.5.1.3 Switching Stage

When LO voltage level is too small the output voltage is dependent on the LO level, which means gain will be larger for larger LO because output voltage will become insensitive to the LO amplitude. Noise is also minimized for large LO. However, when the LO becomes too large, this leads to spikes in the signals, reducing switching speed and increase LO feedthrough. The result of the spikes can also cause transistors to leave

the saturation region. For complete switching, the LO level should be at least 100mV peak and at most 400mV peak.

When one transistor pair is conducting, we want the other pair to be completely off. If two pairs are conducting current at the same time, it will generate noise. Therefore, the overdrive voltage (Vgs – Vt) should be as close to zero as possible. This is the midpoint between turning the transistor on and off. In Figure 1.45, we demonstrate the proper LO switching.

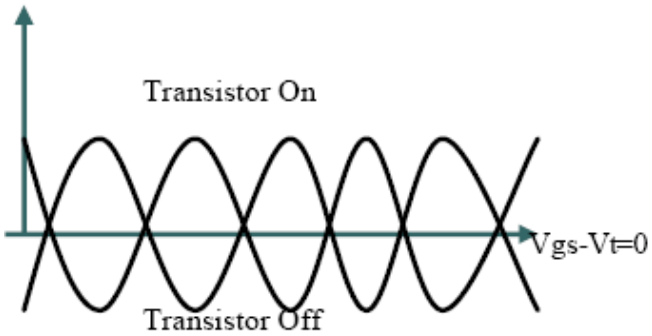


Figure 1.45: Illustration of proper LO switching

CHAPTER 2

Cadence Framework II Installation Procedure

2.1 Installation Pack

Firstly, the user must install the tool ¨Cadence Framework II¨.

The full pack of the tool consists of:

1st packet: 5CD's (software kit for IC5.1.41 USR2 Solaris)

2nd packet: 4CD's (software kit for IC5.1.41 Solaris)

3rd packet: ASSURA

4th packet: MMSIM60

2.2 Installation Procedure

1.2.1 Creating Directories

In a terminal write the following commands:

mkdir cad	→ create the directory CAD
cd CAD	→ access the directory CAD
mkdir CDS	→ create the directory CDS
cd CDS	→ access the directory CDS
mkdir IC5141	→ create the directory IC5141
mkdir ASSURA314	→ create the directory ASSURA314
mkdir MMSIM60	→ create the directory MMSIM60

2.2.2 Menu Installation:

1. The user must insert the first CD in the drive.
2. Open a terminal and type:
   /cdrom/cdrom0/SETUP.SH
3. Specify path of install directory as:
   /export/home/CAD/CDS/IC5141
4. Do you want to start softload? [Y/N]

The user must type ¨Y¨.

5. Now it will ask for the ¨mail files¨, which is a code for specifying the sequence that the installation progress wants the CD's. The usual path for the ¨mail files¨ is: /export/home/CAD/cdsmailfiles/ "the name of the mailfile"

6. Then the user must choose default Choose default installation

7. The installation process will start. For changing the cd's you must follow these steps: File → Eject → Insert the next cd → In the pop window choose the 3rd selection that says ¨Cd Mounted¨ (Be careful, the user must also wait for the progress to pop the ¨File Manager¨ window before choosing ¨Cd Mounted¨).

8. After the installation the user must configure the installed products selecting the corresponding products.

2.3 Design Kits

2.3.1 Design Kid Installation

The next step is to setup the design kits of the corresponding technologies that the user want to use for the design. The design kits are shipped as compressed tar files. The user must create a folder named ¨libraries¨ to the home directory and a folder named tmp to the folder libraries. Then we paste the design kits on the folder tmp and after that we extract the design kits, each one separately. Finally we are ready to make the installation following the commands:

cd /home/libraries → Access the directory ¨libraries¨

. /kit_install → Initializing the kid installation progress

The user must choose as install directory the directory ¨/home/libraries¨

A ¨README¨ file contains installation instructions and other information related to the particular release of the design kit. An installation script, ¨ams_install¨, simplifies the installation procedure. The user is prompted for the installation path, <AMSPath>. A symbolic link, "rel", may optionally be set during install, and is a pointer to the current version of the design kit (usually the latest version). During installation, example files may also be updated to reflect the directory path of the kit installation. The design kit directory structure is under <AMSPath>/IBM_CDS/<technology>/<rel>.

<AMSPath>/IBM_PDK/<technology>/<rel>

2.3.2 Design Kid Contents and Tools

The Design Kit Contents (cdslib) are as shown below:

- **/IBM_PDK/bicmos7wl/<rel>/cdslib/bicmos7wl**

– CDS library containing

  • cellviews for each device (layout, symbol and netlisting views)

  • Symbolic Contacts (defined through the techfile)

  • Ascii techfiles for each number of levels of metal

– Diva checking decks and map files for stream in/out

- **/IBM_PDK/bicmos7wl/<rel>/cdslib/esd7wl**

– CDS library for ESD support

- **/IBM_PDK/bicmos7wl/<rel>/cdslib/doc**
- **/IBM_PDK/bicmos7wl/<rel>/cdslib/examples**
- **/IBM_PDK/bicmos7wl/<rel>/cdslib/Skill**

– Bindkeys for common functions

  • bindkeysCDS.il - standard CDS bindkeys with IBM_PDK functions added

– Compiled Skills

The supported Cadence tools for the IBM A&MS Design Kits are:

- Design Framework II
- Composer with Analog Artist Extensions (schematic entry)
- Spectre and SpectreRF Extensions (simulation)
- HSpice netlisting and simulations
- Virtuoso with Analog Artist Extensions (layout editor)
- Virtuoso XL (schematic driven layout)
- DIVA DRC (design rule checking)

  PRE (parasitic resistance extraction)

  LPE (layout parasitic extraction)

  LVS (layout versus schematic)
- Assura (hierarchical DRC and LVS)
- Data Translations using Stream In/Out

2.3.3 Design Kid Setup and Initialization

In the home directory of the design, the user must set the files *.cdsinit*, *.cdsenv* and *cds.lib*. The use of these of these files is explained below.

The *.cdsinit* file contains the standard Cadence initialization procedures and A&MS unique additions. The global variable *AMSPath* is set to the design kit installation path. Different versions of Cadence require separate compiled context file.

Examples files which are provided with the design kit are customized for the design kit full path during design kit installation. The example .cdsinit file needs to be copied into the users Cadence startup directory. This file contains the code that conditionally loads the compiled Skill global utilities based on which CDS version is used.

Users may also assign commonly used commands to keystrokes ‘’Bindkeys’’ by loading the ¨bindkeyAMS.il¨ or ¨bindkeyCDS.il¨ file, which is located in the Skill subdirectory. Furthermore they can also be loaded from the .cdsinit file as shown below:

load(strcat(AMSPath "bicmos7wl/V1.2.1.0AM/cdslib/Skill/bindkeyAMS.il")).

Two different bindkey files are provided; bindkeysAMS.il has the shortcut keys set up consistent with prior IBM internal tools and bindkeysCDS.il has the basic Cadence default bindkeys. Either of these files can be used as a base and a user or CAD group administer can modify as desired.

.cdsinit

```
;;   This will place the banner, AMSutils, on each edit session
;;   A&MS trigger functions are
;;       schAMSTrigger          for schematic and symbol editors.
;;       leAMSTrigger           for layout and extract editors.
;;
;;   If the A&MS trigger functions are unregistered by the customer site, the
;;   AMSutils banner may be placed by issuing the procedure placeAMSMenu().
;;   This function is bound to the PF11 key, or the "Stop" key on the SUN5
;;   keyboard.
;;

;;Uncomment the following line for Sequence Columbus-RF
;;load(strcat(getShellEnvVar("SEQUENCE_ROOT") "/lib/skill/cdsinit"))

AMSPath="/home/andreou/libraries/IBM_PDK/"

  cond(
      ( rexMatchp("4.4.6" getVersion())      loadContext(strcat( AMSPath "bicmos7wl/V1.3.2.0DM/cdslib/Skill/
procAMS500.cxt"))
                                             hiRegTimer("trInsertMenu()" 1))
      ( rexMatchp("5.0" getVersion())        loadContext(strcat( AMSPath "bicmos7wl/V1.3.2.0DM/cdslib/Skill/
procAMS510.cxt"))
                                             hiRegTimer("trInsertMenu()" 1))
  )

AMSutils()

;;Uncomment the following lines to override AMS utils DIVA switch settings
;;ams->DIVAdrcSwitches = "GridCheck"
;;ams->DIVAextSwitches = "resimulate_extracted"

load(strcat( AMSPath "bicmos7wl/V1.3.2.0DM/cdslib/Skill/bindkeyAMS.il"))
```

Design kit installation path

Bindkeys Path

Figure 2.1: View of “.cdsinit” file.

The example .cdsenv file should be placed in the user's root directory, or important lines from the example file can be added to an existing .cdsenv file.

The cds.lib example file is also customized for the installation path during the design kit install process. This file contains the library definitions for the bicmos7wl technology library, the esd7wl library for electro-static discharge (ESD) devices, and various Cadence libraries such as analogLib and basic. The photo shoot of cds.lib is shown below. The first two blue arrows show the paths for the define of bicmos7wl and the red arrow shows the path for the define of the design we have created, which is "GILBERT_MIXER". The other libraries before the blue arrows, like "analogLib", are default libraries of Cadence that mainly have ideall elements.

Display properties are controlled by the display.drf file (shipped in the technology library) and may either be merged within Cadence or copied into the users "home" path prior to starting Cadence.

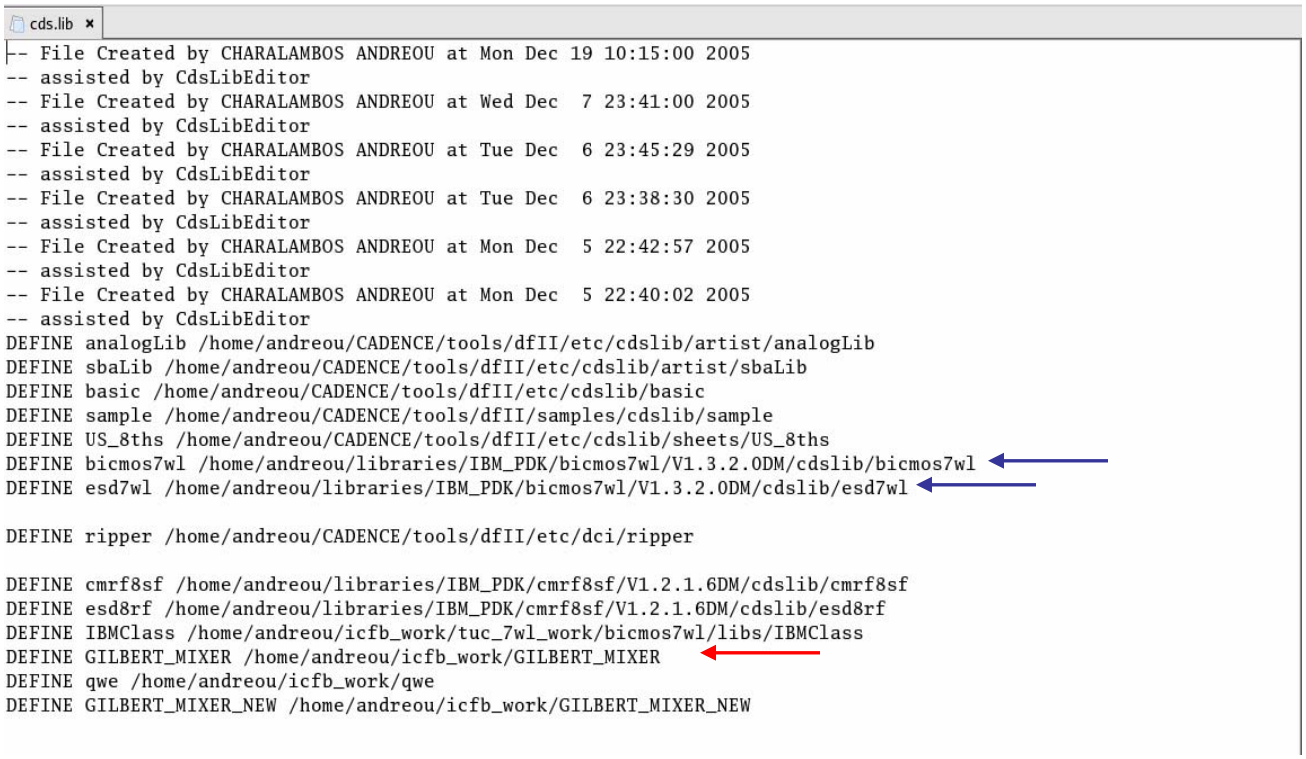


Figure 2.1: View of "cds.lib" file. The blue arrows show the paths for defining bicmos7wl and the red arrow shows the path for defining the design we have created, which is "GILBERT_MIXER".

2.3.4 Library Tech Files

Cadence techfile contains information for pcells and symbolic contacts which are need for doing layout.
Below are the default techfiles compiled in bicmos7wl for six level metal:
• AM last metal option: M1, M2, M3, M4, MT, AM
• DM last metal option: M1, M2, M3, MT, E1, MA
• ML last metal option: M1, M2, M3, M4, MT, ML
Furthermore the user can choose to re-compile bicmos7wl library for desired metals.

CHAPTER 3

The design was done using Cadence Framework II and IBM 7WL 018µm design kit. In the next few pages we will explain the procedure step by step of the whole design. Furthermore, we will explain the verification of the design as also the simulation steps using the same tool.

3.1 NFET 0.18µm Technology Overview

In the design we have used the standard NFET which operates at 1.8 volts and has an effective gate oxide thickness of 35A and a minimum drawn channel length of 0.18µm (0.145 µm effective channel length). Idsat, a figure of merit for FET devices
(defined as the drain current of the minimum channel length device with the nominal supply voltage on gate and drain with the source and body grounded), is 600 µA/µm of device width. The high-Vt version of this device has 170 mV higher threshold voltage with an Idsat of 500 µA/µm.

The NFET length and width are the primary pcell options. Multi-fingered layouts (multiple PC gates in parallel, within the same RX shape) are supported through the "number of fingers" prompt in the Cadence Component Description Format (CDF). The CDF "width" parameter is the total width of the multiple fingers, and the "width (parallel)" reflects the actual RX width. "Connect terminals," wires the multiple sources and drains of multi-fingered FETs.

PARAMETER	NFET
Max Voltage (V)	1.98
Tox (physical, A)	35
Ldrawn (µm)	0.18
Leff (µm)	0.145
Idsat (µA/µm)	600
Vt (long wide mv)	355

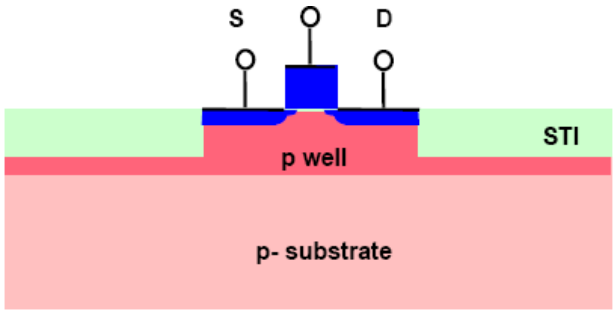
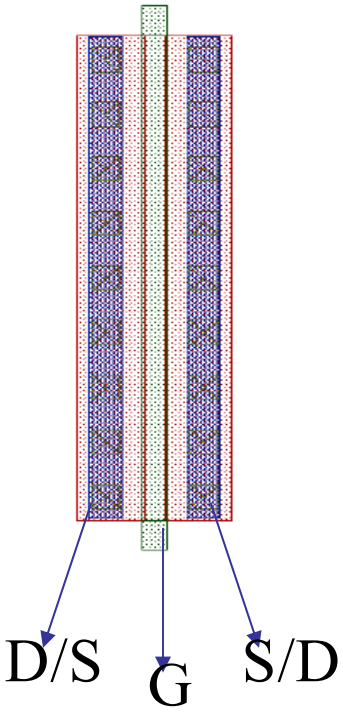


Figure 3.1: NFET 0.18µm characteristics.



3.2 Design

3.2.1 Cadence Design Environment

Before a design is created, a Cadence user environment must be setup by copying some files to our Cadence working directory. These files are .cdsinit, .cdsenv, cds.lib and display.drf, and their use was explained in "2.3.3 Design Kid Setup and Initialization".

The Cadence environment can be initialized using a command prompt in the home directory of the design, and executing the command "icfb". It is important to be in the design home directory when executing the above command, otherwise we will not be able to access the needed libraries and designs from the "library manager". The "icfb" window will be as shown in figure 3.2.

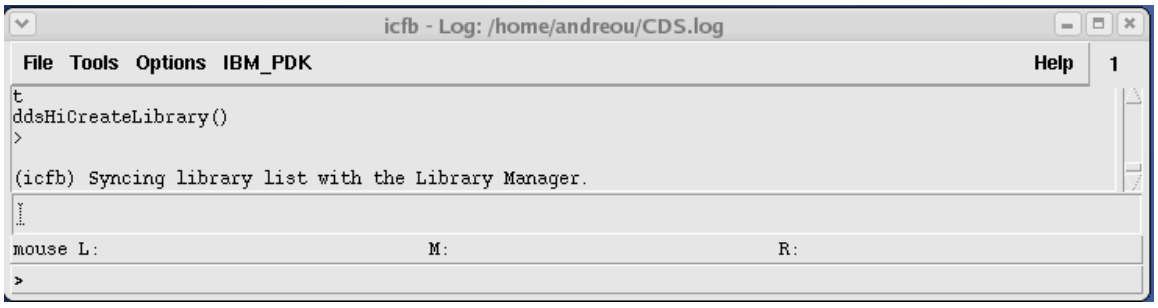
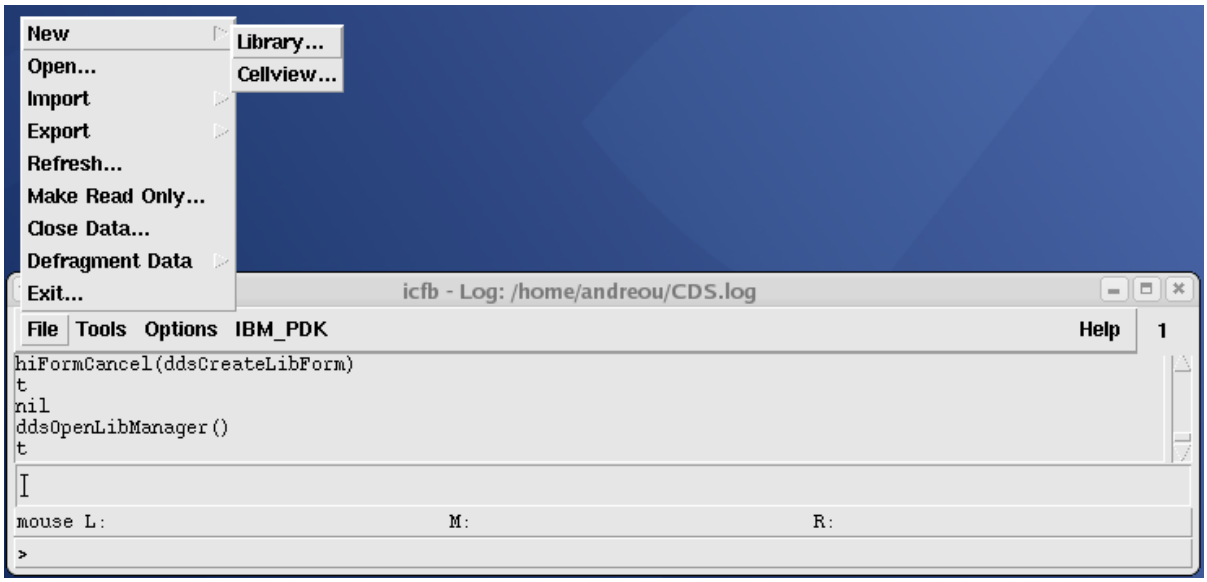


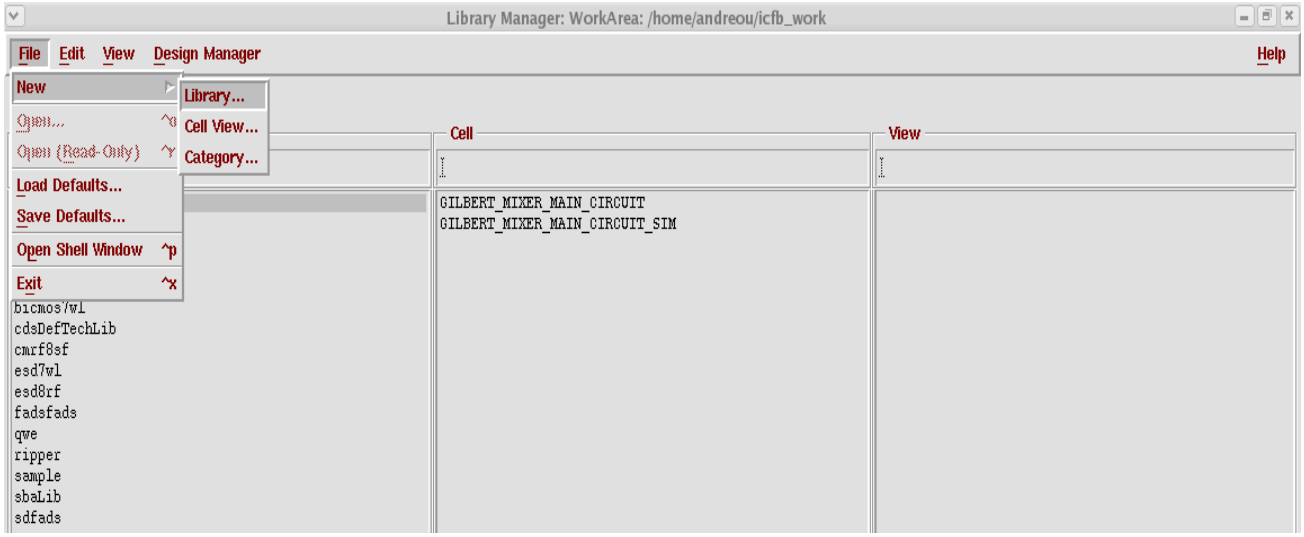
Figure 3.2: ICFB window.

3.2.2 Create a Design Library

From icfb window, choose File → New → Library.

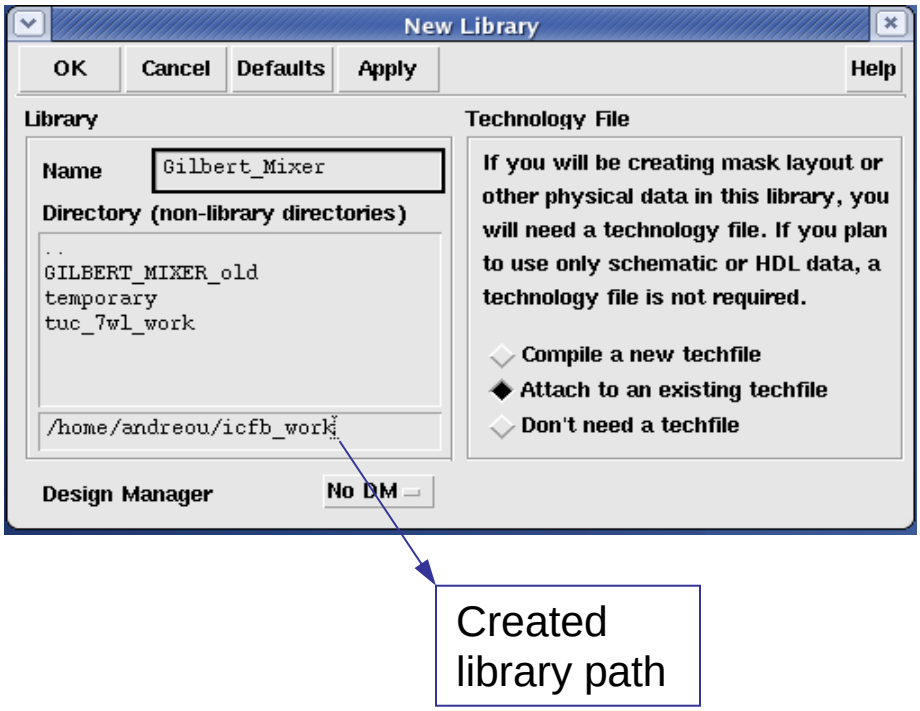


Alternatively from the Library Manager window, select File → New → Library.

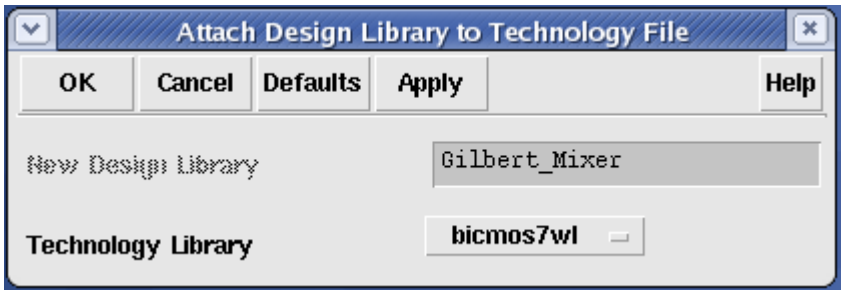


The New Library form appears, where we type the name of the library that we want to create (GILBERT_MIXER) and we choose the path of the directory that the created

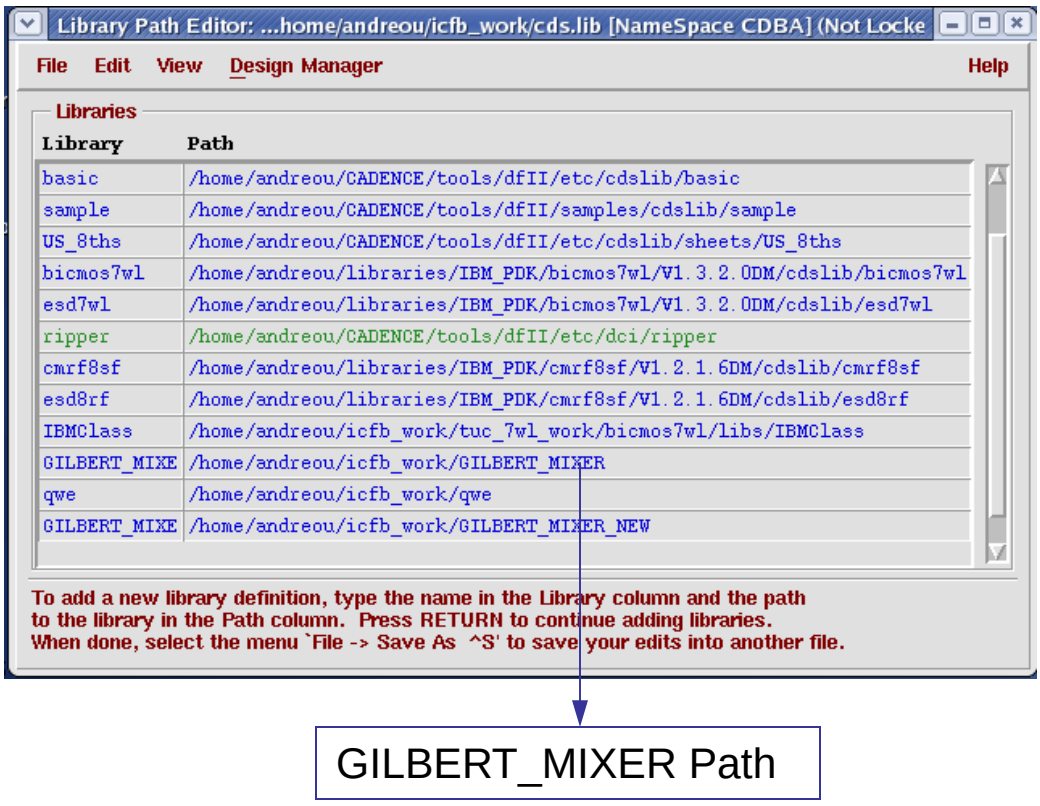
library will belong. As a technology file we choose “Attach to an existing techfile” and then select OK to the form.



At the next form we choose the Technology Library which will be one of the existing as we select that in the previous form “bicmos7wl”. We select OK to the form.

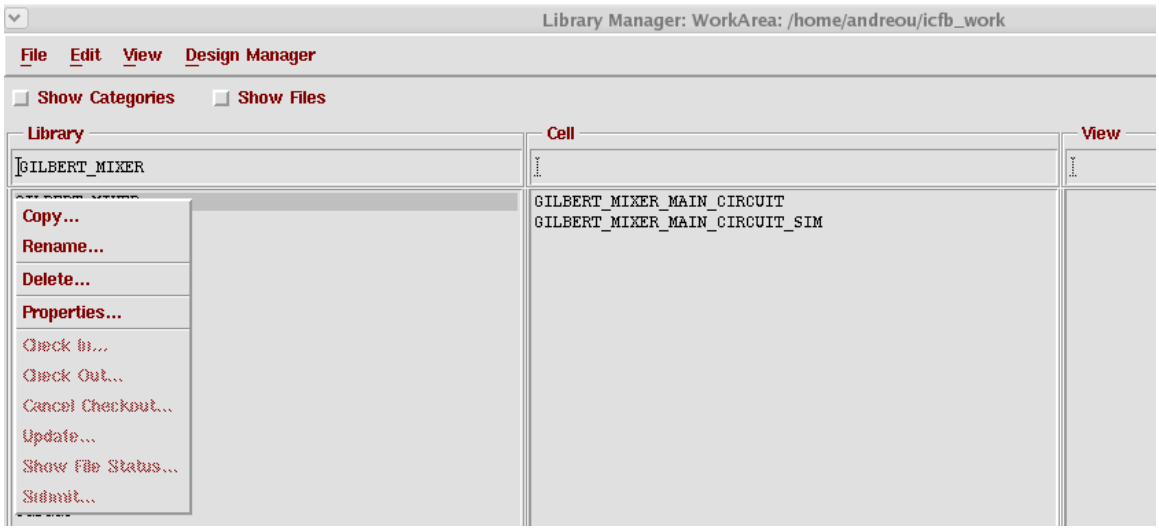


The Library Manager will display now the libraries that are defined in the cds.lib file in our home directory, including the “Gilbert_Mixer” that was just created.

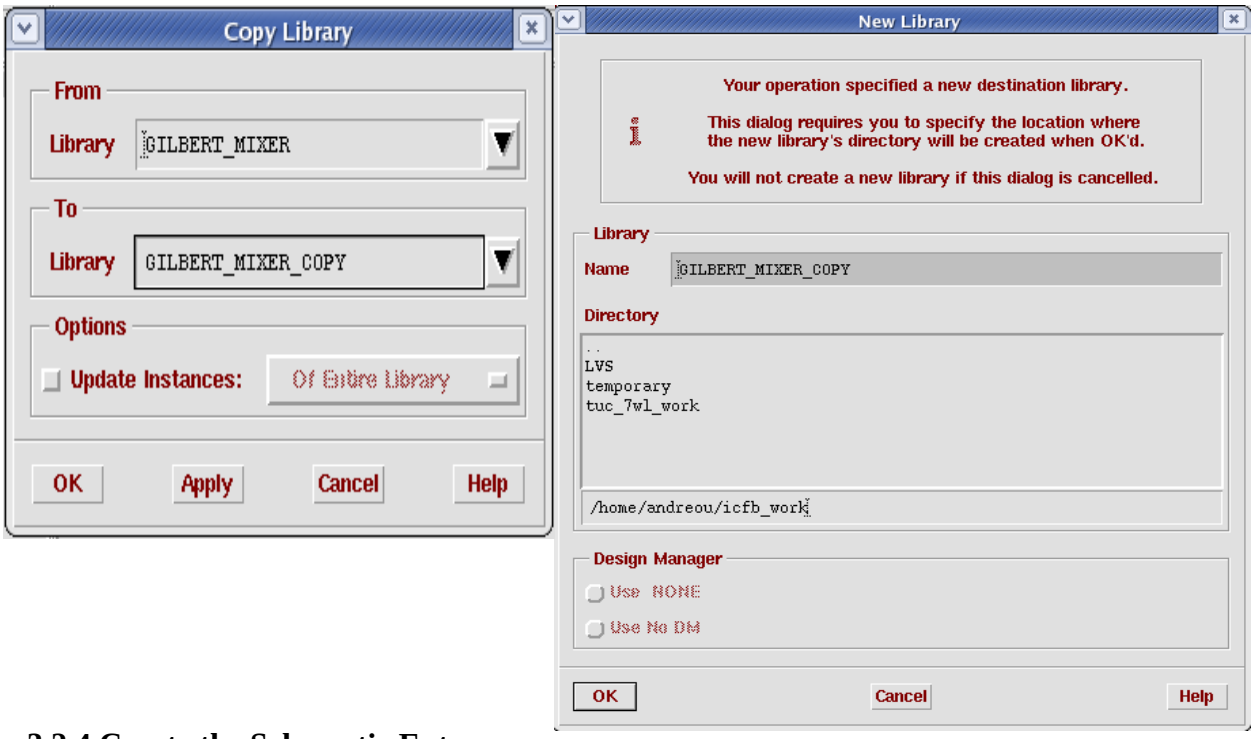


3.2.3 Copy a Design Library

After we have created a library we can also make a copy of that library by right clicking on it and then selecting “copy” as shown below.

