12-2004

RF CMOS receiver front-end using a reflex amplifier

Radovan Kuzet
University of Texas-Pan American

RF CMOS RECEIVER FRONT-END USING A REFLEX AMPLIFIER

A Thesis

by

RADOVAN KUZET

Submitted to the Graduate School of the
University of Texas Pan American
In partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ENGINEERING

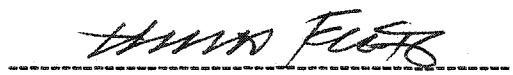
December 2004

Major Subject: Electrical Engineering

RF CMOS RECEIVER FRONT-END USING A REFLEX AMPLIFIER

A Thesis
by
RADOVAN KUZET

Approved as to style and content by:



Dr. Heinrich D. Foltz, Ph.D., P.E.
Associate Professor of Electrical Engineering
Chair of Committee

Dr. Junfei Li, Ph.D.
Assistant Professor of Electrical Engineering
Committee Member

Dr. Robert Jones, Ph.D.
Associate Professor of Mechanical Engineering
Committee Member

December 2004

ABSTRACT

Kuzet, Radovan, RF CMOS Receiver Front End Using a Reflex Amplifier. Master of Science and Engineering (MSE). December 2004, 83 pp.; 49 figures, 7 tables, 6 illustrations, references, 27 titles.

A concept for an RF-CMOS receiver with reflex topology is developed and tested, using OrCad as the primary design and simulation tool. The main idea of the reflex topology is to reuse a Low Noise Amplifier (LNA) stage in the circuit and thus eliminate one high power consumption stage. A Gilbert cell is used for down conversion, with the benefit of low noise and low power while maintaining gain. A set of passive diplexers allows reuse of the LNA at the RF and IF. The complete circuit was simulated using 0.18 micron and 0.5 micron AMI technology. Parameters for the transistors are imported from MOSIS development tools and design specifications for the purpose of production of this device. The overall system was simulated using the 900 MHz frequency for radio frequency (RF) and shows substantial gain and low noise figure. Lastly, the Gilbert cell mixer has been modified in an attempt to further reduce power consumption. Instead of using a local oscillator, we designed the Gilbert cell so it would self-oscillate. Output results and linearity characteristics are shown through simulation.

DEDICATION

This thesis is dedicated to all the great people that supported me and helped me throughout my career. Above all, I dedicate this thesis to my greatest role model, my father Zivko Kuzet.

Ovaj maturski rad je posvecen svim velikim ljudima koji su me podrzavali i pomogli mi kroz moju karijeru. Iznad svih zelim da posvetim ovaj rad mojem najvecem uzoru, mom ocu Zivku Kuzetu.

TABLE OF CONTENTS

Abstract ---------- iii

Dedication---------- iv

Contents---------- v

List of Figures---------- vi

List of Tables---------- ix

Acknowledgments---------- x

Chapter 1. Introduction ---------- 01

Chapter 2. System Description ---------- 04

Chapter 3. Low Noise Amplifier---------- 05

Chapter 4. Diplexer---------- 09

Chapter 5. Mixer--- ---------- 14

Chapter 6. Complete Reflex Simulation---------- 21

Chapter 7. Self-Oscillating Mixer---------- 29

Chapter 8. Implementation---------- 41

Chapter 9. Conclusion---------- 46

References ---------- 47

Appendix A-LNA Parameter Calculation ---------- 51

Appendix B-OrCad Simulation and Transistor Parameter Import---------- 55

Appendix C-Complete Circuit Schematics---------- 72

Appendix D-VLSI Layout of Reflex and SOM---------- 74

Vita---------- 78

LIST OF FIGURES

Figure 1.a: Conventional receiver front-end ---------- 02

Figure 1.b: Receiver front-end employing reflex arrangement---------- 02

Figure 2.1: Basic receiver architecture---------- 04

Figure 3.1: Basic Schematic of amplifier used ---------- 06

Figure 3.2.a: OrCAD schematic of the amplifier used in Reflex Circuit ---------- 07

Figure 3.2.b: OrCAD Simulation of circuit from Figure 3.2.a ---------- 08

Figure 4.1: Two Port representation of the Diplexer ---------- 09

Figure 4.2: Diplexer that uses combination of High Pass and Low Pass Filters ------- 10

Figure 4.3: Diplexer with Band Pass / Band Stop Combination ---------- 10

Figure 4.4.a: Diplexer circuit used at LNA output. ---------- 11

Figure 4.4.b: Diplexer Frequency Response ---------- 12

Figure 5.1: Basic double balanced mixer configuration ---------- 15

Figure 5.2: Basic double balanced mixer with IF and RF reversed ---------- 15

Figure 5.3: Double Balanced / Gilbert Cell Mixer ---------- 16

Figure 5.4: Gilbert Cell with bias circuitry ---------- 17

Figure 5.5.a: Time Domain Response of G.C. Mixer ---------- 18

Figure 5.5.b: Frequency Domain Response of G.C. Mixer ---------- 18

Figure 6.1: Detailed block diagram of the reflex circuit ---------- 22

Figure 6.2. Frequency domain output of reflex front-end ---------- 23

Figure 6.3. Time domain output of reflex front-end circuit ---------- 23

Figure 6.4: Normal and 3rd order curves explained ---------- 25

Figure 6.5: Normal Curve of the 0.18u process used for 1dB compression point calculation ---------- 26

Figure 6.6: FFT simulation of the circuit used for IP3 calculation ---------- 27

Figure 6.7: Crossing of normal and 3rd order curve as IP3 point ---------- 28

Figure 7.1a: Schematic of Basic Colpitts Configuration ---------- 30

Figure 7.2.a: Differential Oscillator ---------- 31

Figure 7.2.b: Time Domain Simulation of Differential Oscillator ---------- 31

Figure 7.2.c: FFT of Differential Oscillator ---------- 32

Figure 7.3: Quadrature Self Oscillating Configuration ---------- 32

Figure 7.4: Self oscillating Mixer using the BJT Technology ---------- 33

Figure 7.5.a: Self Oscillating Mixer using the CMOS Technology ---------- 34

Figure 7.5.b: Time Domain simulation of the CMOS – SOM ---------- 34

Figure 7.5.c: FFT Simulation of the CMOS – SOM ---------- 35

Figure 7.5.e Enlargement of the CMOS SOM 128 MHz component that produces the gain of the 3.5 dB ---------- 35

Figure 7.6: 1 dB compression point of SOM is ~ -16 dB at Output ---------- 36

Figure 7.7a: FFT response of 2 RF inputs showing main and IP3 components ------- 38

Figure 7.7b: 3rd order component enlarged ---------- 38

Figure 7.9: IP3 Point of SOM is approximately – 5 dB at output ---------- 39

Figure 8.1.a: Schematic of Circuit used for test ---------- 41

Figure 8.1.b: Actual SOM Circuit used for testing ---------- 42

Figure 8.2: FFT output captured by oscilloscope ---------- 52

Figure 8.3: Transistor layout for the Self Oscillating Mixer uses 10 fingers ---------- 43

Figure 8.4: Complete Circuit layout for Self Oscillating Mixer ---------- 45

Figure B.1: Or-CAD Simulation set for 200 ns---------- 55

Figure B.2: Or-CAD Simulation set for 200 ms ---------- 56

Figure B.3a: Simulation run 200 ns takes 10 sec to execute ---------- 57

Figure B.3b: Simulation run 2000 ns takes 1 min to execute ---------- 57

Figure B.3c: Simulation run 20000ns takes 30-45min ---------- 57

Figure B.4: Screen shot of the Main building block of system ---------- 58

LIST OF TABLES

Table 5.4.a: Optimum load of the Gilbert cell Mixer---------- 19

Table 5.4.b: Mixer Specifications ---------- 20

Table 6.0: Performance Characteristics of Reflex Receiver---------- 24

Table 6.1: Performance Characteristics of some related BJT receivers---------- 24

Table 7.0: Table of input vs. output values used for normal curve---------- 37

Table 7.10: Measured and calculated results of the SOM---------- 39

Table 7.11: Table of measured values for the Quadrature mixer---------- 40

AKNOWLEDGEMENTS

I would like to thank to my advisor, Dr. Heinrich Foltz, for giving me his support, and sharing his knowledge and experience. I would like to thank Dr. Junfei Li and Dr. Robert Jones for their willingness to serve as the members of my thesis committee. Finally, Sebastian Pupandpouri and Mukesh Chilukuri for their assistance on this graduate project.

CHAPTER 1

Introduction

Reduction of power consumption is a critical goal in many wireless device designs. In a typical CMOS (Complementary Metal Oxide Semiconductor) receiver, the front-end stages (Low Noise Amplifier (LNA), mixer, Local Oscillator (LO)) consume the greatest amount of power despite relatively low device count, due to the need for high bias currents to maintain gain and noise figure at high frequencies. This is true even when there is significant signal processing and decoding in the low frequency back-end of the receiver, and holds even in extremely low power CMOS receivers. For example, Darabi and Abidi [7] report a complete 900 MHz paging receiver that draws a total of 4.5 mW, of which more than 50% is consumed by the LNA, mixer, and LO. In this paper, we present a concept for applying the classical reflex technique, formerly in common use in low-cost receivers, in which a single amplifier stage is used at two points in the signal path. Tutorial material on the subject can be found in [8], and in a series of articles by Hall [9-13].

The original idea behind the reflex technique was to reduce the number of active devices in an era when tubes or transistors constituted the major expense in a design. In our application, the purpose is to reduce the number of high current stages required for a given gain by reusing an amplifier stage in two parts of the signal path. The actual parts

count in our case is higher than in a conventional design, but the power consumption is lower.

The basic reflex concept is illustrated in Figure 1. Figure 1a shows a conventional receiver front end, while Figure 1b represents the same concept implemented using a reflex amplifier. A single amplifier is used twice; the first time at the RF as an LNA and then as the first stage of IF amplifier.

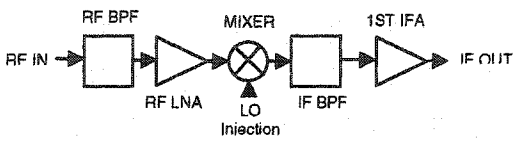


Figure 1a. Conventional receiver front-end.

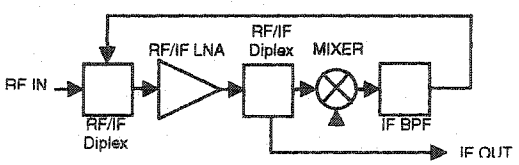


Figure 1b. Receiver front-end employing reflex arrangement..

In this way, a high performance, low-noise first IF stage is gained with no additional power consumption. The main cost is an increase in the passive component count.

For testing purposes, a Radio Frequency (RF) of 900 MHz and Intermediate Frequency (IF) of 100 MHz were picked because of their wide application. Simulations were conducted using the AMI 0.18 μm and 0.5 μm processes.

CHAPTER 2

System Description

The reflex topology has been around since the 1960s. The idea of a reflex means reusing the parts within the system. In original design, people used a reflex idea to reduce part count. Today with new technologies and miniaturization, parts count is not such a concern; but we used the idea to minimize the power consumption. By elimination of the biggest energy spenders in the system, we were able to save energy and achieve desirable performance.

There are different receiver architectures present today. Even though they all have their disadvantages, advantages, similarities, and differences, many have the basic starting point and that is architecture shown in Figure 2.1

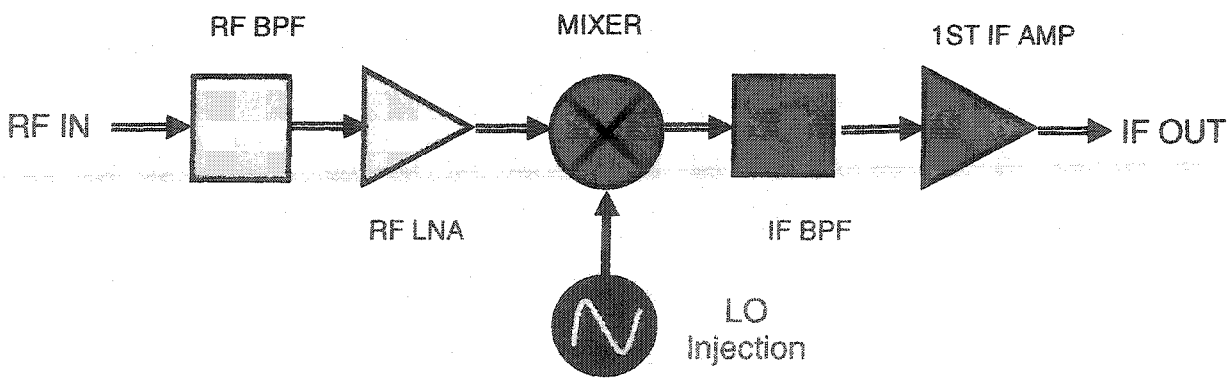


Figure 2.1 - Basic Receiver Architecture

Chapter 3

Low Noise Amplifier

One of the first steps of the system was to design a low noise amplifier. This is a very important device in the system because it provides gain and reduces Noise Figure (NF). The challenge of the LNA, as the part of the reflex system, is the ability to work on two frequencies and still provide enough gain and low noise figure.

So, the first question is: "Do we use wideband or narrowband LNA?" The advantage of the wideband design is that it is simpler and easier to manage. When designing wideband LNA, we first design general wideband amplifier and then we place band-pass filter at the input of the amplifier for the desired frequency. This way, design work is divided into two independent and easily manageable steps. The problem with this approach is that it demands that the amplifier possess wideband response making the circuit structure complicated and power hungry. Also, noise performance of wideband design is poor. On the other hand, narrowband design requires more involvement with the design, due to impedance matching and low noise optimization. As the result, noise figure is improved and power consumption is reduced. In narrowband design, tank circuit is placed on the end so that power gain is peaked out; so, there is no need for additional gain stage, which means simpler circuit. The disadvantage of narrowband design is that it is hard to achieve amplification on accurate band-pass frequency, due to component values variation. This can be resolved by using a tunable tank circuits or placing some parts off the integrated circuit.

For the reasons mentioned above, we decided to use narrowband design and the basic outlines of the design are shown below. The complete calculations are listed in Appendix A.

After the design calculations and proper bias, which we decided to be at 3.3 V, the major remaining task is to design a tank circuit. Because the amplifier needs to work at two frequencies (RF and IF), we designed dual tank circuits placed in series. Figure 3.1 shows the simplified schematic of the circuit in which cascode configuration is used for the amplifier stage [1], [12]. Figures 3.2.a and 3.2.b show actual circuit with simulated output shown.

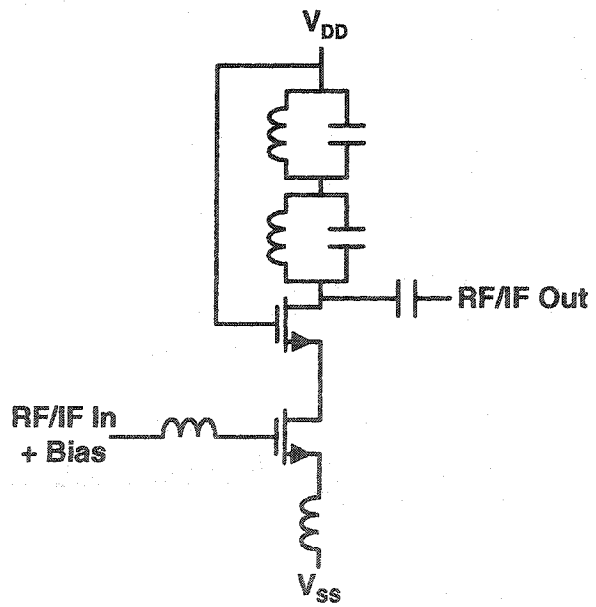


Figure 3.1: Basic schematic of amplifier used

One of the challenges of two tanks circuits is mutual interference between them; so, the standard formula (1) does not give us accurate values of the components.

$$f = \frac{1}{2\pi\sqrt{LC}} \qquad (1)$$

In this case, the solution is to use formula (1) as the primary design tool and then observe the simulation and make adjustments until it matches the desired frequencies. Because S parameters were not available, the standard matching design procedure will not work. As the result, we used a narrowband design procedure derived by Leung [1].

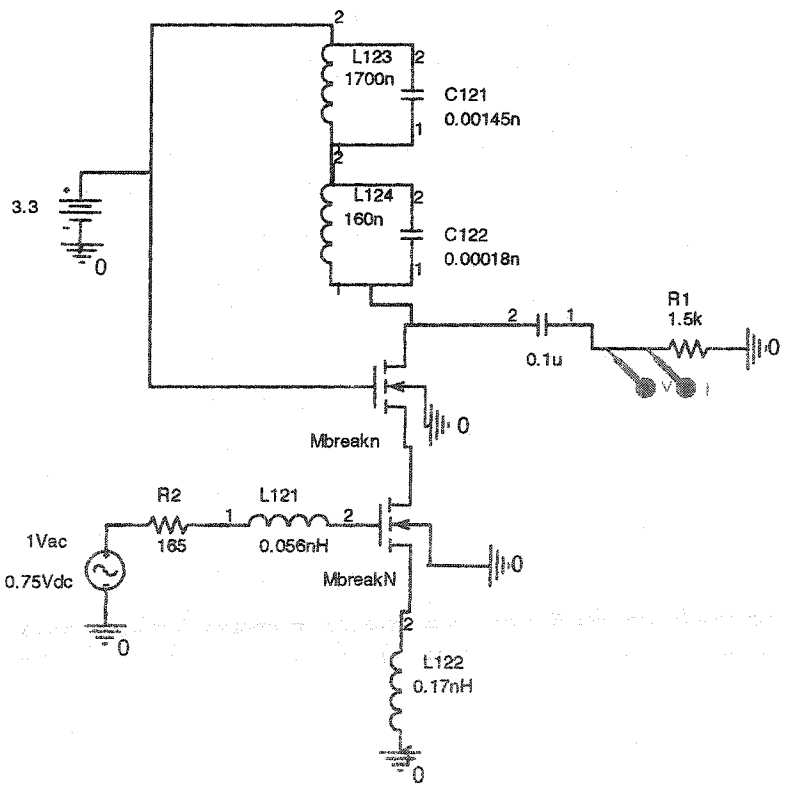


Figure 3.2a: Or-CAD schematic of the amplifier used in Reflex Circuit

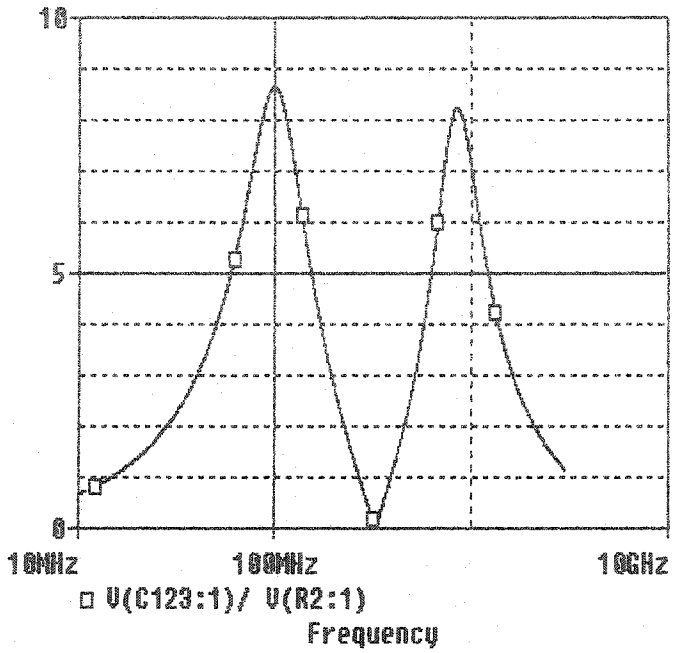


Figure 3.2b: Or-CAD Simulation of circuit from figure 3.2.a

The process can be divided into five basic steps::

1. Designing a drain Inductor using the real part matching condition
2. Designing a base inductor using the NF specifications
3. Designing a Width/Length ratio using real imaginary part matching condition
4. Checking to see if power specification is met, and
5. Using the Gain to find the value of the tank capacitor

The resulting circuitry, along with the bias, is shown in Figure 3.2.a. A noise figure (NF) simulation of the same circuit gave NF=5.7 dB at the two operating frequencies (100 and 900 MHz). This NF is adequate for the purpose of demonstrating the reflex architecture, but does not compare to the state of art for Complementary Metal Oxide Semiconductor (CMOS) LNAs, e.g. [15, 16] and further optimization may be necessary.

CHAPTER 4

Diplexer

The usual amplifier is a two-port circuit. That is, it has an input port consisting of two terminals and an output consisting of two more. One terminal (ground) can be shared between the ports. Many filters are also two-port networks, including most of the ladder networks we use so often. Many other networks have three or even more ports. A common example is a mixer, which has three ports. Another example of a three port network is a diplexer. This linear network is usually designed around two port filters where one end of two different filters is paralleled to form an input port. This is illustrated as Figure 4.1.

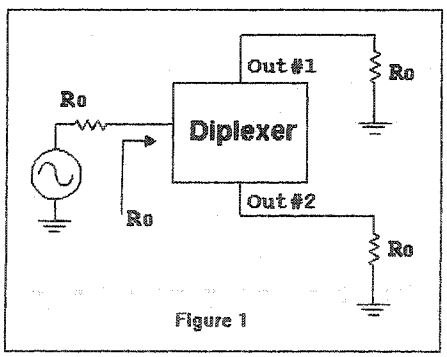


Figure 4.1: Two Port representation of the Diplexer

The purpose of a diplexer is usually to force a frequency constant impedance to occur at the input port, even though we usually only use one of the two output ports for

signals. The simplest form of diplexer uses a pair of one-element filters, a low pass and a high pass. This is shown in Figure 4.2.

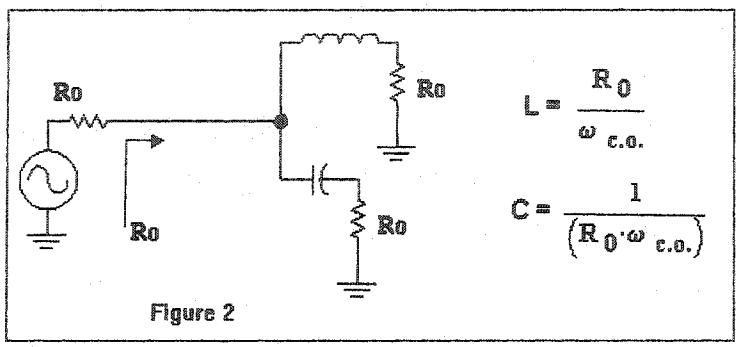


Figure 4.2: Diplexer that uses combination of High Pass and Low Pass Filters

Figure 4.2 shows the basic diplexer circuit layout. Equations that calculate L and C values are derived for a perfect match. The angular frequency (ώ) is called the cross-over. A familiar example is the cross over used in audio systems. The network that splits signals is a diplexer. Here is an example where both outputs are used. Another form of diplexer is the band-pass/band-stop combination. This is shown in Figure 4. 3:

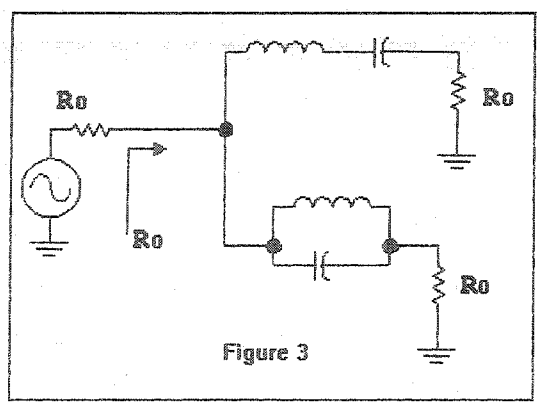


Figure 4.3: Diplexer with Band Pass / Band Stop Combination

The Band-Pass / Band-Stop combination allows us to choose the path for certain frequency of incoming signal. In our case, we have two signals RF and IF; so, we will modify this design so it can direct two frequencies. In this case, single path will have to pass one frequency and reject the other. On the other path, it will do the same thing, but with opposite frequencies. The resulting circuit of this design is shown in Figure 4.4.a and the simulation result of the circuit is shown in Figure 4.4.b.

The diplexer has the function of combining and separating the composite signal from/into its constituent parts, to permit both frequencies to be transmitted through the LNA simultaneously. The characteristics of the diplexer are different from the responses of two individual filters. The main problem that one has to face when designing the diplexer is mutual interaction of the two filters composing the diplexer. In our design, we had two diplexers: one before the amplifier and one after the amplifier.

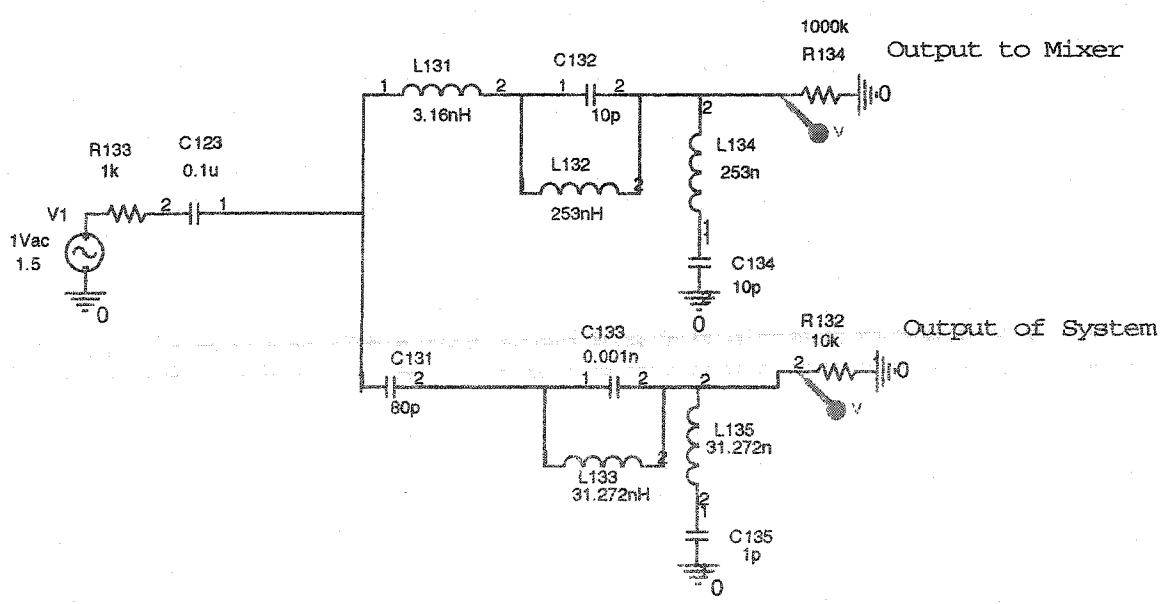


Figure 4.4.a: Diplexer circuit used at LNA output.
*A similar circuit is used at the LNA input.

Figure 4.4.a shows the basic design of the diplexer as used at the LNA output. This diplexer takes the signal from LNA and splits it into two signals. The top branch passes 900 MHz but presents an open circuit to the LNA at 100 MHz, so that only 900 MHz arrives at the mixer.

The bottom section passes 100 MHz and is open for 900 MHz, so only the 100 MHz arrives at the output of the entire circuit. The first section of diplexer is used for impedance matching. The second section has two paths, each one acts as an open circuit for the undesired frequencies in each path: 100MHz in the top and 900 MHz in the bottom. The third section in each path is a shunt series resonant to short 900 MHz in the top path and 100 MHz in the bottom path, thus providing further rejection. The diplexer frequency response is shown in Figure 4.4.b.

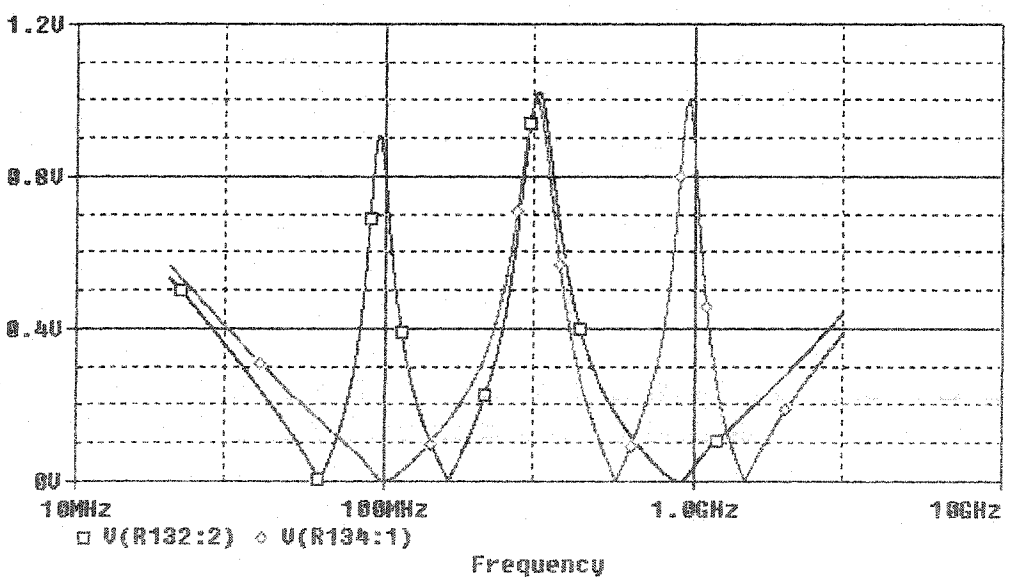


Figure 4.4.b: Diplexer Frequency Response

Three peaks are visible, those at 100 MHz (IF) and 300 MHz for the lower branch and 900 MHz (RF) and 300MHz for the upper branch. The 300 MHz response is spurious

but, since 300 MHz is not one of the product frequencies of the mixer, the response does not affect the circuit operation.

A similar configuration is used on the input of the LNA. This diplexer is the mirror image of one shown in Figure 4.4.a. It takes two signals: the output from the mixer stage at 100 MHz and the antenna input at 900 MHz and combines them as the single input to LNA.

As in the case of the tank circuit, one of the main problems of the circuit is the mutual interference and, as in the previous case, the problem is solved by slight adjustment of the L and C values, until the desired center frequencies match the values in simulation.

CHAPTER 5

Mixer

A mixer, or frequency converter, converts the signal from one frequency (RF) to another frequency (IF). The choice of the mixer used in design was based on the application. Passive mixers can have advantages in the form of dynamic range, IP3 point, and power consumption in the mixer itself; however, the trade-off is in conversion loss and increased power from the local oscillator (LO) and conversion loss. Active mixers achieve conversion gain and need less LO power, an important consideration in low voltage applications.

One of the simplest active mixer designs is the unbalanced mixer in which output is taken from only one side of the differential pair. The problem with this construction is that RF signal appears at the output (RF feedthrough), which is undesirable for our application.

The next level of design would be single balanced mixer in which output is taken differentially from both branches of the differential pair. In this manner, we eliminated the RF throughput, but it still has the LO throughput. This is seen in Figure 5.1. This structure is also reversible which means that RF and LO injection points can be interchanged. The circuit will still perform as the single balanced mixer, except it will reject LO this time and have RF feedthrough. The resulting circuit is seen in Figure 5.2.

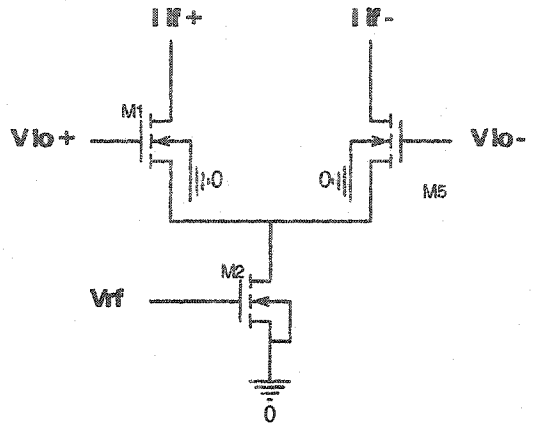


Figure 5.1: Basic single balanced mixer configuration

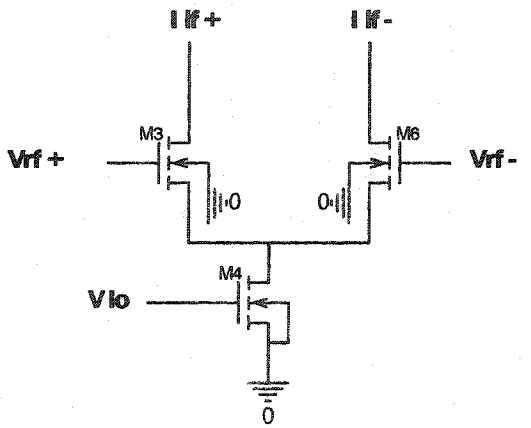


Figure 5.2: Basic single balanced mixer with IF and RF reversed

Next logical design step is combining two single balanced transistors. The result is a double balanced mixer, also called quad mixer or a Gilbert cell mixer. Gilbert cell mixer ideally eliminates both RF and LO feedthrough.

How does the Gilbert Cell actually work? Using the Figure 5.3 we will explain the main principles of operation.

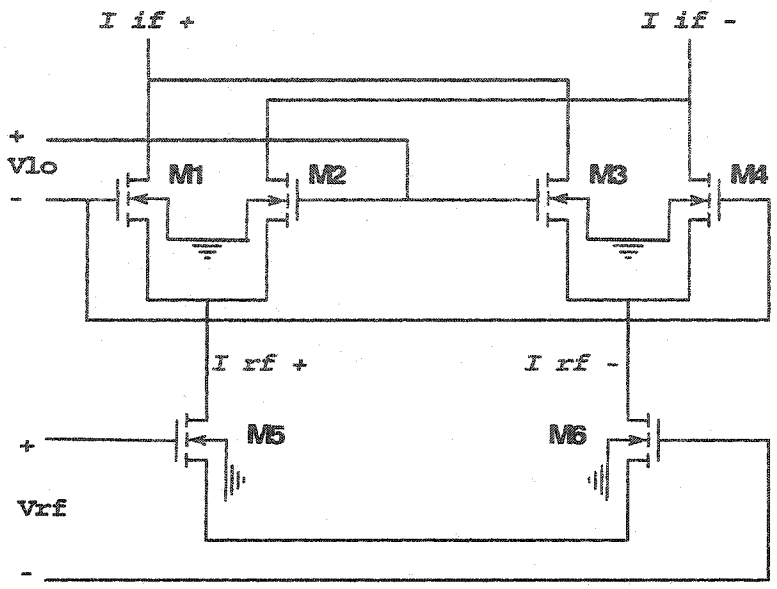


Figure 5.3: Double Balanced / Gilbert Cell Mixer

For this example, we will assume that pulse wave is applied to the LO with amplitude high enough to turn transistors on. During the positive cycle (N channel), transistors M2 and M3 are on. During this, M1 and M4 are completely off. At the same time, $I\,if+$ is equal to sum of the currents from M1 and M3 and that is equal to:

$$I\,if+ = 0 + I\,rf-$$

Now, when pulse switches to the negative part of the cycle, transistors M3 and M6 will be on, and $I\,if+$ is still equal to sum of currents from M1 and M3; but, this time,

$$I\,if+ = I\,rf + 0.$$

The same procedure can be used to trace the $I\,if-$; so, we can see that advantage of double balanced mixer is that we doubled the efficiency in comparison to the single

balanced mixer. This means that conversion gain (Gc) is double the Gc of single balanced mixer and four times the Gc of the unbalanced mixer.

Consequently, a doubly-balanced mixer is necessary in this application, in order to reduce the number of spurious frequency components at the mixer output, which would complicate the design of the diplexers. After testing several different active mixer configurations, the best performance was achieved with the CMOS Gilbert Cell configuration seen in Figure 5.4, where bias transistors and voltage followers are added to bias the circuit and improve the performance [3] of the mixer.

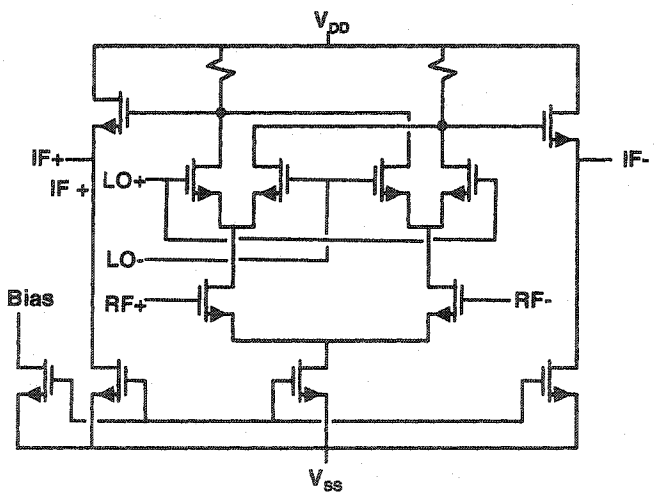


Figure 5.4: Gilbert Cell with bias circuitry

After determining the optimum and minimal bias of the mixer, results are measured and recorded in table 5.5.

Simulation of the output in time domain and in the frequency domain are shown in Figure 5.5.a and 5.5.b for the RF of 900 MHz and LO of 1 GHz.

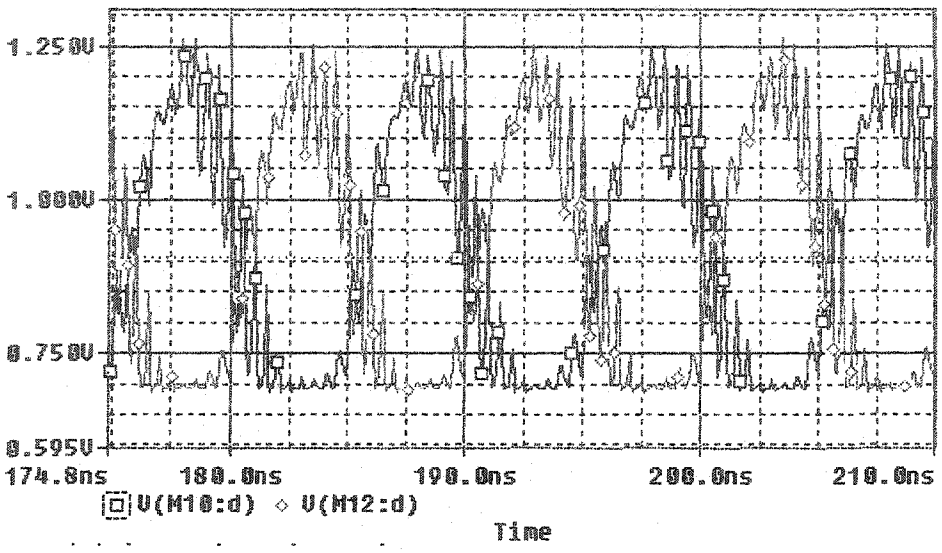


Figure 5.5.a: Time Domain Response of G.C. Mixer

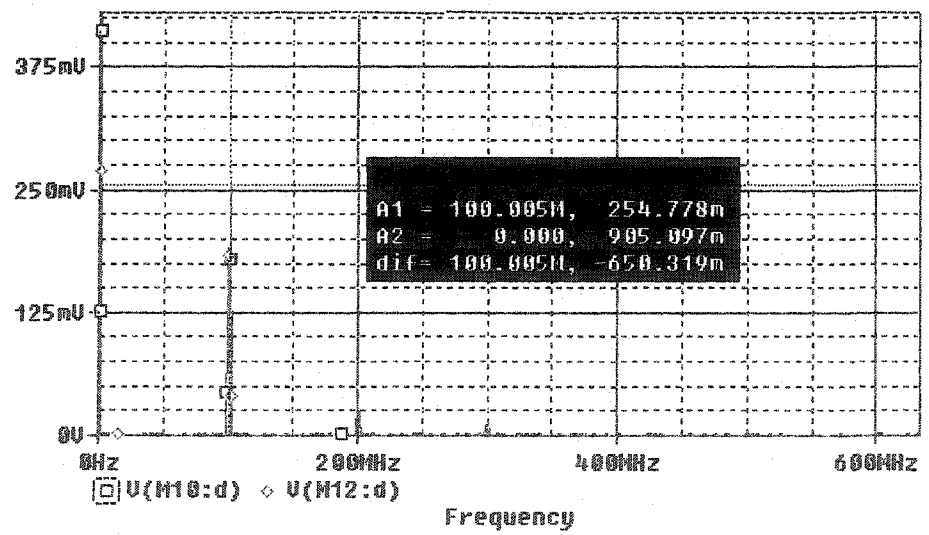


Figure 5.5.b: Frequency Domain Response of G.C. Mixer

Gain of the mixer was initially measured according to the formula derived by Sullivan [2]:

$$Gc = \frac{4}{\pi}(g_m \times R_L) \qquad (2)$$

After this general formula is used to determine the exact gain

DaSilva [6]

$$Gc = \frac{Pdel.load}{Pavil.source} = 10\log_{10}\left(\frac{\frac{V_{out}^2}{R_{load}}}{\frac{V_{in}^2}{4R_{in}}}\right) \quad (3)$$

Obviously, in order to optimize gain, we had to find the optimum input and output impedance and then maximum gain. For that reason, table 5.4.a and 5.4.b show the results that were tested in order to optimize the impedance levels.