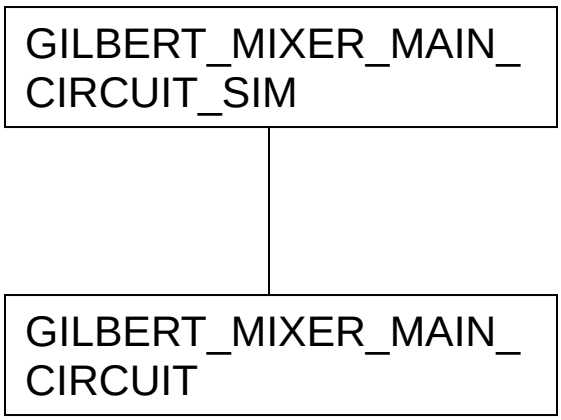


A form like the left form below will appear, where we can choose the library we want to copy and the library we want to create or update. We select OK at the form and a new form like the right one below will appear. In the new form we select again OK ignoring the message and we have ready the copy of our library.



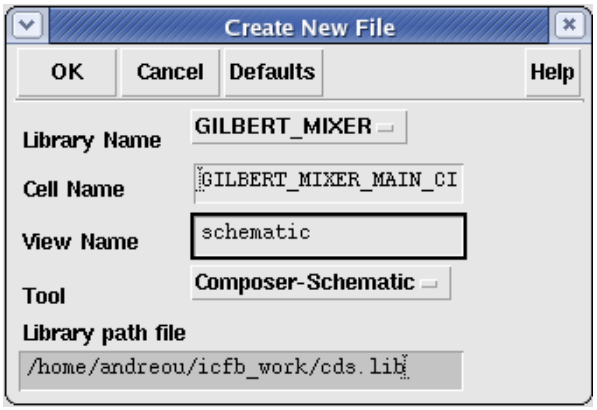
3.2.4 Create the Schematic Entry

In this step a Gilbert mixer schematic will be entered into Cadence using the composer schematic entry tool and the bicmos7wl technology library components. The overall hierarchy of the design will be as shown in the figure below. The "GILBERT_MIXER_MAIN_CIRCUIT" Cell is the circuit of the mixer and the "GILBERT_MIXER_MAIN_CIRCUIT_SIM" Cell is the circuit of the simulation including bias and ground as also the implementation of the Transconductor.



For creating the "GILBERT_MIXER_MAIN_CIRCUIT" Cell we follow the steps that are shown below?

1. From the icfb menu, we select Tools → Library Manager
2. From the Library Manager menu, select File → New → Cell View.
3. At the appeared form we can enter the name of the Cell View that we want to create, as also the View Name and the tool which at the moment is "Composer-Schematic".
4. Finally we select OK to the form.



After the last step of the previous form, a Virtuoso Schematic Editor will appear, where we can start our design of the schematic.

Before starting the design it would be very helpful to know some important "bindkeys". During the design and the simulations we can use those "bindkeys" rather than the toolbar of library manager for simplicity and manageability. The shortcuts for the main "bindkeys" are listed below depending of which kind of bindkeys ("bindkeysAMS.il" or "bindkeysCDS.il") we have loaded to the ".cdsinit" file in our home directory:

I	→	Add instance
Shift + Q	→	Properties
W	→	Wire
Esc	→	Cancel last command
Ctrl + D	→	Deselect an object
Q	→	CellView properties editor
E	→	Enter Hierarchy

Shift + Home → Return to top of the Hierarchy

Ctrl + E → Return Top

R → Rotate (only for layout "virtuoso view")

Alt + r → Redo

S → Stretch

C → Copy

M → Move

Del → Delete

F → Normal size center view

Alt + E → Current CellView

Alt + v → LVS

V → World view

P → Add pin

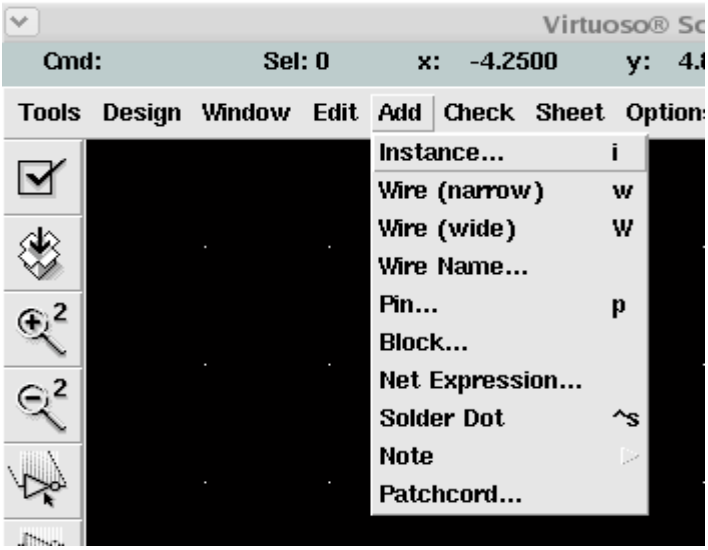
L → Add note text

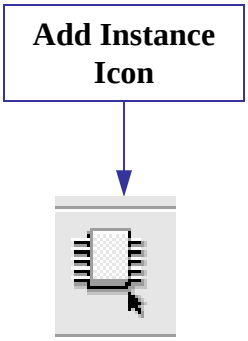
T → Add terminal

N → Add net

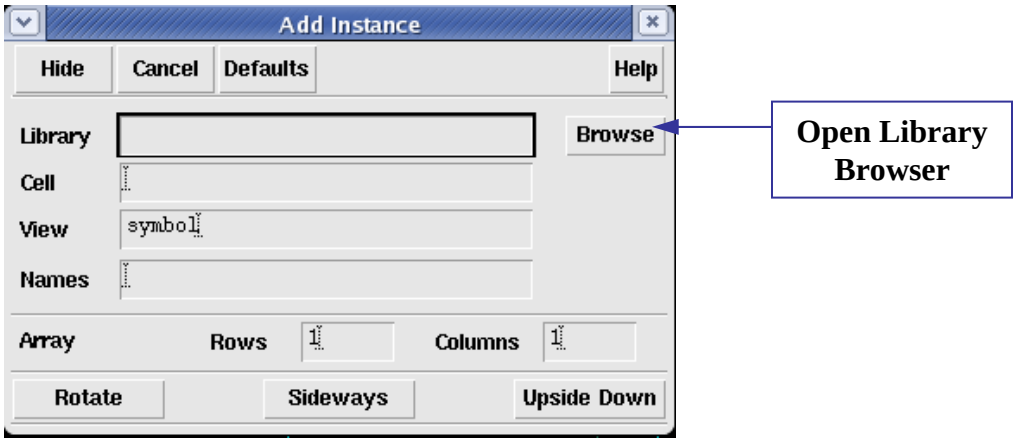
3.2.5 Adding Instances to the Schematic Entry

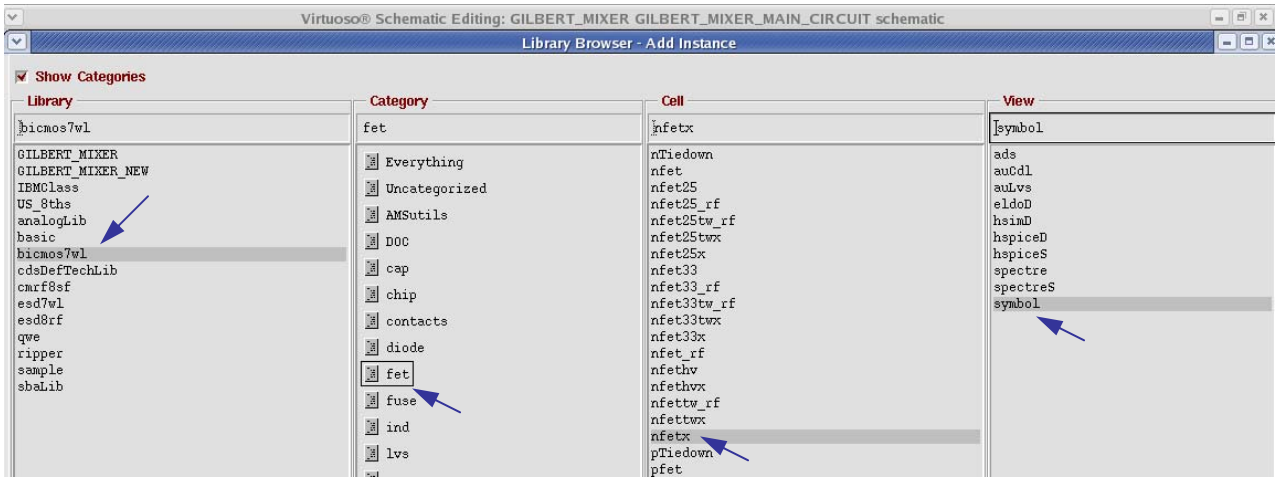
1. Using the menu from Virtuoso Schematic Editor, we select **Add → Instance**, or we can use the bindkey "i", or the "instance icon" at the left menu bar.





In the “Add Instance” form that appears, we select “Browse”, and then in the next form we choose the library bicmos7wl, category as “nfet” and View as “symbol” as shown below.





2. After completing the last step the symbol of nfet will appear at the Virtuoso Schematic Editor. We can rotate the symbol and with a left click to enter it in the Schematic Editor. With the same way, or using “copy-paste” we can set the eight “nfetx” transistors that are needed for our design. Using the same way we add the instances of two “res” and two “ind” that are needed for the design as also “Gnd”.

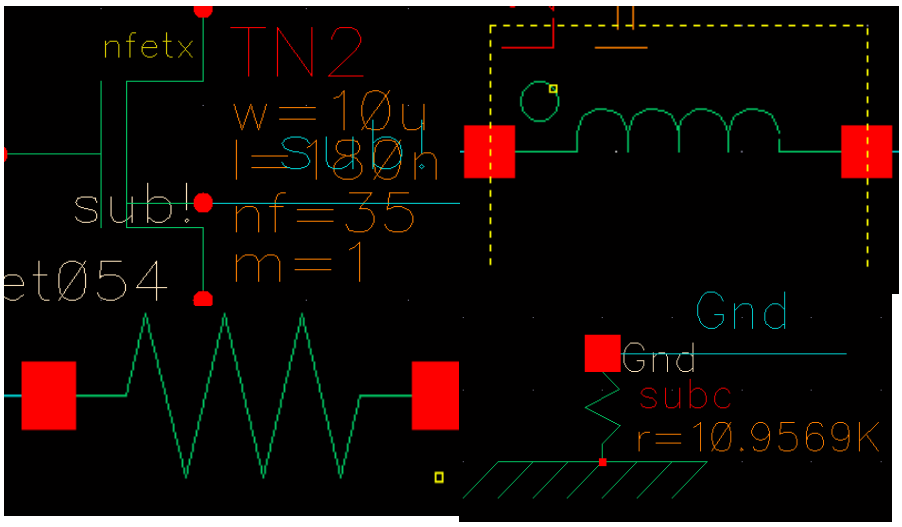
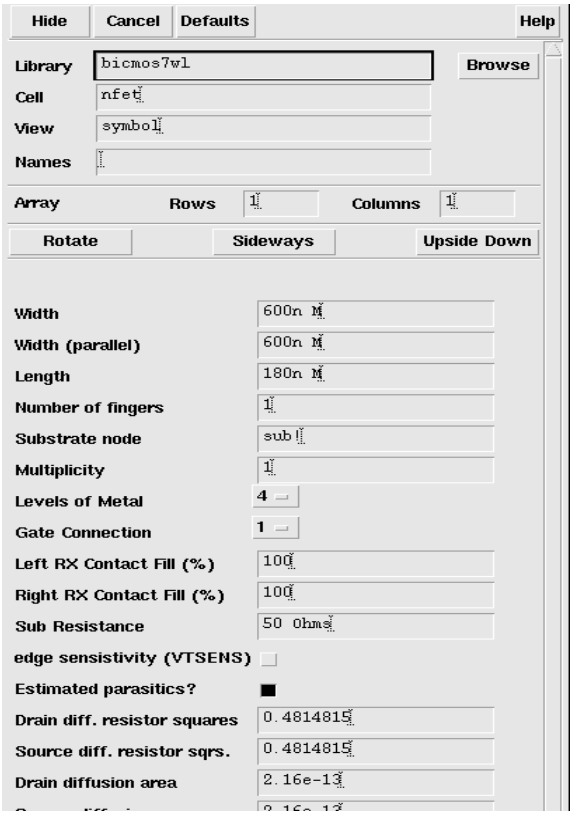


Figure 3.2: Mixer components.

3. Then using the bindkey "Shift+Q", or the properties icon  at the left site bar, or **Edit → Properties →Objects** from the menu banner , we can adjust the properties of the devices, like width, length, number of finger and multiplicity that are important for the device as shown below.

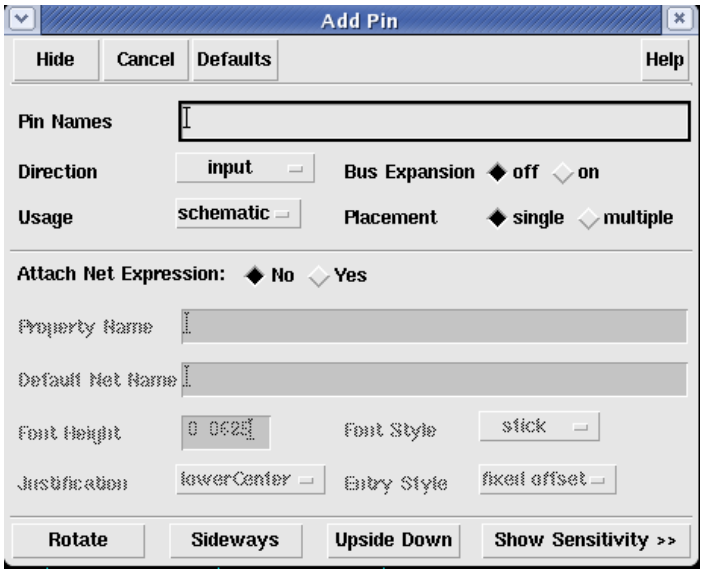


4. Once all devices are placed we must do the connections between the devices through wires using the icon,  at the left side bar, or **Add → Wire** from the menu banner.

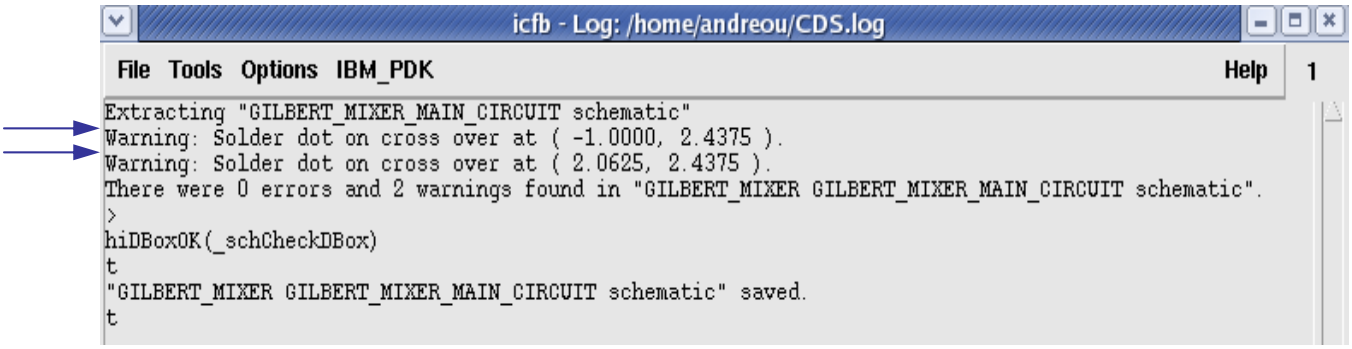
5.Furthermore we must add the pins, using **Add → Pin** or using the toolbar icon,  . The pins that we must add are:

Inputs: Vdd, BIAS, LOp, LOn, RFp, RFn, Gnd.

Outputs: OUTp, OUTn.



6. Once we have finished the schematic, we must check and save the design using, **Design → Check and Save**, from the menu banner.



The two warnings that are shown in the "icfb" after Check and Save, and are shown above are not something important, as they may be bugs of cadence (when more than two wire cross each other a "solder dot on cross" warning appears).

The final schematic is shown below in normal and world view.

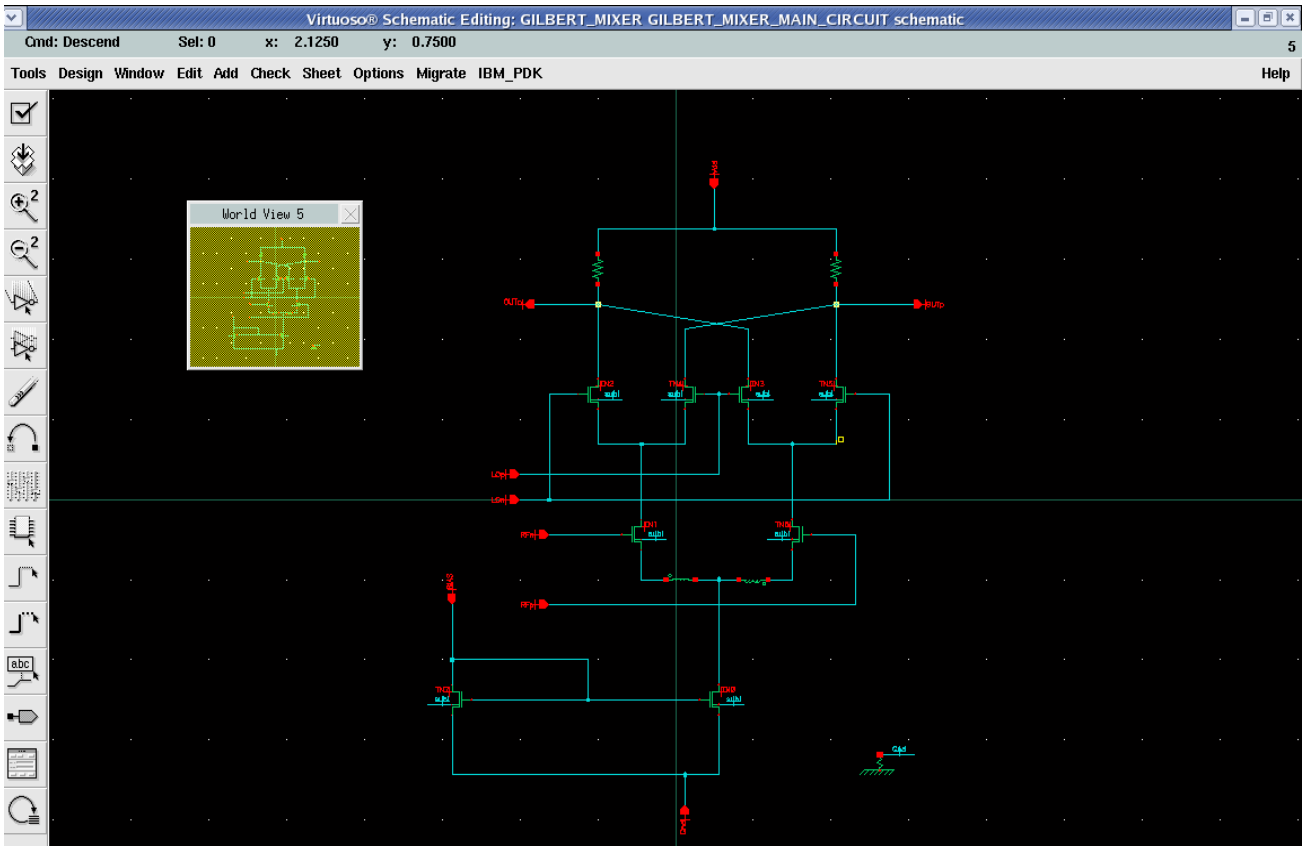
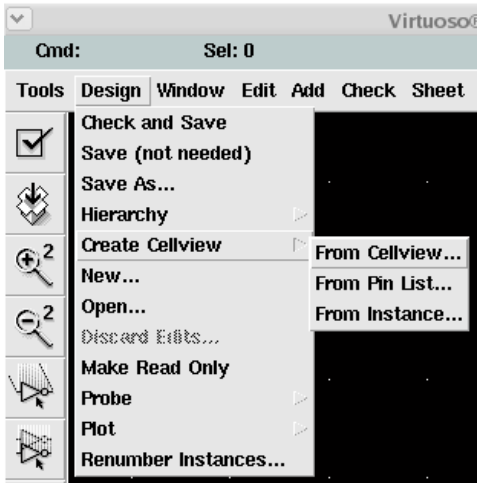


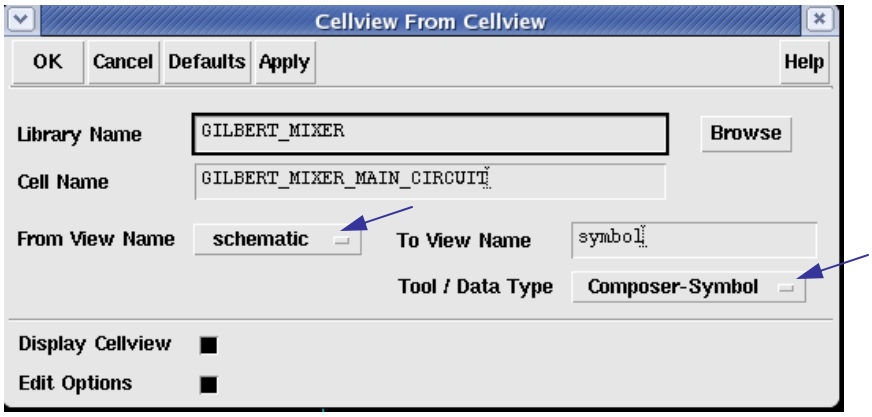
Figure 3.3: Gilbert Mixer schematic.

3.2.6 Create Symbol from the Schematic Entry

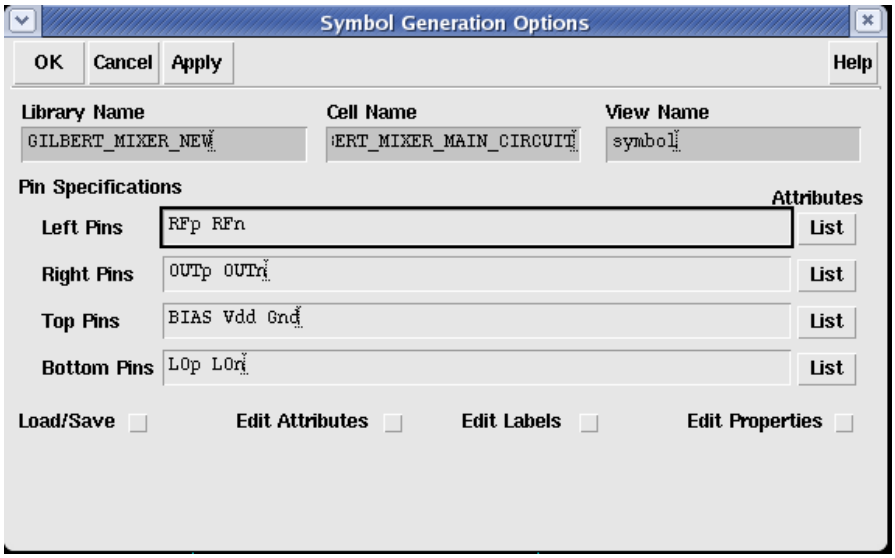
1. Having the Virtuoso Schematic Editor of "GILBERT_MIXER_MAIN_CIRCUIT" Cell open we select **Design → Create Cellview → From Cell View** from the menu banner.



2. In the “Cellview From Cellview” form that appears we select “Composer-Symbol” as Data Type as shown below and then OK with the default settings.



3. Then, the Symbol Generation Options form will appear and we may rearrange the pins specifications according to our needs.



5. After rearranging, we select OK to the form and the Composer Symbol editor will appear with the “GILBERT_MIXER_MAIN_CIRCUIT” symbol in view as shown below.
6. Finally we save the symbol to complete the procedure.

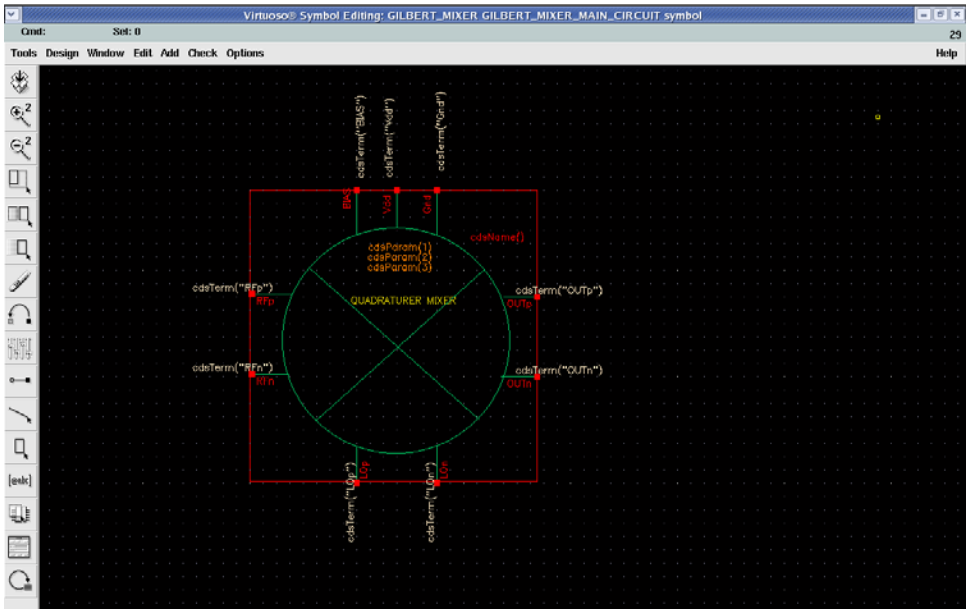


Figure 3.4 : Gilbert Mixer symbol.

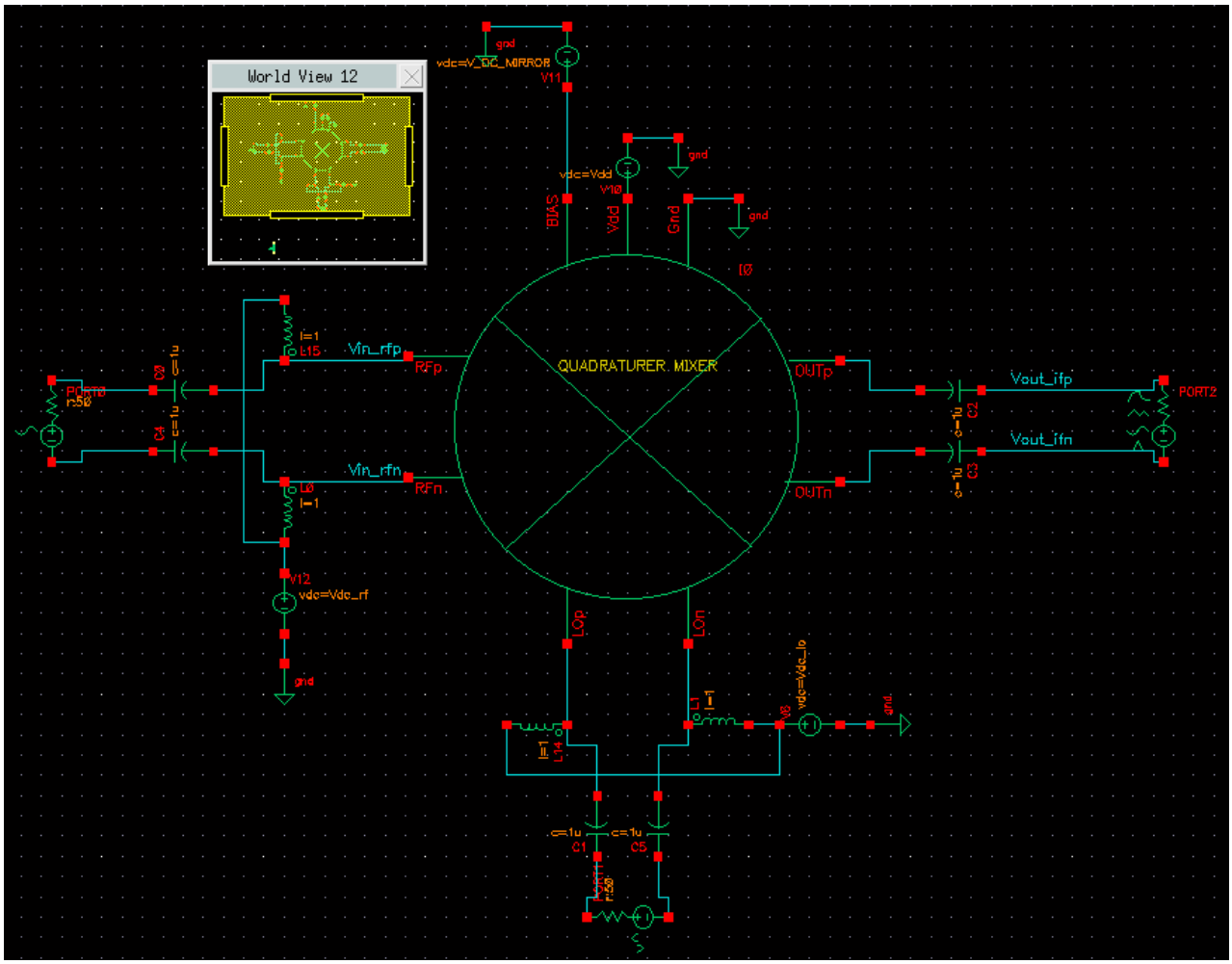
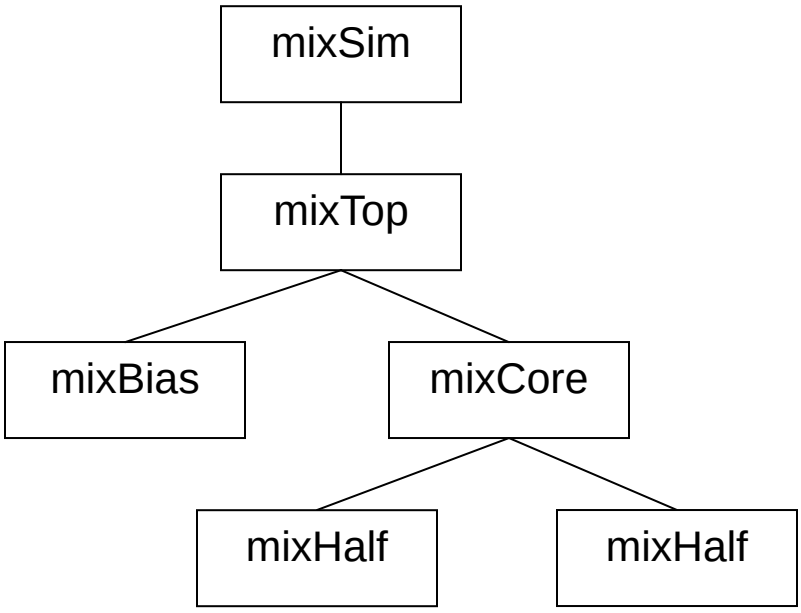


Figure 3.5: MixSim schematic

3.2.7 Create the Layout

For the Gilbert cell mixer we didn't design a layout, but we have design it for a passive mixer that was used before the design of Gilbert for the training and the familiarization with Cadence Framework II tool. Below we will show the schematics without the procedure of their design as it's similar to the design of schematics of Gilbert and then we will explain step by step the Layout design procedure. The design was done with the same technology as Gilbert mixer, IBM 0.18μm CMOS technology.

The hierarchy of that mixer is:



The schematics of the mixer are shown below in order to the hierarchy:

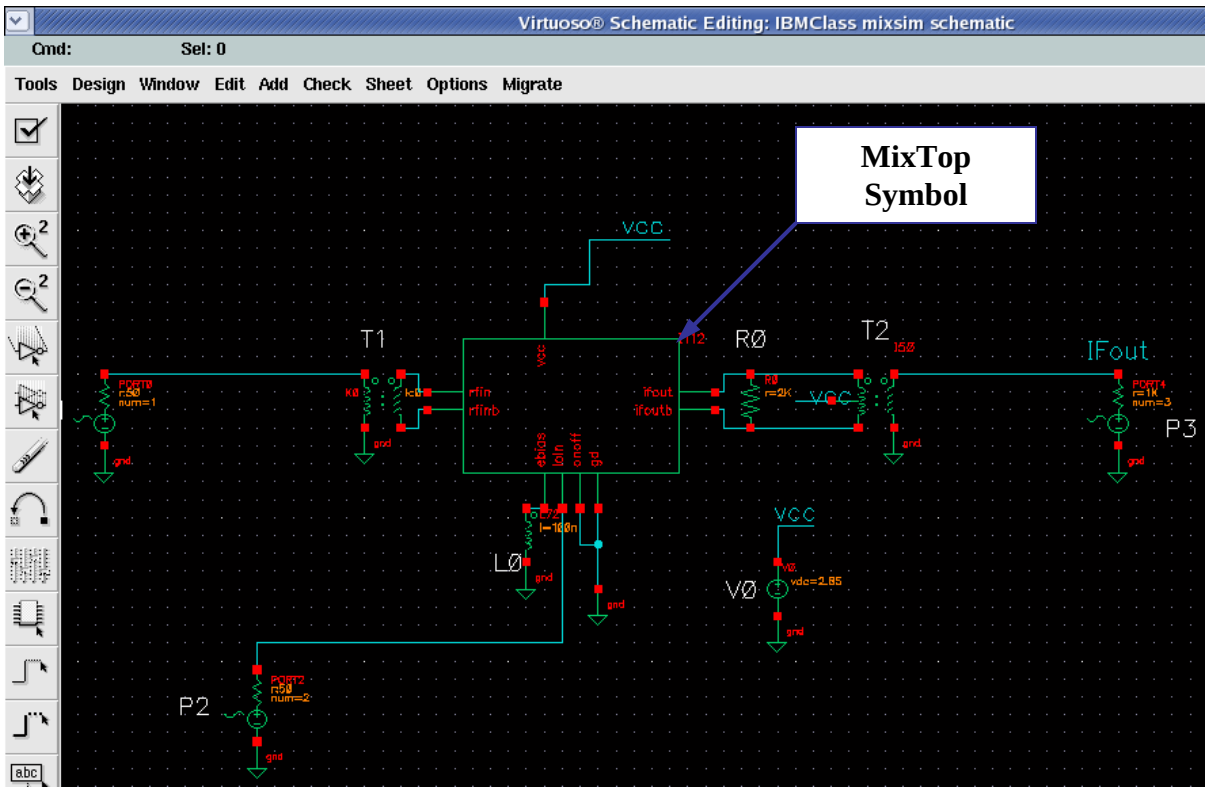


Figure 3.6: Mixsim

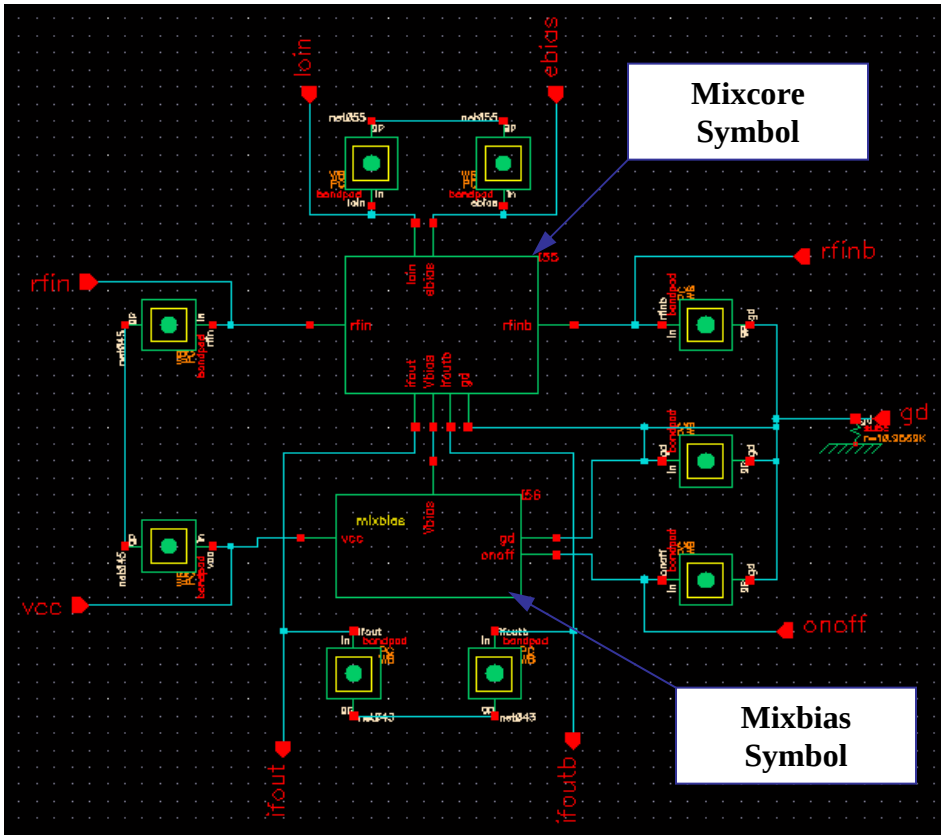


Figure 3.7: Mixtop

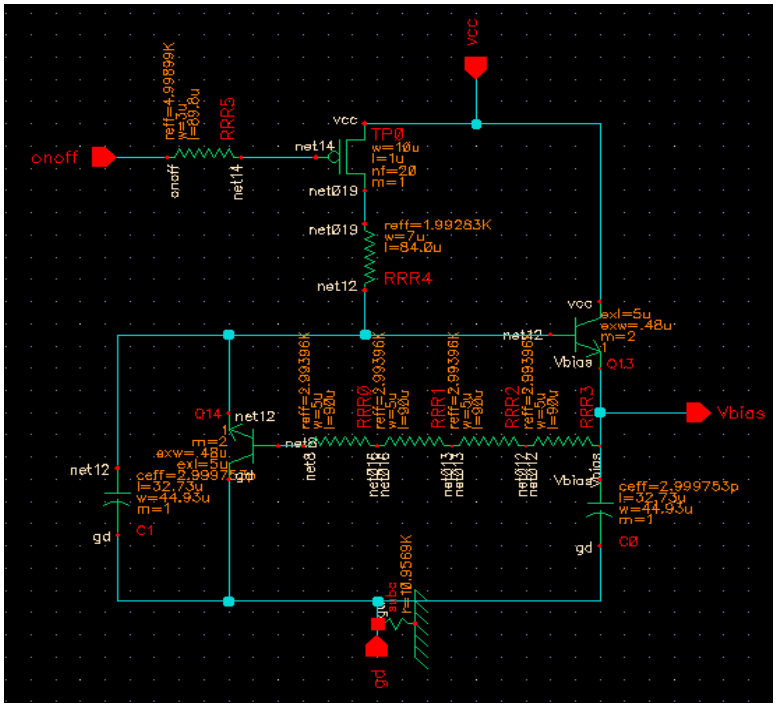


Figure 3.8:Mixbias

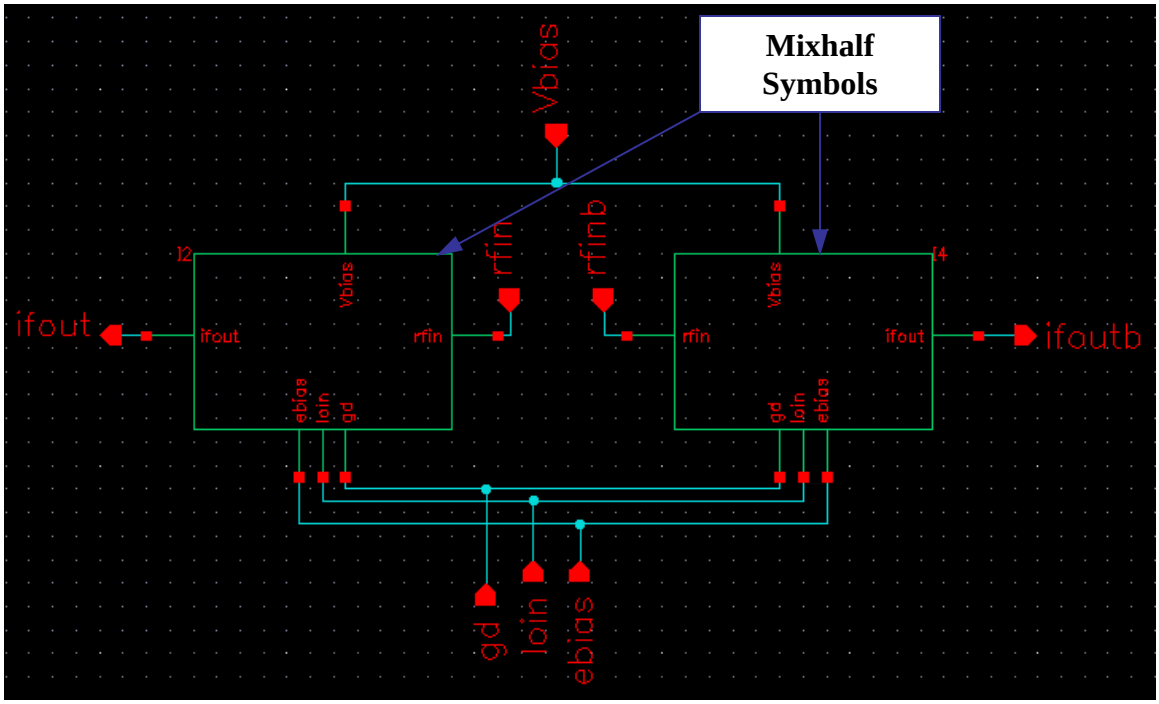


Figure 3.9: Mixcore

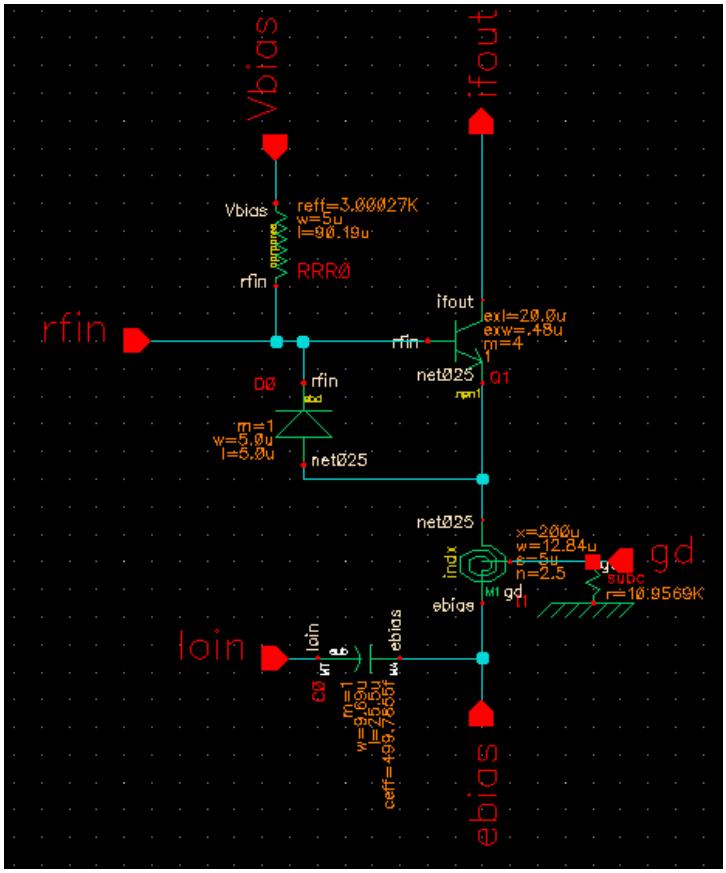


Figure 3.10:Mixhalf

After finishing design flow of the schematics we are ready to start the design flow of Layout. A very important thing before starting the design is to be sure that we have the correct “display.drf” file in our design home directory, otherwise the devices won’t be able to be displayed correctly.

Objectives

· Use Virtuoso XL to generate a layout from a schematic

· Use Virtuoso XL to aid in routing nets

· Place the *Image* PCell guard ring

· Establish bulk connectivity of inductors and bondpads

3.2.7.1 Create the Layout using Virtuoso XL

Virtuoso XL tool partially automates the physical design process by automatically placing layout PCells from the schematic view into the Layout XL editor. Once the cells are placed in the layout editor, visual aids are provided to assist in the routing of physical interconnects between layout PCells. The mixer layout hierarchy will follow that of the

schematic, including layout cells for *mixTop*, *mixCore*, *mixHalf* and *mixBias*. An alternative way is to add the devices manually using the menu banner, **Add → instance**, and choose the proper device from library manager. Here we will explain the Virtuoso XL tool because the manual method is very simple and similar to the way of adding devices to the schematic entry.

1) From the Library Manager, we open the *mixTop* schematic cellview from our library.
2) In the Composer banner menu, we select,
   Tools → Design Synthesis → LayoutXL
3) The "Virtuoso XL Startup Option" form appears.
   At that form we select Create New, and click OK.
4) Then we fill out the "Create New File" form.
   Select OK to the form
5) The Virtuoso XL layout window will appear, and the schematic Composer and layout windows should resize and position themselves next to one another. The Virtuoso XL window design area appears blank at this point.
6) In the Composer schematic window, we select:
   Design → Check and Save
7) In the layout window, we select:
   Connectivity → Update → Components and Nets.
   A new form, "Layout Generation Options", will appear and will allow as to choose which metal layer to use for our layout pins (the layout terminals that corresponds to pins in the schematic). The layout pins will be used to define connectivity of the layout cell. Now we can also change the pin width and height.
8) We select the "Create Labels" button to add labels to the layout pins and fill out the form.
9) "Display Pin Name Options" button can be used to change the properties of pins, like height and orientation.
10) We select OK to both forms and Use the bindkey **f** to fit the design to the window.

11) The layout window will now consist of all the elements in the schematic (pins, pads and circuit blocks). To see all of the levels we can press the **F** key. To place all of the elements in the "prBound by" shape we selected

Edit → Place From Schematic.

12) For the wiring of the layout devices, Layout XL can highlight the networks that need to be connected to match the schematic of the design. Therefore, in the Layout XL window, we select

Connectivity → Show Incomplete Nets.

A form, "Show Incomplete Nets", appears, which will allow as to interactively turn on and off 'flight-lines' on nets that are not yet wired. We can leave this form open while we wire the circuit.

The final design of the layout (mixtop) will be as shown below:

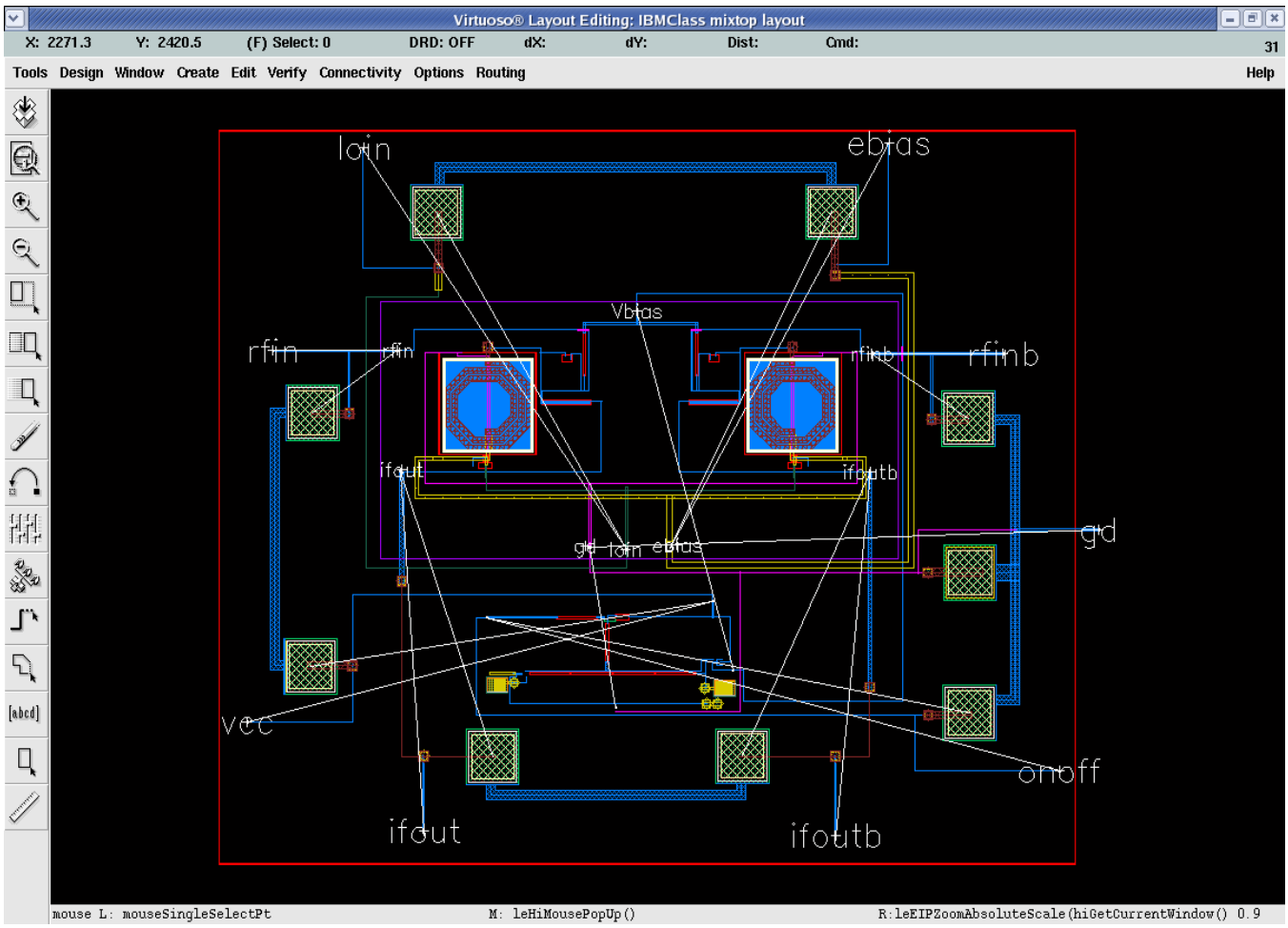
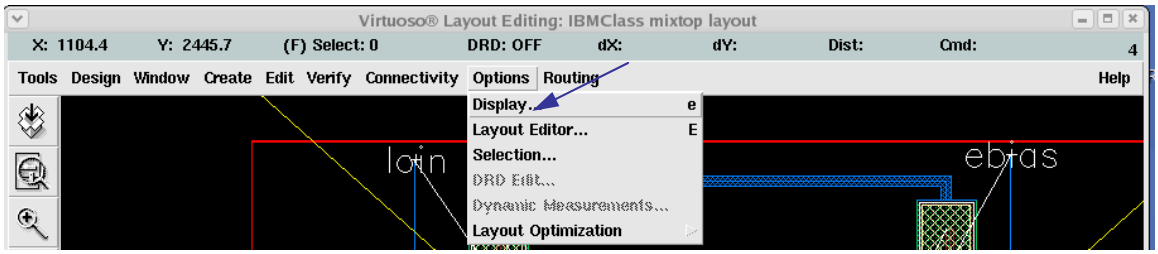
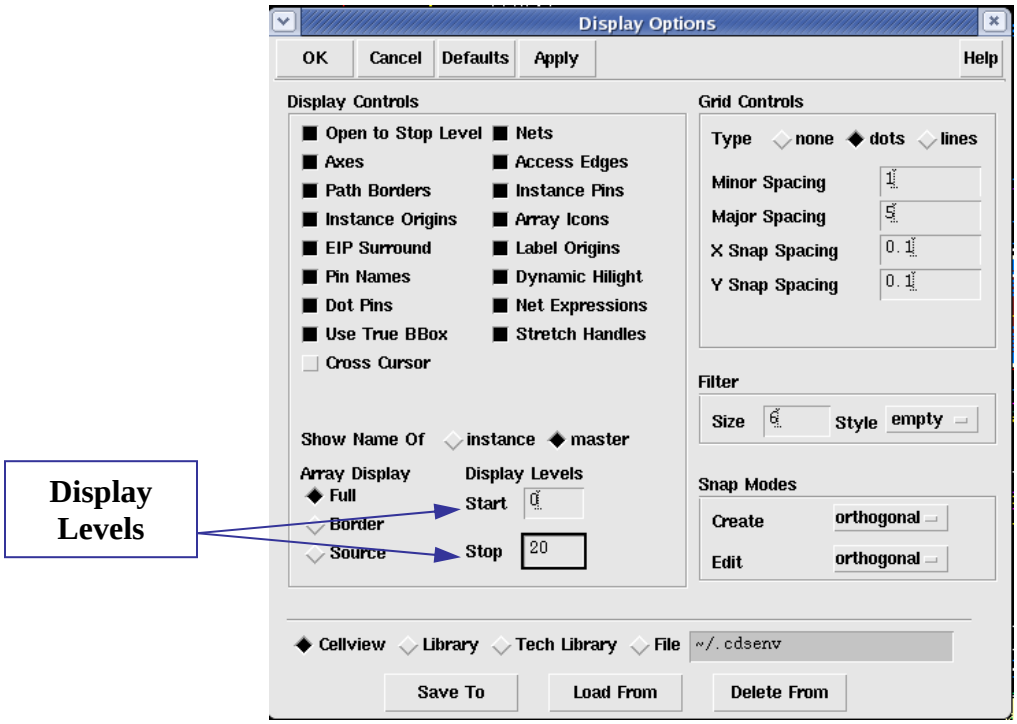


Figure 3.11: Mixtop layout.

If we have the correct “display.drf” file and we still have problems with the display colors we should check the display options: Options → Display



Then the display options window will appear, as shown below. At the “Display Levels”, we choose Start as zero and also we choose as stop, a number greater than zero as show. Using this form we can also adjust many other Display Controls of Virtuoso Layout.



3.2.7.2 Verification Tools

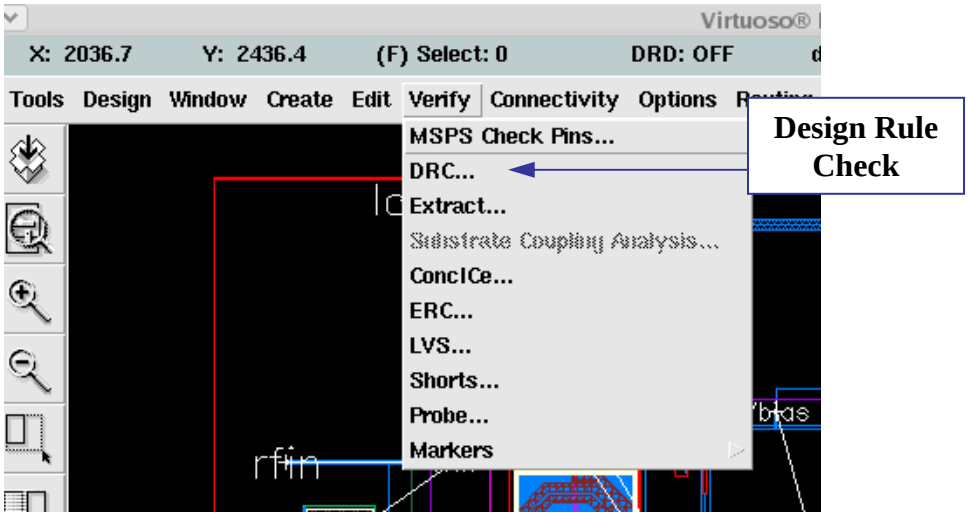
Physical designs submitted for manufacturing must meet several verification criteria.

- DRC (Design Rule Checking)
- Layout Extraction
- LVS (Layout Vs Schematic)
- Pattern Density

3.2.7.2.1 DRC (Design Rule Check)

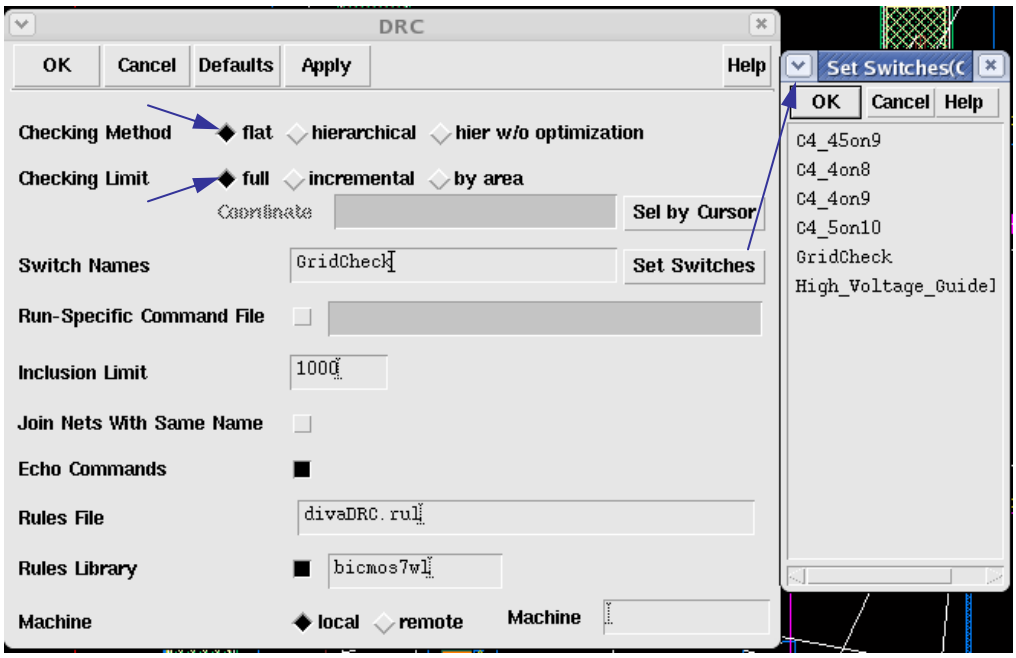
1. From the Library Manager, we open the layout view of cell mixTop.

2. From the Virtuoso menu banner, select **Verify → DRC.**



3. A form will pop-up asking if we want to run a PRE-DRC check. We choose "Yes". After the PRE-DRC check is run, in the CIW, there will appear a report that contains information on stacked points, non-orthogonal lines, and poorly formed paths. This is only an indication that the pre-DRC check has run.

After the pre-DRC check is complete, the Cadence *DRC* form will appear with default options for switches and rules file. These can be modified for a custom DRC run.



At “DRC” form we choose “flat” as a checking methot, “full” as checking limit and “GridCheck” as Switch Names. The Switch Names can be set using the button “Set Switches” that will open the “Set Switches” form as shown in figure*****above.
Furthermore we must choose the correct “Rules File” and “Rules Library” corresponding to the technology that was used.

4. We select “OK” to this form to run DRC.
DRC violations create error and warning markers to highlight the violations of the defined ground rules. Cadence provides a methodology for looking at these markers. The AMS utilities provide another alternative through the
AMS utils → Display → Marker Select.
The AMS function will allow cyclic field choices of unique marker error message text that is written to the log as a summary at the completion of the run. Once a group of markers is selected, there is a second AMS function that will fit (zoom) the display to one error marker at a time from the selected set.
For further explanations of a design rule violation, we can search the Design Manual for the specific design rule number. To do that, in the CIW, we select
AMS utils → Documentation → Design Manual.

5. A clean DRC run contains some standard errors that are mentioned as “#INFO” as shown in figure3.12.

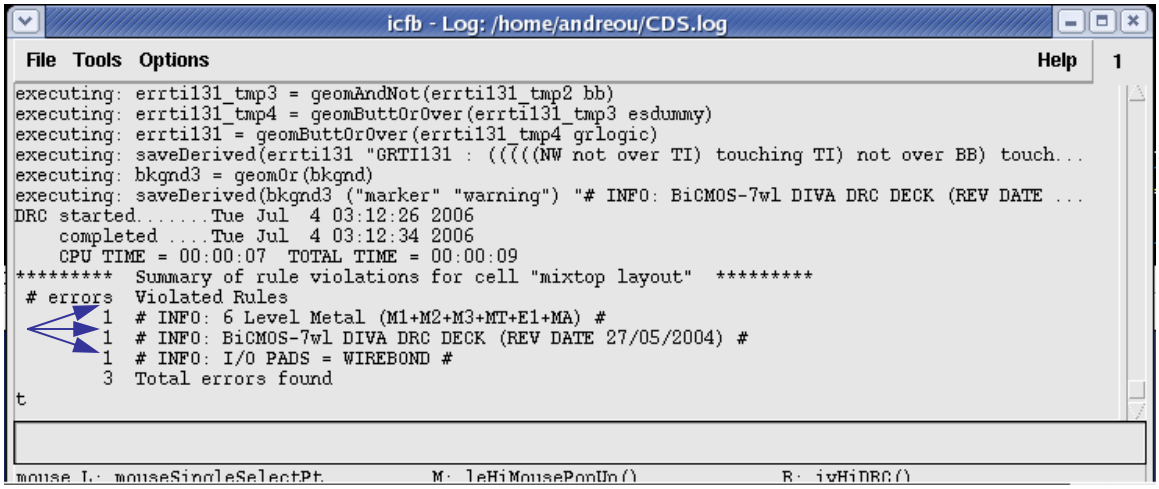


Figure 3.12: Design Rule checking results, showing the three standard errors that we should ignore.

We can delete error markers by using the ALT+DEL bindkey in order to be able to make the appropriate corrections.

3.2.7.2.2 Layout Extraction

1. With the "mixTop" layout cell open, we can extract the cell by selecting

AMS utils → Checking → Extract from the Virtuoso menu banner. Using the defaults in the Extract form, we select OK to run the extraction.

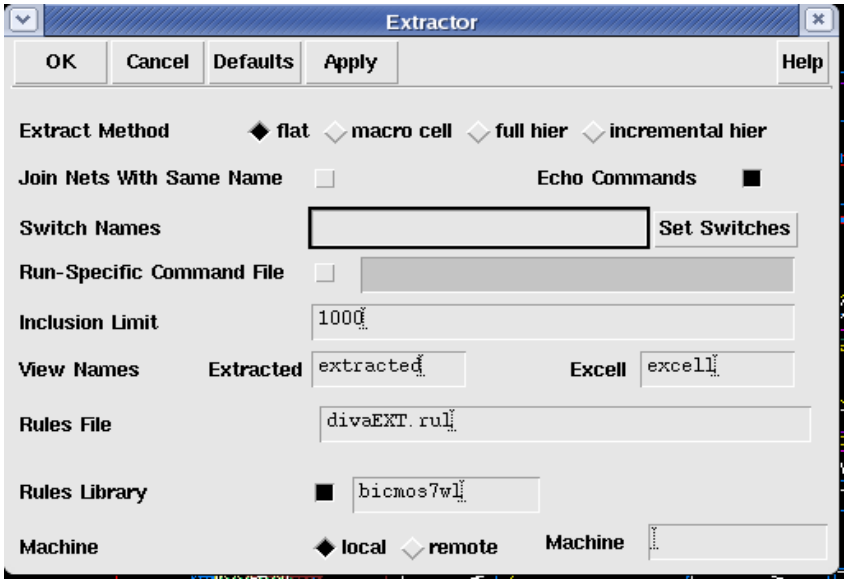


Figure 3.13: Extraction of Layout.

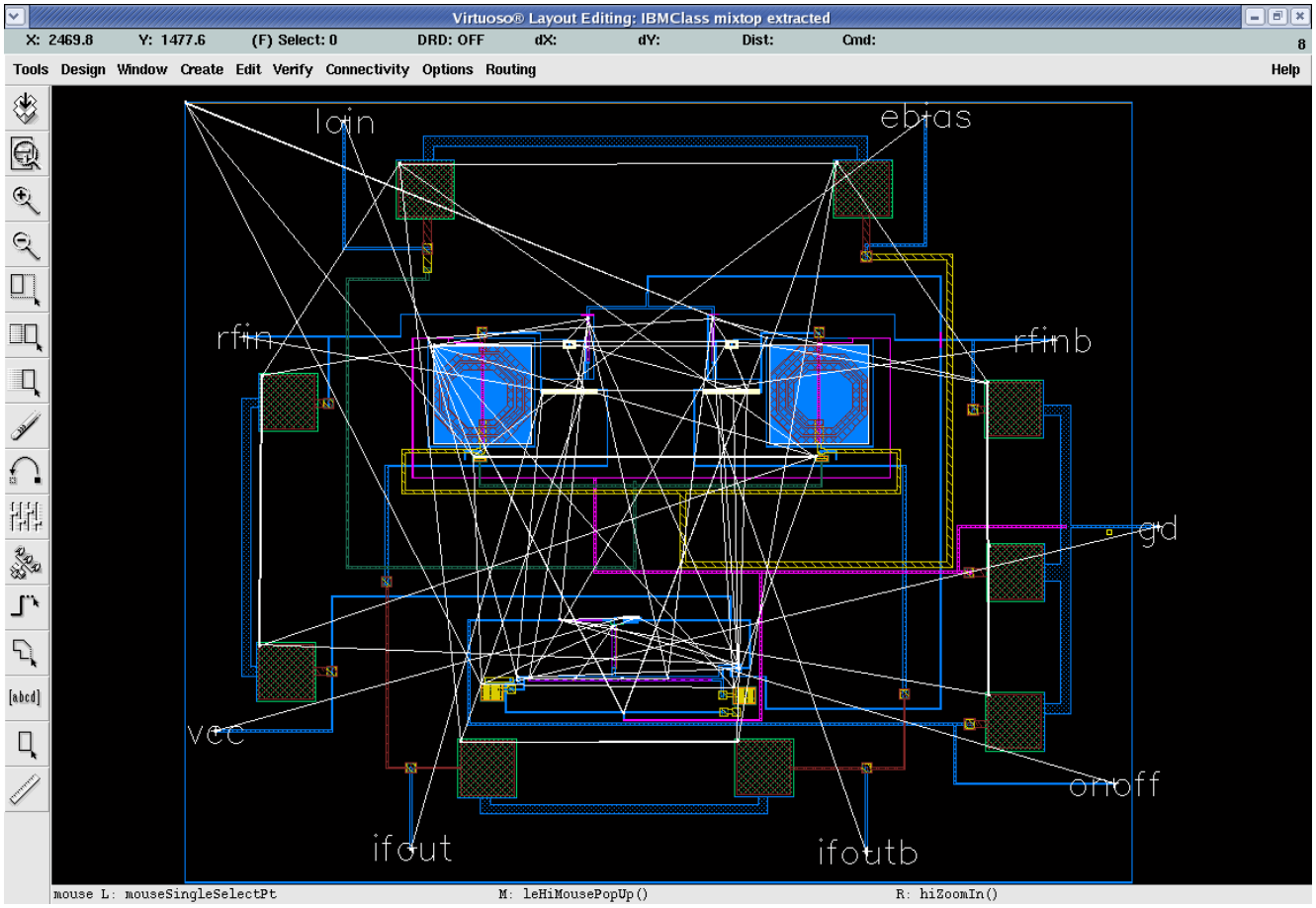


Figure 3.14: Mixtop Extracted View.