Load Resistance *Ohms*	Output Voltage *mV*	Output Power *mW*	Power *dB*
100	17.50	1400.00	3.146
200	34.50	1380.00	3.140
300	48.80	1301.33	3.114
400	66.80	1336.00	3.126
500	85.20	1363.20	3.135
600	100.60	1341.33	3.128
700	117.50	1342.86	3.128
800	130.90	1309.00	3.117
900	147.60	1312.00	3.118
1000	161.50	1292.00	3.111
1100	172.90	1257.45	3.099
1200	187.90	1252.67	3.098
1300	197.40	1214.77	3.084
1400	212.70	1215.43	3.085
1500	220.10	1173.87	3.070
1600	231.01	1155.05	3.063
1700	240.60	1132.24	3.054
1800	252.80	1123.56	3.051

Table 5.4.a: Output power for different loads of the Gilbert cell Mixer

MIXER

Specifications	*0.18 micron*	*0.6 micron process*
Conversion Gain	16 dB	16 dB
NF	9 dB	
Power	8.7mW	9.7mW

Table 5.4.b. Mixer Specifications

As we can see from the Table 5.4.a, there is not much change in the gain as the output impedance changes. One of the things that we can notice is that as the resistance lowers the conversion gain increases. Output impedance needs to be matched to the load and on these frequencies it is common to see impedance ranging from a few hundreds to several kilo-ohms. From the table we decided to use 200 Ohms resistance. This way we can get relatively high resistance and gain at the same time.

CHAPTER 6

Complete Reflex Simulation

Connecting the circuit blocks together to form a complete reflex circuit was not a straightforward assignment. Individual parts had to be matched and the circuit optimized based upon the results. The process of joining the circuit was done in several stages:

- Output diplexers joined to the Mixer
- LNA connected to the diplexers and mixer
- Input Diplexer Joined to the LNA, Output Diplexer and Mixer
- Loop Established between mixer and input diplexers
- Additional circuitry added to suppress the oscillations

The design was tested as individual stages, as described previously; then, they were combined into sub-blocks prior to assembling the entire RF front end.

After designing the mixer, the diplexer was added and the mixer and diplexer were tested together to resolve the matching issues and observe the total gain. Then LNA and second diplexer were added to the input of the circuit. The resulting configuration is shown in Figure 6.1.

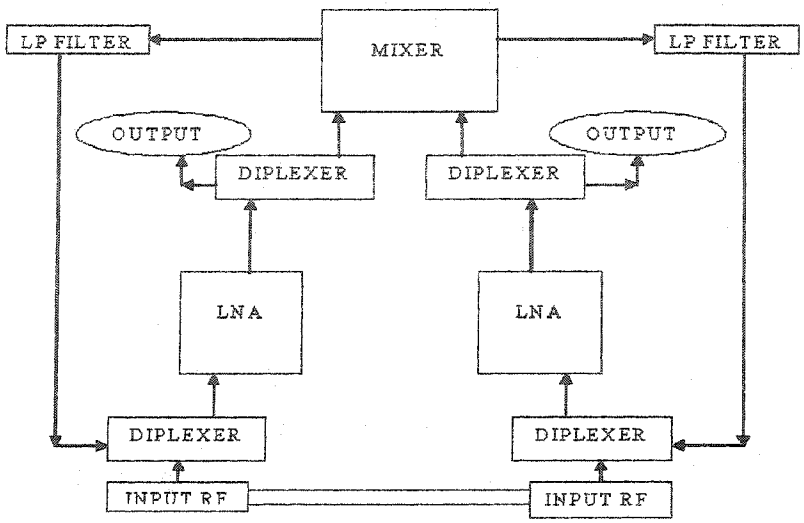


Figure 6.1: Detailed block diagram of the reflex circuit

The circuit is differential throughout, with the LNA split into left and right sections, each with their own diplexers.

A simple RC low pass filter (f_0 = 250 MHz) was introduced into the feedback circuit from the mixer output back to the LNA input. This served as compensation to eliminate undesired oscillation that occurred when we connect the feedback circuit back to the input of the diplexer.

The resulting signal spectrum coming from this setup is shown in Figure 6.2, under large signal conditions [324 mV for IF and 175 mV for 900 MHz]. Figure 6.3 shows the time domain response of same IF signal. FFT simulation shows that the second and third harmonics are suppressed and that no significant spurious signals are present. This forms low harmonic, intermodulation distortion, and stability in the feedback loop by the reflex arrangement.

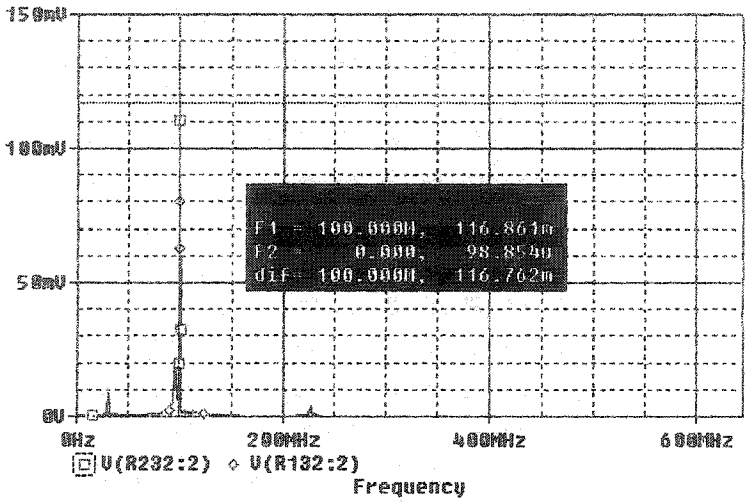


Figure 6.2. Frequency domain output of reflex front-end.

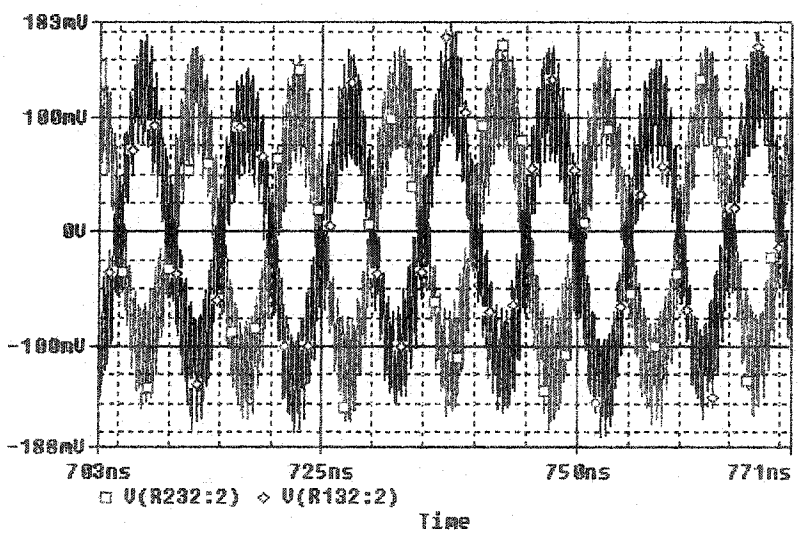


Figure 6.3. Time domain output of reflex front-end circuit

After putting the pieces together and optimizing the parameters we measured the characteristics shown in the Table 6.0

REFLEX Rin=1k / RL=10k

Specifications	*0.18 micron*	*0.6 micron process*
Conversion Gain	37 dB	26 dB
NF	6 dB	5.7 dB
Power Consumption	14mW	42mW

Table 6.0: Performance Characteristics of Reflex Receiver

	Receiver	Frequency	Norm. Sensitivity	IIP3	Power Diss.	Technology
[25]	Philips UAA2082	900 MHz (POCSAG)	-117 dBm	-27 dBm	6.6 mW, 2V	Bipolar
[26]	Plessey SL6610	470 MHz (POCSAG)	-123 dBm	-39 dBm	3.4 mW, 1.3V	Bipolar
[27]	Wireless Access	900 MHz (ReFLEX)	-122 dBm	-43 dBm	24 mW, 3V	27 GHz BJT 1.2μm CMOS

Table 6.1: Performance Characteristics of some related BJT receivers

One of the most important points of analog design is linearity or the dynamic range of the system. Regarding the dynamic range of a receiver, two definitions can be applied: spurious-free dynamic range (SFDR) and blocking dynamic range (BDR) [7] (Figure 6.1). SFDR is the input signal range from the noise floor up to the input power that creates intermodulation products equal to the noise power. BDR is the input power range from the noise floor up to the 1-dB gain compression point (P_{-1dB}) [7]. Intermodulation products are undesired harmonics arising from the non-linearity of the receiver components [8], including the LNA and the mixers. In most RF receivers, third-order intermodulation dominates and the corresponding distortion is specified by the third-order intercept point (IP3). IP3 is the intersection point of the fundamental frequency component gain curve and the third-order harmonics gain curve (see Figure 6.4). In homodyne systems, even-order distortion can be severe [5] and the second-order intercept point (IP2) is also

specified. The 1-dB gain compression point is the input power which compresses the linear gain (fundamental frequency component) by 1dB. Knowing this, we can move on to finding our IP3 point.

The procedure for deriving the IP3 point of this reflex amplifier is as follows:

1. Create a normal gain curve of the reflex system using a RF frequency of 900 MHz and IF of 1 GHz.
2. Replace the 900 MHz RF with two sources 890 MHz And 880 MHz and use those to find the output gain of the 110 MHz component (Figure 6.3).
3. Use this component to find 3rd order curve
4. Extend linear pats of both curved and find intersection point which is the IP3 point.

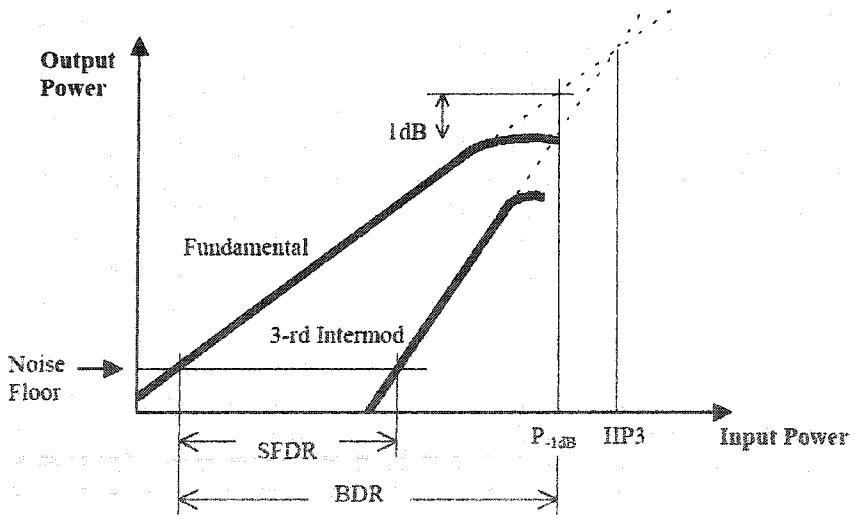


Figure 6.4: Normal and 3rd order curves explained

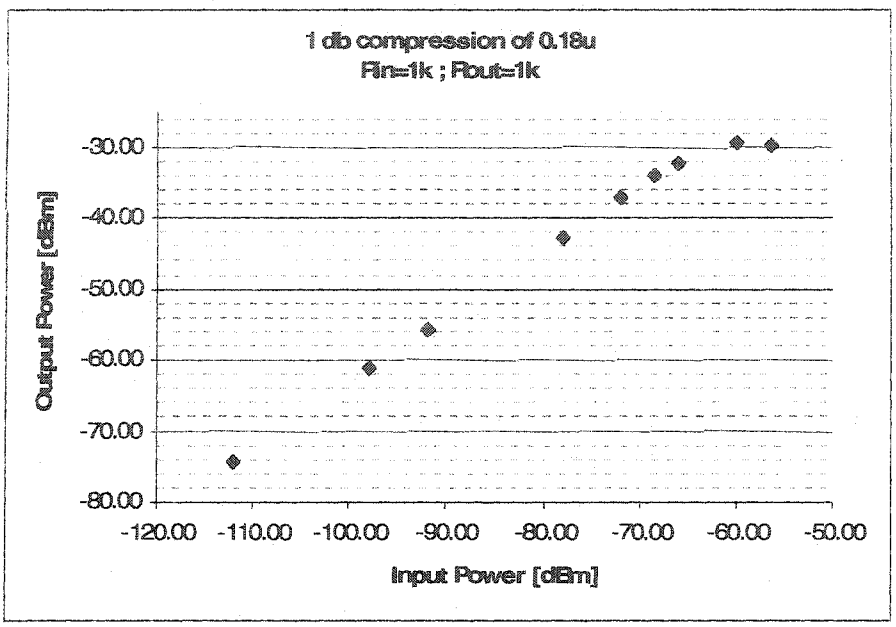


Figure 6.5:
Normal Curve of the 0.18u process used for 1dB compression point calculation

Figure 6.2 shows the graph of output vs. input power. We know that gain of the system is calculated using the formula:

$$G_c = 10 \log \left(\frac{\frac{V_{out}^2}{R_l}}{\frac{V_{in}^2}{4 \cdot R_{in}}} \right) \qquad (4)$$

Using Or-Cad simulation tools and FFT, we can generate all the points shown in Figure 6.2 and estimate that the 1db compression point occurs at about -30 dB. This means that, after this point, our power gain is not linear. The procedure was repeated for 3rd order curve, except two frequencies are injected 880 and 890 MHz and the difference component of 110 MHz is observed as seen in Figure 6.3. Here, we have to use simulation as accurate as possible: accuracy of the simulation is directly proportional to the length of

the simulation. The simulation that produced Figure 6.6 had the file size of 1Mb and it took about 45 minutes to execute on Pentium IV 2.8 MHz processor.

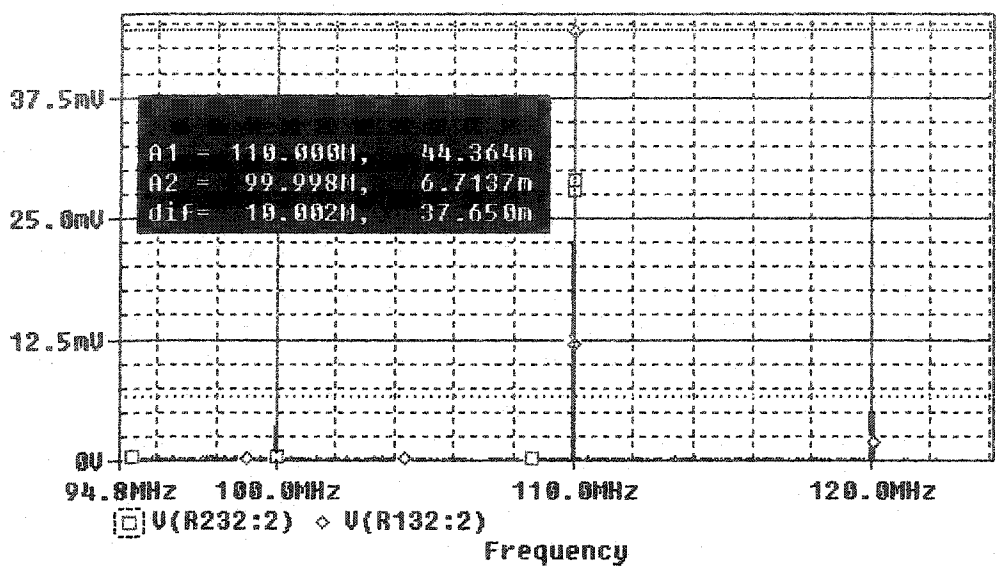


Figure 6.6: FFT simulation of the circuit used for IP3 calculation

Given the simulation from Figure 6.6, and knowing that the slope of the third order curve has to be three, we can set up the following system of equations:

$y=x+41$ ---------normal

$y=3x+164$ ---------3rd order.

Now, to find the point of interception at the input, we have to solve the system above and determine the input and output gain using the formula (4) followed by simple number to dBm conversion. Figure 6.7 illustrates these two lines/curves and approximates the IP3 point to be about -20 dBm at the output.

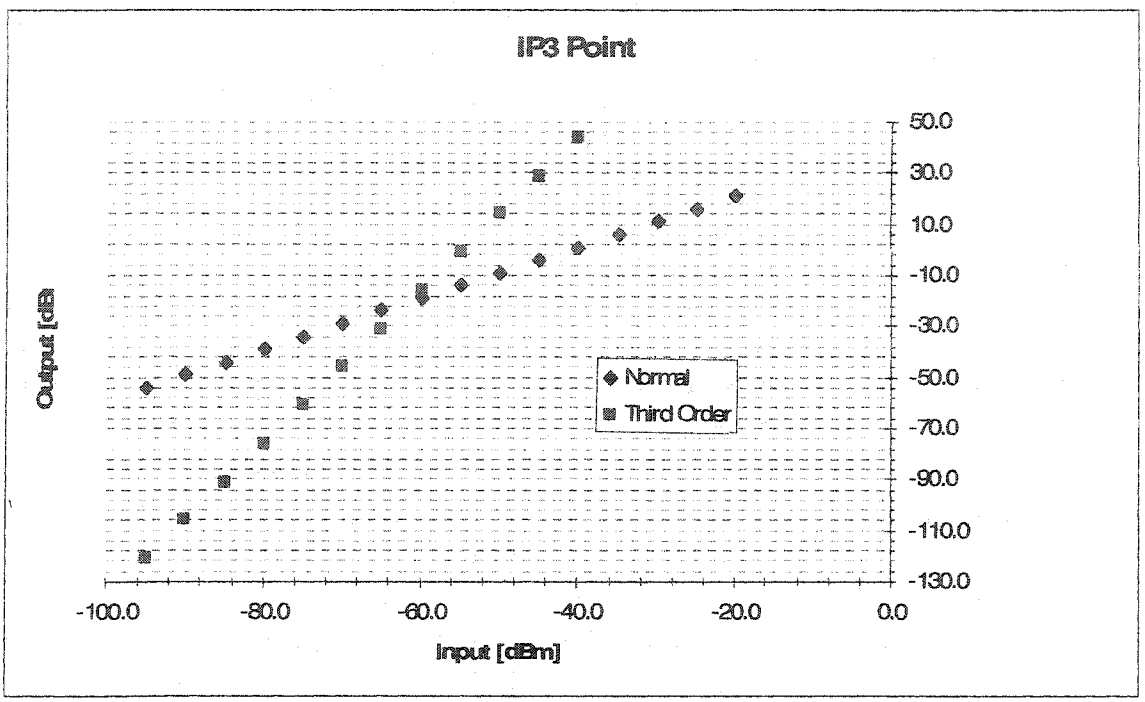


Figure 6.7: Crossing of normal and 3rd order curve as IP3 point

For complete set of values used for Figure 6.4 as well as the all the calculations refer to Appendix D.

CHAPTER 7

Self-Oscillating Mixer

As an improvement on the existing design, we decided to try to further improve the mixer. This is done by eliminating the power needed for the LO by making the mixer oscillate on its own. Basically, we are merging two circuits together, LO and Gilbert Cell Mixer, instead of stacking them on top of each other. The design of the circuit was taken in several stages:

- The first stage was to build a basic oscillator and make sure that we could apply this to our design.
- Several oscillators were tested and finally the following structure, shown in Figure 7.2a, was adopted.
- After unsatisfactory results with regular feedback oscillators, we decided to utilize the Colpitts feedback oscillator that will connect drain back to the gate.

The frequency of the oscillation of the Colpitts oscillator and basic colpitts configuration is given in Figures 7.1a and equation (5).

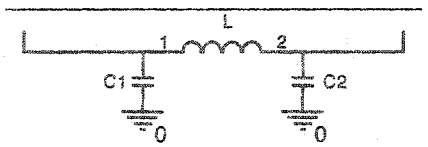


Figure 7.1a: Schematic of Basic Colpitts Feedback Configuration

$$f_{osc} = \frac{1}{2\pi\sqrt{L\left(\frac{C_1 C_2}{C_1 + C_2}\right)}} \qquad (5)$$

This basic formula can be used for our differential amplifier with small changes. Instead of C_2 we have combination of C_2, C_3 and C_{gs} from the transistor. So, the new equation (6) has the form:

$$f_{osc} = \frac{1}{2\pi\sqrt{L_1\left(\frac{C_1\left(C_3 \| (C_2 + C_{gs})\right)}{C_1 + \left(C_3 \| (C_2 + C_{gs})\right)}\right)}} \qquad (6)$$

Because our Gilbert Cell Mixer is made of the two differential amplifiers for the LO signal, we will have to incorporate two of the differential oscillators shown in Figure 7.2.a. How much of the interface will the oscillators have and how precise does the frequency lock need to be? A system has to oscillate as clean as the original single differential pair, whose time and frequency domain outputs are shown in Figure 7.2.b and 7.2.c

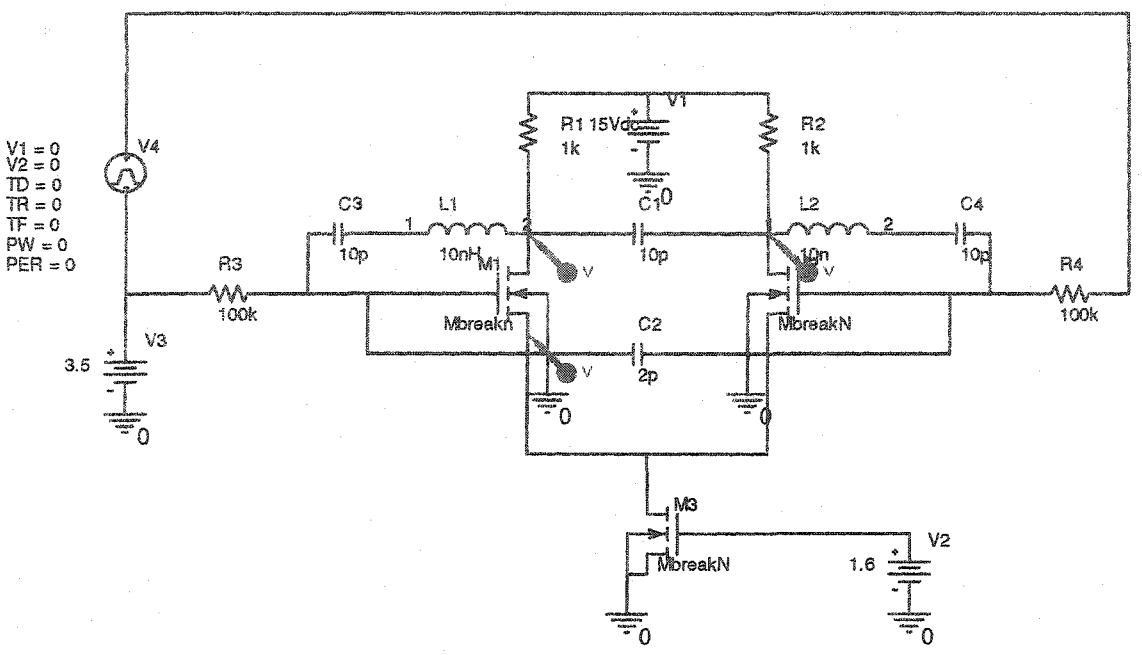


Figure 7.2.a: Differential Oscillator

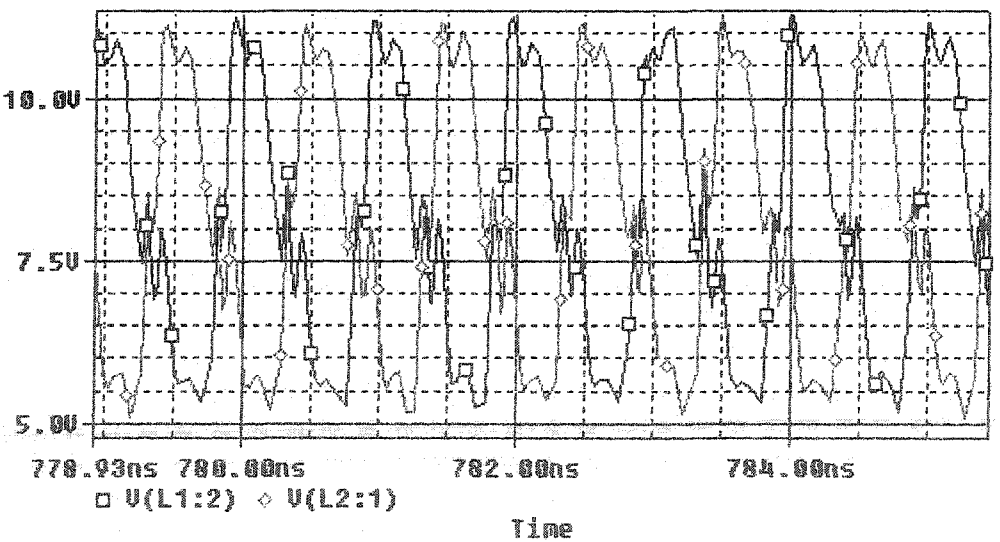


Figure 7.2.b: Time Domain Simulation of Differential Oscillator

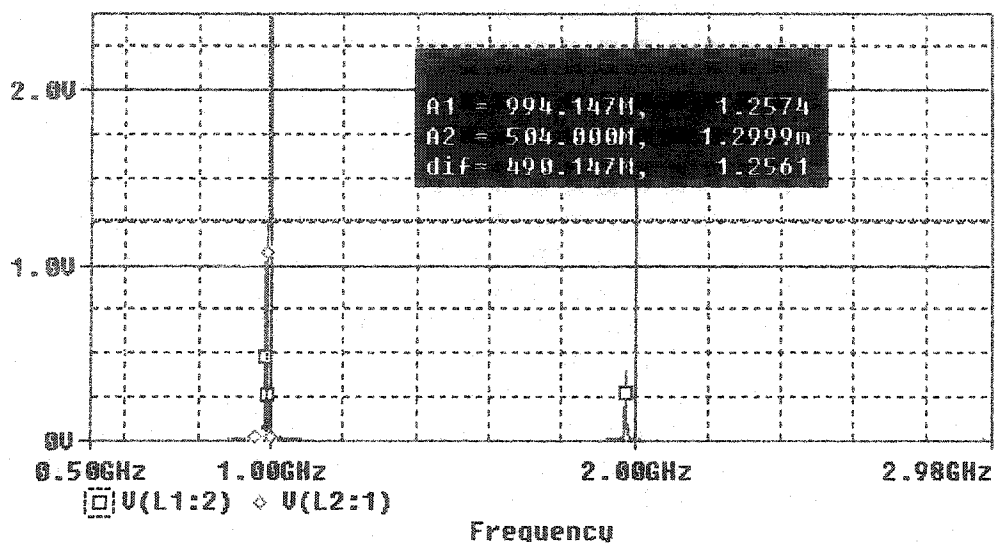


Figure 7.2.c: FFT of Differential Oscillator

Now, in order for this system to be effective, we had to make one more adjustment. The adjustment was caused by differential system inability to oscillate without the initial stimulation in the form of a simple pulse or similar wave. This problem was solved by incorporating the "quadrature topology" [11] shown in Figure 7.3

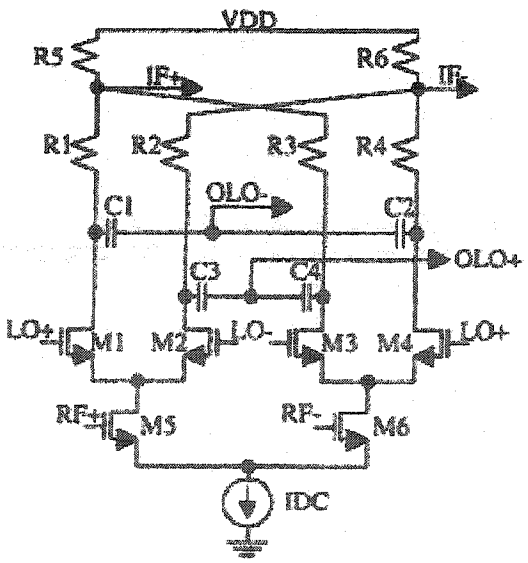


Figure 7.3: Quadrature Self Oscillating Configuration
(taken from [11])

The technique for achieving self oscillation is based on observation of the sums of drain currents for transistors M1 and M4. RF signal and product of RF and LO inputs cancel, leaving only the LO component. This LO output component can then be fed back to the LO input port, in order to produce the self oscillation. In Figure 7.3 resistors R1 and R4 isolate the intermediate frequency (IF), RF and LO output signal components at the drain of transistors M1 and M4. The two pairs of capacitors formed by C1 through C4 sum the in-phase LO components and cancel the out-of-phase RF and IF components.

In our design, we modified the capacitor system C1 – C4 and replaced it with Colpitts oscillators previously described. Results of this transformation are circuits shown in Figure 7.4 and 7.5.a.

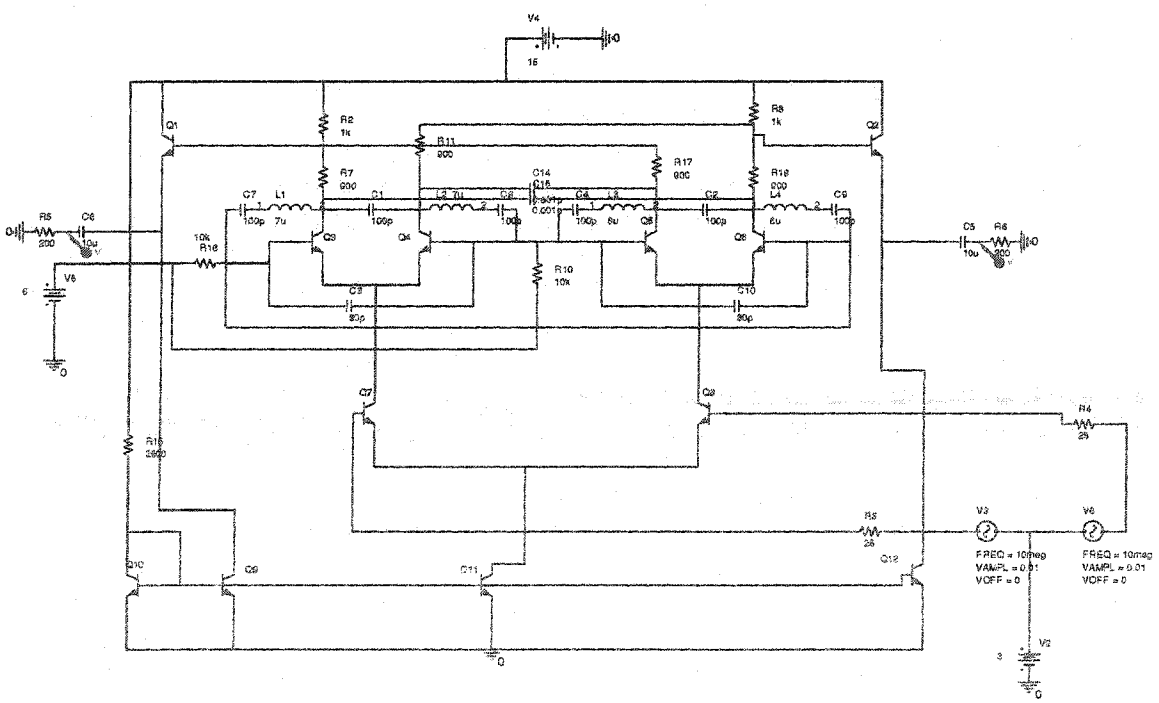


Figure 7.4: Self oscillating Mixer using the BJT Technology

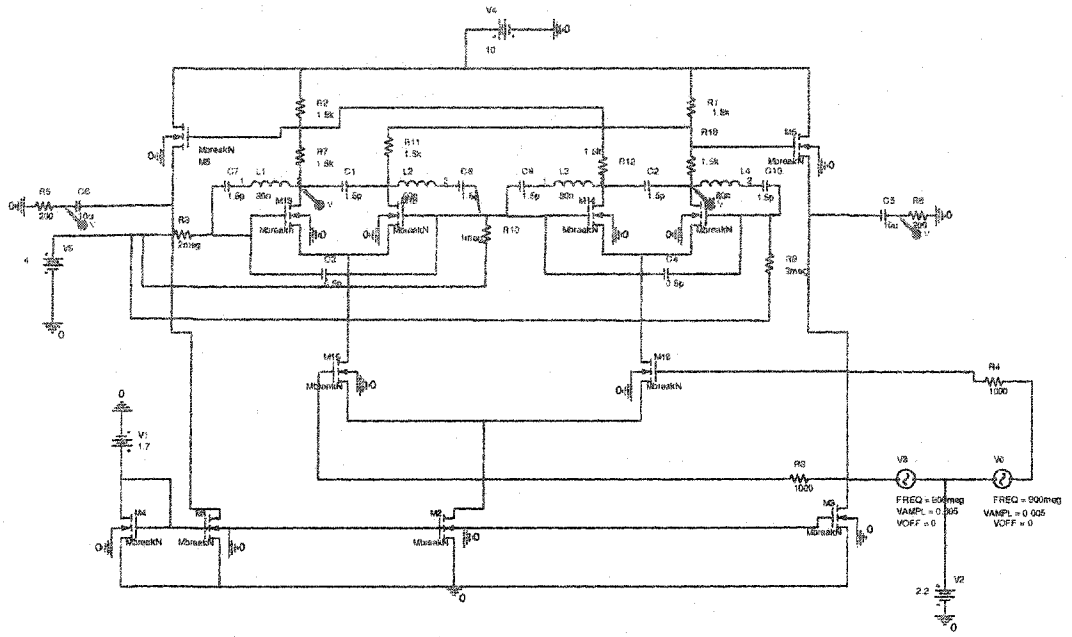


Figure 7.5.a: Self Oscillating Mixer using the CMOS Technology

Figure 7.5 b proves to us that the circuit is oscillating and that two outputs are out of phase. This is very useful because it will cancel the unwanted frequencies and, if output is taken differentially, the amplitude will double.

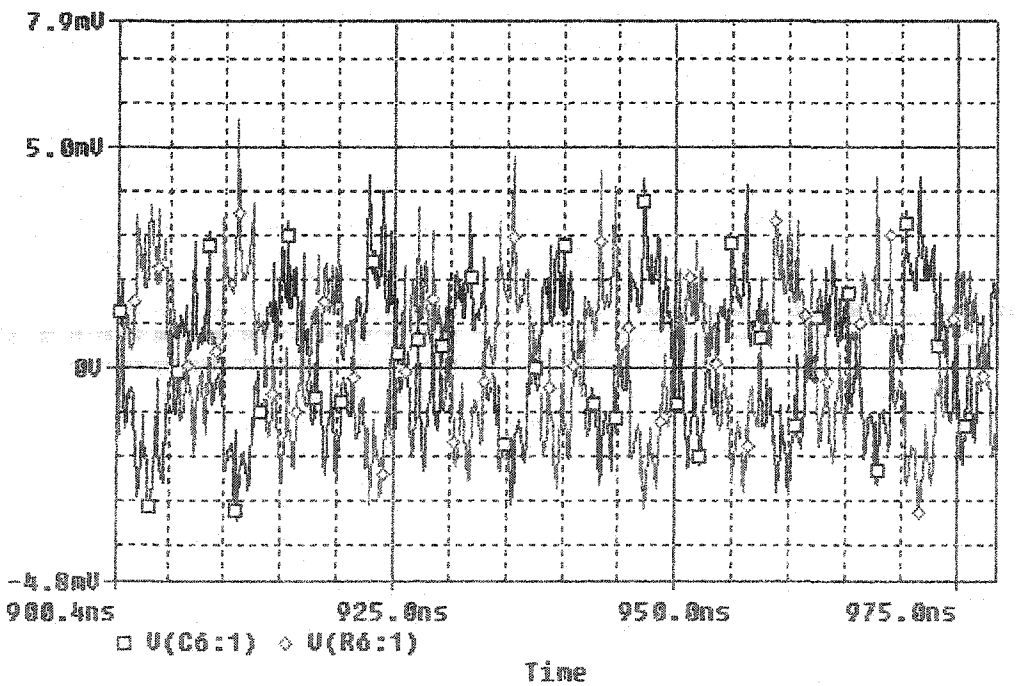


Figure 7.5.b: Time Domain simulation of the CMOS – SOM

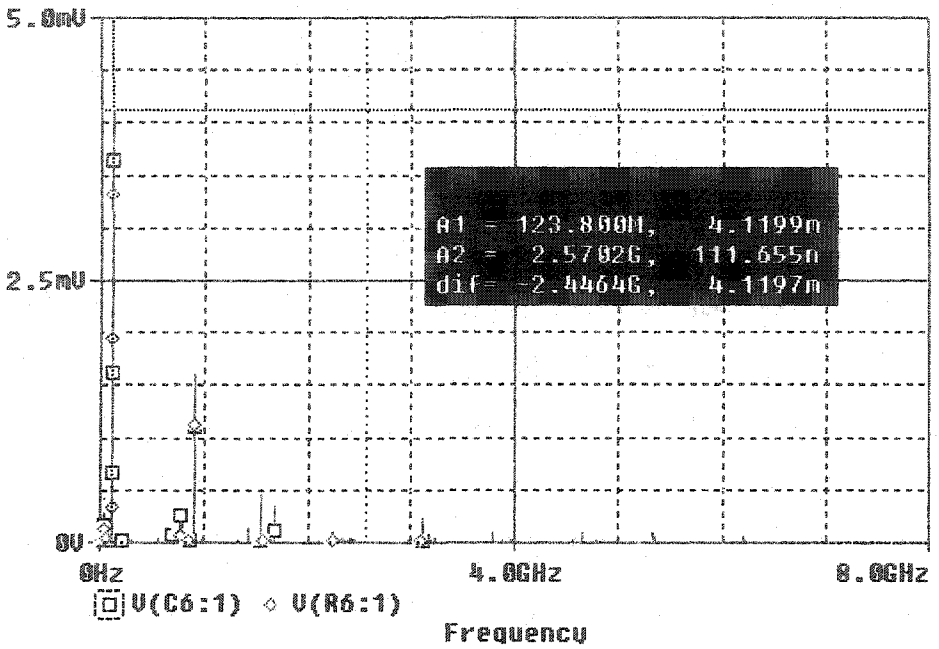


Figure 7.5.c: FFT Simulation of the CMOS – SOM

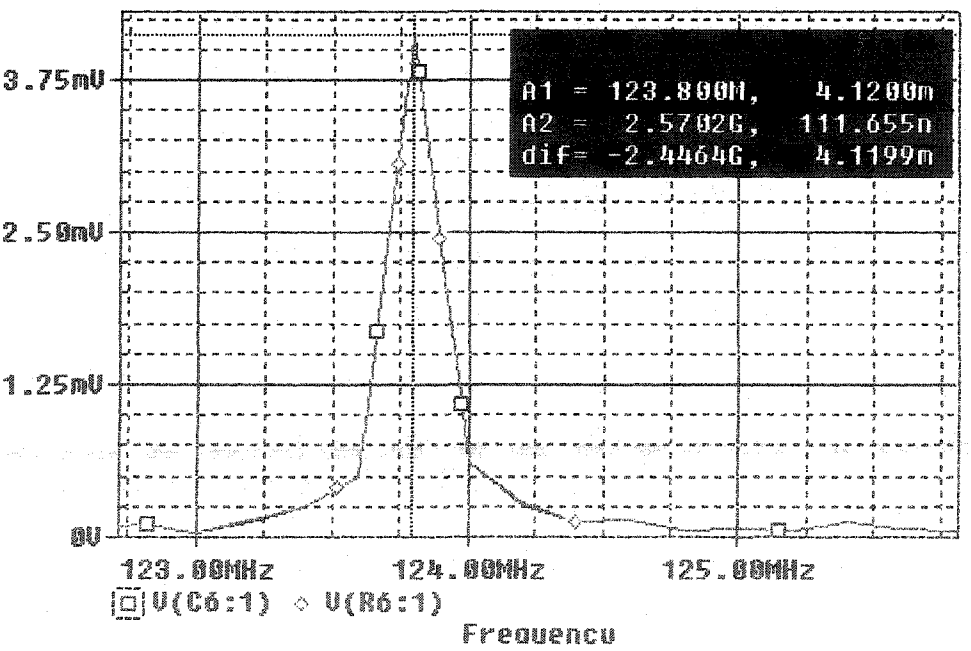


Figure 7.5.e Enlargement of the CMOS SOM 128 MHz component

that demonstrates the gain of the 5.31 dB

In this case, resolution is very important because it can halve the gain value, if the simulation is shorter then 5 ms.

IP3 Calculation

In order to perform the IP3 calculation, we had to alter our circuit and see the output when two frequencies are applied. In our case, we applied RF frequencies of 890 and 895 MHz and self –LO of 760 MHz, which gave us the 146μV output, while the input was still 10mV.

Before we can define IP3 point, let's find out what is 1 dB compression point. For that purpose, as seen at table 7.0, we varied the input of the SOM until our response was not linear anymore. From Figure 7.6, we can extrapolate the value is about -16 dBm.

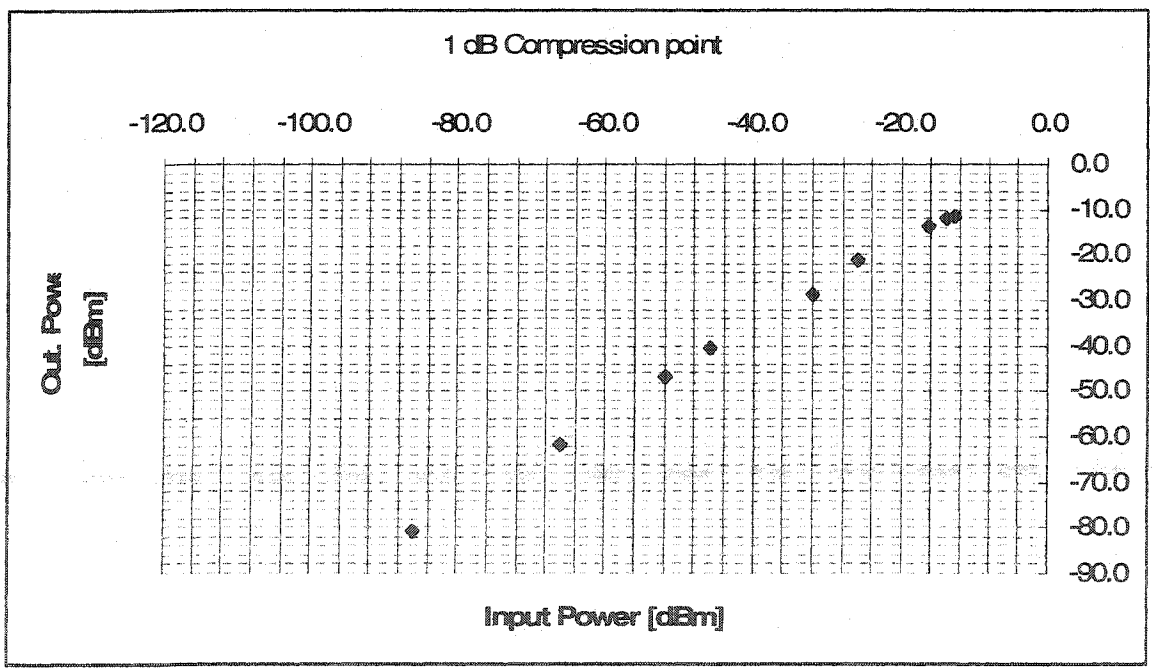


Figure 7.6: 1 dB compression point of SOM is ~ **-16 dB** at Output

The same situation can be more accurately read from the table 7.0, where the last good value is ~ 10 mV. After this value, the gain drops by about 2 dB for input of 3 mV (greater then 1 dB).