The final extracted view is shown in figure3.14 and will be used for the Layout versus Schematic verification.

3.2.7.2.3 LVS (Layout Vs Schematic)

LVS is required to verify that the schematic and layout represent the same netlist.

1. From the Library Manager, we open the schematic view of *mixTop*.
2. **Design → Check and Save** the schematic data.
3. From the extracted view of *mixTop*, we select **Verify → LVS**

   A couple of forms will appear. One will give a choice between using the Form Contents or the Run Directory Contents. We choose the Form Contents and select OK at the form.

   The second form has all the options for the LVS compare execution.

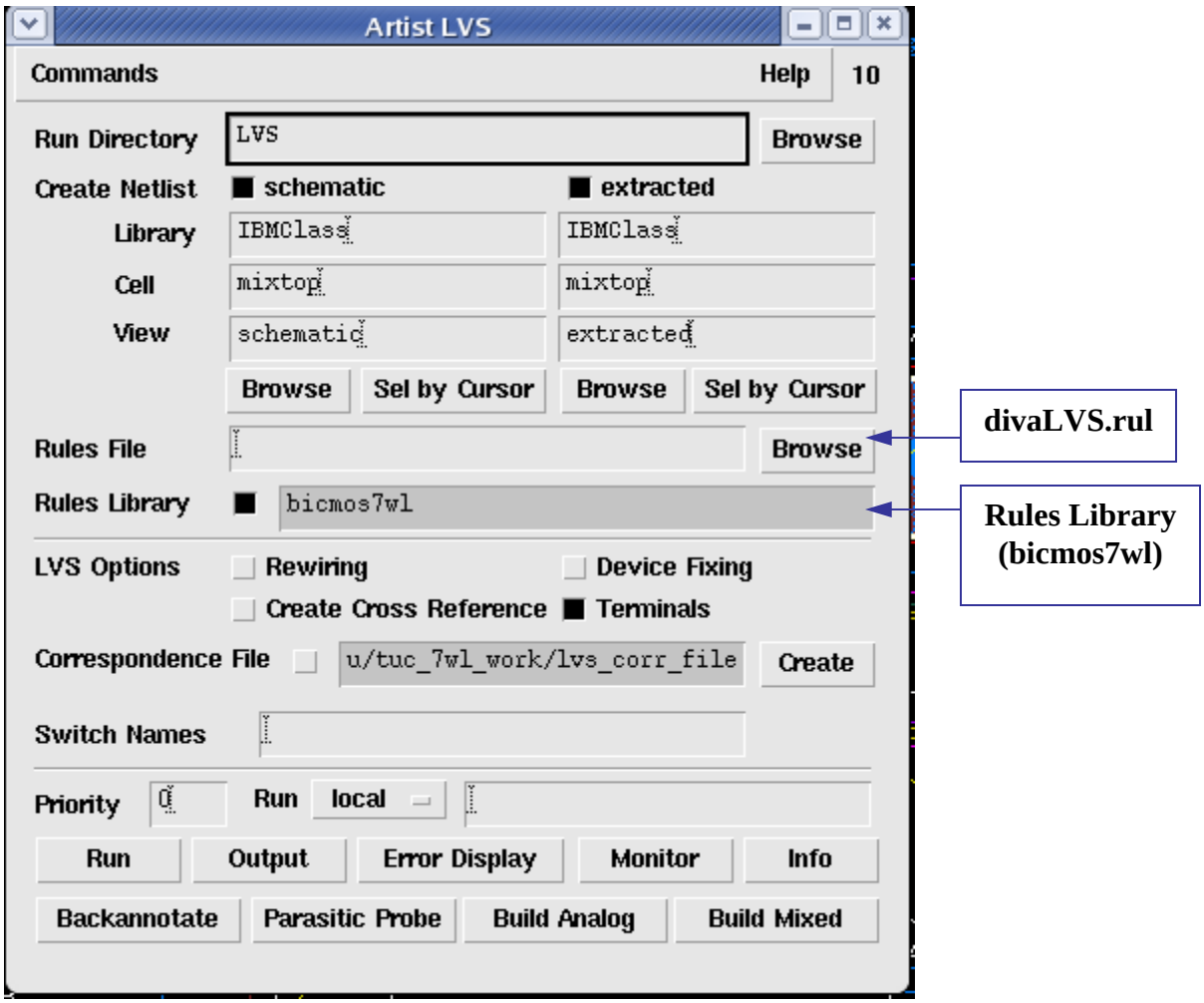
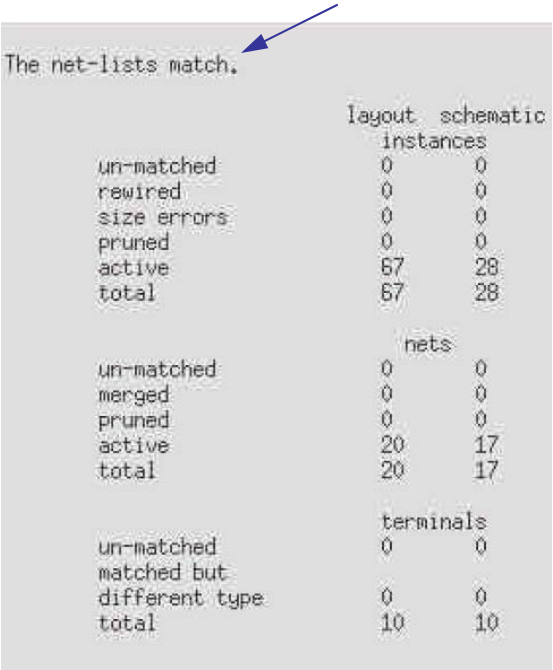


Figure 3.15:Artist LVS form.

We must set the appropriate Rules File (divaLVS.rul) and the appropriate Rules Library (bicmos7wl).

4. Finally we click the "Run" button in the *LVS* form, which is near the bottom left of this form. When the LVS job completes, a form will pop up saying the job either has finished successfully or has failed. If the job completes successfully, this does not mean there were no errors, only that the job did not abort. If the LVS job fails to complete, we can use the LVS form to find out why. From the *LVS* form, we click on the "Info" button. In the *Display Run Information* Form that appears, we click the "Log File" button to view the reason for the failure. Results of the LVS job can be found in the LVS run directory specified at the top of the LVS form. The output files in this directory will get a copy of all errors.

5. After a successful LVS run, we browse the output file by clicking the “Output” button in the LVS form. Netlists that match will include output, including the line “The net-lists match”:



```
The net-lists match.

                          layout  schematic
                            instances
      un-matched            0       0
      rewired               0       0
      size errors           0       0
      pruned                0       0
      active                67      28
      total                 67      28

                              nets
      un-matched            0       0
      merged                0       0
      pruned                0       0
      active                20      17
      total                 20      17

                            terminals
      un-matched            0       0
      matched but
      different type        0       0
      total                 10      10
```

Figure 3.16: LVS net-lists.

An easy way to identify LVS mismatches is the use of “Error Display” utility found on the bottom LVS form. This will cycle through the LVS mismatches and hi-light schematic or layout devices or nets that do not match.

3.2.7.2.4 Pattern Density

A pattern fill check will be performed and the *aFil* library device will be placed in the layout. The view layout_pd contains both the layout and the shapes required to pass pattern density.

1. First of all we open the layout view of top cell for edit
2. With the mixTop layout cell open in Virtuoso, we run:

AMS utils → Checking → Pattern Density.

This will pop-up a “DRC” from with the rules file changed to divaDensity.rul.

We choose “OK” to the form to run the job, and view the results in the CIW.

The output should look similar to the following:

********* Summary of rule violation for cell "mixTop layout"

****

0 Total errors found

12 Antenna errors

t

Pattern density checking for "bicmos7wl" technology

q2 percent area = 0.30 *** minimum 0.5%

lm percent area = 35

m2 percent area = 8.25 *** 30.0% < M2 < 70.0%

m1 percent area = 19.4 *** 30.0% < M1 < 70.0%

pc percent area = 3.50 *** 15.0% < PC < 30.0%

np percent area = 1.10 *** 4.0% < NP < 20.0%

ma percent area = 8.18

dt percent area = 17.0

m3 percent area = 3.18

m4 percent area = 2.05

- "Antenna errors" are standard and should be ignored.
- Percent areas are given for all levels that are required to meet pattern density ranges.

Mask levels that do not meet the requirement are followed by the required range. In the above example, the *q2*, *m2*, *m1*, *pc*, *ne*, *np* and *rx* layers do not meet density requirements. These layers must be added to the layout.

Therefore, we place a few instances of *aFil* in vacant portions (inside the *Image* guard ring) of the *mixTop* layout.

3. Finally we rerun the pattern density check to see how the *aFil* cells changed the density requirements.

CHAPTER 4

GILBERT CELL MIXER SIMULATIONS

Setting Up Model Libraries

1. In the Simulation window, we choose Setup - Model Libraries and the Model Library Setup form appears.

2. In the "Model Library File" field, we type the full path to the model file of the technology we have used for the design. For our design we must include the files "design.scs" and "process.scs" as shown in figure4..

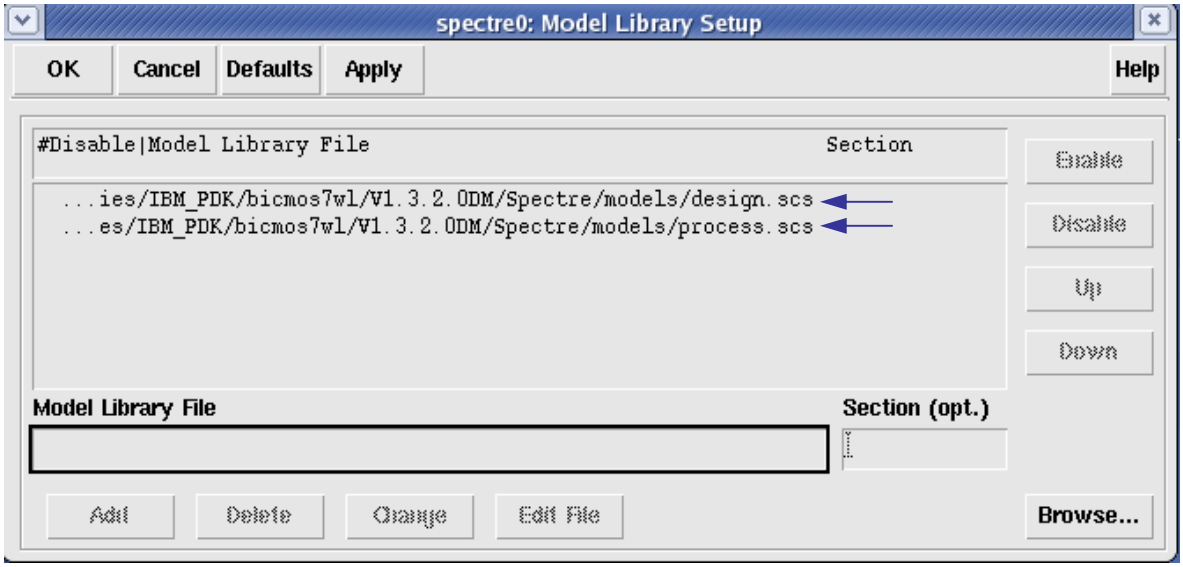


Figure 4.1: Model Library Setup.

3. In the Model Library Setup form, we click on *Add*.

The completed form appears like the one below.

4. In the Model Library Setup form, we click *OK*.

5. And finally we must disable any analyses we may have ran previously.

4.1 DC ANALYSIS:

1. Open the schematic window "GILBERT_MIXER_MAIN_CIRCUIT_SIM".

2. In the Schematic window, we choose *Tools- Analog Environment*.

The Simulation window opens. This window is also called the Cadence® Analog Circuit Design Environment.

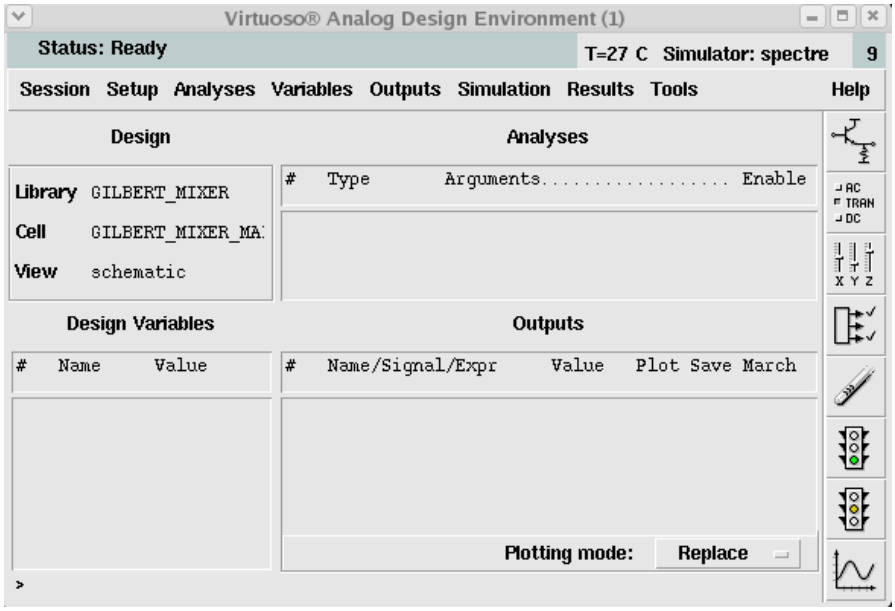


Figure 4.2: Virtuoso Analog Design Environment.

3. From the Analog Environment menu banner, we select **Variables →Edit,** and the window in figure will appear. We can set one by one all the design variables in respect to the specifications of our design and the needs of the different kind of simulations that will follow.

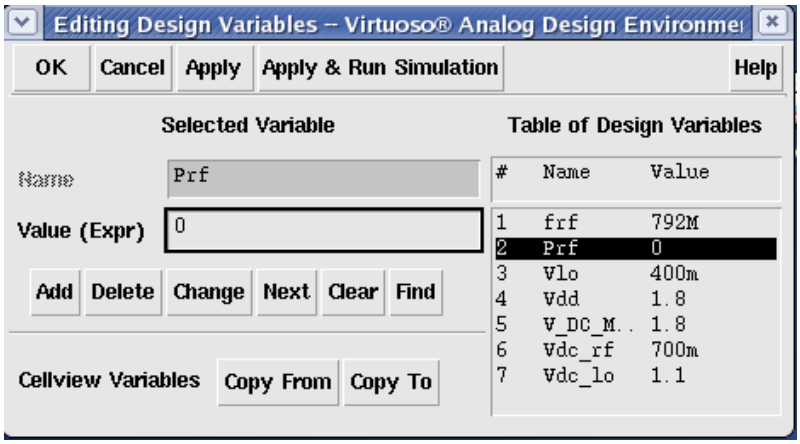


Figure 4.3: Edit Design Variables.

4. From the Analog Environment menu banner, we select **Analysis → Choose**, and the form of figure will appear. We choose "dc" analysis and "Save DC Operating Point". By choosing OK to the form we will see the DC analysis to appear in the Analog Environment, giving as also the capability to modify it in the future by "double cliking" on it. Finally we select "Netlist and Run" or  , to the Analog Environment.

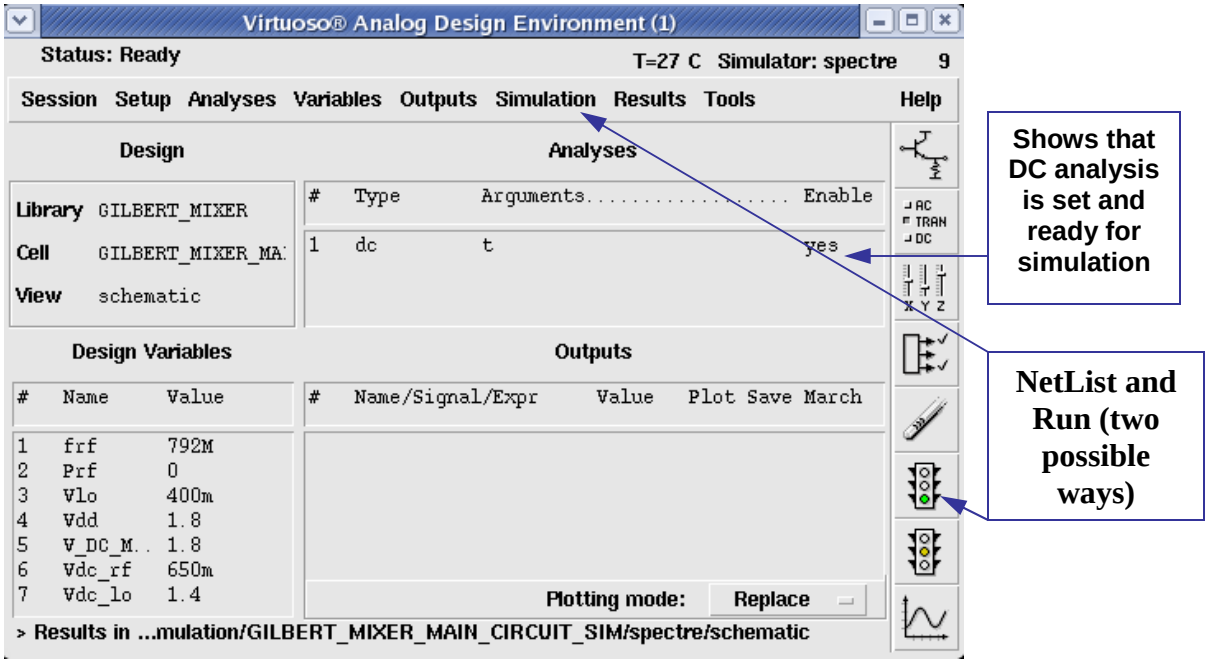


Figure 4.4: Analog Environment DC analysis settings.

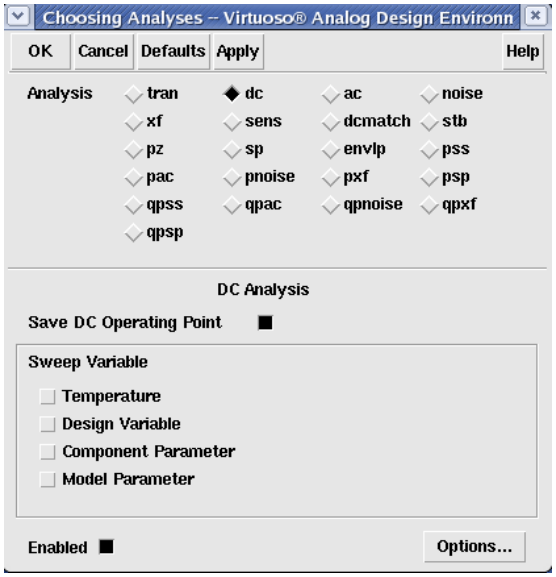


Figure 4.5: DC analysis setup.

5. After choosing “netlist and run” a simulation text window will pop-up that will show the progress as also the end of our simulation, like the one in figure . If simulation fails due to errors, the errors will be displayed here.

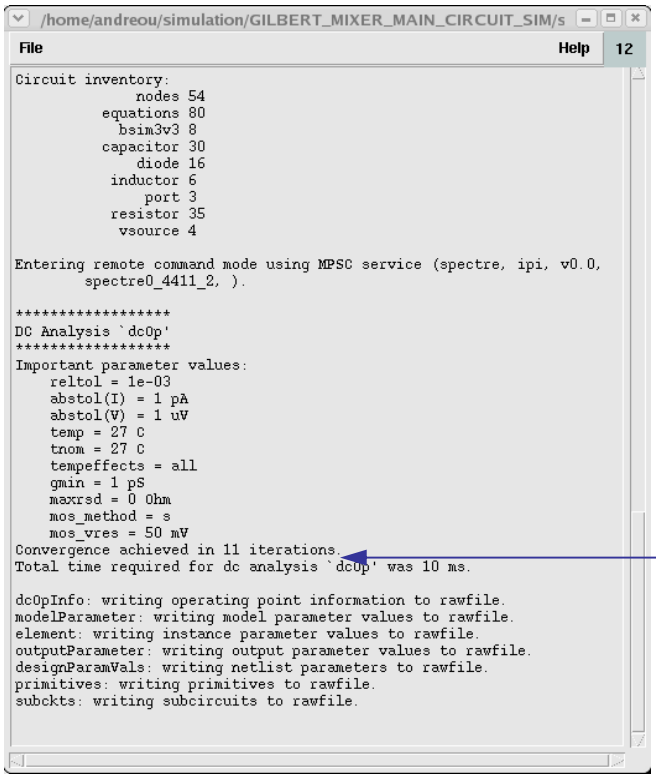


Figure 4.6: DC analysis progress text window.

6. When the DC analysis successfully completes, we bring the “GILBERT_MIXER_MAIN_CIRCUIT_SIM” schematic window to the foreground and descend the hierarchy to view one of the differential transistors in the mixer.

To descend into the hierarchy of the circuit, with the mouse, select the “GILBERT_MIXER_MAIN_CIRCUIT” instance and then we select

Design → Hierarchy → Descend Read or alternatively we use the bindkey “**e”** from the Composer menu banner. In the *Descend* form that appears, we keep the View Name as schematic and we select **OK** this form.

7. To view the operating point of the circuit, we can go to the Analog Environment window and select

Results → Annotate → DC Operating Points.

The operating points informations are displayed next to all devices.

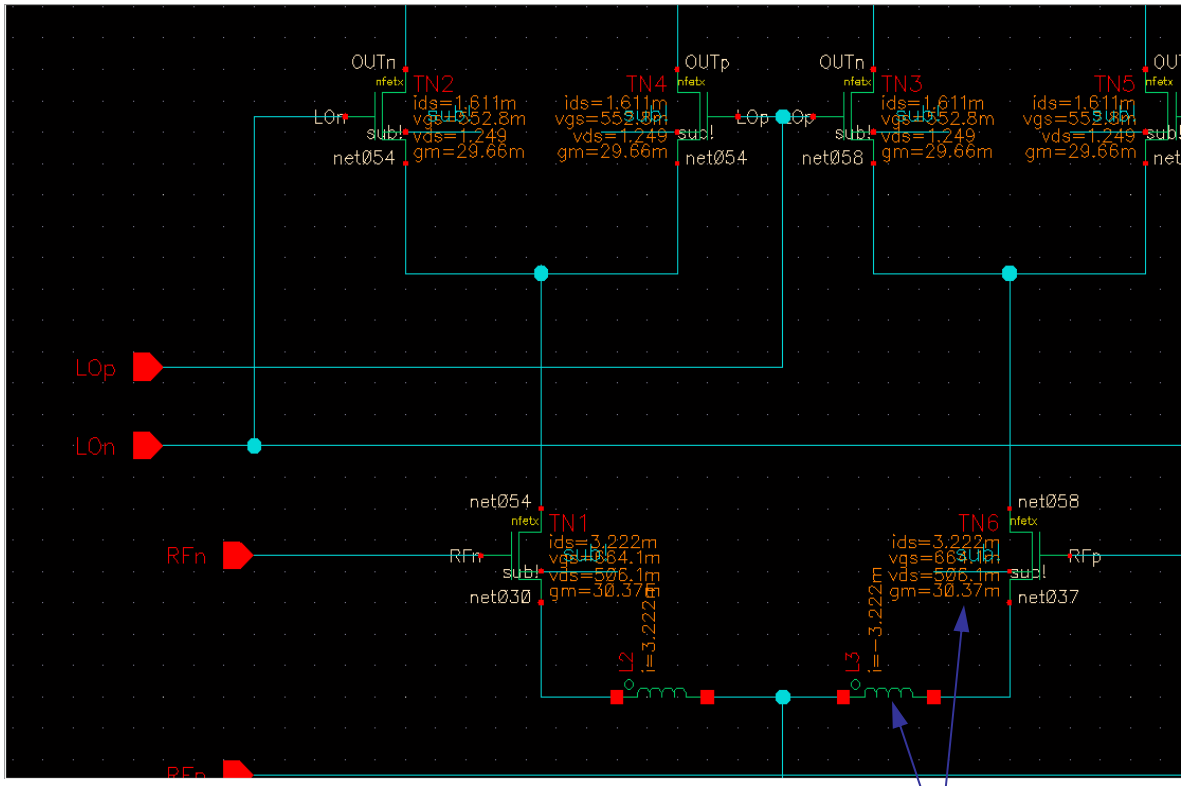


Figure 4.7: DC operating points after DC Analysis

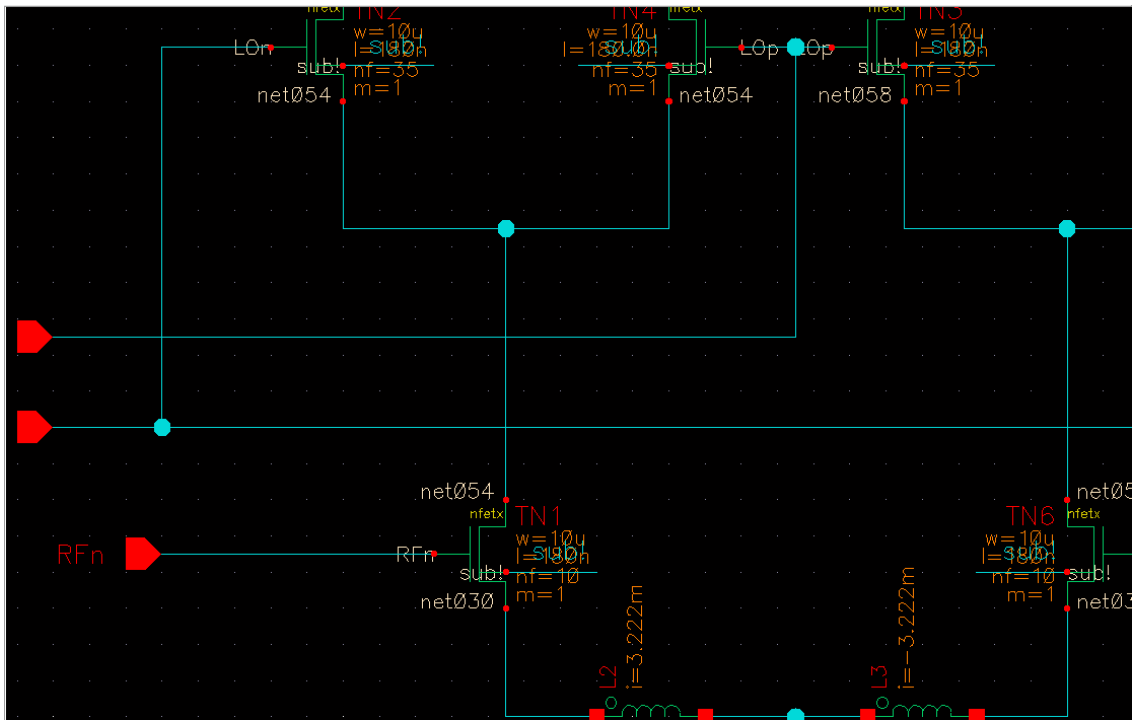


Figure 4.8: Model Parameters of the devices in the schematic window.

8. To return to the top of the hierarchy, we can select

Design → Hierarchy → Return to Top

in the Composer menu banner.

9. To disable the DC analysis "Double-click" the DC analysis line in the Analyses portion of the Analog Environment window and Toggle the Enabled button on the bottom of the form, and finally select OK the form.

4.2 Harmonic Distortion using PSS Analysis:

Setting Up the Simulation

1. First of all we must open the "GILBERT_MIXER_MAIN_CIRCUIT_SIM" mixer circuit.

2. In the Simulation window, we use "Analysis – Disable" to disable any analyses we ran previously.

4. If necessary, we set the design variables as it was explained before to the values shown in figure 4.9.

Design Variable	Value
frf	792MHz
Prf	-30db
Vlo	400mV
VDD	1.8V
V_DC_Mirror	1.8V
Vdc_rf	700mV
Vdc_lo	1.1V

Figure 4.9: Design variables values.

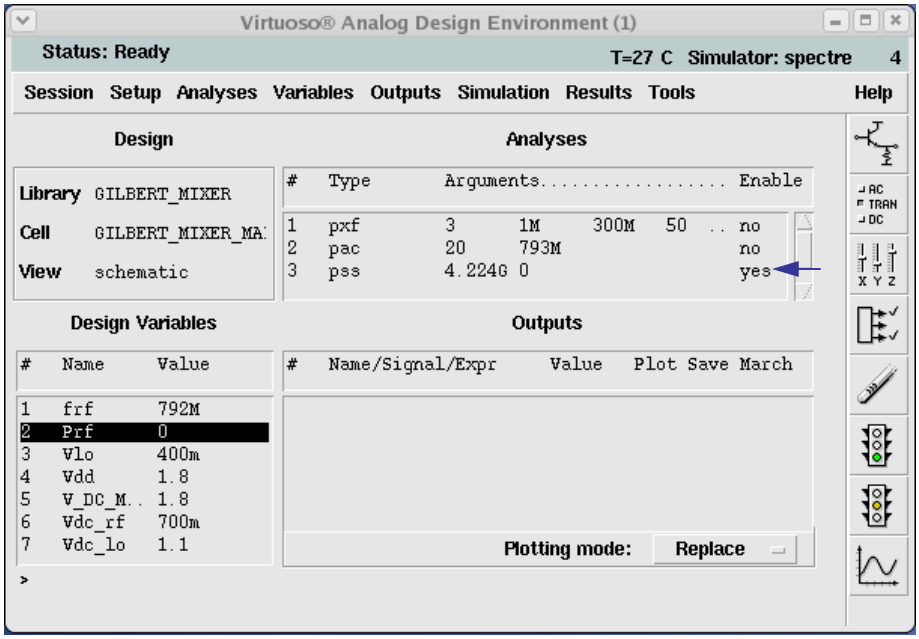


Figure 4.10: pss settings.

1. We choose “Edit - Properties – Objects” in the Schematic window.

The Edit Object Properties form appears. We use this form to change the list of CDF (component description format) properties for the *rf* voltage source and modify the schematic for this simulation.

2. In the Schematic window, we click on the *rf* voltage source.

The Edit Object Properties form changes to display information for the voltage source.

3. In the Edit Object Properties form, we must be sure that “sine” is specified as the Source type.

4. In the Edit Object Properties form, click *OK*.

5. In the Schematic window, choose “Design - Check and Save”.

Setting Up the PSS Analysis

1. We choose “Analyses – Choose” in the Simulation window.

The Choosing Analyses form appears.

2. In the Choosing Analyses form, we click on “pss”.

3. In the “*Fundamental Tones”* list box, *Beat Frequency* is highlighted by default. Be sure *“Auto Calculate”* is also highlighted.

The "*Beat Frequency*" is now displayed. The 264MHz value is the minimum period for which both RF at 792 MHz and the LO at 4.224 GHz are periodic (or for which RF and LO are integer multiples of the fundamental frequency).

792/264=3 4224/264=16

4. In the "Output harmonics" cyclic field, we choose "Number of harmonics" as 16 in the "number of harmonics" field. We can choose more than 16 but not less, as this entry expands the plotted frequency range to include the areas of interest around both 792 MHz and 4.224 GHz. (256 x 16 MHz equals 4.224 GHz, or 0 to 4.224 GHz, and includes all important frequencies).