Normal Response of SOM with LO of 766 MHz			
Input	Input		Output
900MHZ	900MHZ		124MHZ
[Volts]	[dBm]		[dBm]
0.0001	-86.021		-80.8184
0.001	-66.021		-61.8602
0.005	-52.041		-46.9110
0.010	-46.021		-40.7546
0.050	-32.041		-28.7373
0.100	-26.021		-21.3493
0.300	-16.478		-13.5687
0.400	-13.979		-12.0854
0.450	-12.956		-11.4784

Table 7.0: Table of input vs. output values used for normal curve

Once we determined normal curve, we can move on to finding 3rd order curve. Figures 7.7a and 7.7b illustrate the output created by combining all the frequencies. This is the basic calculation of the process:

RF = 890 MHz LO = 766 MHz → IF = 114 1st order

RF = 895 MHz LO = 766 MHz → IF = 119 1st order

IF = 124 3rd order

IF = 109 3rd order.

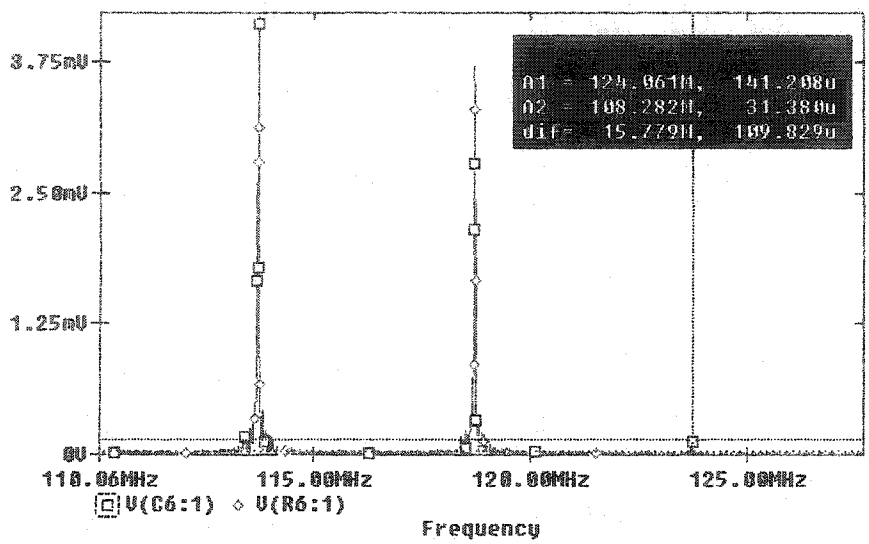


Figure 7.7a: FFT response of 2 RF inputs showing normal and IP3 components

Now, we can all see there is a slight difference in the amplitude of the main and 3rd order curve. This difference is due to the slight imbalance of the transistors.

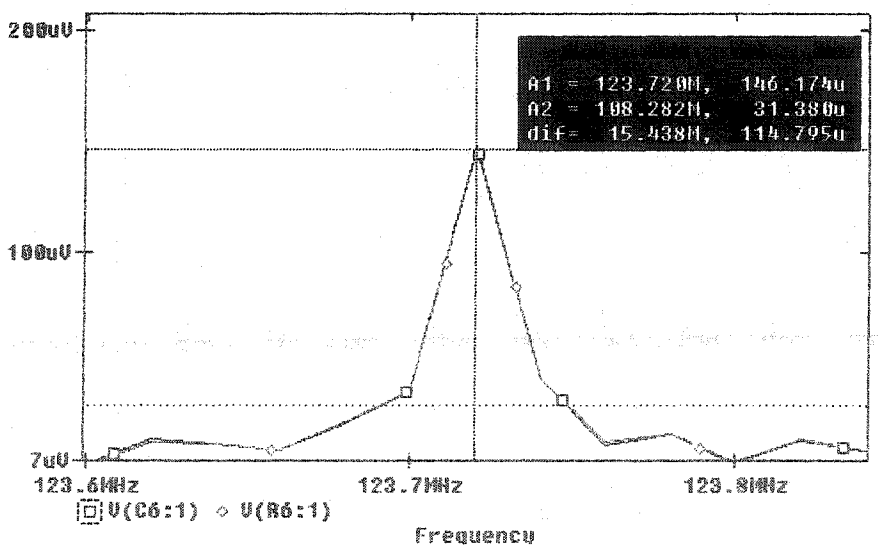


Figure 7.7b: 3rd order component enlarged

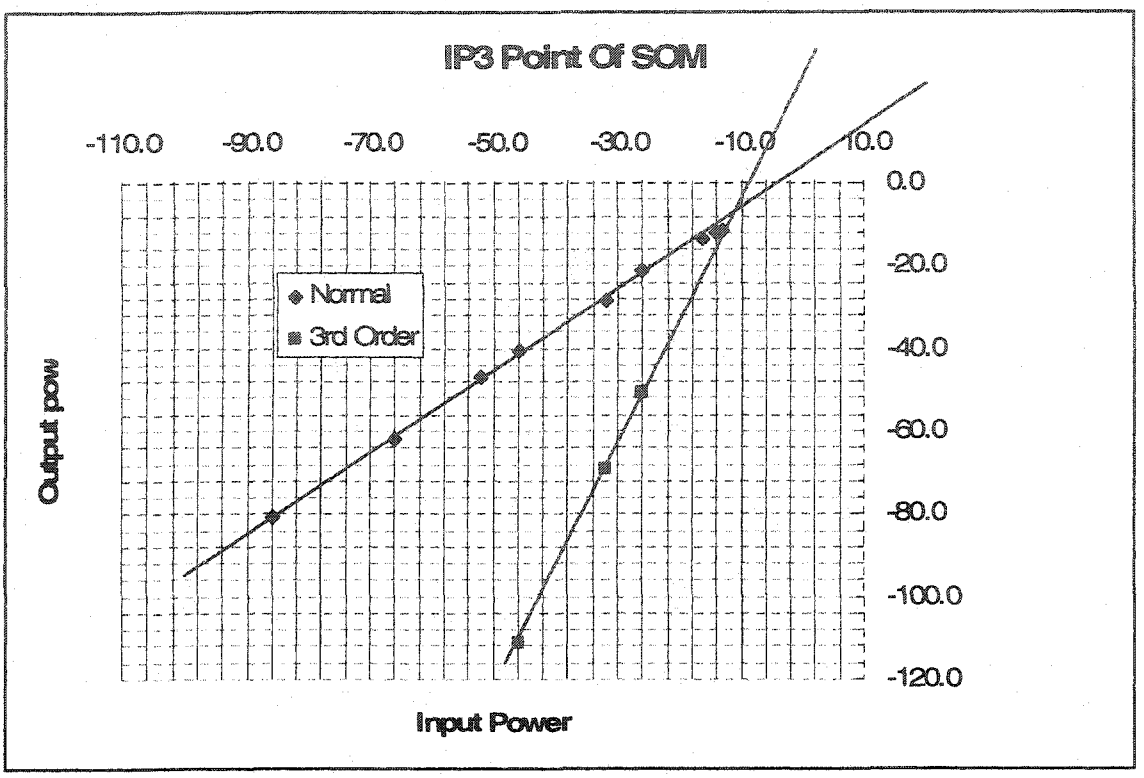


Figure 7.9*:* IP3 Point of SOM is approximately **– 5 dB** at output

Table 7.10 shows a summary of the results that were previously calculated and measured.

VDD	**10 V**
Power Dissipation	**240 mW**
Self Oscillating Frequency	**900 MHz**
Conversion Gain	**5.31 dB**
1 dB Compression Point	**-16 dBm**
IP3 Point	**-5 dBm**

Table 7. 10 - Measured and calculated results of the SOM

Just for the comparison, we will show the table of the values for a related design from the literature. [9]

VDD	3.3 V
Power Dissipation	60 mW
Self-Oscillation Freq.	1.2 GHz
Tuning Range	60 MHz
Conversion Gain	-1 dB
1 dB compression point	-9 dBm
IIP3	-11.5 dBm
Noise Figure	20 dB

Table 7.11: Table of measured values for the Quadrature mixer [9]

CHAPTER 8

Implementation

As the circuit was developed, VHDL implementation had to be done so that a circuit can be manufactured and made into the system. First, we had to choose the software for the layout. We used several software tools in the development stage and a large majority of the design was developed using two major tools: Electric and, the most important, Cadence. We used Electric to get familiar with basic procedures of the layout. We used knowledge acquired to create the final circuit layout in Cadence. For testing purposes, before we designed all the layouts, we tried to implement our circuit using regular "off-shelf " components. So, we took the self oscillating mixer and implemented it using 7.2 MHz as the RF frequency. The resulting circuit is shown in Figure 8.1.

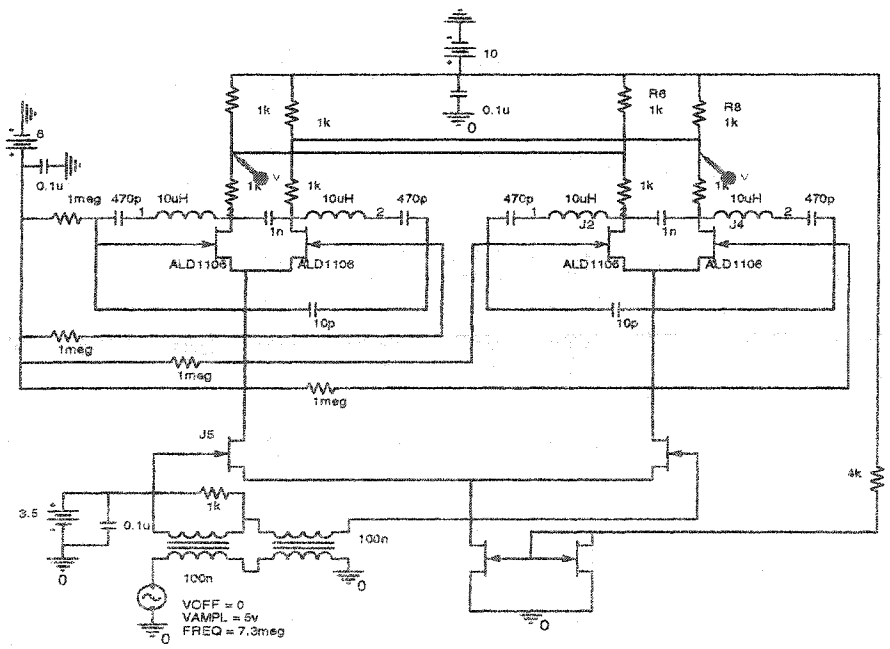


Figure 8.1.a: Schematic of Circuit used for test

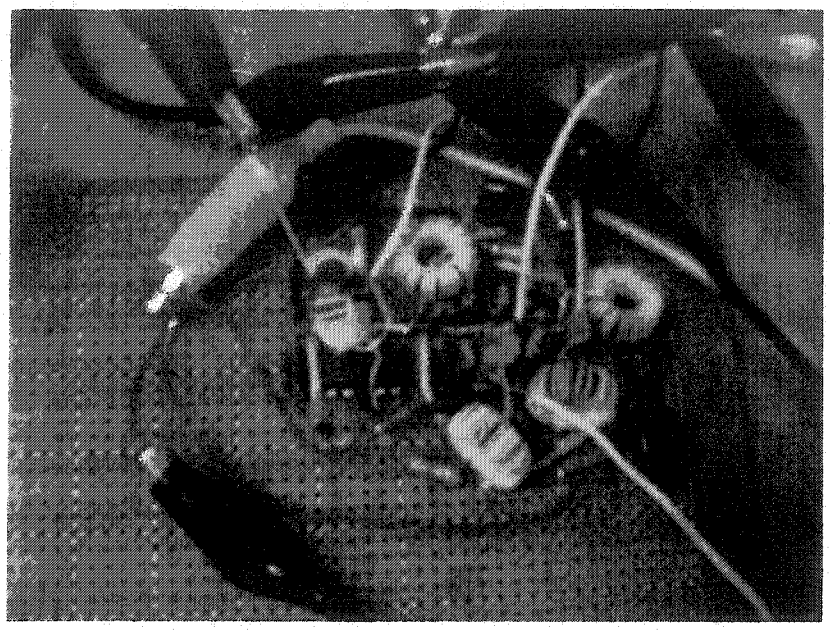


Figure 8.1.b: Actual SOM Circuit used for testing

This circuit will be set up to self oscillate on 1 MHz, with the IF frequency of ~6MHz. The circuit does not have great gain, but that is due to transistors that we use. In the VLSI design, we can change the width of the transistors and that way increase the current flow which will improve our gain.

An FFT of the output was captured using a digital oscilloscope. This circuit works properly and, most importantly, it shows that our simulations are valid and gives us more confidence in continuing to develop the design.

In order to draw the circuit, we have to resolve and learn some basic characteristics of the component drawing. One of the most important parts of drawing is the transistor. The reason we do this is to minimize the area and the design and implementation errors due to resistances, capacitances, or anything similar.

When designing a transistor, we have to pay attention to several elements of design [23]. The first is *fingering*. In order to place a transistor on the cell that is smaller than normal

transistor width, we use a technique called fingering. The normal transistor width can be 100 or more times greater than the length. In that case, a transisistor would not fit into the normal size cell. So, to draw such transistor, we split the total width in several parts called fingers. This method is, for obvious reasons, called fingering. But, the most important reason for fingering is to optimize the resistance of the gate poly. At the same time we need to pay attention on actual number of fingers and odd vs. even number of fingers.

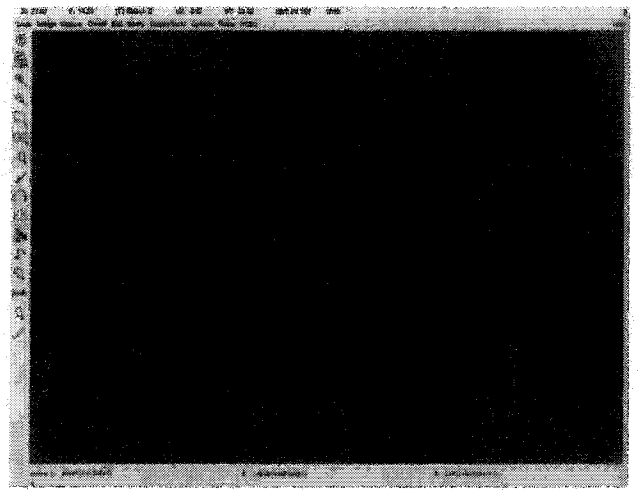


Figure 8.3: Transistor layout for the Self Oscillating Mixer uses 10 fingers

Figure 8.3 shows us the actual transistors drawn in Cadence Layout where we incorporated ten fingers seen in the red color.

Another consideration is to optimize the number of contacts. The most commonly used and reliable rule is to use as many contacts as you can using the minimum design rule. This assures the maximized performance of the transistor and the only down side is limited routability of the transistor.

The type and angle of paths is also important. 90 degrees polygons and paths are most commonly used. The reason is that orthogonal shapes require smallest amount of data and layout. Some specific designs use 45 degree bends because of limited space and pitch, like memory cells, but, in those cases, the maintenance and modifications become that much harder.

We should try to avoid any soft connect nodes which are ones that are being connected through nonrouting layer. This kind of error will pass the standard design checks but will cause poor circuit performance.

Figure 8.5 shows the complete layout done by cadence of the Self Oscillating Mixer. All of the inductors and capacitors are taken off the chip for the purpose of saving space and cost of the production. This gives us more freedom for adjustment and fine tuning. Appendix E presents some of the Cadence files showing parts of the circuit as well as complete circuit.

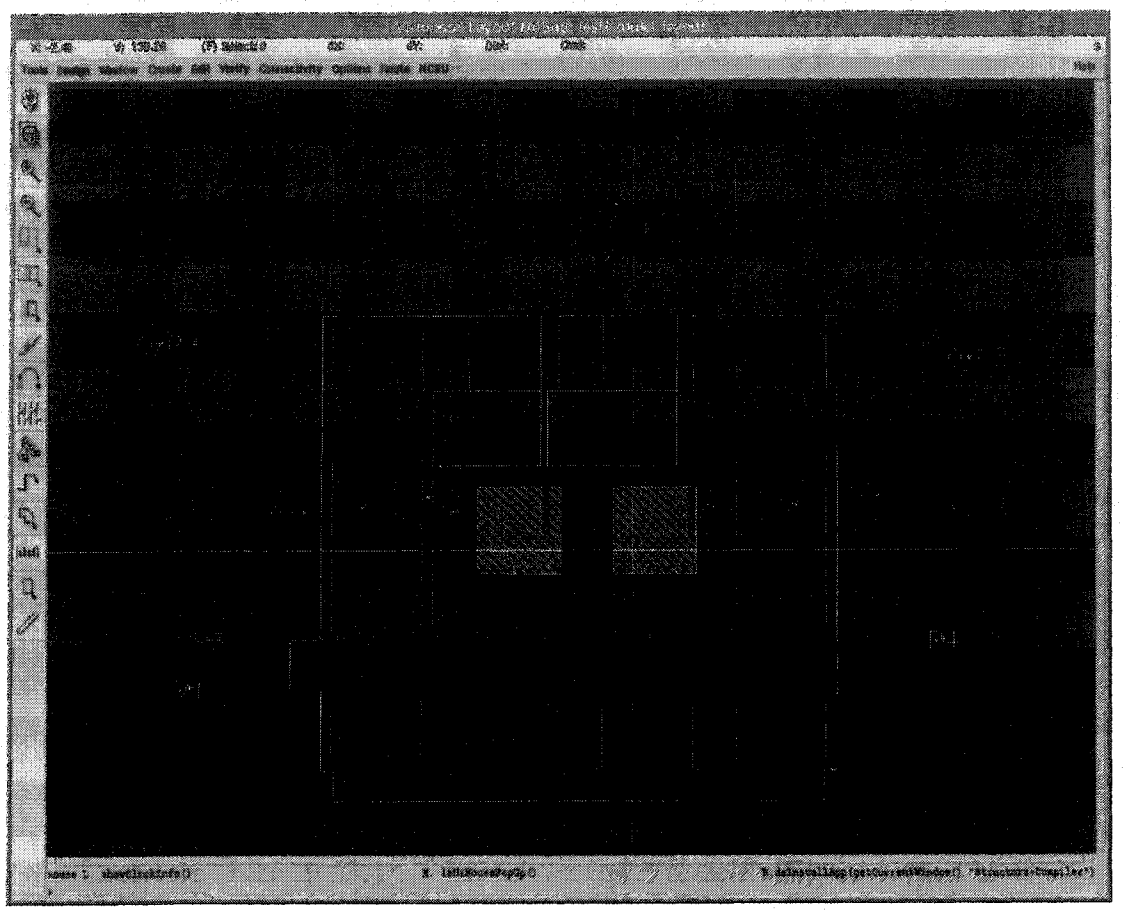


Figure 8.4: Complete Circuit layout for Self Oscillating Mixer

CHAPTER 9

Conclusion

The reflex amplifier proof of concept was successfully simulated. The usage of the reflex amplifier will result in a reduction in power level through stage reuse. We have shown that stability issues can be resolved through compensation and a double balanced, fully differential configuration. The reflex topology does result in an increase in passive parts count with the diplexer implemented off-chip. We believe this to be the first published CMOS reflex receiver design and the first use of reflex topology in an RF/IF configuration.

A self-oscillating mixer was simulated, and a low-frequency model was built (6 MHz). This circuit can result in reduction of system power by eliminating the local oscillator. It has good gain, and the linearity figures are acceptable. The mixer does require high bias voltage due to ensure proper startup conditions. To the best of our knowledge this is the first demonstration a self-oscillating Gilbert Cell Mixer.

REFERENCES

[1] B. Leung, *VLSI for Wireless Commun-ications*, Prentice Hall Electronics and VLSI Series, 2002

[2] P.J. Sulivan, B.A. Xavier, W. H. Ku, "Low Voltage Performance of a Microwave CMOS Gilbert Cell Mixer", IEEE Journal of Solid-State Circuits, Vol. 32, No. 7, July 1997

[3] P. J. Sulivan, B. A. Xavier, W. H. Ku, "Doubly Balanced Dual-Gate CMOS Mixer", IEEE Journal Of Solid State Circuits, Vol. 34, No. 6, June 1999

[4] Vikas Chandra, "A 2.5V High Lineaity CMOS Mixer for 1.9 GHz Application", Dept. of ECE, Carnegie Mellon Univerisity, Pittsburg, PA, July 2001

[5] B. Razavi, *RF Microelectronics*, Prentice-Hall, Upper Saddle River, 1998.