5. Highlight "*moderate*" for the "*Accuracy Defaults*" *(errpreset)* setting.

6. We must verify that "*Enabled*" is highlighted

7. Click *OK.*

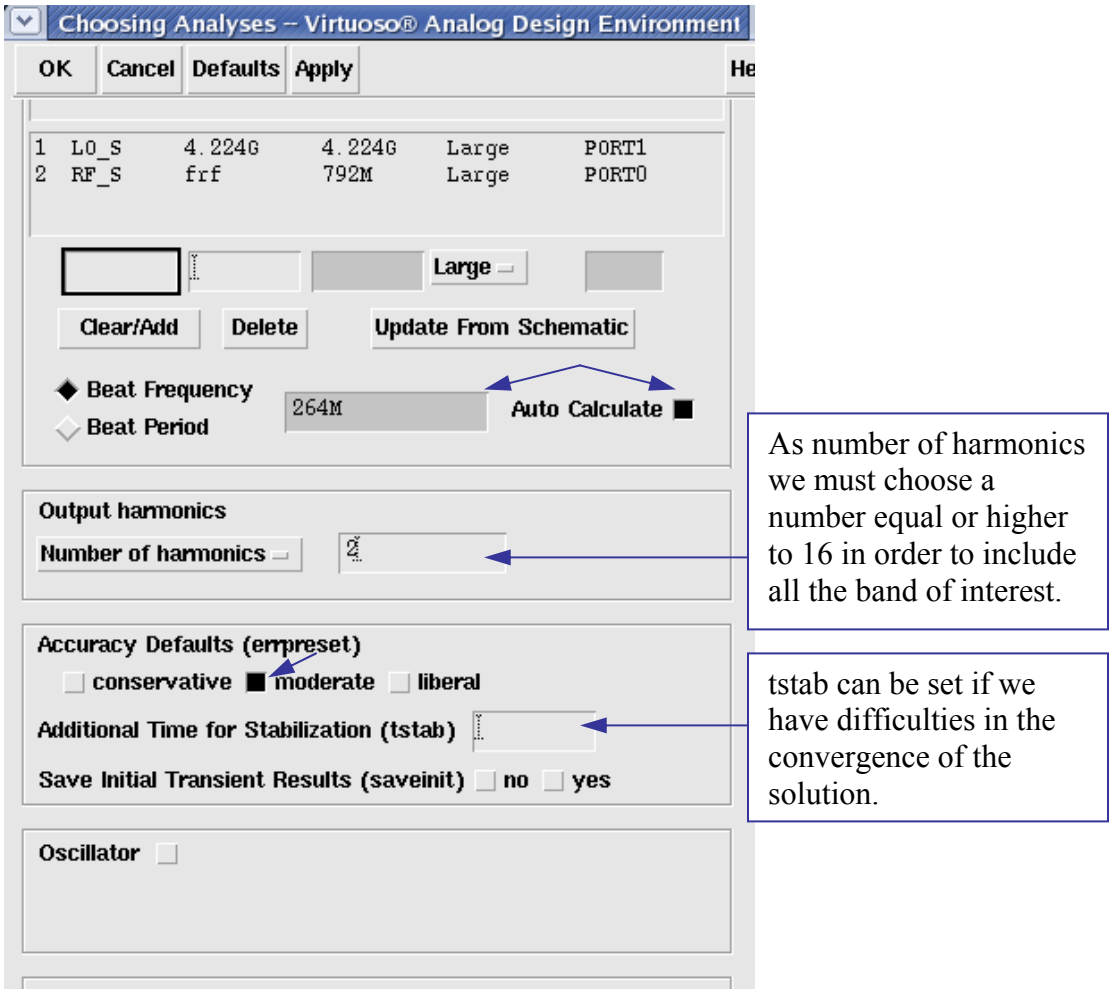


Figure 4.11: PSS analysis setup.

Running the Simulation

1. To run the simulation, we choose "*Simulation - Netlist and Run"* in the Simulation window. The output log file appears and displays information about the simulation as it runs as in figure 4..

2. We can also look in the CIW for a message that says the simulation completed successfully.

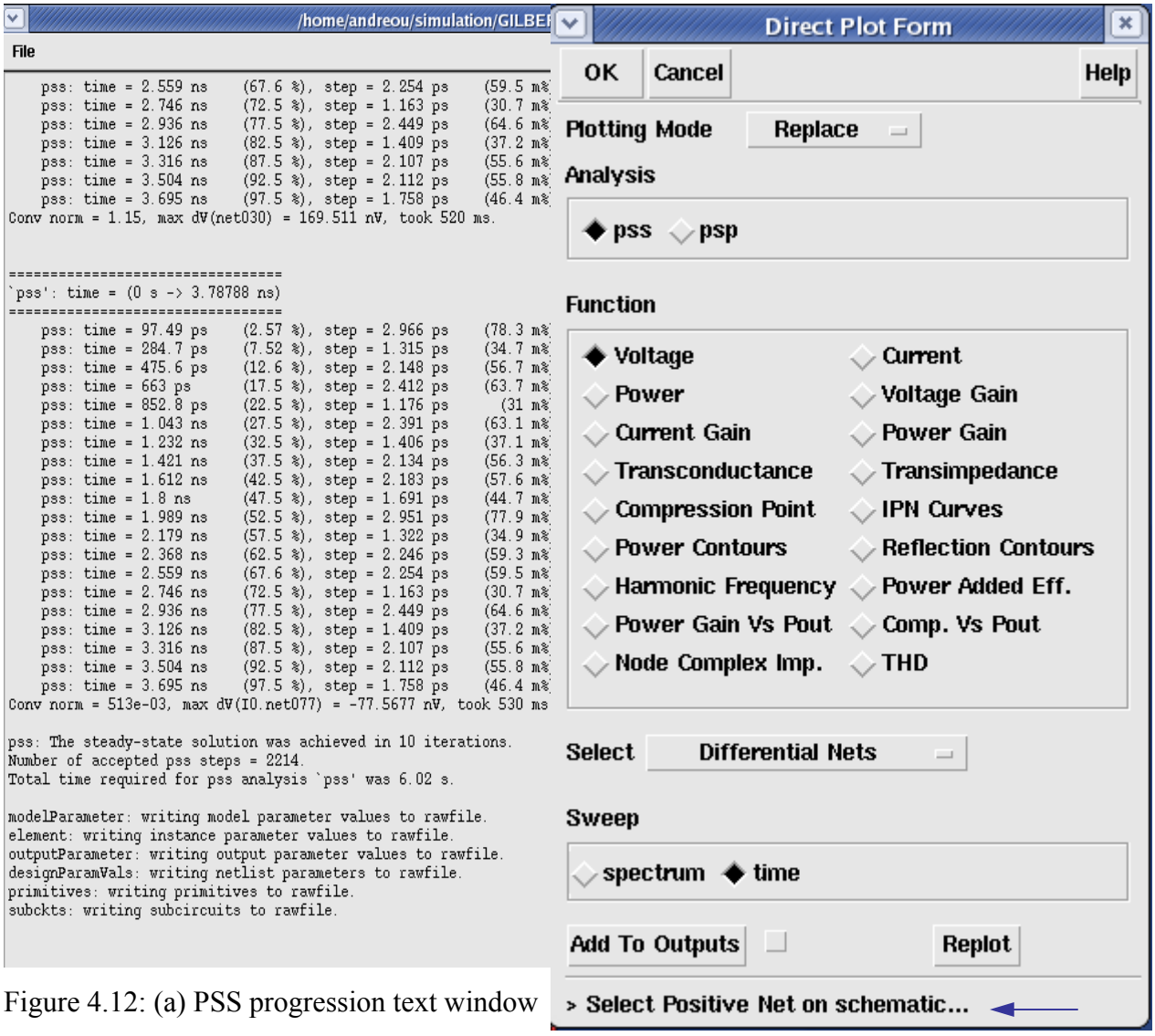


Figure 4.12: (a) PSS progression text window

(b) Direct plot form setup.

Plotting and Calculating Harmonic Distortion

We choose “Results - Direct Plot - Main Form” in the Simulation window.

The Direct Plot form appears.

In the Direct Plot form, do the following:

1. Highlight *Append* for *Plot Mode*.

2. Highlight *pss* for *Analysis*.

3. Highlight *Voltage* for *Function*.

4. Select *Differential Nets* in the *Select* cyclic field to plot voltage against frequency.

(The message at the bottom of the form changes and is a guide to the nets that we must select.)

5. Highlight *spectrum* for *Sweep*.

6. Highlight *peak* for *Signal Level*.

7. Highlight *dB20* for *Modifier*.

8. To plot the voltage against frequency, click on the *Pif* and *Prf* wires in the Schematic Window.

a. Following the instructions at the bottom of the Direct Plot Form we select the positive net, *if* , on the schematic.

→ *if* is highlighted in the schematic window and the instructions at the bottom of the Direct Plot form change.

b. Following the instructions at the bottom of the Direct Plot Form we select the negative net, *rf* , on the schematic.

→ *rf* is highlighted in the schematic window.

The Waveform Window display appears like the one below.

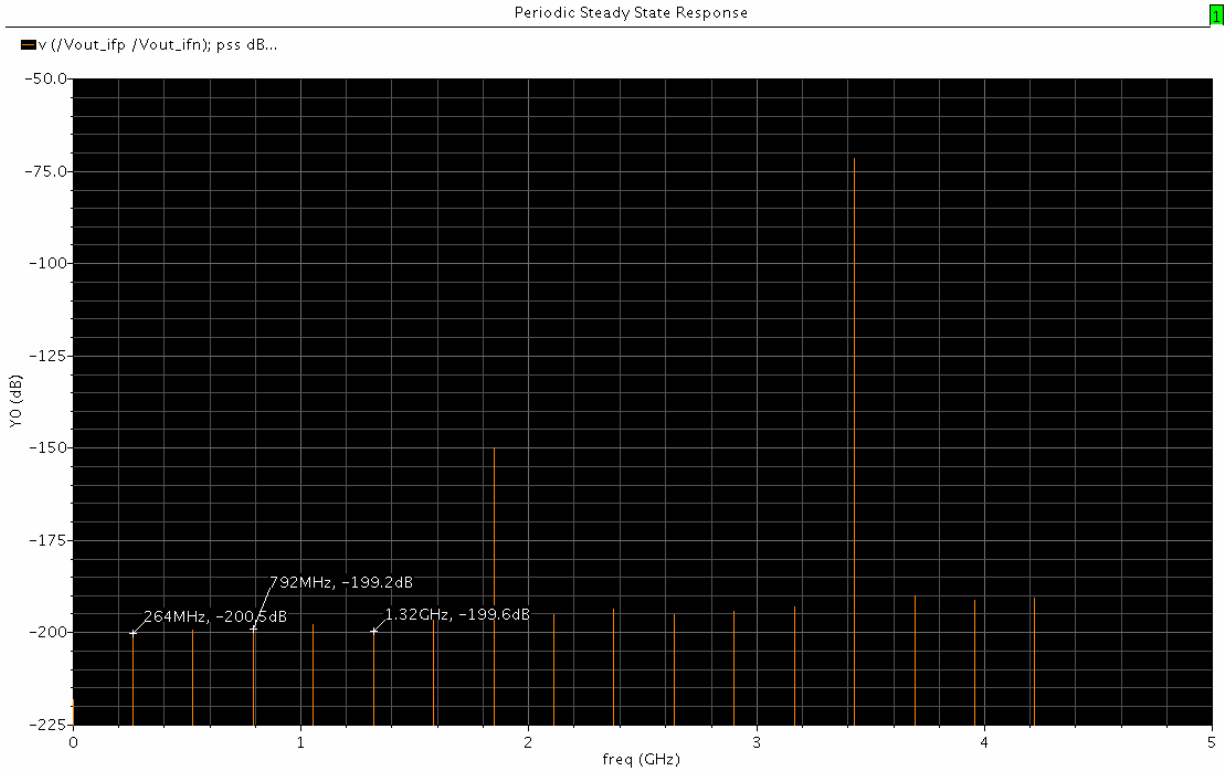


Figure 4.1: PSS differential nets Voltage Spectrum (rms,db20). The markers are on 4^{th}, 4^{th} and 6^{th} harmonic that are needed in order to calculate the harmonic distortion.

We can calculate the total harmonic distortion (THD) with the following formula:

$$THD=db10\left[(h4)^2+(h6)^2\right]\text{-}db20(h2)$$

where

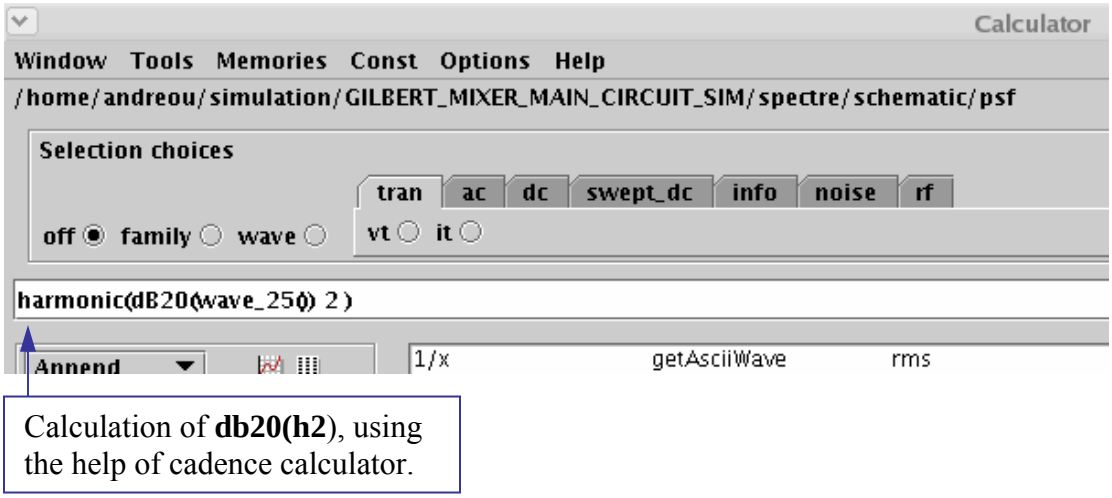
h2 is 264MHz (Vif is -200.6 dB)

h4 is 792 MHz (Vif is -199.2 dB)

h6 is 1.32 GHz (Vif is -199.6 dB)

⇨ THD = -45.98 - 46 + 46 => **THD = -45.98 db**

The calculations of the different parts of THD were done by the help of calculator of Cadence as shown below for db20(h2).



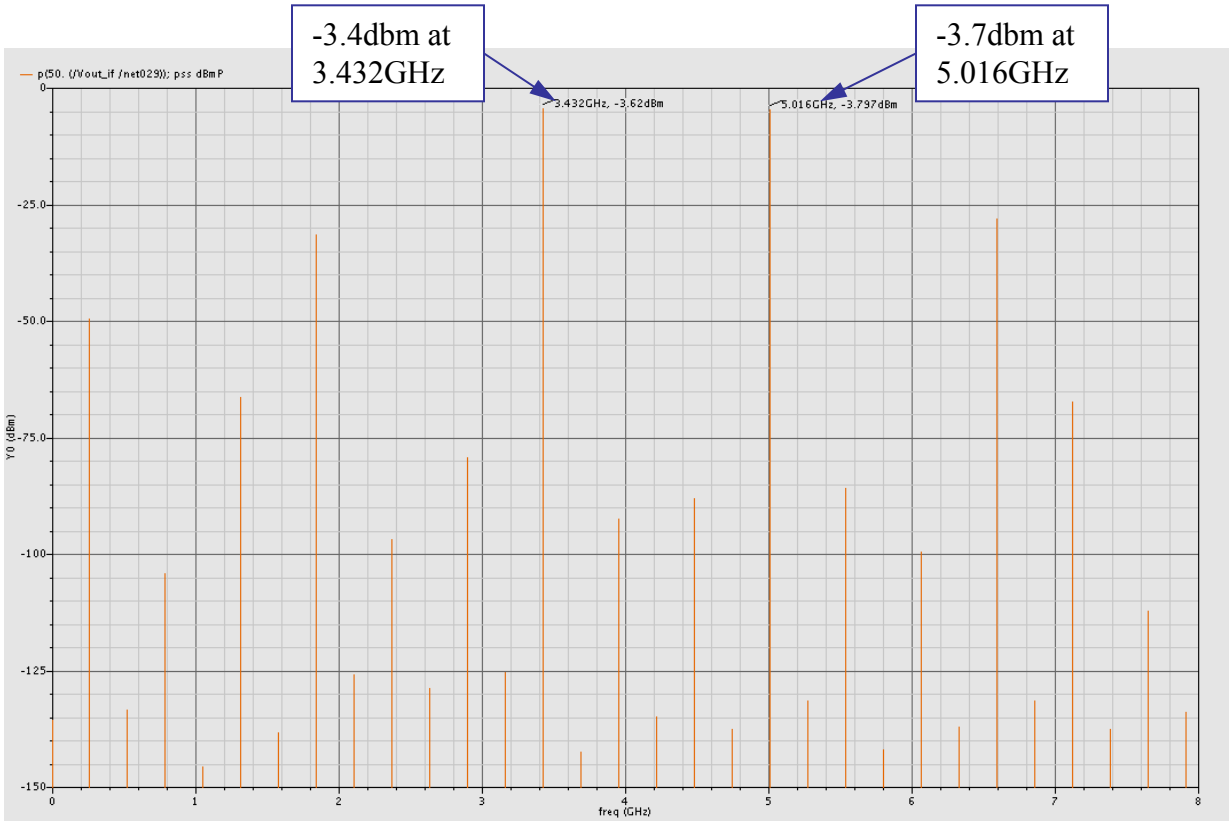


Figure 4.13:PSS Spectrum of IF_POWER (OUTPUT).

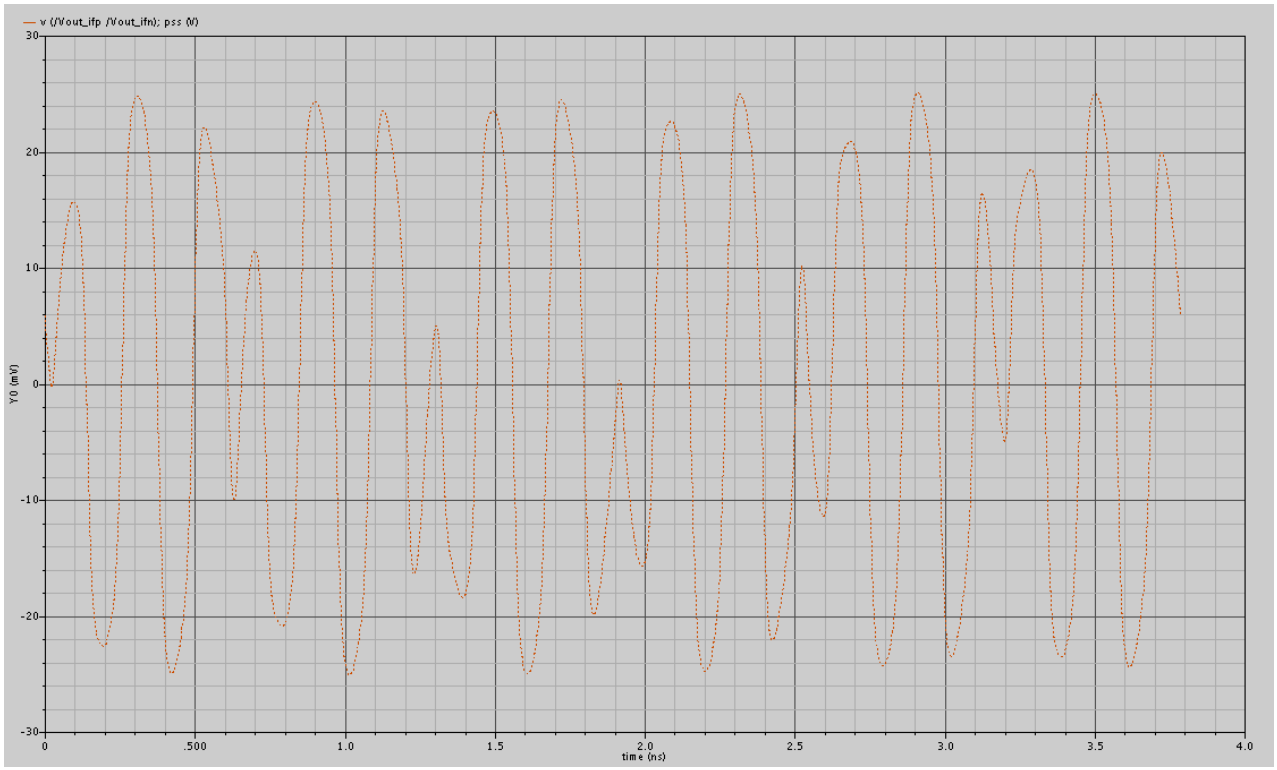


Figure 4.14: PSS IF OutputVoltage (DifferencialNets) with time sweep.

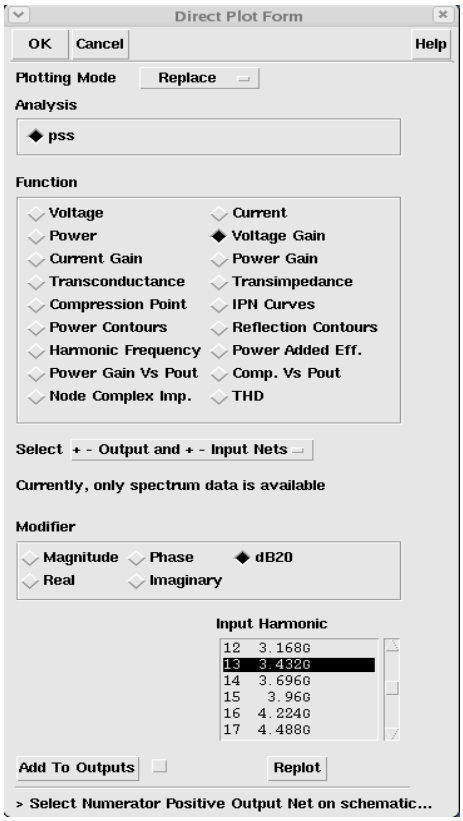


Figure 4.15 :PSS voltage Gain settings of Direct Plot form.

4.3 Noise Figure Measurement with PSS and Pnoise

Pnoise analysis calculates the total noise at the output of the circuit. The SpectreRF Pnoise analysis computes the single sideband noise figure. The total noise can vary with the number of harmonics we choose because each harmonic contributes a noise component.

Setting Up the Simulation

1. In the Simulation window, we use *Analysis - Disable* to disable any analyses we ran previously. (Check the Simulation window to verify whether or not an analysis is enabled.)

Editing the Schematic

Modify the schematic to be sure the PSS analysis is the response of the mixer to *only* the LO signal.

1. We choose *Edit - Properties - Objects* in the Schematic window.

The Edit Object Properties form appears. We use this form to change the list of CDF properties and modify the schematic for this simulation.

2. In the Schematic Window, click on the *rf* voltage source.

3. In the Edit Object Properties form, we choose *dc* as the *Source type*, if necessary, and click *OK*.

4. In the Schematic window, we choose *Design - Check and Save.*

Setting Up the PSS Analysis

1. In the Choosing Analyses form, click on *pss* for the *Analysis*.

2. At the lower right corner of the Fundamental Tones section, we highlight *Auto Calculate*. The *Beat Frequency* is now displayed as 4.224GHz.

3. In the *Output harmonics* cyclic field, we choose *Number of harmonics* and type 0 in the field.

Pnoise requires that we set the number of harmonics value to 0 to determine the circuit's response to LO only.

4. We highlight *moderate* for the *Accuracy Defaults (errpreset)* setting.

5. Highlight *Enabled*.

Setting Up the Pnoise Analysis

1. At the top of the Choosing Analyses form, we highlight *pnoise.*

The Choosing Analyses form changes to let us specify data for a *Pnoise* analysis.

2. In the *Frequency Sweep Range (Hz*) cyclic field, choose *Start-Stop*.

3. Type 2048MHz in the *Start field* and 4096MHz in the *Stop* field.

This frequency range covers the frequencies of interest.

4. In the *Sweep Type* cyclic field, we choose *Logarithmic* for the sweep type, then highlight *Points Per Decade* and we type 10 in the *Points Per Decade* field.

5. In the *Sidebands* cyclic field, we choose *Maximum sideband* and type 30 for the number of sidebands.

With this setting, we specify that 30 sidebands contribute noise to the output.

6. In the *Output* cyclic field, we choose *voltage* for the *Output* value.

7. Highlight the *Positive Output Node Select* button. Then click on the appropriate wire in the Schematic window to choose *Pif* .

→ /PORT2 appears in the *Positive Output Node* field.

8. We repeat step (7) with the negative node.

9. In the *Input Source* cyclic field, we choose *voltage*.

10. Click on the *Input Voltage Source Select* button. Then click on the appropriate component in the Schematic window to choose *rf* .

/PORT0 appears in the *Input Voltage Source* field.

11. In the *Reference Side-band* cyclic field, we choose *Enter in field*.

12. We type -1 in the *Reference Sideband* field.

In this field, we specify the difference between the input and output frequencies in the whole *frf* . The *Reference Sideband* must be -1 because this is a down converter. Other possible choices are 0 and +1.

The Pnoise Choosing Analyses form looks as in figure 4..

13. Highlight *Enabled* and click *OK*.

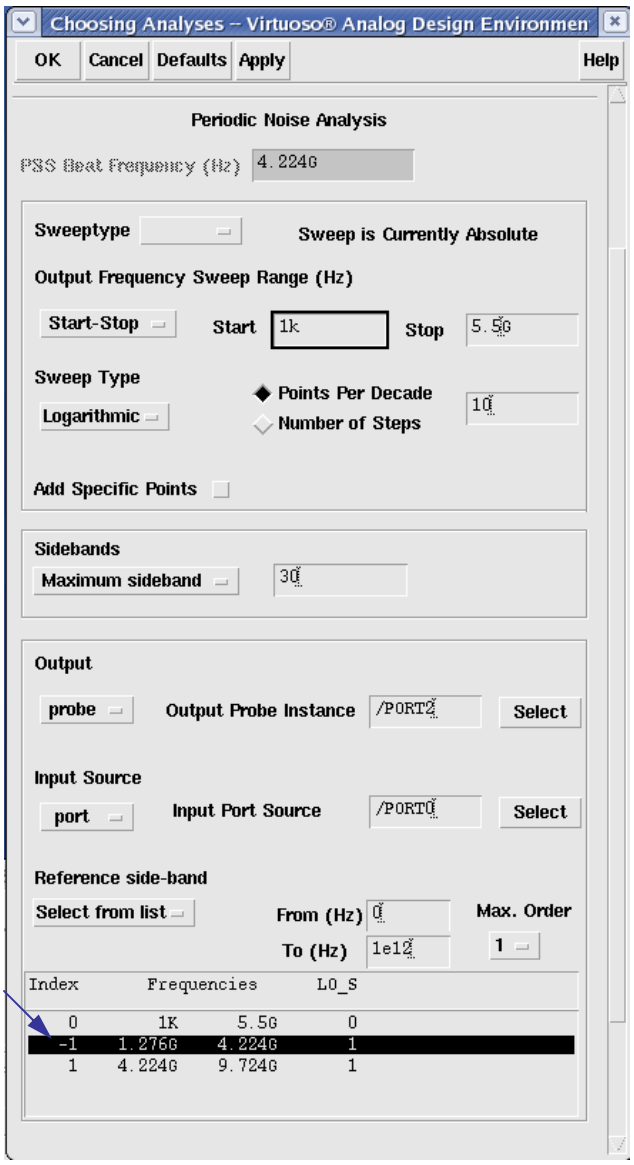


Figure 4.17(a): Pnoise + Periodic Noise Analysis settings (sweep from 1kHz up to 5.5GHz) .

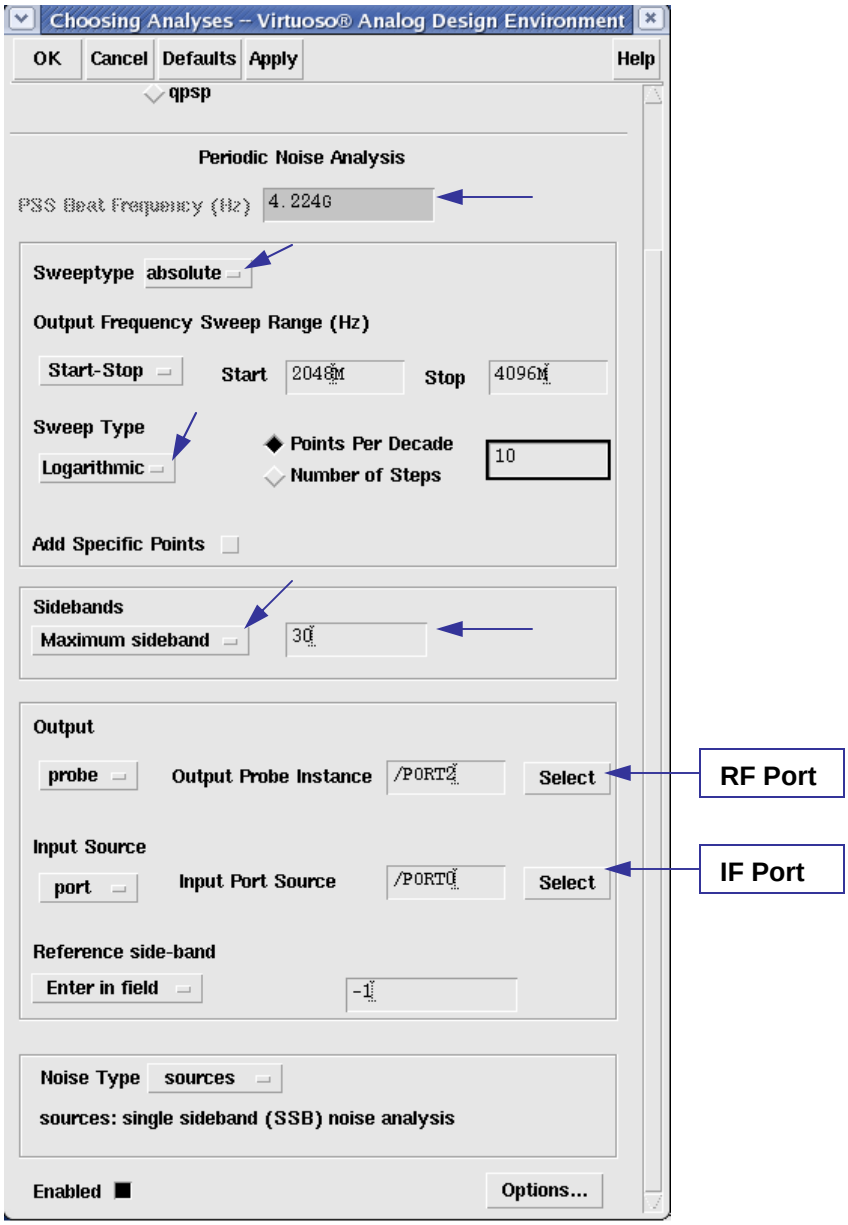


Figure 4.17(b): Pnoise + Periodic Noise Analysis settings (sweep from 2048MHz up to 4096MHz) .

Running the Simulation

1. To run the simulation, we choose *Simulation - Netlist and Run* in the Simulation window.

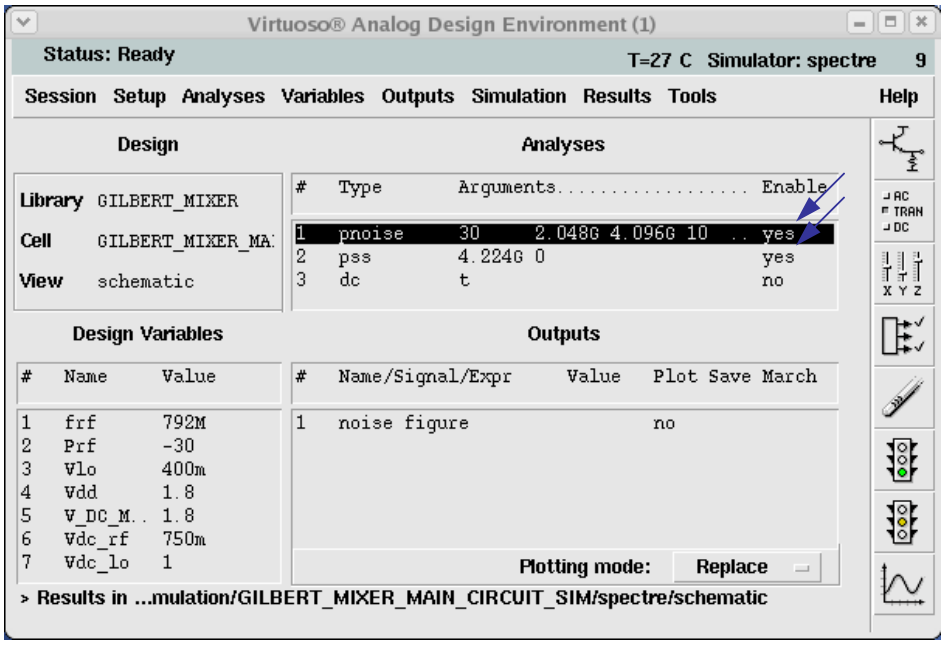


Figure 4.18: PSS and PNOISE setup for the measurement of Noise Figure.

2.The output log file appears and displays information about the simulation as it runs.

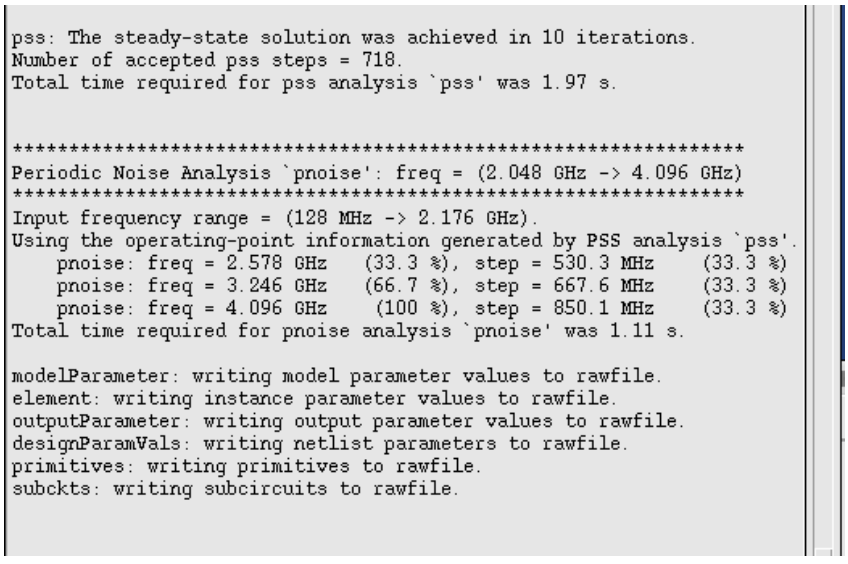


Figure 4.19: Output log file after the simulation of pss and pnoise.

3. We can look in the CIW for a message that says the simulation completed successfully.

Plotting the Noise Figure

To open the Direct Plot form, we choose *Results - Direct Plot - Main Form* in the Simulation window.

In the Direct Plot form, we do the following:

1. Highlight *Replace* for *Plot Mode*.
2. Highlight *pnoise* for *Analysis*.
3. Highlight *Noise Figure* for *Function*.
4. Follow the prompt at the bottom of the form

Press plot button on this form...

Click on *Plot* in the Direct Plot form.

The plot appears in the Waveform window.

The Waveform window displays the noise figure:

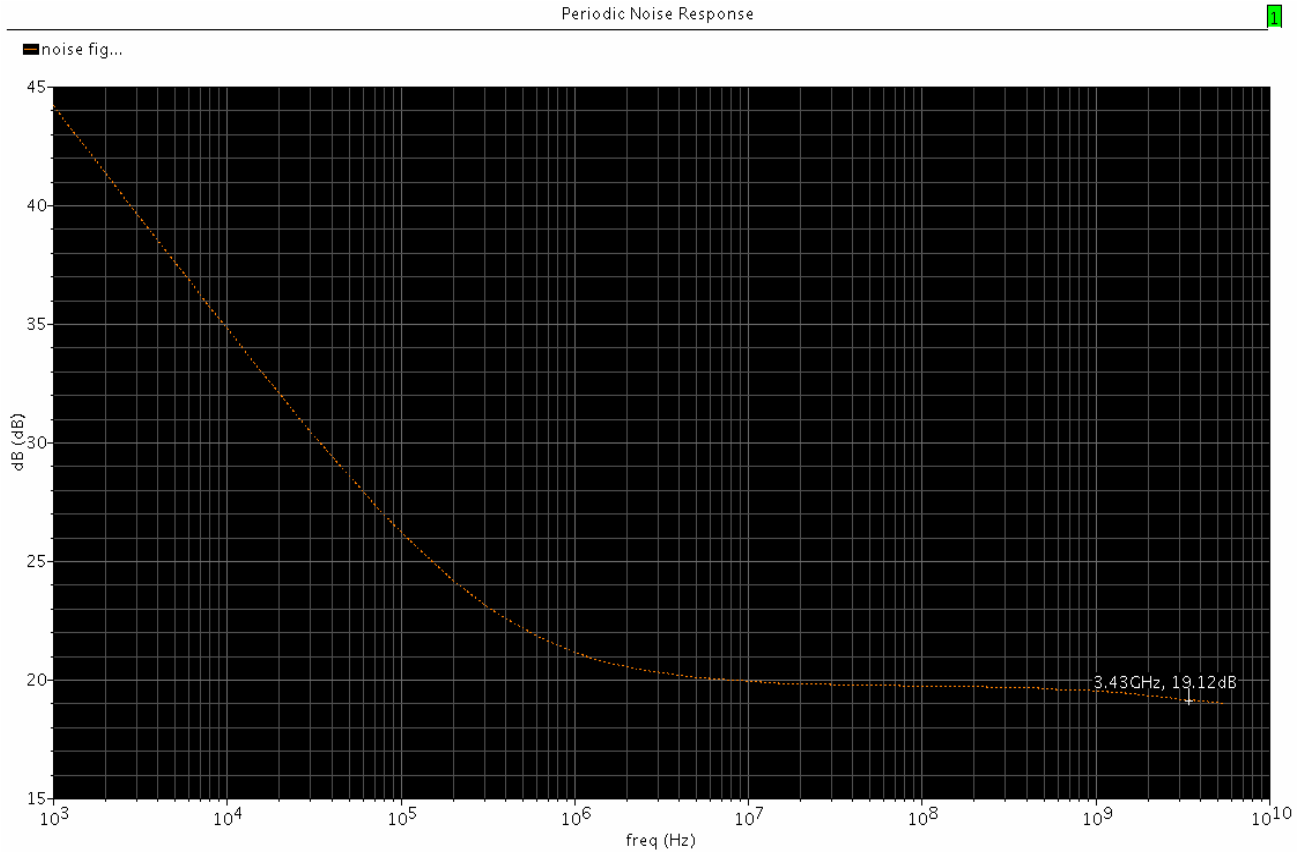


Figure 4.20: SSB Noise Figure using periodic noise response and PNoise (1k - 5.5G & 30 sidebands) at 3.432GHz is 19.12db.

This is the single-sideband noise figure (SSB NF) of the mixer, due to the fact that the desired signal spectrum resides on only one side of the LO frequency. Typical noise figure meters measure the DSB NF and predict the SSB value by simply adding 3 dB.
So the double-side band Noise Figure (DSB) here will be:

NF(DSB) = NF (SSB) – 3dB = 19.12db - 3db = 16.12db

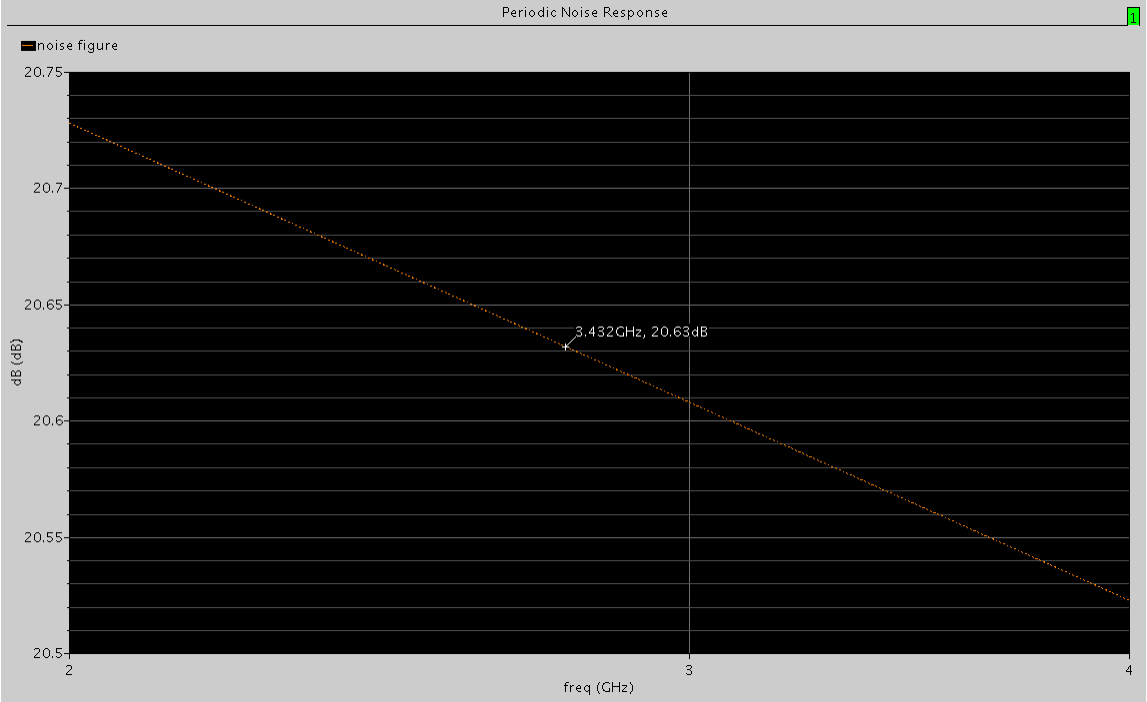


Figure 4.21: SSB Noise Figure using periodic noise response and pnoise (2048M-4096M & 30 sidebands)

4.4 Conversion Gain Measurement with PSS and PXF

We can combine a PSS analysis with a periodic small-signal transfer function PXF analysis to determine the conversion gain of the down converter mixer.

Setting Up the Simulation

In the Simulation window, we use *Analysis - Disable* to disable any analyses we ran previously.

Editing the Schematic

We must modify the schematic to set the RF source to a DC source to be sure the PSS analysis is the response of the mixer to only the LO signal.

1. Choose *Edit - Properties - Objects* in the Schematic window.

The Edit Object Properties form appears. We use this form to change the list of CDF properties and modify the schematic for this simulation.

2. In the Schematic window, we click on the *rf* voltage source.
3. In the Edit Object Properties form, we choose *dc* as the *Source type*, and click *OK*.
4. In the Schematic window, we choose *Design - Check and Save*.

Setting Up the PSS and PXF Analyses

1. In the Simulation window, we choose *Analyses - Choose*.

The Choosing Analyses form appears.

Setting Up the PSS Analysis

1. In the Choosing Analyses form, we highlight *pss* for the *Analysis*.
2. In the *Fundamental Tones* section, the *Auto Calculate* button must be highlighted.

The *Beat Frequency* is now displayed as 4.224GHz. LO is the only time-varying signal in the circuit, so it becomes the fundamental frequency. The *Beat Frequency* button is highlighted by default.

3. In the *Output Harmonics* cyclic field, we choose *Number of Harmonics* and type 0 in the *Number of Harmonics* field.

4. Highlight *conservative* for the *Accuracy Defaults (errpreset)* setting.

5. Highlight *Enabled*.

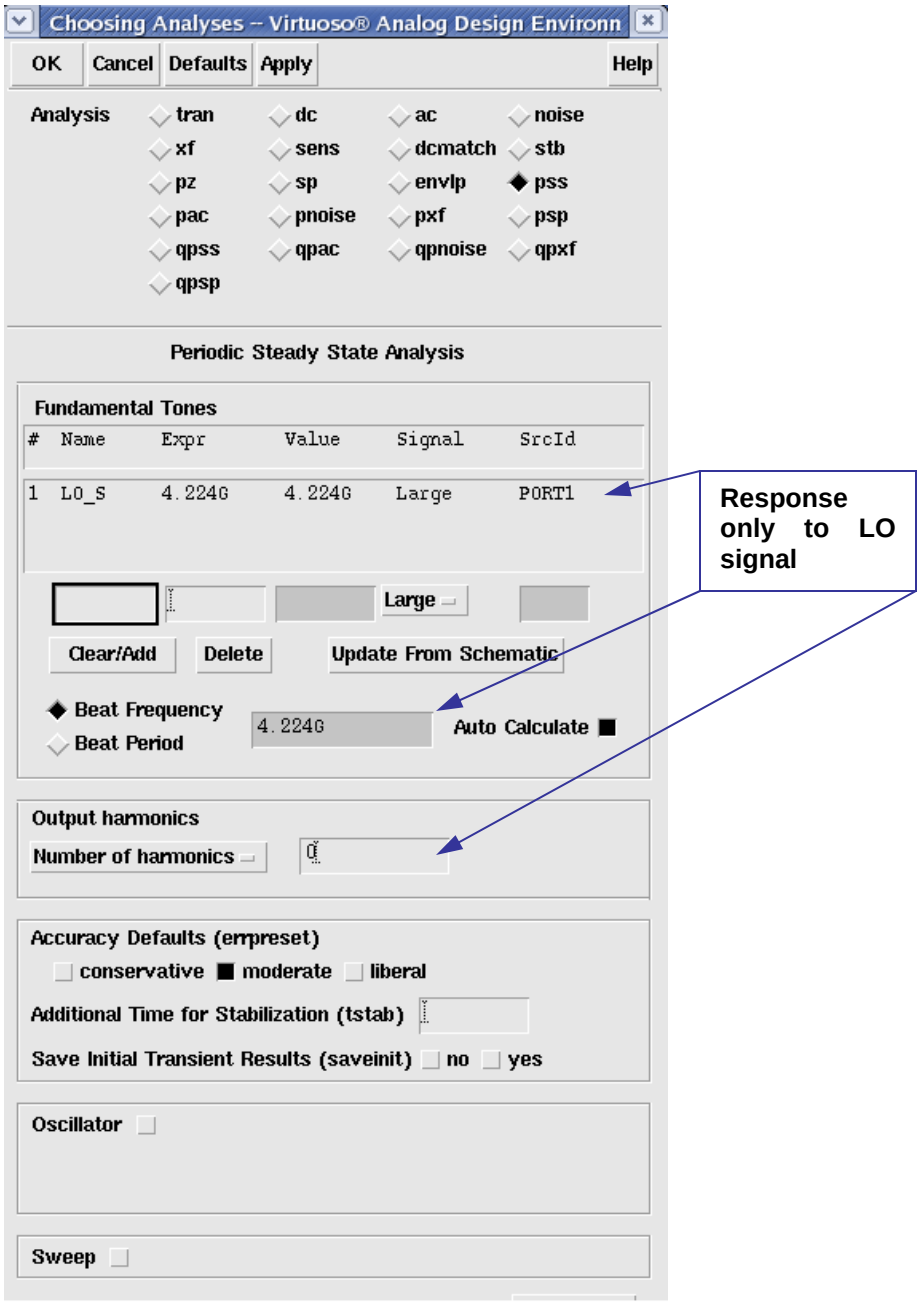


Figure 4.22: PSS setup for Conversion Gain measurement.

Setting Up the PXF Analysis

1. At the top of the Choosing Analyses form, we click on *pxf* .

The Choosing Analyses form changes to let us specify data for the PXF analysis.

2. In the *Frequency Sweep Range (Hz)* cyclic field, we choose *Start - Stop*, and type 1M in the *Start* field and 300M in the *Stop* field.

This range specification leaves space between the plots, but still shows the dramatic gain changes in the first 100 Mhz of each plot.

3. In the *Sweep Type* cyclic field, we choose *Linear* and highlight *Number of Steps*. We type 50 for the *Number of Steps*. Larger numbers of total points increase the resolution of the plot but also require a longer simulation time.

4. In the *Sidebands* cyclic field, we choose *Maximum sideband* and type 3 as the value.

5. In the *Output* section, we must highlight *voltage*.

6. Then we must click on the *Positive Output Node Select* button. Then click on the appropriate wire in the Schematic window to choose *Pif* .

/Vout_ifp appears in the *Positive Output Node* field.

7. Repeat step (6) for the N*egative Node*

/Vout_ifn appears in the N*egative Node Node* field.

8. Highlight *Enabled*.

The bottom of the PXF Choosing Analyses form looks like this.

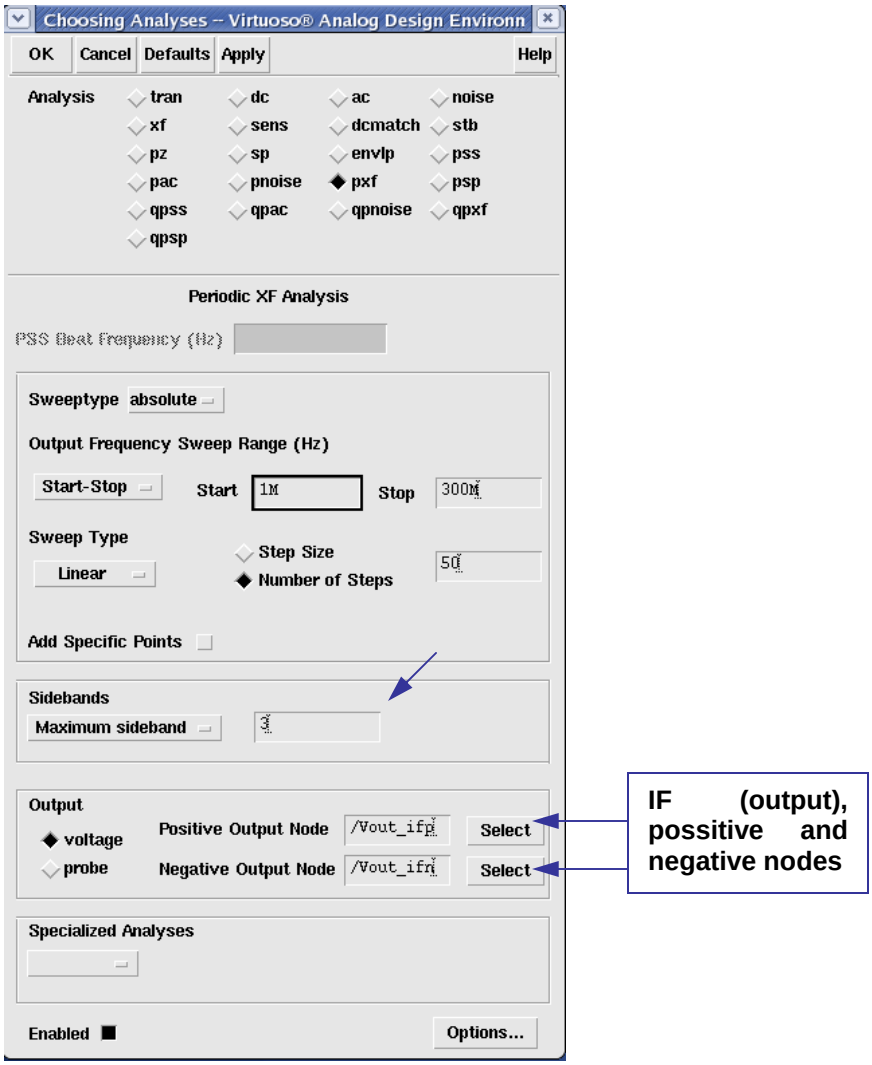


Figure 4.23: PXF setup for Conversion Gain measurement.

Running the Simulation

To run the simulation, we choose *Simulation - Netlist and Run* in the Simulation window.

The output log file appears and displays information about the simulation while it runs.

Plotting the Conversion Gain

1. We choose *Results - Direct Plot - Main Form* in the Simulation window.

The Direct Plot form appears.

2. Highlight *Replace* for *Plot Mode*.

3. Highlight *pxf* for *Analysis*.

4. Highlight *Voltage Gain* for *Function*.

5. Highlight *dB20* for *Modifier*.

The completed PXF Direct Plot form looks like this.

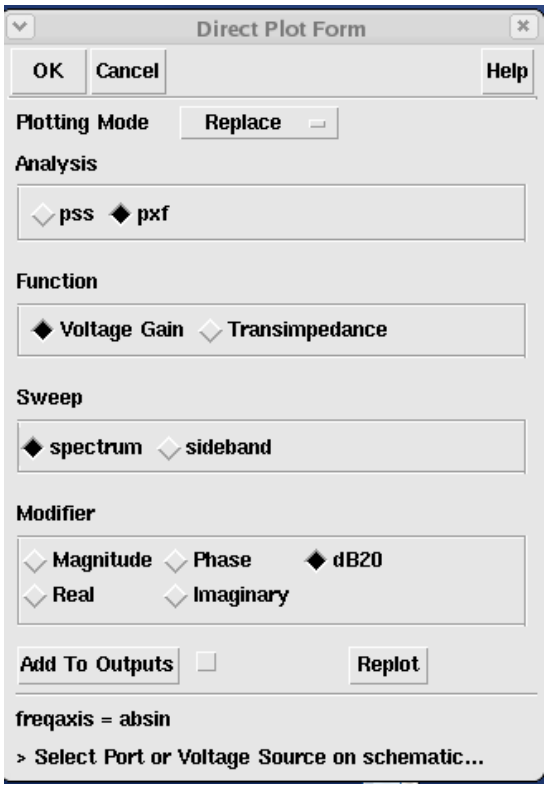


Figure 4.24: Direct Plot Form setup for Conversion Gain.

6. We follow the prompt at the bottom of the form

Select Port or Voltage Source on schematic...

Click on the Vout_ifp source on the schematic.

The plot appears in the Waveform window.

7. To determine the gain at different frequencies, we can move the cursor along the curve and use the readout at the top of the Waveform window.

The Conversion Gain was 3.61db at 3.432GHz.

4.5 Calculating the 1 dB Compression Point with Swept PSS

Setting Up the Simulation

In the Simulation window, we use *Analysis* - *Disable* to disable any analyses we ran previously.

Editing the Schematic

1. In the Schematic window, we click on the *rf* voltage source.
2. Choose *Edit* - *Properties* - *Objects* in the Schematic window.

The Edit Object Properties form appears. We use this form to change the list of CDF properties and modify the schematic for this simulation.

3. We choose *sine* for *Source type*.
4. We type prf in the *Amplitude 1 (dBm)* field because we want to sweep this variable.
5. Click *OK*.
6. In the Schematic window, we choose *Design* - *Check and Save*.

Setting Up the Swept PSS Analysis

1. In the Simulation window, we choose *Analyses* - *Choose*.

The Choosing Analyses form appears.

2. In the Choosing Analyses form, we choose *pss* for the *Analysis*.
3. In the *Fundamental Tones* list box, the *Auto Calculate* button must be highlighted.

The *Beat Frequency* is now displayed as 264M. Now that the RF signal is active again, the fundamental frequency of the circuit goes back to 264 MHz. The *Beat Frequency* button is highlighted by default.

4. In the *Output harmonics* cyclic field, we choose *Number of harmonics* and type 2 in the *Number of harmonics* field.

Only two harmonics are required to determine the 1 dB compression point.

5. Then we highlight *moderate* for the *Accuracy Defaults (errpreset)* setting.
6. Highlight the *Sweep* button.

The form changes to let us specify data for the sweep.

7. In the *Sweep* cyclic field, we choose *Variable*.

8. We click on the *Select Design Variable* button.

The Select Design Variable form appears.

9. In the Select Design Variable form, we highlight *prf* and click *OK*.

The *Variable Name* prf appears in the Choosing Analyses form. Selecting the prf variable sweeps RF input.

10. We choose *Start-Stop* for the *Sweep Range*, and then type -30 in the *Start* field and 10 in the *Stop* field. Both the signal source and the sweep are done in dBm.

11. We choose *Linear* for the *Sweep Type*, and specify 10 for the *Number of Steps*.

12. Highlight *Enabled*.

13. In the Choosing Analyses form, we click *OK*.

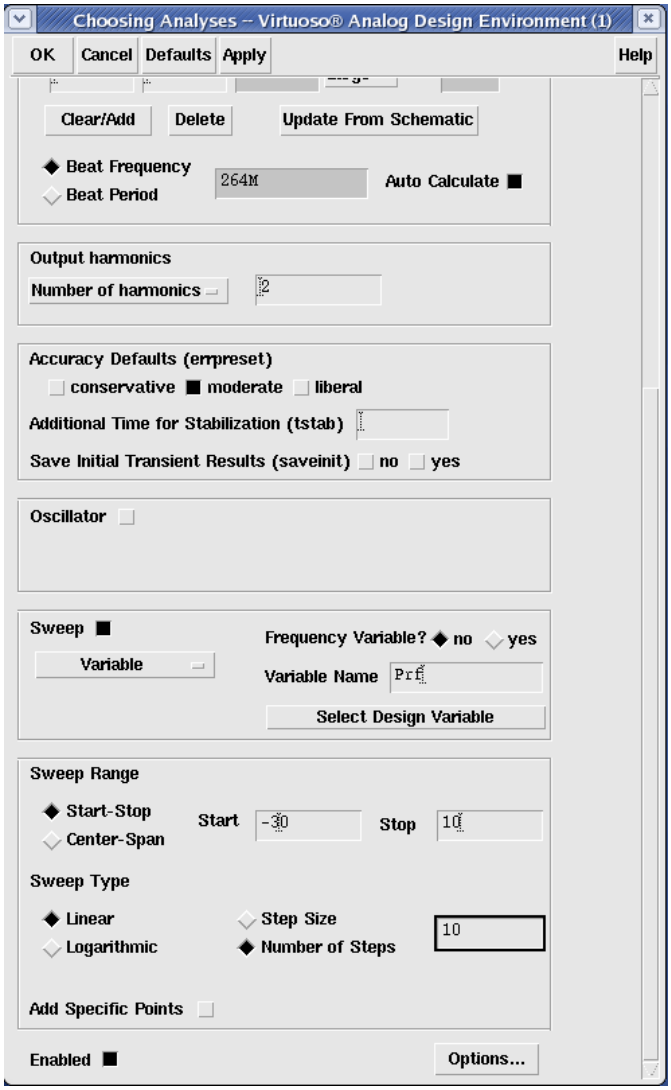


Figure 4.25: PSS setup for 1 db Compression Point.

Running the Simulation

1. To run the simulation, we choose *Simulation - Netlist and Run* in the Simulation window.

The output log file appears and displays information about the simulation as it runs.

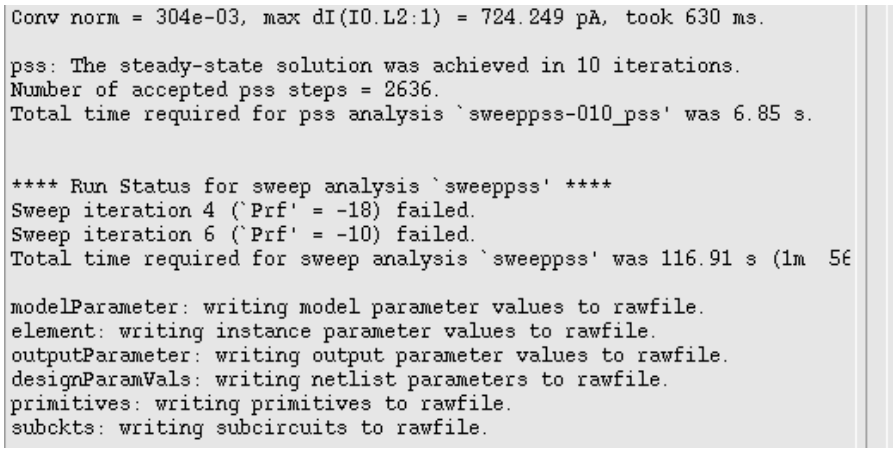


Figure 4.26: Simulation output log file for for 1 db Compression Point.

2. We look in the CIW for a message that says the simulation completed successfully.

Plotting the 1 dB Compression Point

The first sideband of the IF signal is the second harmonic of the 256 MHz fundamental.

To plot the 1 dB compression point,we perform the following:

1. We choose *Results - Direct Plot - Main Form* in the Simulation window.

The Direct Plot form appears.

2. Highlight *Replace* for *Plot Mode*.

3. Highlight *pss* for *Analysis*.

4. Highlight *Compression Point* for *Function*.

5. Type 1 for *Gain Compression (dB)*.

6. Type -20 (-30 until 10) for *Input Power Extrapolation Point (dBm)*.

This value specifies the point where the ideal amplification curve intersects the output curve. If we do not specify a value, the plot defaults to the minimum variable value.

7. In the cyclic field, we choose *Input Referred 1dB Compression*.

8. We follow the prompt at the bottom of the form

Select 1st Order Harmonic on this form...

In the *Harmonic* list box, we highlight harmonic *2 (528 MHz)*.

This is the second harmonic of the 256 MHz fundamental frequency, which is the IF frequency.

The completed Direct Plot form looks like this.

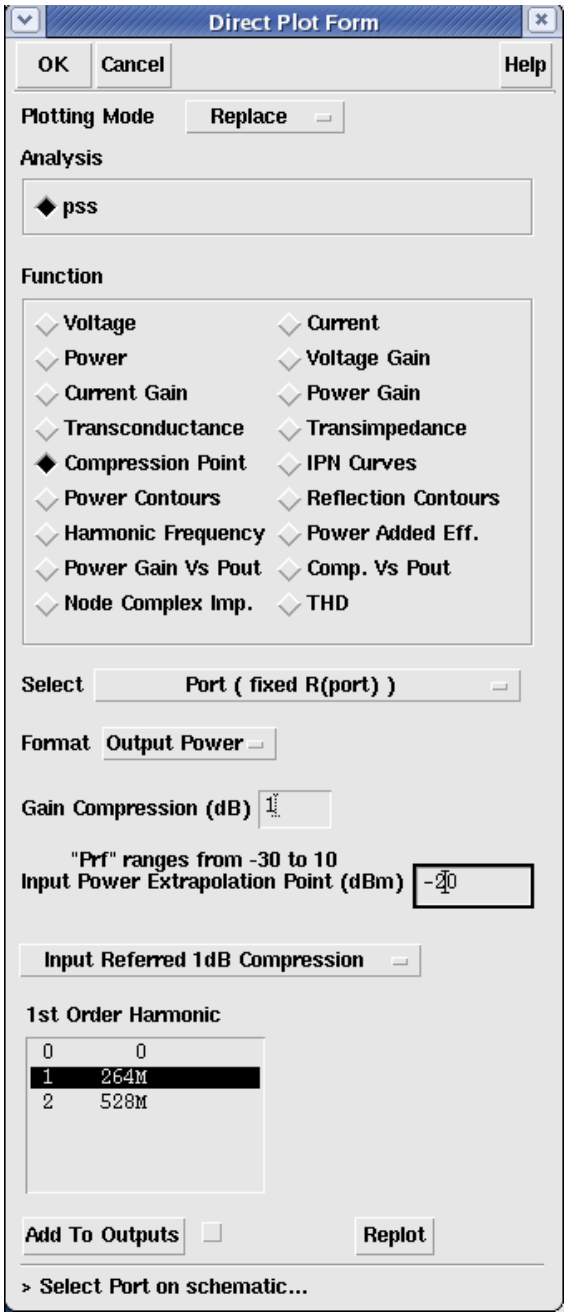


Figure 4.27: PSS Direct Plot Form setup for 1 db Compression Point.

9. We follow the new prompt at the bottom of the form

Select Port on schematic...

In the Schematic window, we click on the *if* node.

The Waveform window display looks like below for different "input Prf" (-30db until +10db) and different 1st order frequency (264MHz and 528MHz):

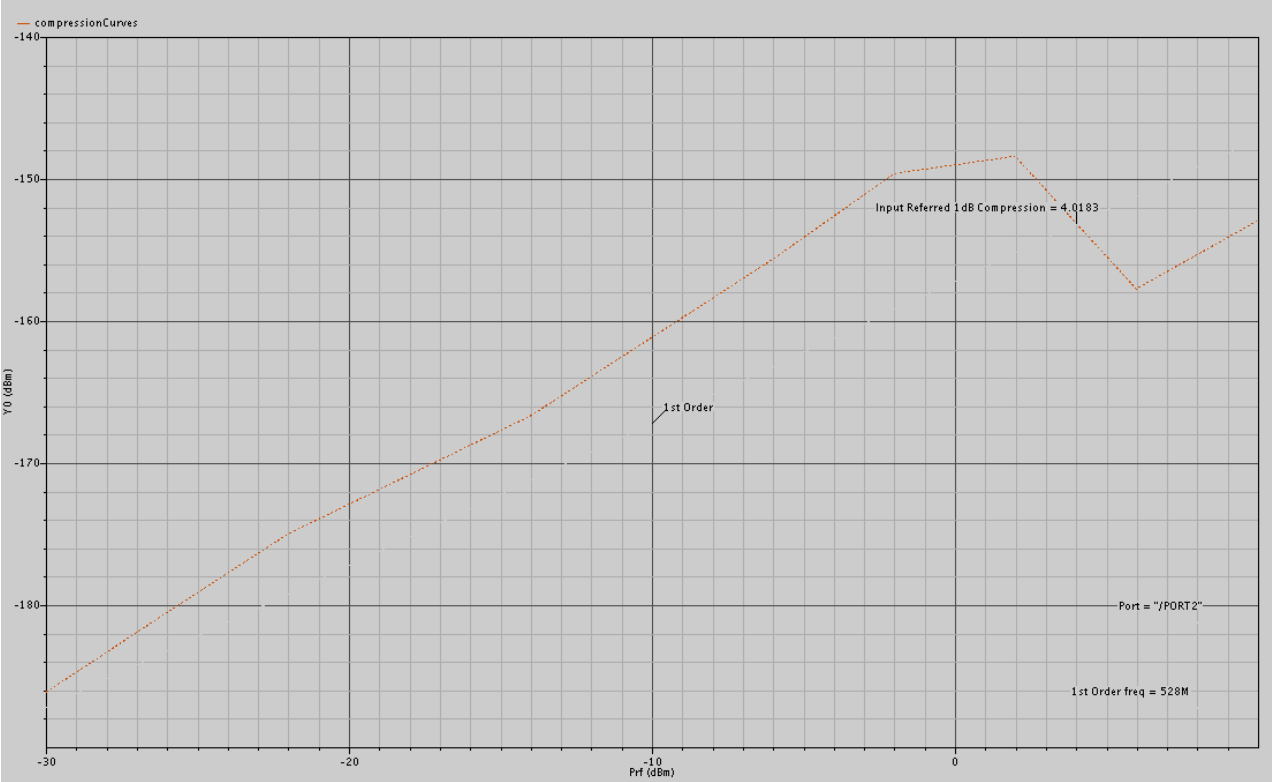


Figure 4.28: PSS 1db Compression point 528MHz (inputPrf = -30db)

→ 1db comp. point =- 4.0183dBm.