[6] E. Da Silva, High Frequency and Microwave Engineering, Read Elsevir PLC Group, 2001

[7] A. Rofougaran, G. Chang, J. Rael, J. Chang, M. Rofougaran, P. Chang, M.Djafari, J. Min, E. Roth, A. Abidi, H. Samueli, "A Single-Chip 900-MHz Spread-Spectrum Wireless Transceiver in 1-um CMOS - Part II: Receiver Design", *IEEE Journal of Solid-State Circuits*, vol 33, pp. 535-547, Apr. 1998.

[8] D. Pozar, *Microwave Engineering*, John Wiley & Sons, 1997

[9] H. Darabi, A. Abidi, "A 4.5mW 900-MHz CMOS Receiver for Wireless Paging", *IEEE Journal of Solid-State Circuits*, vol. 35, pp. 1085-1096, Aug. 2000

[10] Cotter W. Sayre, "Complete Wireless Design", the McGraw Hill Companies 2001

[11] Michael B. Bendak, Bernard A. Xavier, Paul M. Chau, " A 1.2 GHZ CMOS Quadrature Self-Oscillating Mixer"

[12] B. Razavi, *Design of Analog CMOS Integrated Circuits*, McGraw-Hill 2001

[13] G. Gonzales, *Micoelectronic Transistor Amplifiers, Analysis and Design*, 2nd ed., Prentice Hall, 1997

[14] P. Gray, R. Meyer, Analysis and Design of Analog Integrated Circuit, 3rd ed.,John Wiley and Sons, 1993

[15] P.Leroux, J.Janssens, M.Steyaert, "A 0.8dB NF ESD-Protected 9mW CMOS LNA," *ISSCC Dig. Tech. Papers*, vol. 26, pp.410-411, 471, Feb 2001.

[16] B.Floyd, J.Metha, C.Gamero, and K.K.O, "A 900 MHz 0.8 um CMOS Low Noise Amplifier with 1.2 dB Noise Figure," *Proc. 1999 Custom Integrated Circuits Conference*, pp.661-664, May 1999.

[17] H.Darabi and A.A.Abidi, "An Ultralow Power Single-Chip CMOS 900 MHz Receiver for Wireless Paging," *Proc. IEEE 1999 Custom Integrated Circuits Conference*.

[18] General Electric Transistor Manual, 1964.

[19] D.B.Hall, "A Battery-Operated Communication 3-Valver," *Practical Wireless*, pp.4-6, December 1943.

[20] D.B.Hall, "A Single Transistor Loudspeaker Receiver," *The Radio Constructor,* pp.57-59, August 1963.

[21] D.B.Hall, "Mains Reflex Circuit with Transformerless Full-Wave Rectification," *The Radio Constructor*, pp.208-209, October 1963.

[22] D.B.Hall, "Reaction Circuit for Reflexed Transistors," *The Radio Constructor*, pp.310-312, December 1963.

[23] Clein, Dan, " CMOS IC Layout: Concepts, Methodologies, and Tools," Butterworth and Heinmann, 2000

[24] C. A. Balanis, *Advanced Engineering Electromagnetics*, John Wiley & Sons Inc., 1989

[25] J. F. Wilson, R. Y. Youell, T. H. Richards, G. Luff, and R. Pilaski, "A single-chip VHF and UHF receiver for radio paging," *IEEE J. Solid-State Circuits*, vol. 26, pp. 1944–1950, Dec. 1991.

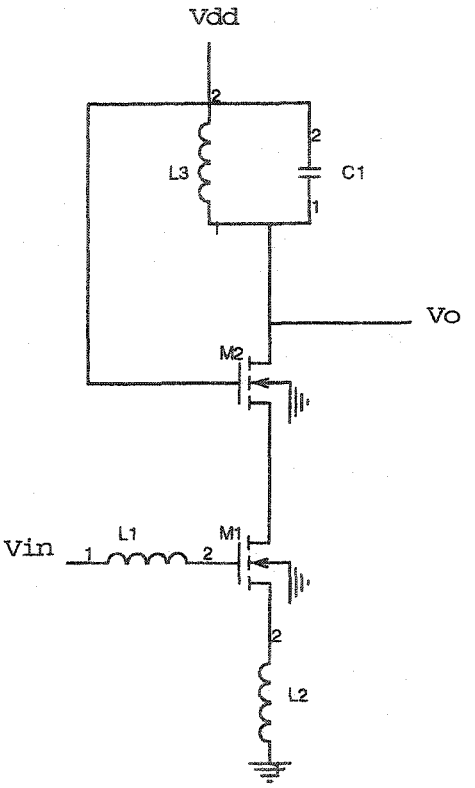
[26] GEC Plessey Application Note for SL6619, Apr. 1996.

[27] S. A. Sanielevici, K. R. Cioffi, B. Ahrari, P. S. Stephenson, D. L. Skoglund, and M. Zargari, "A 900 MHz transceiver chipset for two-way paging applications," *IEEE J. Solid-State Circuits*, vol. 33, pp. 2160–2168, Dec. 1999.

APPENDICES

APPENDIX A

LNA Calculations



Step 1.

Rin power = Rs

$$1000 = g_m \frac{L_2}{C} = \frac{g_m}{C_{gs}} L_2 = 2\pi f_T L_2$$

Where f_t is unity gain frequency of the MOS transistor and given by

$$2 \cdot \pi f_T = \frac{g_m}{C}$$

In most cases f_T is given by manufacturer so in our case for 0.5u process $f_T = 4$ GHz

So:

$$L_2 = \frac{1000\Omega}{2\pi \times 4GHz} = .398nH$$

Step 2.

Lets say that NF specification say that our noise Figure NF<3dB

$$NF = 1 + \frac{2}{3}\frac{1}{1+\frac{L_1}{L_2}} < 2$$

$$\therefore \frac{L_1}{L_2} < \frac{1}{3}$$

$$\therefore L_1 < 1.200nH$$

$$L_1 = 1.000nH$$

Step 3.

Designing C=Cgs and (W/L) using the imaginary part matching condition

$$w_c(L_1 + L_2) = \frac{1}{w_c C}$$

knowing that : $w_c = 2 \times \pi \times 1.9$ Grad/Sec we can determine following

$$C = \frac{1}{w_c^2(L_1 + L_2)} = \frac{1}{(2\pi \times 1.9Grad/\sec)^2(1nH + ..398nH)}$$

$$C = 5pF$$

Now we can find (W/L) using the formula:

$$C = C_{gs} = \frac{2}{3} WLC_{ox}$$

We let L be minimum or 0.6 um and Cox is given to be in the process 2.6 fF/um^2

So we can substitute in the following equation:

$$W = \frac{C}{\frac{2}{3} C_{ox} L} = \frac{5pF}{\frac{2}{3} \cdot 2.6fF \times 0.6} = 4762um$$

$$\rightarrow \left(\frac{W}{L}\right) = \frac{4762\mu m}{0.6\mu m}$$

Step 4.

Now we have to se of the power specs are met:

$$g_m = 2\pi f_T \times C = 2\pi \times 4GHz \times 5pF = 125.7m\Omega^{-1}$$

To find the drain current of M1 and total power we use fallowing steps [13, 14]

$$g_m = \sqrt{2k'(\frac{W}{L})_1 I_{D_1}}$$

For this process we average the k' = 300[uA/V^2]

So by solving for $I_{D_1} = \frac{g_m^2}{2k'\left(\frac{W}{L}\right)} = \frac{(125.7m\Omega^{-1})^2}{2 \times \frac{300\mu A}{V^2} \times \frac{4762}{0.6}} = 3.318mA$

So the power in the transistor M1 is: *3.318 mA x 3.3V = 11 mW*

So the total power is about 4 times the M1 and that is about 44 mW

Step 5.

At last we will use the gain formula to find the capacitance C1 with the establishment that gain has to be more the 20 dB which is gain of 10:

So :
$$A_v = \frac{L_3}{L_2(1 - w_c L_3 C_1)} > 10$$

Now for simplicity we will assume that L3 = L2 which we previously calculated to be 0.398nH. (Step 1)

So:
$$1 - (2\pi f_c)^2 (0.398nH) C_1 < \frac{1}{10}$$

$$\therefore C_1 > \frac{0.9}{(2\pi f_c)^2 \times 2nH} = 15.87\,pF$$

$$\therefore C_1 = 20\,pF$$

Initially we will assume that width to length ratio will be the same for both transistors and latter we will se if we can make the adjustments that make it work more efficient.

APPENDIX B

Usage of OrCad Simulation Tools

OrCad simulation tools were used in development of this project we used. As with any software based research it is impossible to find software that will have all the features necessary for the process of design. While using the Or-Cad attention had to be paid to how the simulation is behaving. Due to the size of the circuitry and accuracy needed for some of the runs additional attention was given to the step size of output results. Details like sampling time for simulation runs and output scales had to be set manually in order to get desired or useful output.

Figures B.1 and B.2 show the difference in the same circuit simulation set automatically (Figure B.1) and set manually (Figure B.2). This difference appears due to OrCad's tendency to chose the step size in such a way to make the simulation as fast as possible. One way to eliminate too short and inaccurate simulations is to set the step size to a value needed. The problem with setting the step size is eventual memory overload.

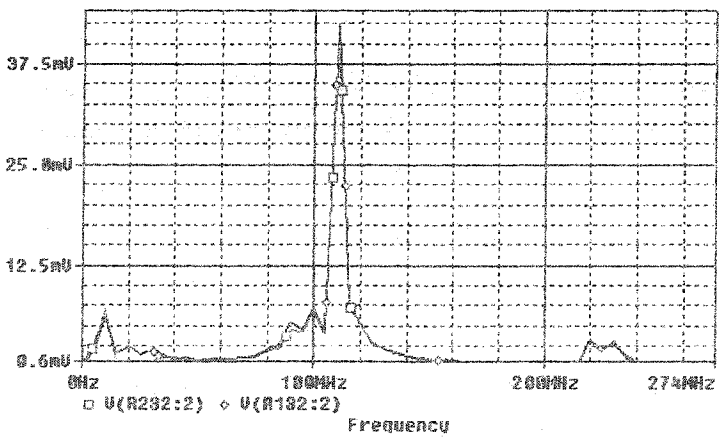


Figure B.1. Or-CAD Simulation set for 200 ns

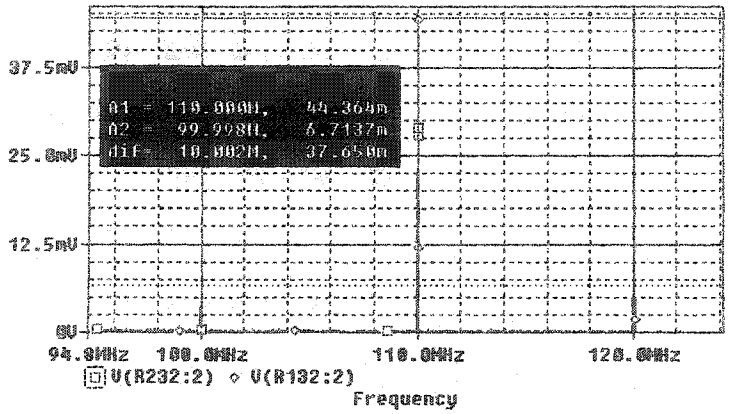


Figure 2.2: Or-CAD Simulation set for 200 ms

Consequently, we notice that there is a difference in the output quality of shorter and longer simulations. In case of shorter simulation two close signals could not be seen and simulation output did not reflect a realistic result. Shorter runs did not have a step size small enough to differentiate between the two close signals (Figure B.3a). On the other hand if simulation is too long it takes to much time to simulate single run. In our case we used 20 μs run which took approximately 45 minutes on Pentium 4 2.66 GHz processor. In this case optimum solution is to use medium resolution with a step size long enough to differentiate close signals and have enough accuracy. After several trial runs a scale of 1-2 μs was chosen for all simulation runs. (Figure B.3b). This will give us about 1000 cycles which is accurate enough for basic rounding. For more accurate readings we used 10 μs which gave us about 9000 cycles.

Figures B.3a, B.3b, and B.3c show the examples of these cases:

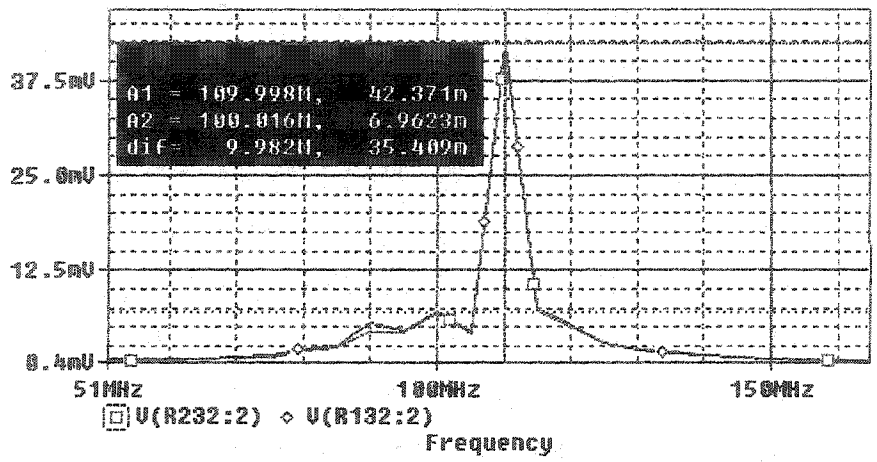


Figure B.3a: Simulation run 200 ns takes 10 sec to execute

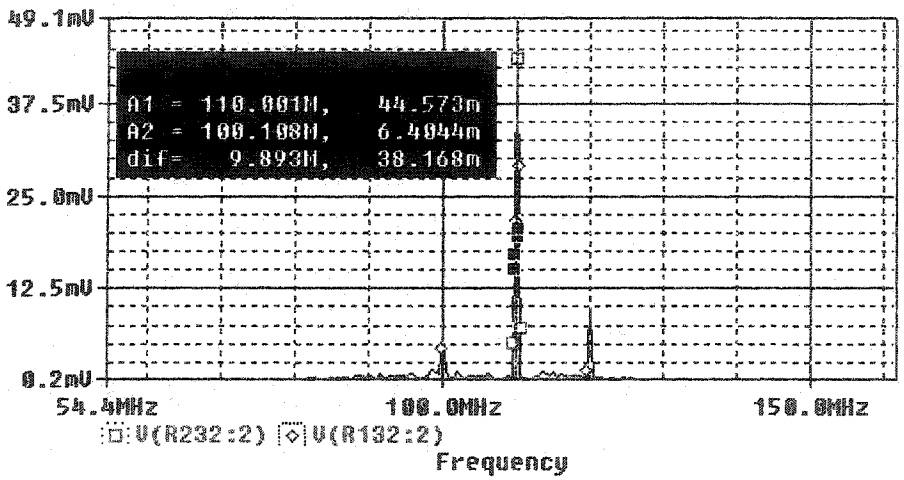


Figure B.3b: Simulation run 2000 ns takes 1 min to execute

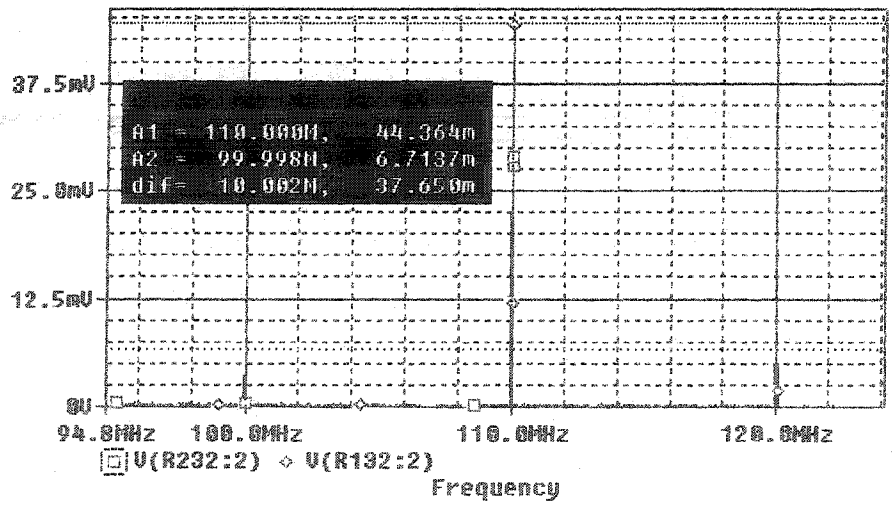


Figure B.3c: Simulation run 20000 ns takes 30-45min

One of the biggest challenges of the successful use of the Or-Cad simulation tools is to be able to use a standard model that can be manufactured. The problem here is that manufacturer uses a specific set of processes with parameters that are not available in the Or-Cad library. For that reason we had to find a way to import transistor parameters from the manufacturer data bank to Or-Cad in a format that can be used for simulation.

This problem was resolved by analyzing the Or-Cad parameter coding and implementing these basic rules for import of the manufacturer transistor parameters. The following is the summary of the procedure to import:

Or-cad stores its transistor parameters in the form of text files. Each file has the title and the body. By default the title is break plus extension. If the title of any new file is kept the same as the Or-Cad default, all transistors in that file will assume same parameters. On the other hand, if the title is changed only the transistor with the changed title will have the new parameters.

As an example the default CMOS N Channel transistor has the name NBREAK and Figure B.4 shows this part as shown by Or-Cad.

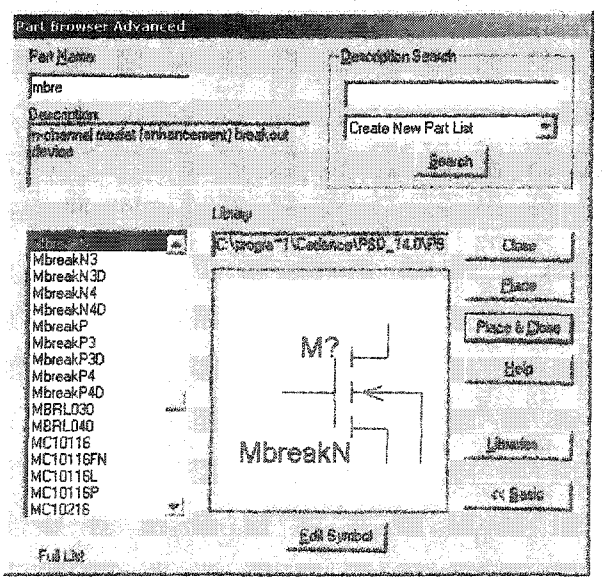


Figure B.4. Screen Shoot of the Main building block of system

Once the part is placed on schematic we can right click the device and go to EDIT PSPICE MODEL and that is where all the attributes of the transistor are found. Though the system does not like this type of adjustments as long as you save all the changes there should be no problem. The best result and readability is achieved if device has all the parameters starting with + and lined up in single row column because in that case there is no need for comas or spacing behind each parameter.

A detailed description and a tutorial on how to import parameters from MOSIS and plot transistor curves are given in next section.

OR-Cad P-Spice Parameter Import

In this appendix it will be shown how to import MOSIS parameters in 7 steps. At the same time I want to emphasize that this is also possible by using the shortcut

-Right Click on MBREAKN Transistor and go to EDIT parameters

-Copy parameters form MOSIS

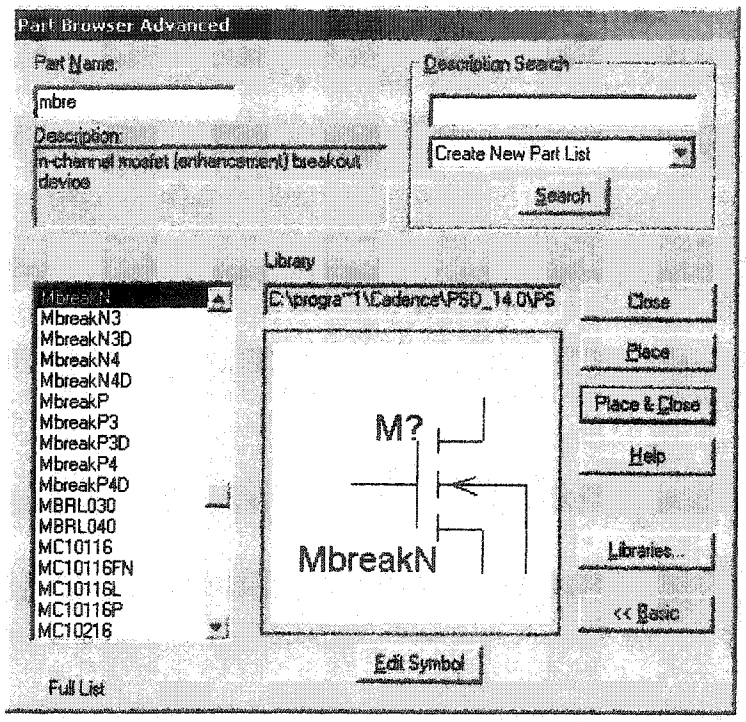
-Paste parameters and change the name of the file to new name

Problem with this is that sometimes P-Spice acts up and does not want to take the new files but instead saves the new file in addition to the old one and this way messes up all simulation. For that reason 7 steps are given which are more work but reliability is worth of little extra work.