In the following graphs we are also presenting the 1db Compression Point for other values of input Prf:

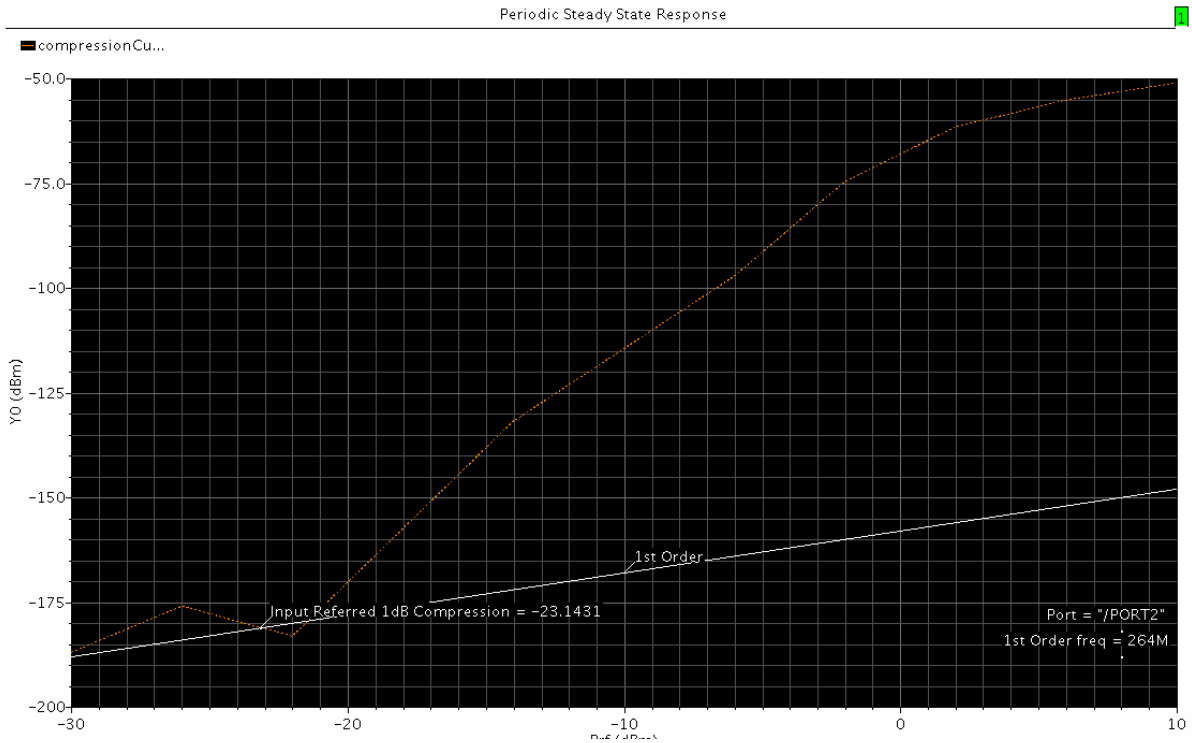


Figure 4.29: PSS 1 dbcompression Point (inputPrf=-30db) → **1db comp. point=-23.1431**

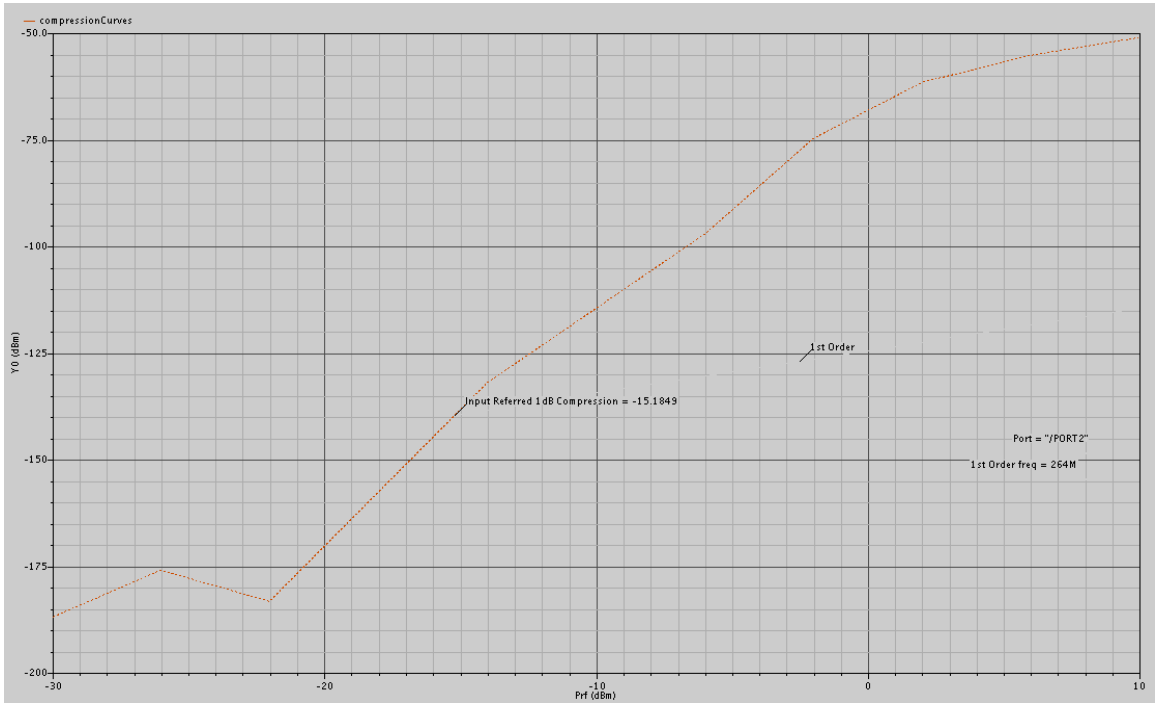


Figure 4.30: PSS 1 dbcompression Point (inputPrf=-20db) → 1db comp. point=-15.1849

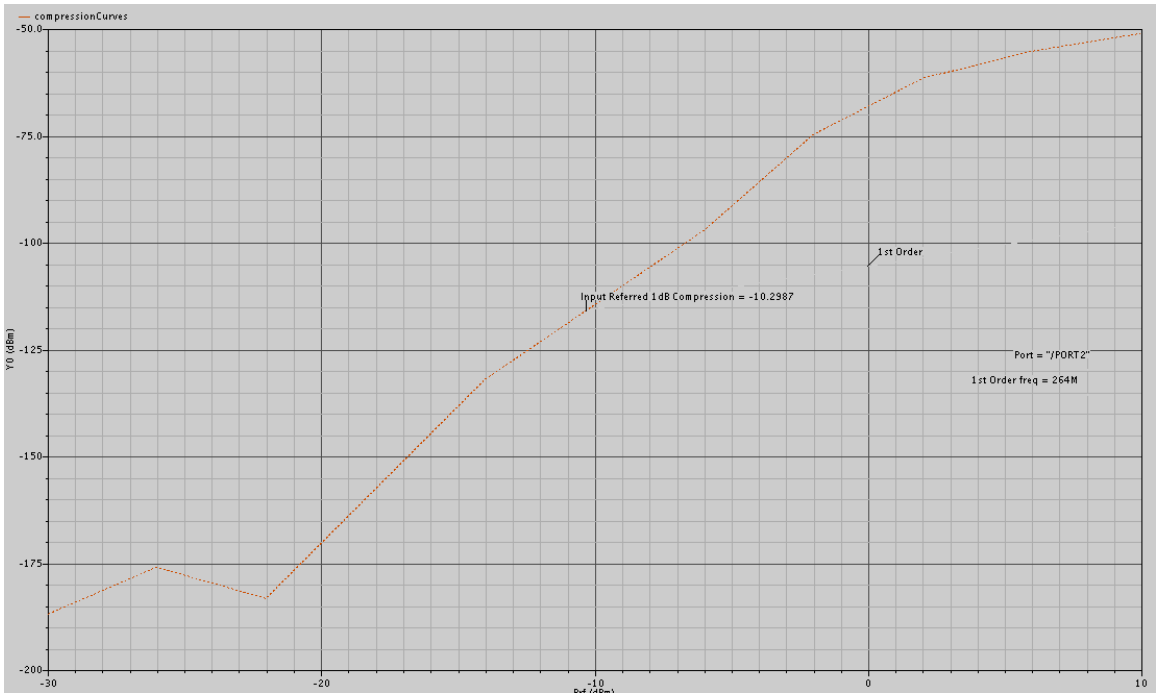


Figure 4.31: PSS 1 dbcompression Point (inputPrf=-15db) → 1db comp. point=-10.2987

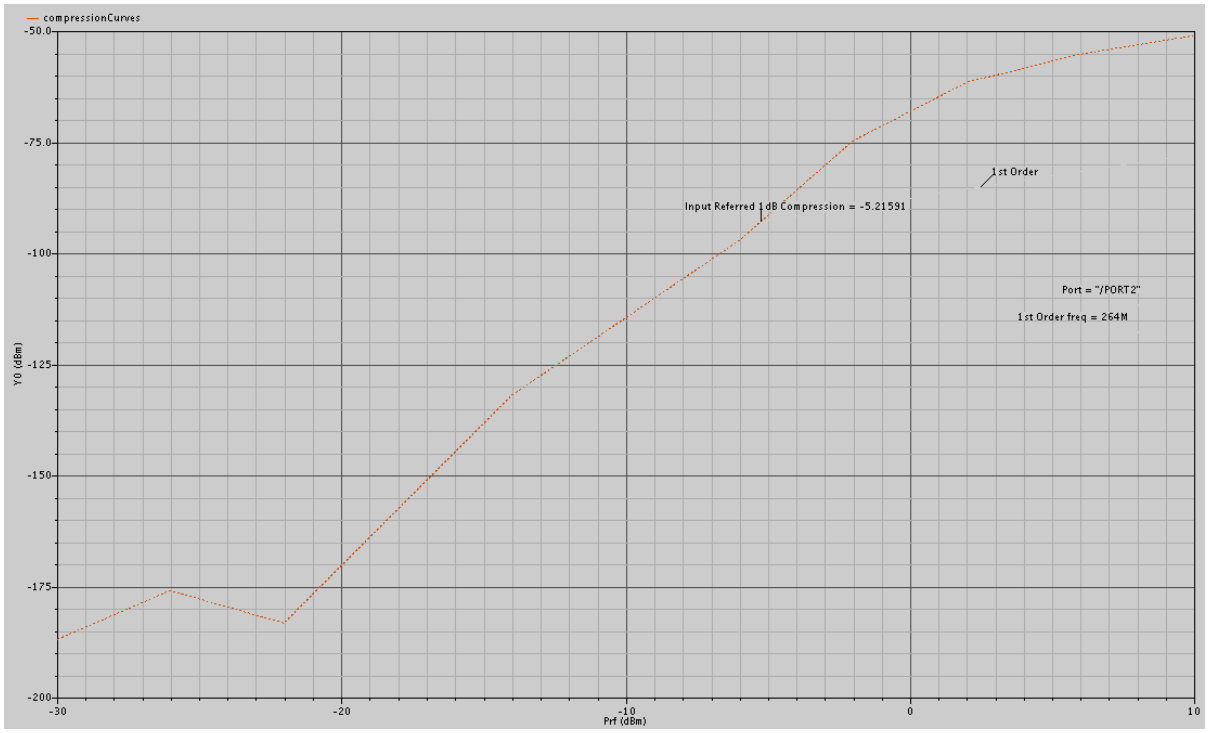


Figure 4.32: PSS 1 dbcompression Point (inputPrf=-10db) → 1db comp. point=-5.21591

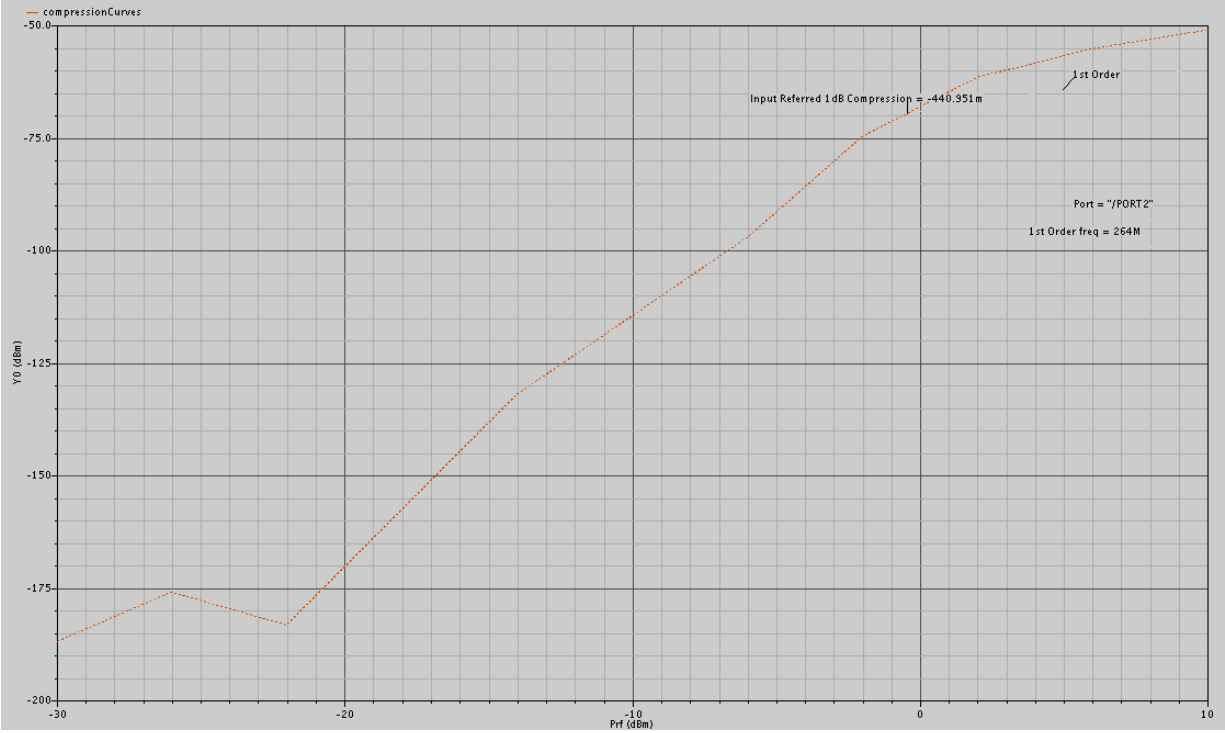


Figure 4.33: PSS 1 dbcompression Point (inputPrf= -5db) → 1db comp. point=-440.951m

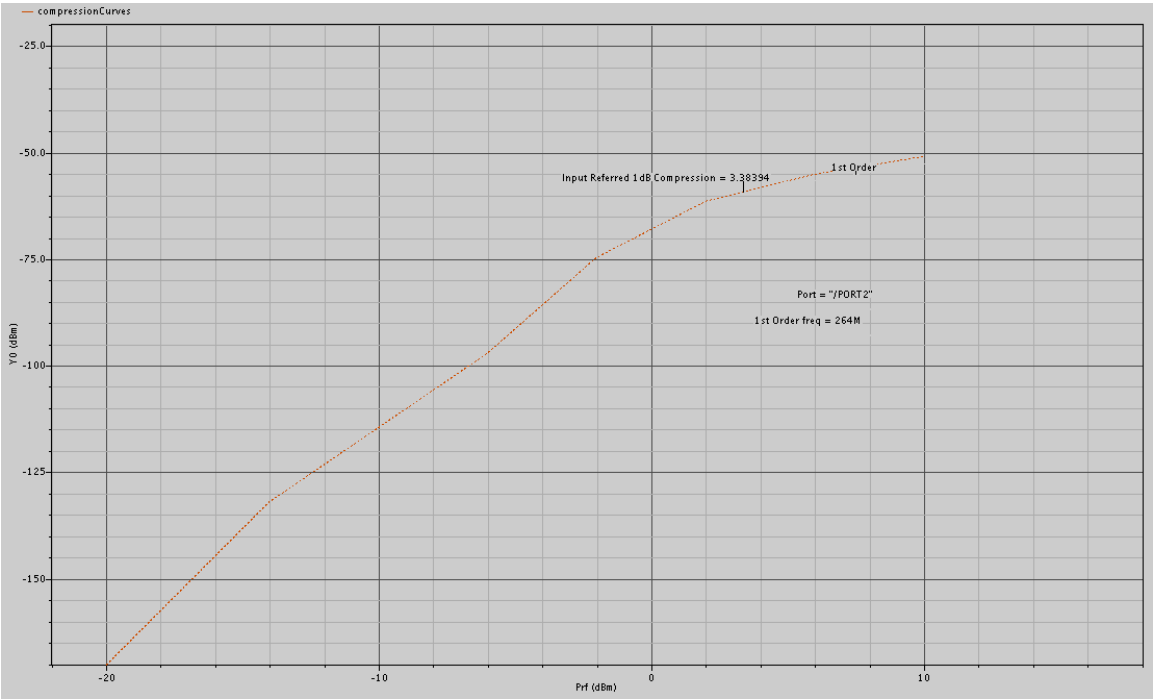


Figure 4.34: PSS 1 dbcompression Point (inputPrf = 0db) → 1db comp. point = 3.38394

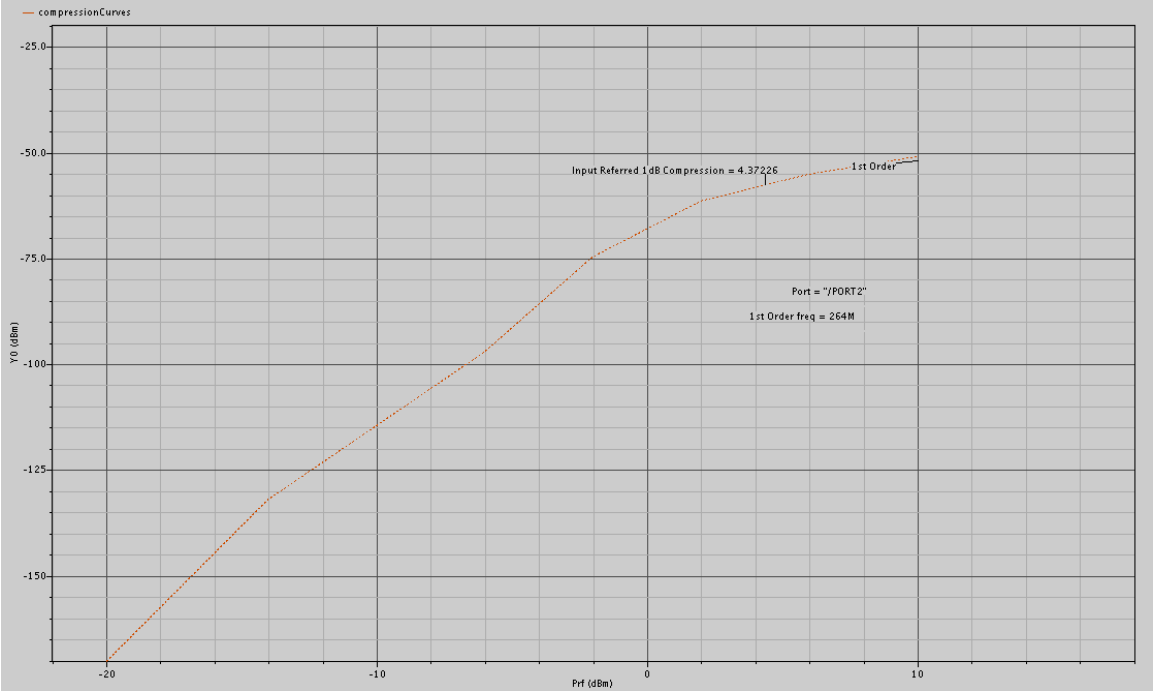


Figure 4.35: PSS 1 dbcompression Point (inputPrf = 5db) → 1db comp. point = 4.37226

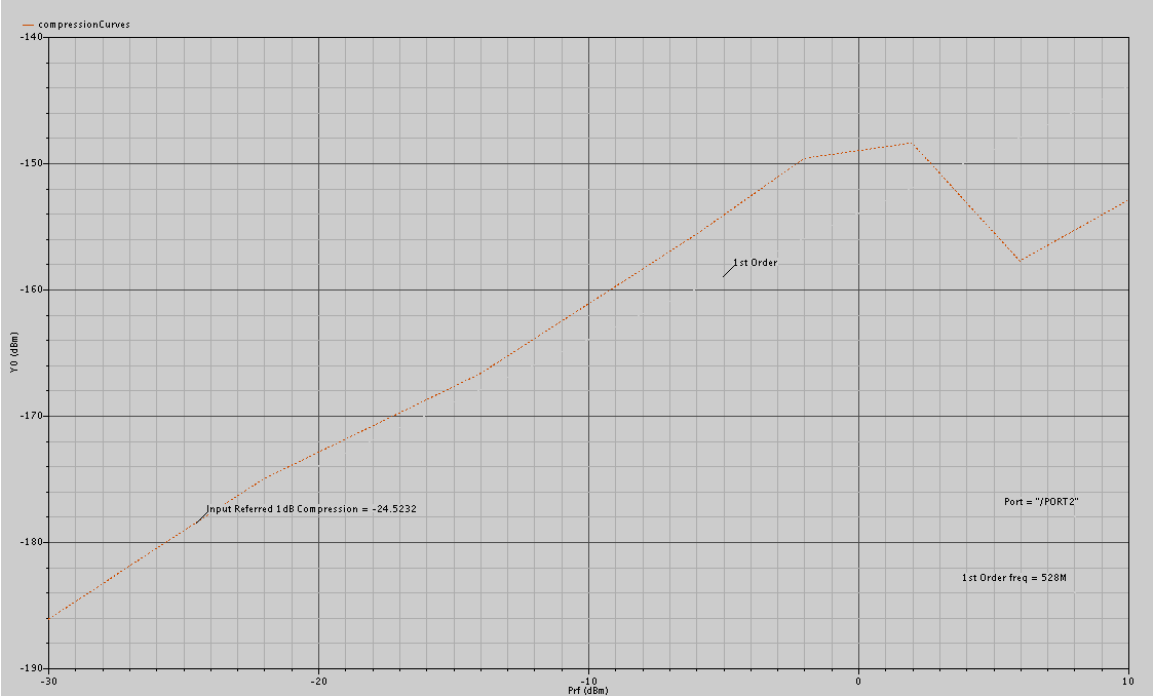


Figure 4.36: PSS 1db Compression point 528MHz (inputPrf = -25db)

→ 1db comp. point = -24.5232.

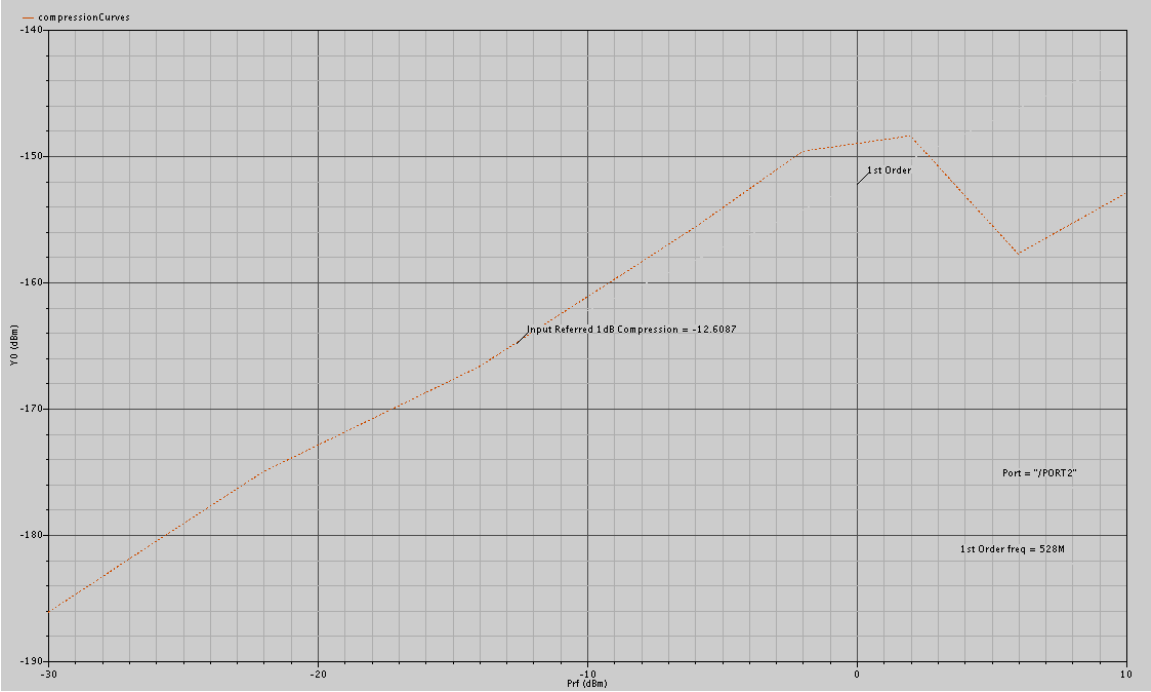


Figure 4.37: PSS 1db Compression point 528MHz (inputPrf = -15db)

→ 1db comp. point = -12.6087

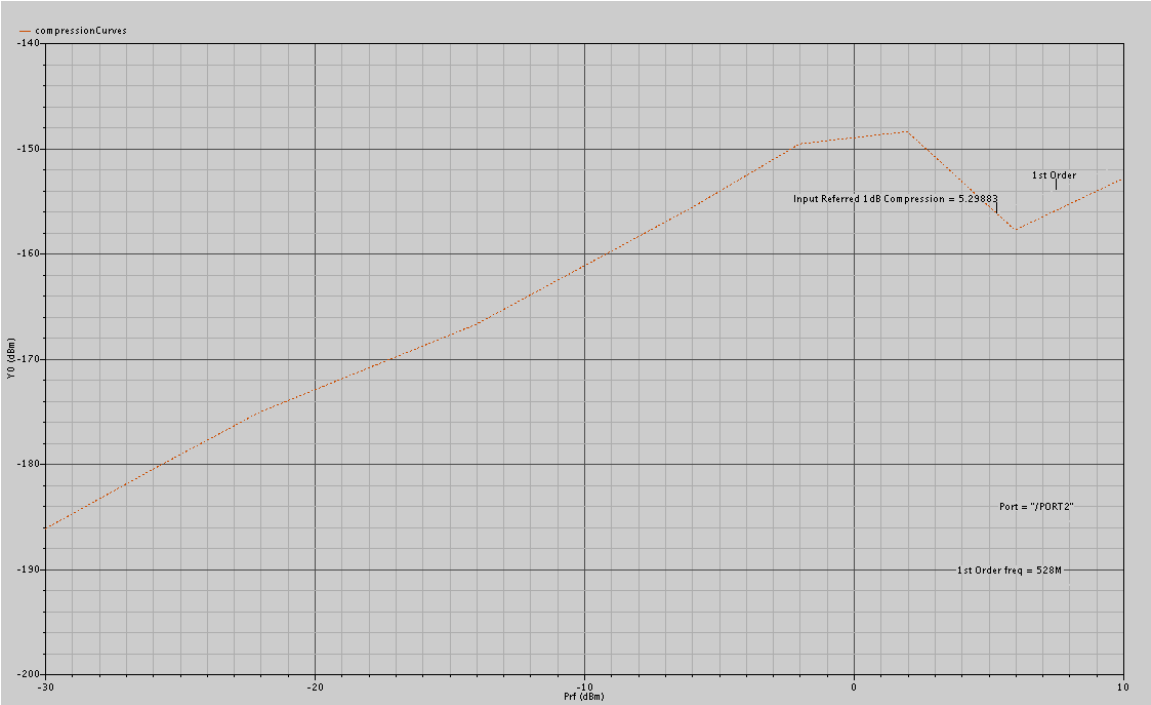


Figure 4.38: PSS 1db Compression point 528MHz (inputPrf =1db)

→ 1db comp. point = 5.29883

4.6 Third-Order Intercept (IP3) Measurement with Swept PSS and PAC

Here we combine a swept PSS analysis with a Periodic AC (PAC) analysis to produce data for an IP3 plot. This IP3 Calculation obtains the same information as the 1 dB compression point, but this simulation runs more quickly for the reasons that it processes the second tone only during PAC analysis and it considers only two of the second tone sidebands. The swept PSS analysis for the 1 dB compression point considered all sidebands for all signals.

Setting Up the Simulation

In the Simulation window, we use *Analysis - Disable* to disable any analyses we ran previously.

Editing the Schematic

1. In the Schematic window, we choose *Edit - Properties - Objects*.

The Edit Object Properties form appears. We use this form to modify the schematic by changing the list of CDF properties.

2. To modify the schematic for this simulation, we click on the rf voltage source in the Schematic window.
3. Highlight *Display second sinusoid* and remove any values that are set from previous analyses.
4. We choose *sine* for *Source type*.
5. We type frf for *Frequency 1*.
6. We type prf in the *Amplitude 1 (dBm)* field.
7. Type prf for the *PAC Magnitude (dBm)* value.

This simulation uses dBm values rather than magnitude.

8. Click *OK*.
9. In the Schematic window, choose *Design - Check and Save*.

Setting Up the PSS and PAC Analyses

1. In the Simulation window, we choose *Analyses - Choose*.

The Choosing Analyses form appears.

Setting Up the PSS Analysis

1. In the Choosing Analyses form, we choose *pss* for the *Analysis*.
2. In the *Fundamental Tones* section, *Auto Calculate* button must be highlighted.

The value 264M is specified as the *Beat Frequency*. The PAC analysis is responsible for the two tones, and the PSS analysis is now a single signal analysis. Consequently, the fundamental frequency is set to the original 256 MHz. The *Beat Frequency* button is highlighted by default.

3. In the *Output harmonics* cyclic field, we choose *Number of harmonics* and type 2 in the field.
4. We highlight *moderate* for the *Accuracy Defaults (errpreset)* setting.
5. Then we highlight the *Sweep* button.

The form changes to let us specify data for the sweep.

6. In the *Sweep* cyclic field, we choose *Variable*.
7. We click on the *Select Design Variable* button.

The Select Design Variable form appears.

8. We highlight prf in this form and click *OK*.
9. We choose *Start-Stop* for the *Sweep Range* value, and then type -25 in the *Start field* and 5 in the *Stop* field.

The sweep is skewed downward for a down converter.

10. Highlight *Linear* for the *Sweep Type*, highlight *Step Size,* and then type 5 in the *Step Size* field.
11. Highlight *Enabled*.

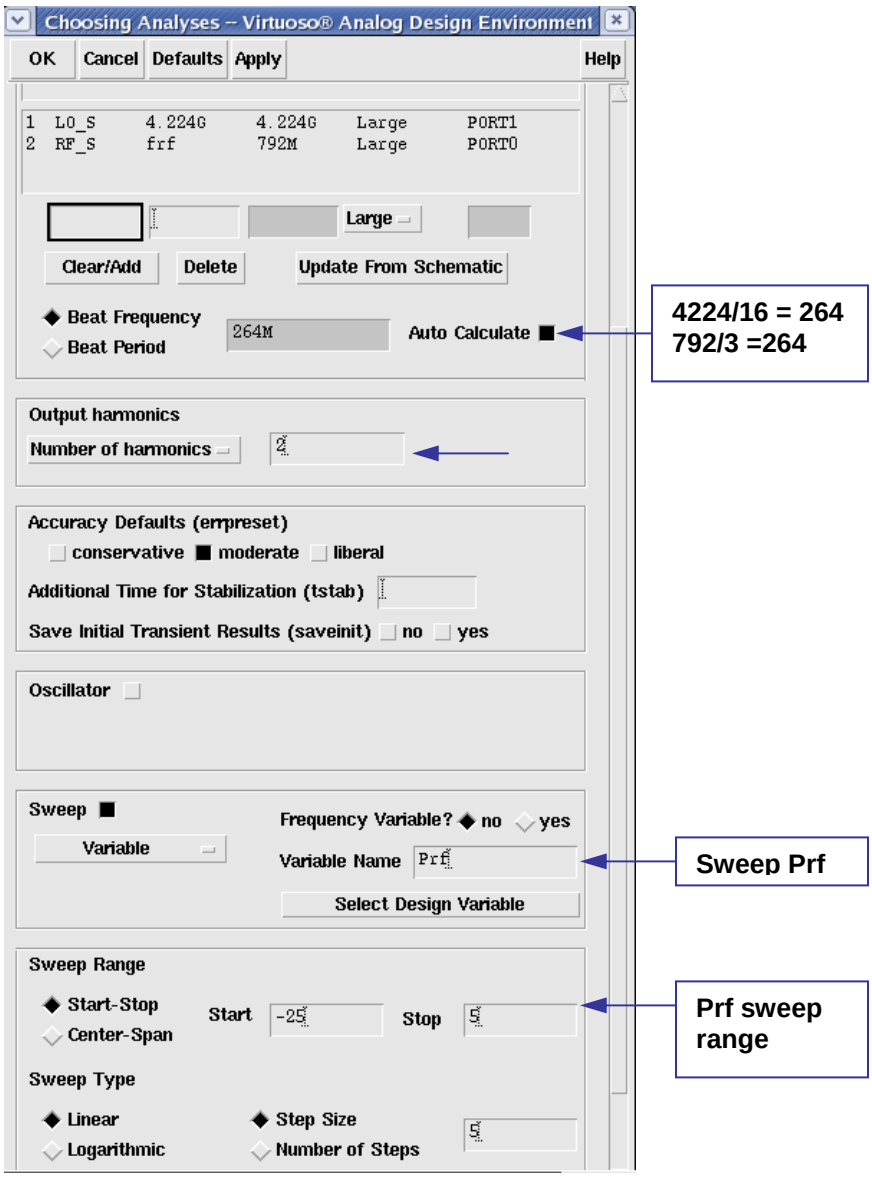


Figure 4.39: PSS setup for IP3 measurement.

Setting Up the PAC Analysis

1. At the top of the Choosing Analyses form, we highlight *pac*.

The Choosing Analyses form changes to let us specify data for a *pac* analysis.