The steps are:

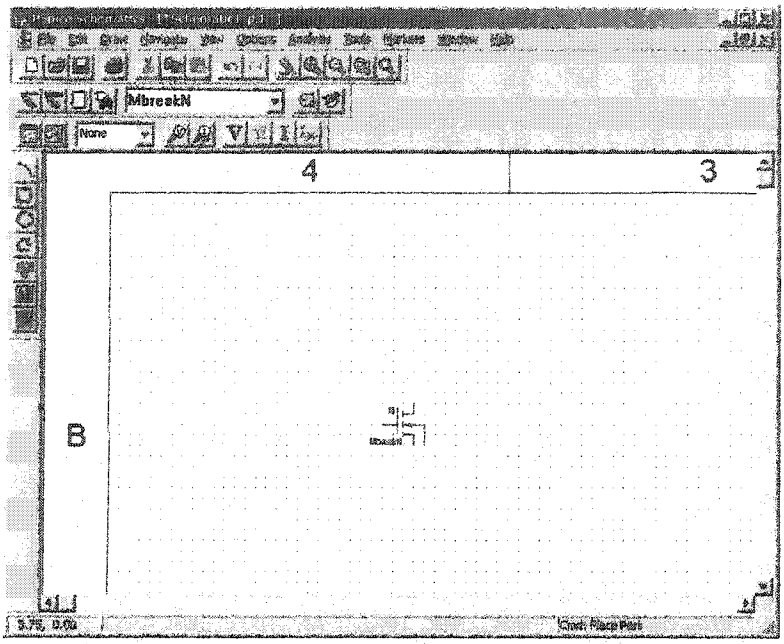
Step 1:

Go to the **Parts** menu. In this case you have need to select the part called MbreakN, as illustrated by the screen shown



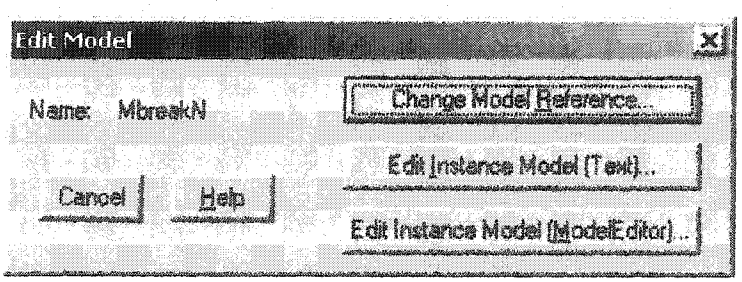
Step 2:

Place and Close a copy of this part on the Schematics screen and leave it highlighted, since the next step will call for you to make this part into one that is appropriate to performing designs in a current manufacturing technology.

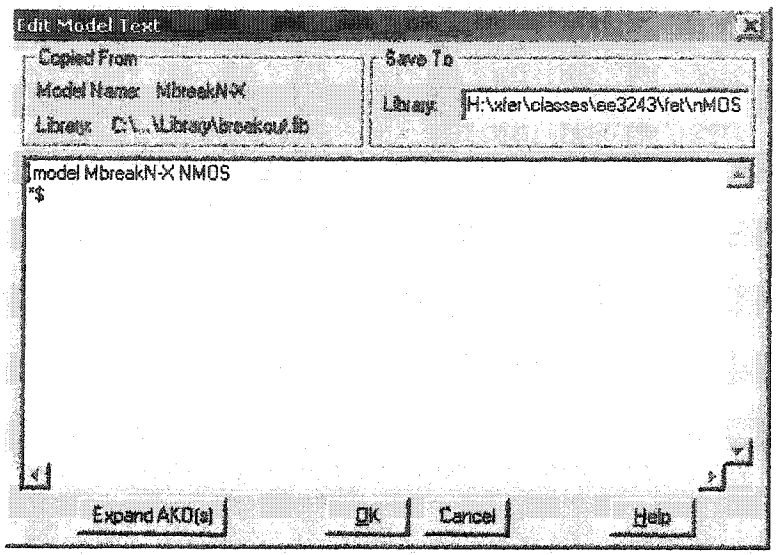


Step 3:

Pull down the **Edit > Model** menu, for which you will immediately get a message that says 'New schematics must first be saved'. So you will now need to save you file under whatever name you choose. For this illustration, we will save it under 'nMOSa7' and move on. Once you have saved your file, then the 'Edit model' option will be enabled for which you should see the pop-up screen



and for which you should select 'Edit Instance Model (Text)' and for which you will now have the screen

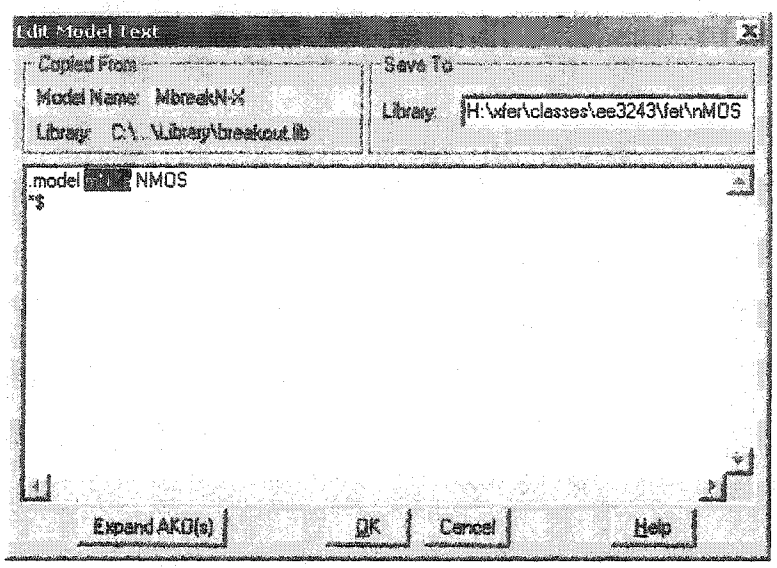


Since this screen is an 'edit' window, you can make changes that will stick. This is accomplished by highlighting the text and overwriting, which is the usual operation for editors created in the MS windows environment.

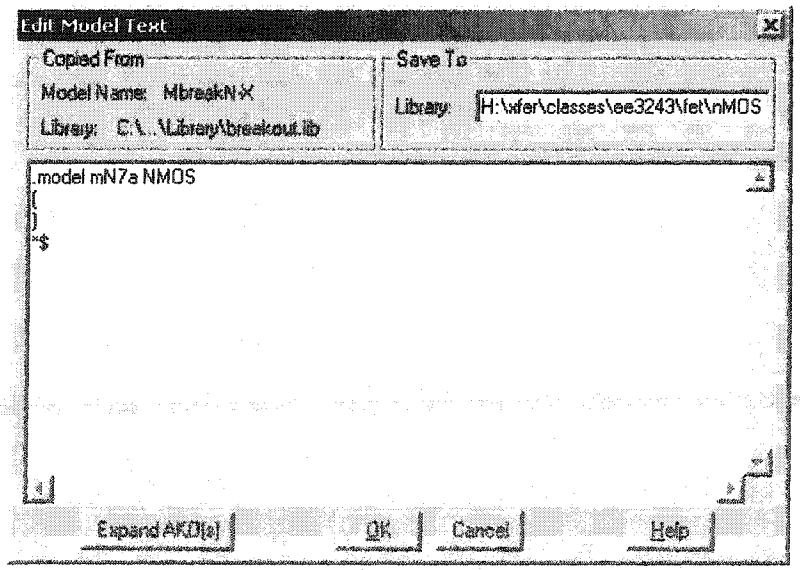
-

Step 4:

First we will change the name from MbreakN-X' to some other name. For purposes of illustration we will let it be 'mN7a', as shown:



Now move the cursor so that it falls after the 'NMOS' text and add a couple of parentheses, as indicated.



In between these parentheses you will eventually add the parameters that are necessary to define the FET as a working device in a particular technology.

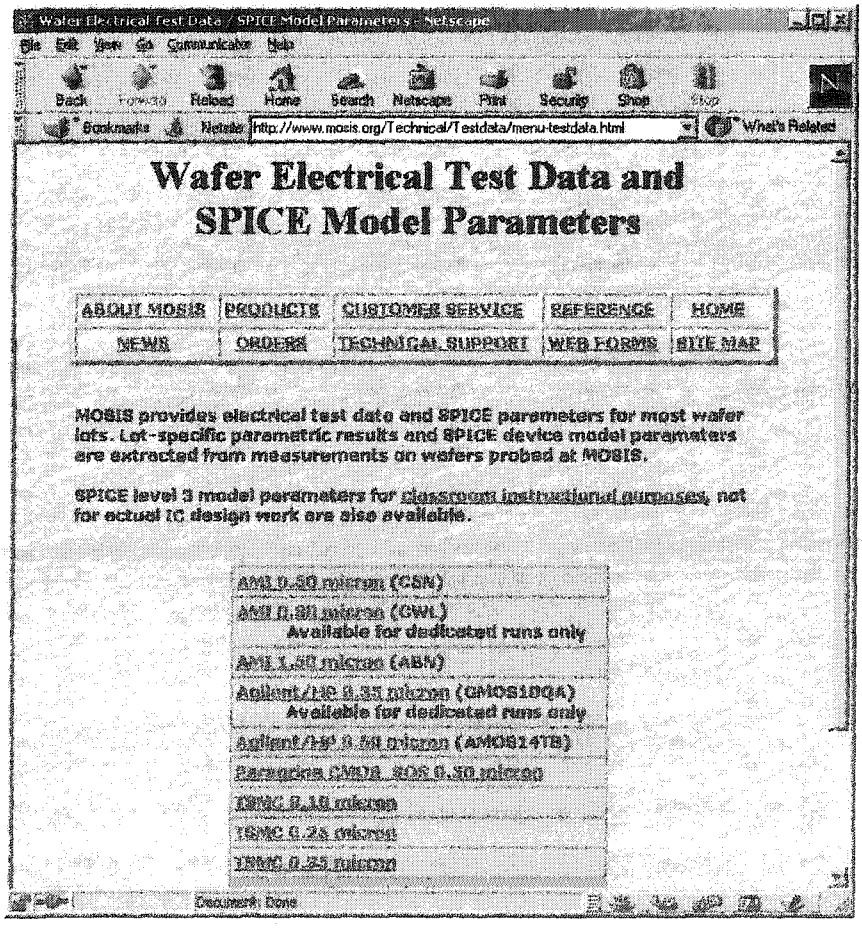
Leave this screen alive while you proceed to the next step.

Step 5:

Where do we get the parameters? We go to internet resources. In this case we will go to a site that provides CMOS fabrication services to universities, among others. It is called 'MOSIS' and we will capture and paste some of their parameters. The link that gets us where we need to be most quickly is:

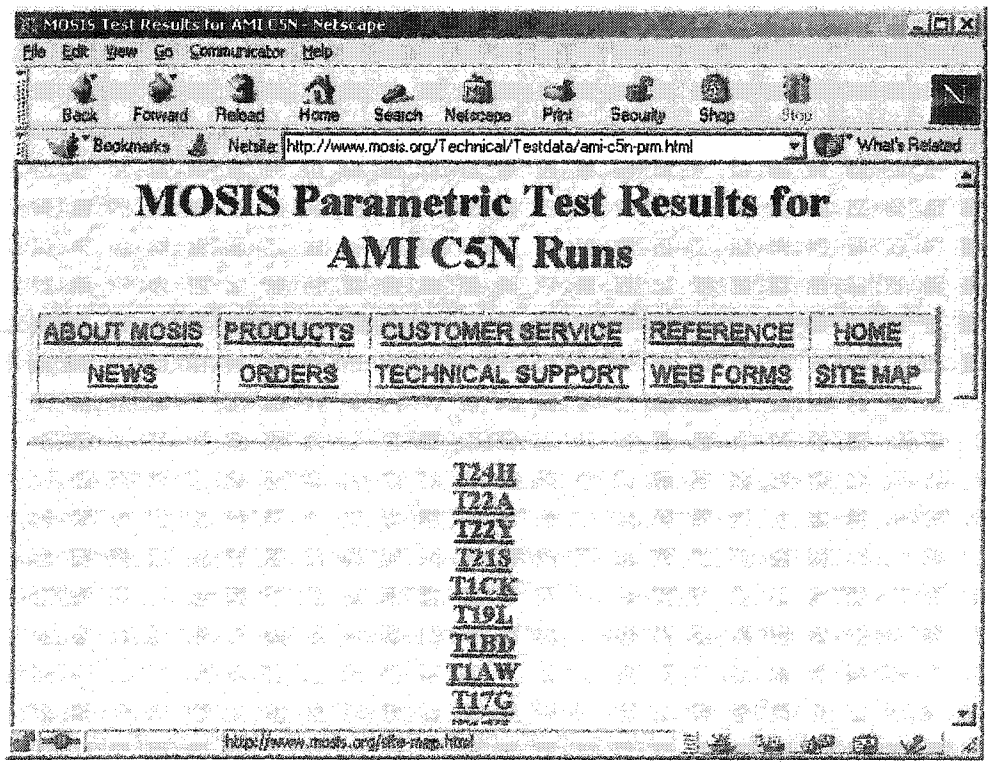
http://www.mosis.org/Technical/Testdata/menu-testdata.html

and looks like



Although the appearance of this site page may change, and surely the parameter and technologies will change, you note that there are several fabrication processes identified. Since universities are usually confined to only a few processes, let us choose the one identified as 'AMI 0.50 micron (C5N) and see what is there. This technology is one for which the smallest feature size is L (length) = 0.5m. As well as the active n- and p-layers this technology has 5 interconnect layers, two of which are polysilicon (poly) and three of which are metal, probably of various refractory alloys conducive to high temperatures and that have a reasonable conductivity. The two polysilicon layers are used to form embedded capacitances.

Under this menu we will find that there is a long list of process runs, and we will select the one called T22Y.



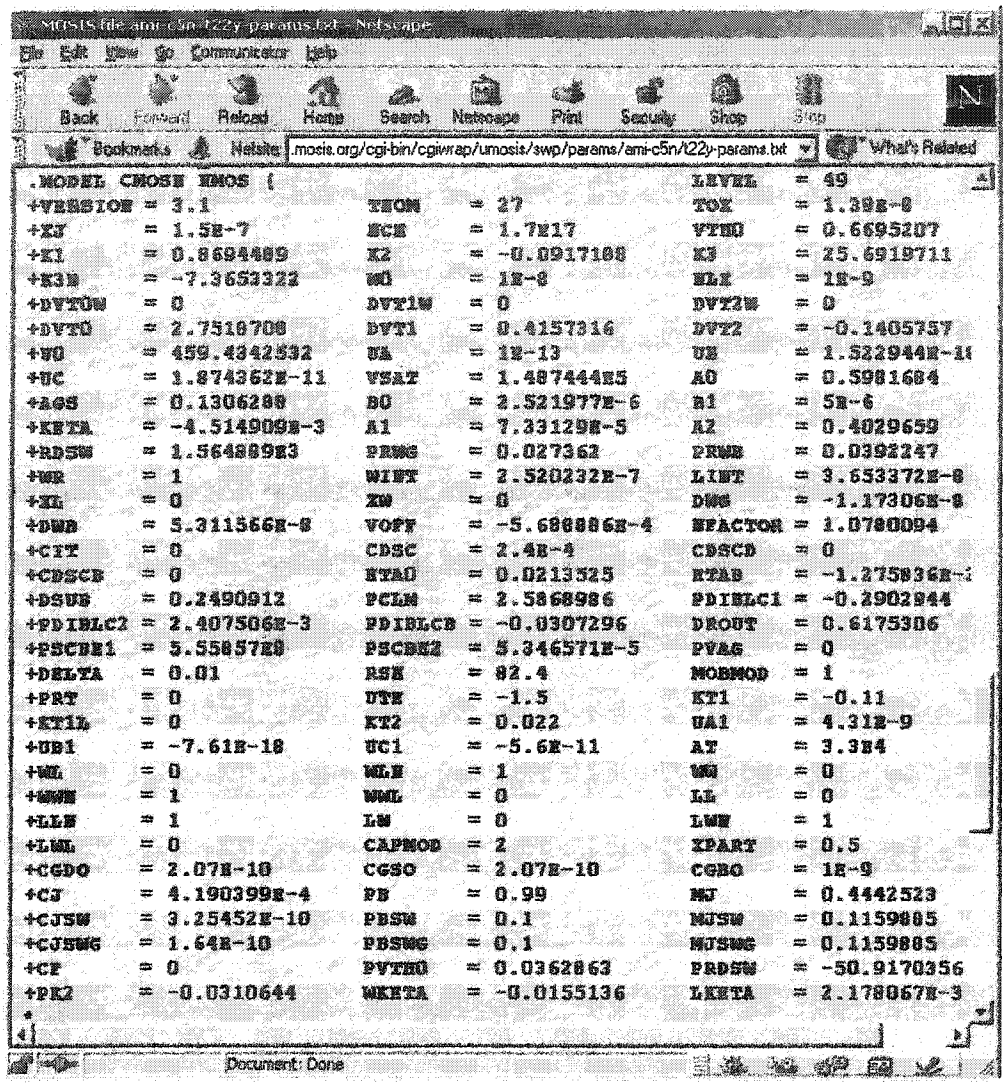
Selection of this process run displays a data sheet that indicates values of various extracted data and parameters, such as sheet resistance, area capacitances, and various thresholds. But the part of the data sheet that is of primary interest to the transistor is the part that contains the transistor parameters. And here we encounter a situation.

Given the very small size of the transistor, the electric field therein are enormous, and therefore there are many physical effects that take place. And each effect has its own equation. And each equation has its own parameter. And there are several different mathematical models that can be assumed. And often these are a mathematical mess. And so the parameter list is often long.

Well let's not get entangled by the descriptions of MOS transistor models. We will cut to the chase and pick one. The one that is most often used is a semi-empirical, semi-statistical model called BSIM3V3 (or a later version called BSIM3v4). It is awful and often disliked. But it is available.

And its parameters, as available under the file that we have accessed are shown below.

Notice that they fill the page. There are 108 parameters for BSIM3v3. Each has a purpose, but in many cases it is somewhat obtuse.

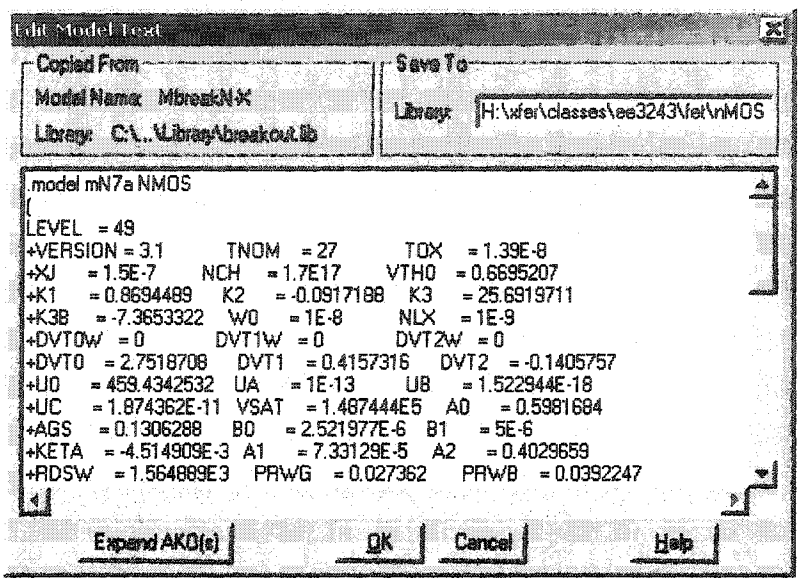


but if they can be displayed, then they can also be highlighted and captured (Ctrl-c). Do so, and be sure that you have them all, beginning with 'LEVEL = ' and down to '*'. Capture only the parameters associated with the NMOS device.

Step 6:

Now find the edit window for the nMOS transistor that you left active, position your cursor, and paste (Ctrl-v) these parameters into the space after the left parenthesis.

Assuming that the copy/paste operation has been successful, you should see them in place.