2. We type 793M for the *Single-Point [ ] Frequency (Hz)* value.

3. In the *Sidebands* cyclic field, we choose *Array of indices* and type 10 and -16 with a space between them.

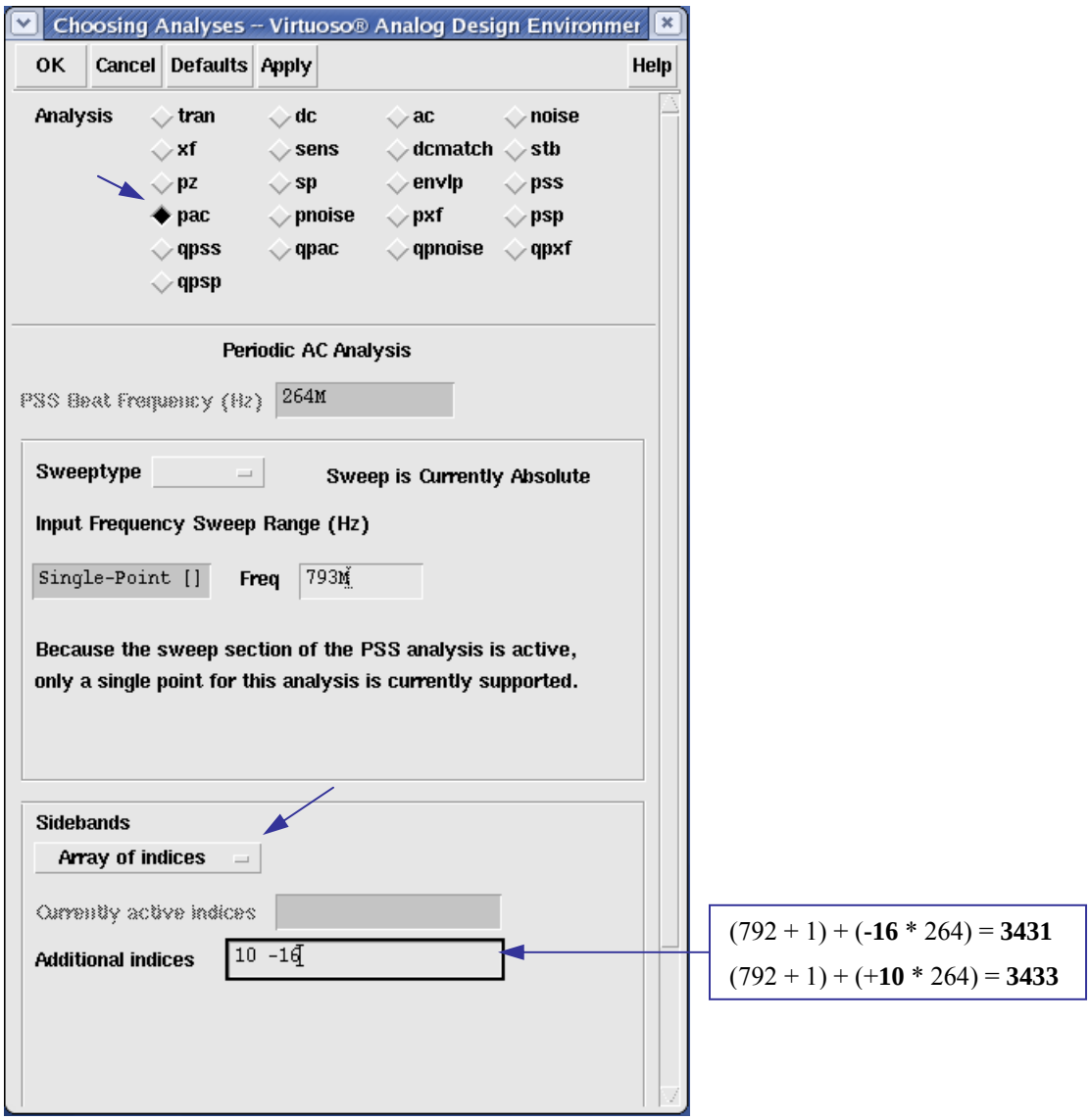


Figure 4.40: PAC setup for IP3 measurement.

Given a fundamental tone of 264 MHz, the LO at 4.224 GHz, and two RF tones at 792 MHz and 793 MHz, the sidebands of 10 and -16 represent respectively the first-order harmonic of the IF output at 3.431 GHz and the third-order harmonic at 3.433 GHz.

(RF + 1) – (harmonic_No * Beat_Frequency) -1 = (792 + 1) + (**-16** * 264) = **3431**

(RF + 1) – (harmonic_No * Beat_Frequency) -1 = (792 + 1) + (+**10** * 264) = **3433**

4. Highlight the *Enabled* button and Click *OK*.

Running the Simulation

1. To run the simulation, we choose *Simulation - Netlist and Run* in the Simulation window. (This example compares two signals only 1MHz apart. An analysis of two signals so close together would have taken much longer with the previous analysis.)
The output log file appears and displays information about the simulation as it runs.
2. We can check the output log file to be sure the simulation is completed successfully.

```
Conv norm = 828e-03, max dI(V10:p) = 1.97377 nA, took 680 ms.

pss: The steady-state solution was achieved in 7 iterations.
Number of accepted pss steps = 2824.
Total time required for pss analysis `sweeppss-006_pss' was 5.15 s.


*******************************************************
Periodic AC Analysis `sweeppss-006_pac': freq = 793 MHz
*******************************************************
Using the operating-point information generated by PSS analysis
        `sweeppss-006_pss'.
Total time required for pac analysis `sweeppss-006_pac' was 1.02 s.


**** Run Status for sweep analysis `sweeppss' ****
Sweep iteration 2 (`Prf' = -20) failed.
Sweep iteration 4 (`Prf' = -10) failed.
Total time required for sweep analysis `sweeppss' was 79.93 s (1m

modelParameter: writing model parameter values to rawfile.
element: writing instance parameter values to rawfile.
outputParameter: writing output parameter values to rawfile.
designParamVals: writing netlist parameters to rawfile.
primitives: writing primitives to rawfile.
subckts: writing subcircuits to rawfile.
```

Figure 4.41: Progression Output log file for IP3 measurement.

Plotting the IP3 Curve

1. In the Simulation window, we choose *Results - Direct Plot - Main Form*.
The Direct Plot form appears.
2. We highlight *Replace* for *Plot Mode*.
3. We highlight *pac* for *Analysis*.
The form changes to display information for the PAC analysis.
4. We highlight *IPN Curves* for *Function*.

The form changes again.

5. We choose *Variable Sweep ("prf")* for *Circuit Input Power*.

6. We type -15 (-25 up to 5) for *Input Power Extrapolation Point (dBm)*.

This value is the intercept point for the ideal amplification extrapolation. If we do not specify a value, the plot defaults to the minimum variable value.

7. In the cyclic field, we choose *Input Referred IP3*.

8. Follow the prompt at the bottom of the form

Select 3rd Order Harmonic on this form...

Highlight ***10 3.433G*** in the *3rd Order Harmonic* list box.

9. We follow the new prompt at the bottom of the form

Select 1st Order Harmonic on this form...

Highlight ***-16 3.431G*** in the *1st Order Harmonic* list box.

10. We follow the new prompt at the bottom of the form

Select Port on schematic...

In the Schematic window, we click on the *if* signal net.

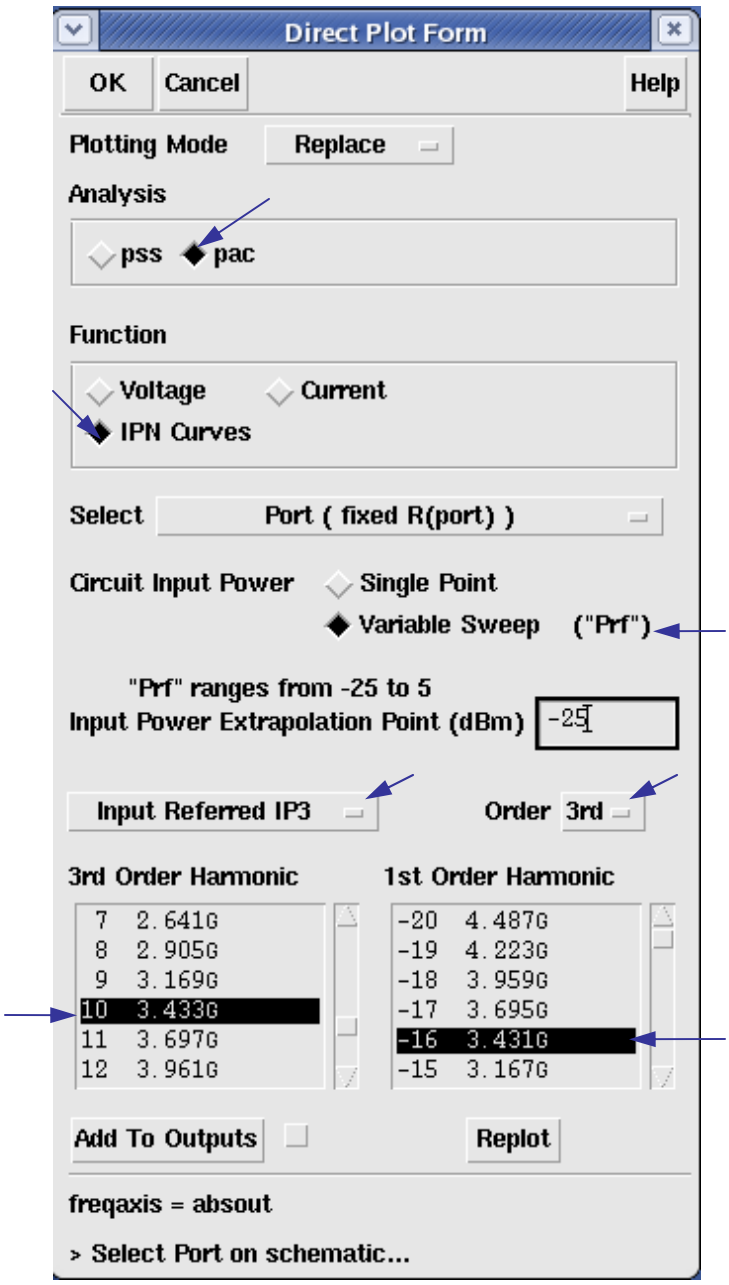


Figure 4.42: Setup of Direct Plot Form for IP3 measurement.

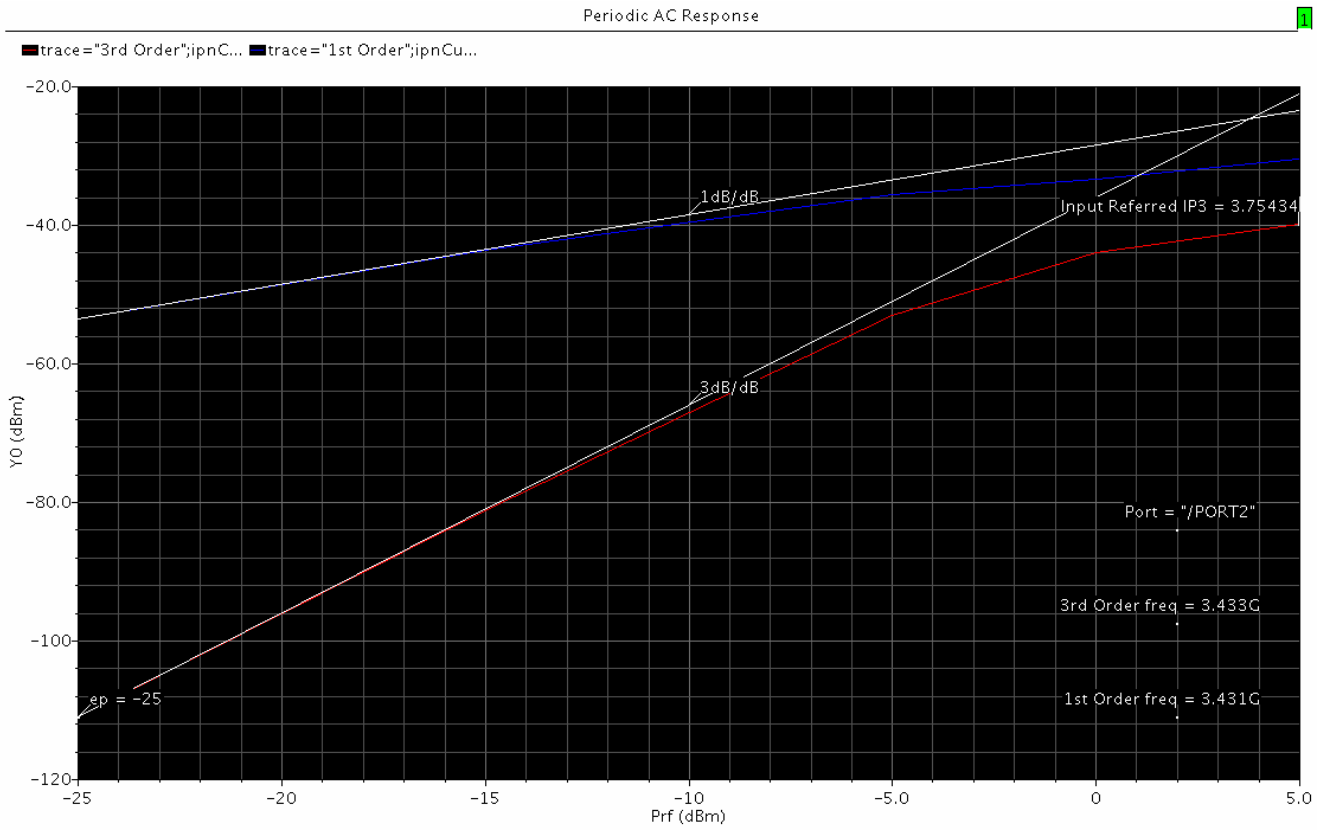


Figure 4.43: IP3 = 3.75434 for Prf=-25dbm (PSS+PAC)

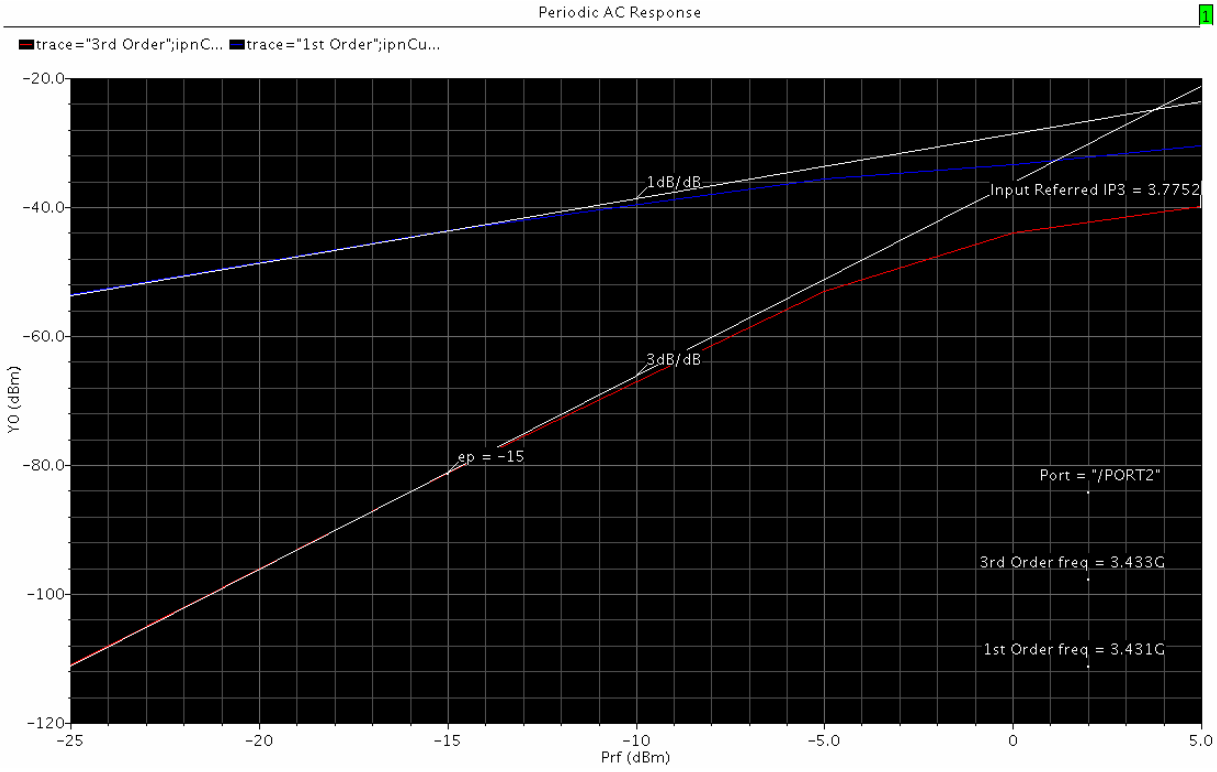


Figure 4.44: IP3 = 3.75434 for Prf=-15dbm (PSS+PAC)

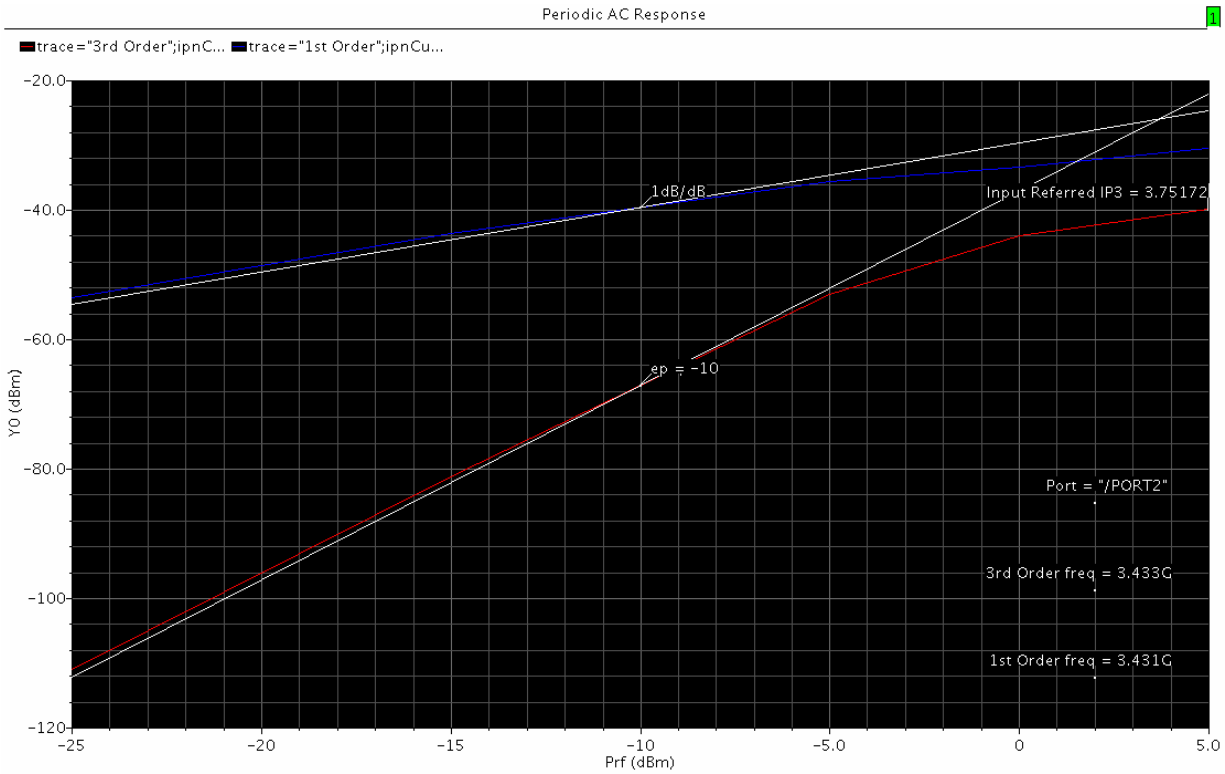


Figure 4.45: IP3 = 3.75434 for Prf=-10dbm (PSS+PAC)

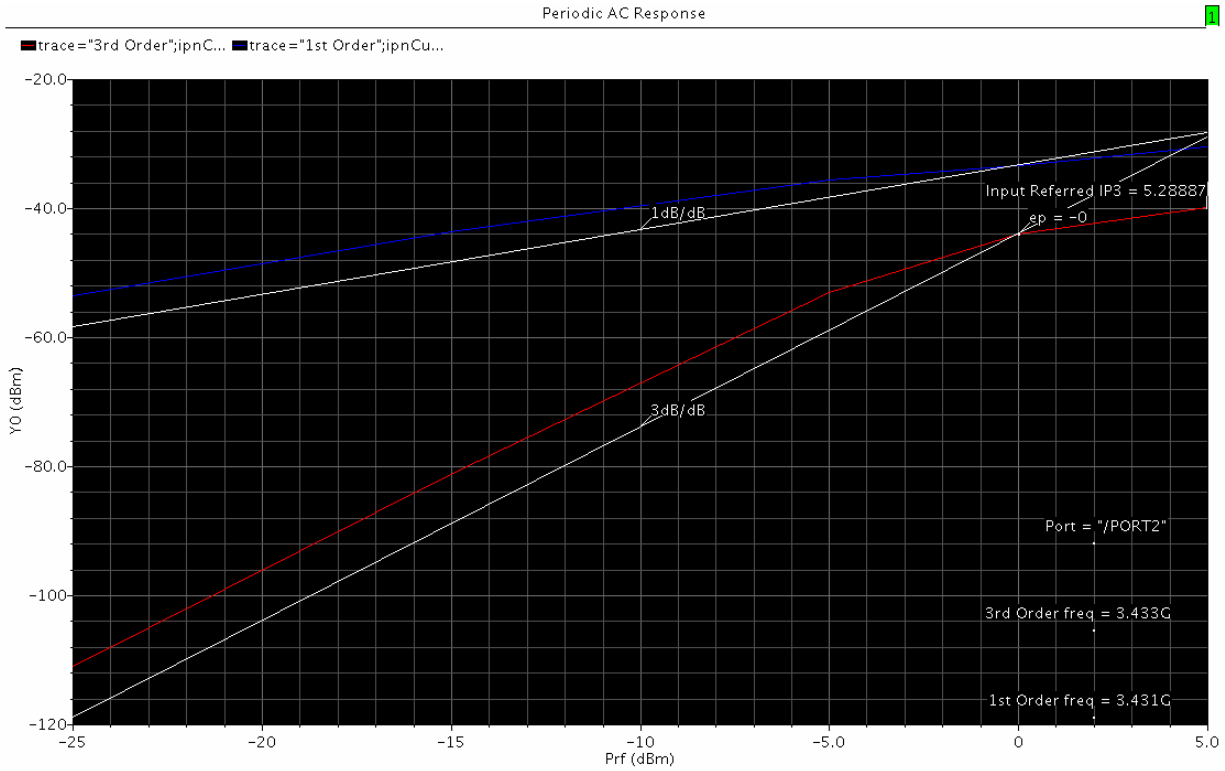


Figure 4.46: IP3 = 3.75434 for Prf=0dbm (PSS+PAC)

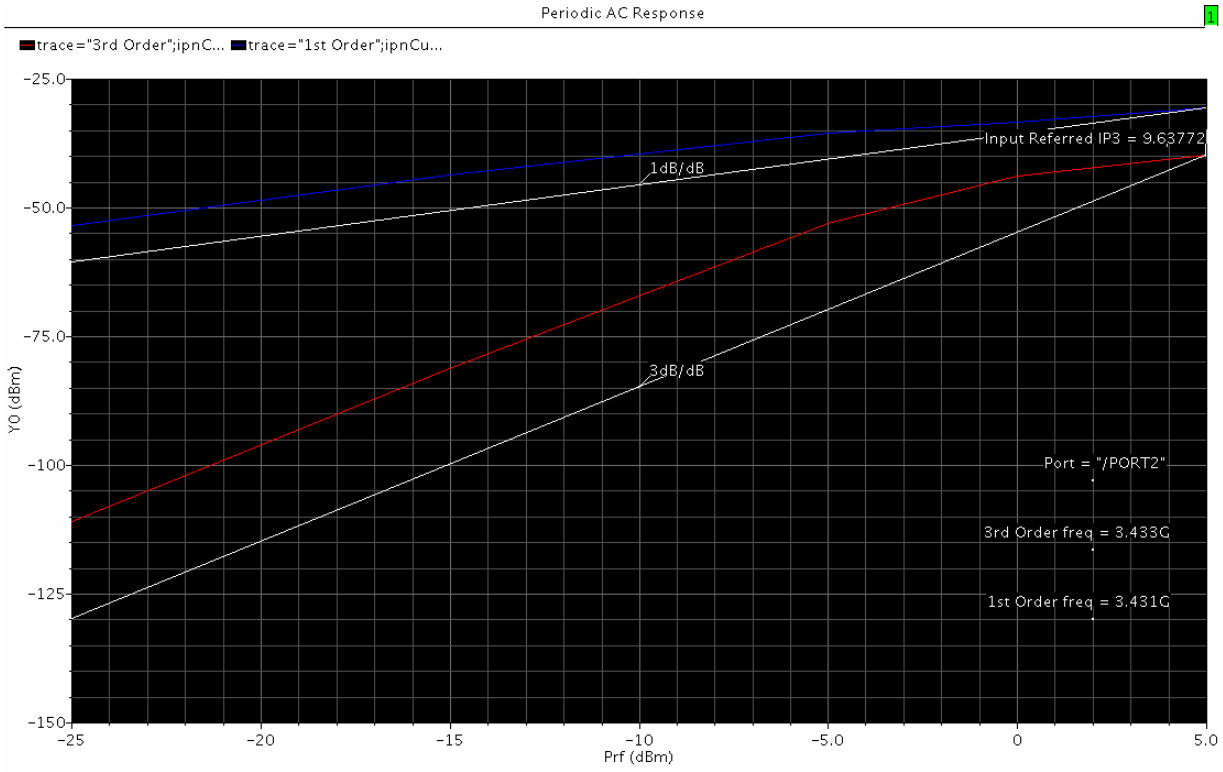


Figure 4.47: IP3 = 3.75434 for Prf=5dbm (PSS+PAC)

4.7 MIXER CONSUMPTION

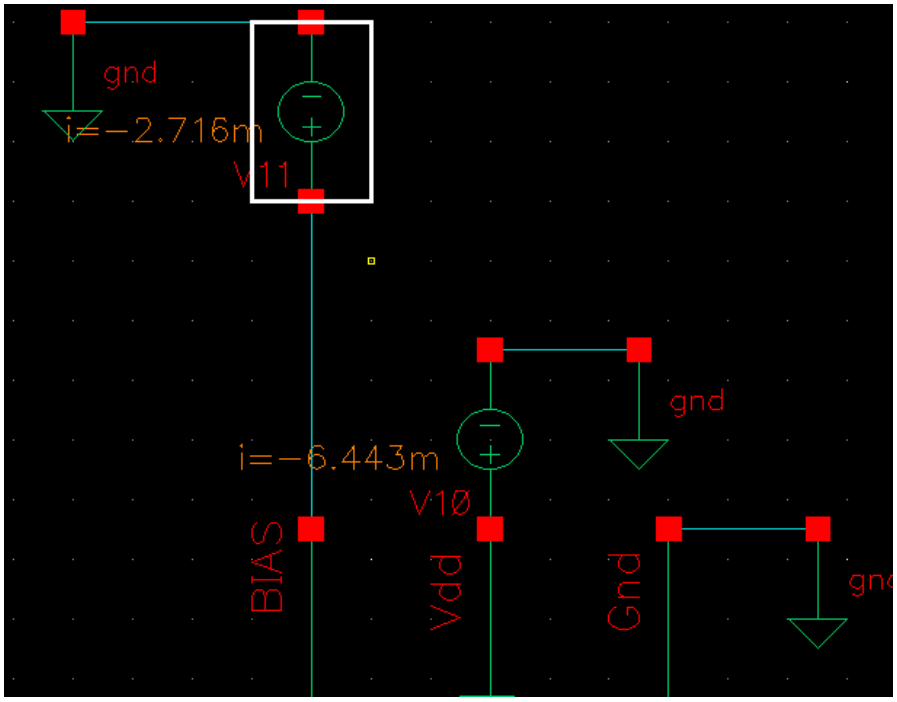


Figure 4.48: Mixer Consumption

The consumption of the mixer as we can see from figure 4.48 is:

Iconsumption= Ibias + Idc_mirror = (-6.443mA) + (- 2.716mA) = >

Iconsumption = -9.159mA

4.8 MIXER OVERALL PERFORMANCE

MIXER PERFORMANCE	SIMULATION RESULTS
IP3	3.75434 dBm
NF	16.12 dB
1dB Compression Point	- 4.0183 dBm
HTD	- 45.98 dB
Conversion Gain	- 3.61 db
Iconsumption	9.159mA

PARAMETER	VALUE
Vdd	1.8V
V_DC_MIRROR	1.8V
Vdc_rf	700mv
Vdc_lo	1.1V
Prf	-30dB
VLO	400mV
fRF	792MHz
fLO	4.224GHz
fIF	3.432GHz
BEAT FREQUENCY	264MHz

INFERENCES – FUTURE EXTENSIONS

This mixer or a combination of it, could be used for the design for a fast-hopping frequency synthesizer for Ultra WideBand or for other WLAN transceivers or receivers with respect on the frequencies that the device will need to operate. This mixer can operate with almost the same performance for IF frequency at 3.432GHz and 5.016GHz.

An extension that must be to the mixer is the design of the layout that was not performed in this thesis project.

REFERENCES

[1] D.A. Johns, Ken Martin, "Analog Integrated Circuit Design", John Wiley & Sons, 1997.

[2] Behzad Razavi, "RF Microelectronics", Prentice Hall, 1998.

[3] Apostolos Samelis "Ειδικά Θέματα Σχεδίασης Ηλεκτρονικών Κυκλωμάτων και Συστημάτων".

[4] Matthias Bucher "Μοντέλο του MOS Τρανζίστορ για Σχεδίαση Αναλογικών Ολοκληρωμένων Κυκλωμάτων", November 2004.

[5] Hong Zhang, Guican Chen, Member, IEEE, "A Monolithic Fast-Hopping Frequency Synthesizer for MB-OFDM UWB".

[6] Sedra & Smith, "Microelectronic Circuits", 3RD Edition.

[7] Gray, Hurst, Lewis, Meyer, "Analysis and Design of Analog Integrated Circuits" 4TH Edition.

[8] Behzad Razavi, "Design Of Analog Cmos Integrated Circuits", 2001

[9] Design Manual, Cadence.

[10] Hae-SeungLee and Michael H. Perrott, "High Speed Communication Circuits and Systems", Massachusetts Institute of Technology, Copyright ©2005

[11] Christian Lindholm,Todorce Petkovski, "Dual-gate MOSFET blandare för FM-mottagare", Lunds University, Februari 2003.

[12] C.-H Liao, Y.-K. Chu, D.-R. Huang, and H.-R. Chuang "A 5.7-GHz Low-Power and High-Gain 0.18-μm CMOS Double-Balanced Mixer for WLAN " 35th European Microwave Conference-Paris 2005.

[13] Glenn Watanabe, Henry Lau and Juergen Schoepf, "Integrated Mixer Design", Motorola Inc., Semiconductor Products Sector, WSSG RF/IF Division.

[14] Agilent Technologies "Noise Figure Performance of an Active 3G Gilbert Cell Mixer:Design & Measurement".

[15] CADENCE "Mixer Design".

[16] Stephen Long, "CMOS RFIC Upconversion Mixer Design", University of California, Santa Barbara.

[17] Chia-Wei Wu, Ming-Chun Su, Pai-Shan Hsiao, Kuo-Pin Lan, and Klaus Yung-Jane Hsu, “A Direct-Conversion CMOS Receiver for 5GHz Wireless LAN”.

[18] K. Vavelidis**,** I. Vassiliou, T. Georgantas, A. Yamanaka, S. Kavadias, G. Kamoulakos1, C. Kapnistis, Y. Kokolakis, A. Kyranas, P. Merakos, I. Bouras, S. Bouras, S. Plevridis, N. Haralabidis, “A Single-Chip, 5.15GHz-5.35GHz, 2.4GHz-2.5GHz, 0.18mm CMOS RF Transceiver for 802.11a/b/g Wireless LAN”.

[19] Vojkan Vidojkovic, Johan van der Tang, Arjan Leeuwenburgh and Arthur van Roermund, Mixer Topology Selection for a 1.8 - 2.5 GHz Multi - Standard Frond-End in 0.18 μ*m* CMOS.

[20] Hans Camenzind “Designing Analog Chips” February 2005.

[21] Chinchun Meng*, Member, IEEE*, Tzung-Han Wu, and Mingchi Lin, “Compact 5.2-GHz GaInP/GaAs HBT Gilbert Upconverter Using Lumped Rat-Race Hybrid and Current Combiner”.

[22] Agilent Technologies, “Fundamentals of Mixer Design”, Design Seminar 2001.

Masoud Zargari*, Member, IEEE*, David K. Su*, Member, IEEE*, C. Patrick Yue*, Member, IEEE*, Shahriar Rabii, David Weber, Brian J. Kaczynski*, Member, IEEE*, Srenik S. Mehta*, Member*, *IEEE*, Kalwant Singh, Sunetra Mendis, and Bruce A. Wooley*, Fellow, IEEE*, “A 5-GHz CMOS Transceiver for IEEE 802.11a Wireless LAN Systems”, IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. 37, NO. 12, DECEMBER 2002.