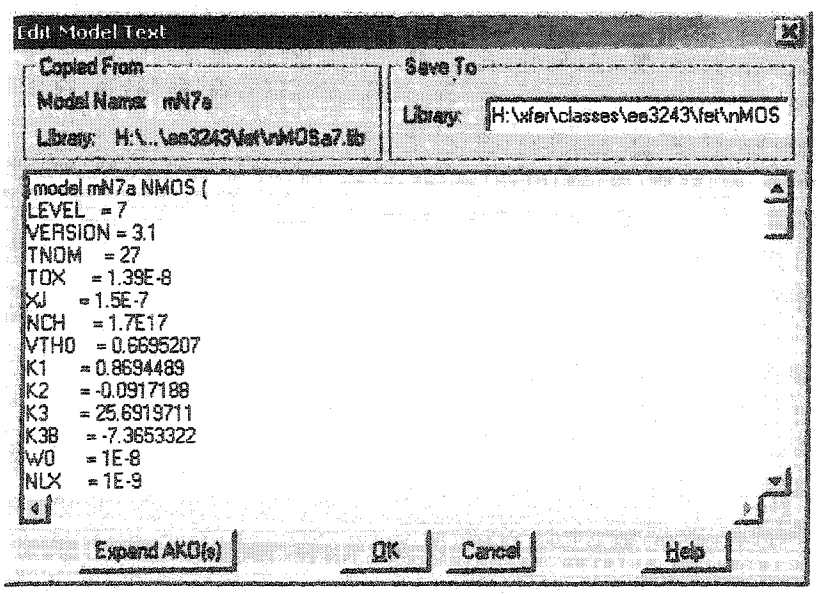
But there is still some work that must be done before you have completely instantiated these parameters into the model file. This file was designed to be downloaded into an F77 environment (different environment besides pSPICE), and therefore has a continuation character at the beginning of each line (the '+') that must be parsed and deleted before saving this file. This process will probably have to be executed line-by-line, unless the editor improves sufficiently to let you make a search and replace. So bite the bullet and do so. If you do not purge these continuation characters, then pSPICE will become very confused and your circuit simulation will crash.

And you will also need to make some changes. **Change LEVEL = 49 to LEVEL = 7.** When operated in the VLSI environment, BSIM3 is level 49. But pSPICE is a maverick,

and calls the BSIM3v3 model level LEVEL = 7. If you digress to some other simulation platform you will have to ascertain for yourself which level corresponds to which model.

Not so long ago there were only 2- 3 MOS transistor models. These are the ones that we use in the classroom, since they are simple and direct, but are also very inaccurate for small feature size MOS transistors. So as the technology has evolved we have accumulated more MOS device models, and now we have more MOS transistor models than a dog has fleas. The BSIM3v# models are public domain models, and so that is what we will use even though it may have its vicissitudes.

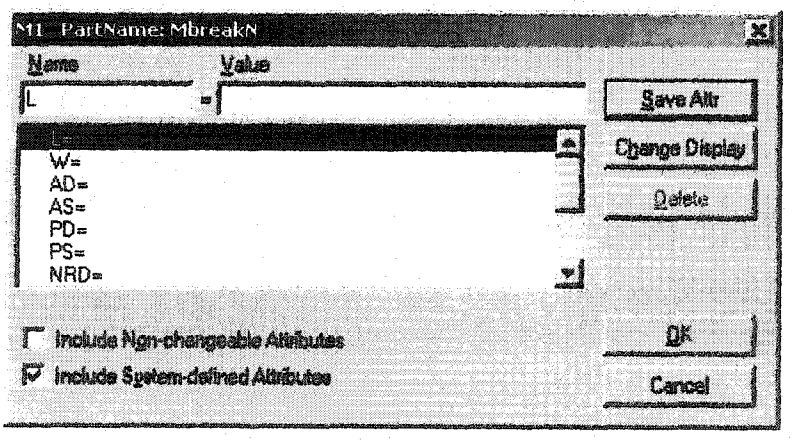
Once you have made these changes you can now close the model edit window by an 'OK', which should save your model parameters in a long list. If you then pull the list back up via **Edit > Model > Edit Instance**.. you will see the list in a more negotiable form:



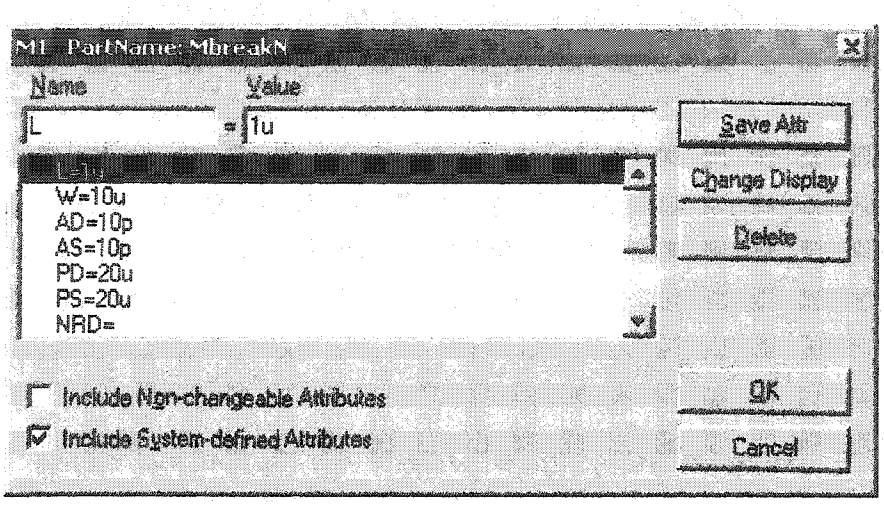
This list may require further editing, since it is entirely possible that some updates have not reached either the fabrication service or the simulation environment. In the event of clashes, we will return to the screen above and 'comment-out' parameters which pSPICE does not recognize, if any.

Step 7:

Now return to the Schematics screen, for which you have the one little nMOS part. Before any further use of this part, double click on it to present its sizing options:



We will assume that this transistor will have W/L = 10μm/1μm, which is pretty small, but large relative to the minimum feature sizes. We will also assume source and drain areas to both be 10μm x 1 μm = 10 p, and the source and drain perimeters to both be 20 μm,. Note that we use 'u' as the prefix for 'micro'. The values are assigned by typing in the values and executing a 'Save Attr' in each case.



Use the 'OK' to save this sizing feature for this transistor.

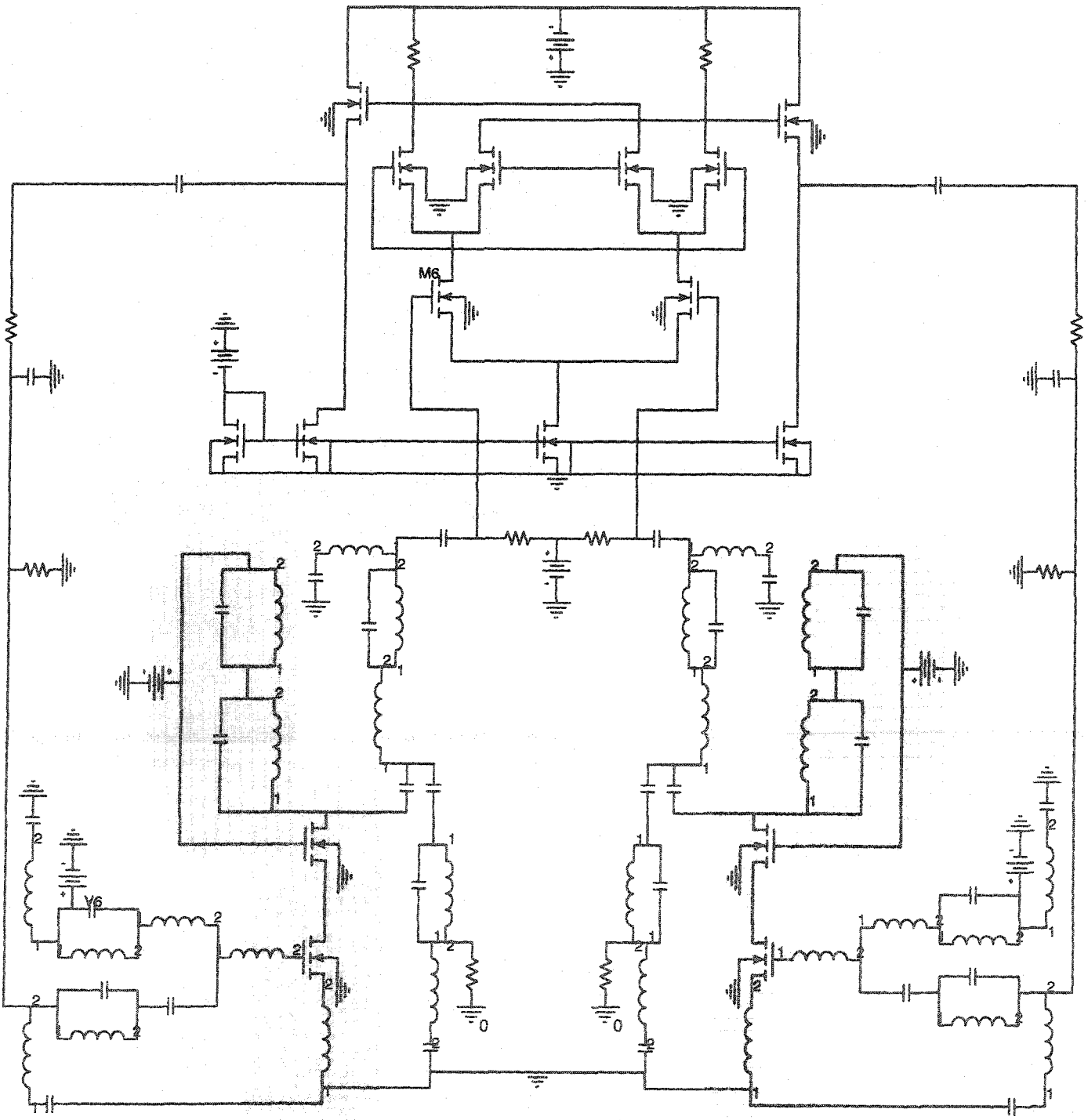
Now it is ready and you are ready to put it to work. It can be copied into any other schematic that you might desire, and will retain these process parameters and sizes.

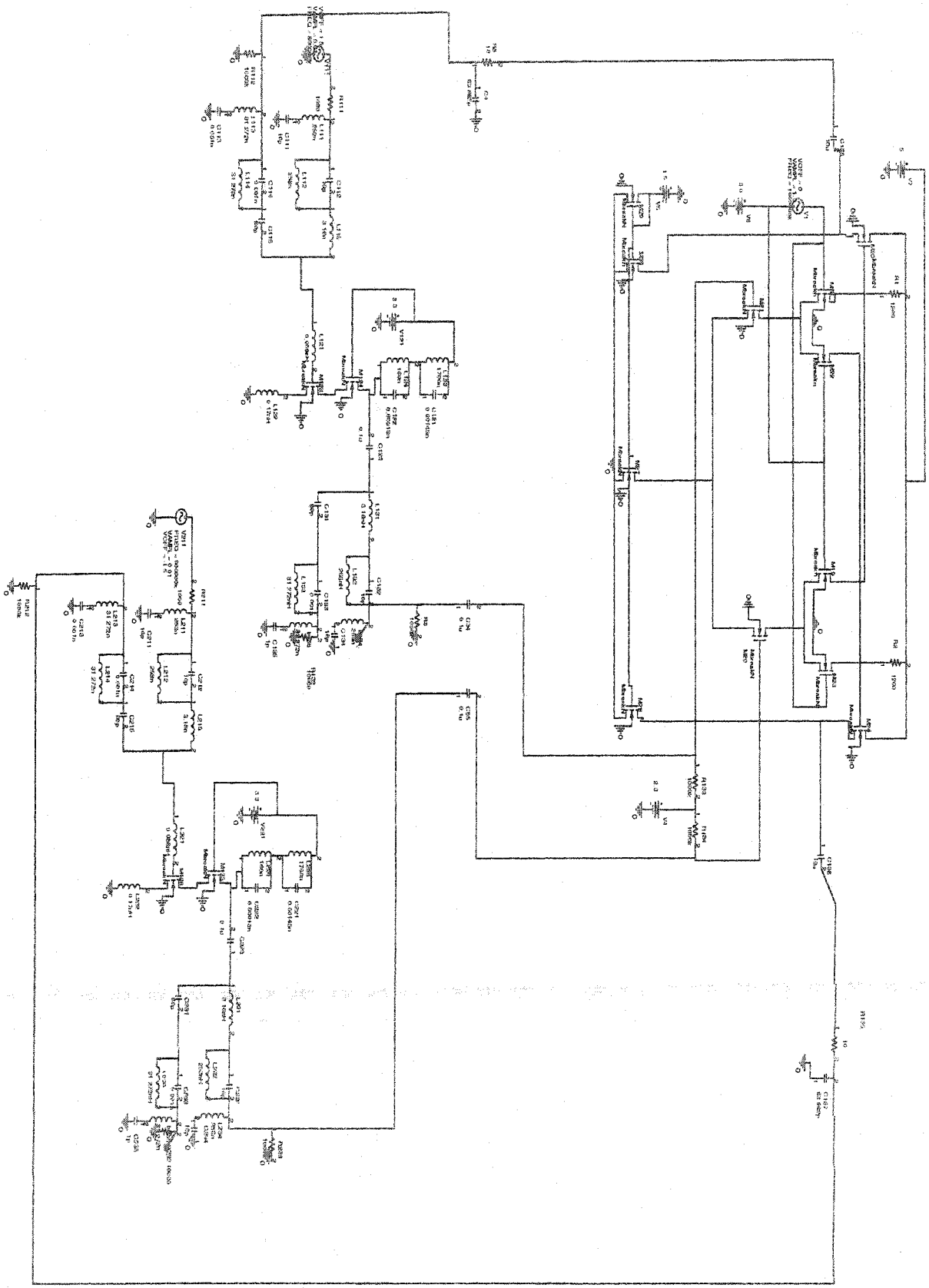
You have created an instance ('instantiated') this transistor into your working environment.

APPENDIX C

Schematic 1: Complete Circuit Schematic

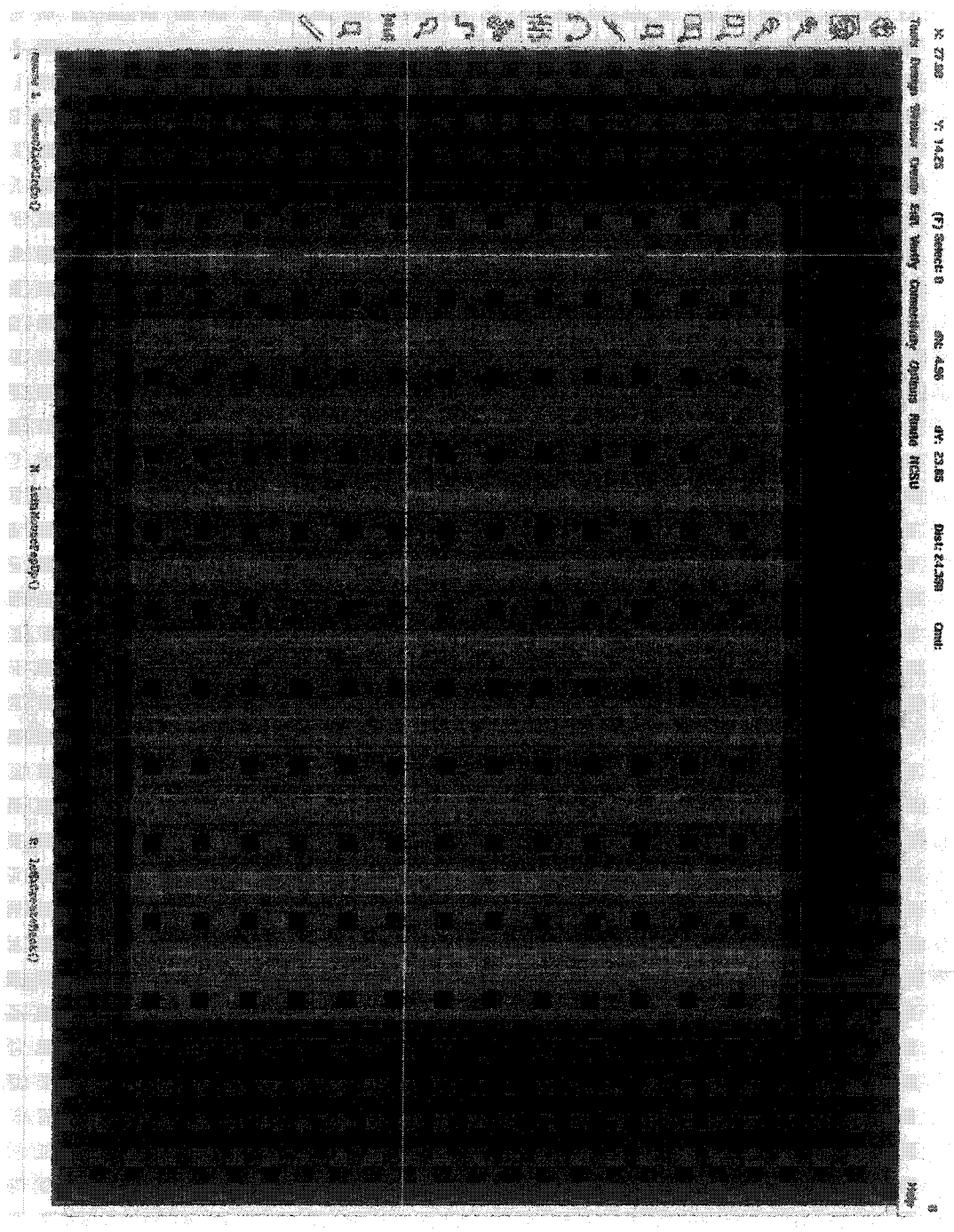
Low Noise Amplifier Gilbert Cell Mixer



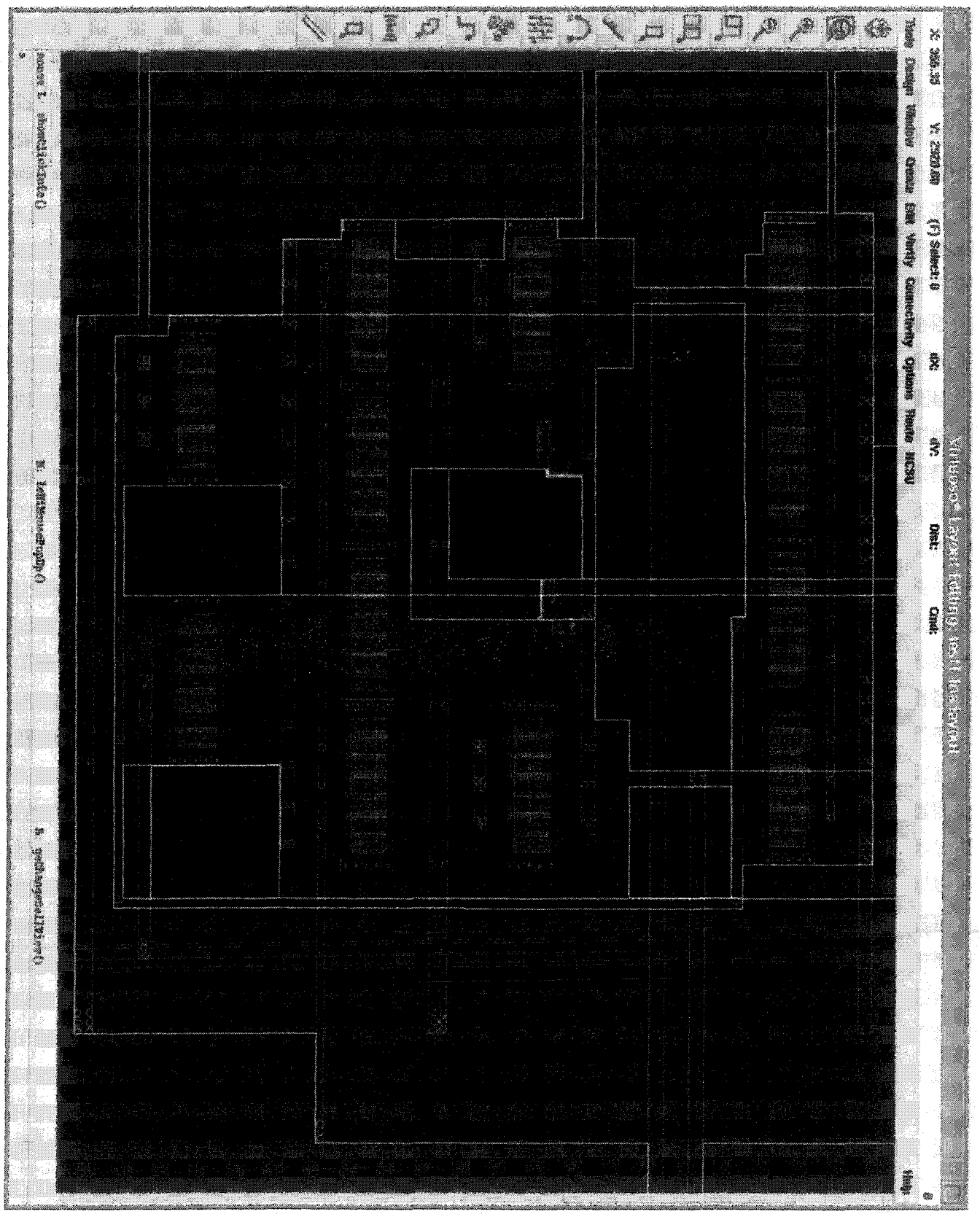


Schematic 2: Complete Circuit Schematic as used for design and testing

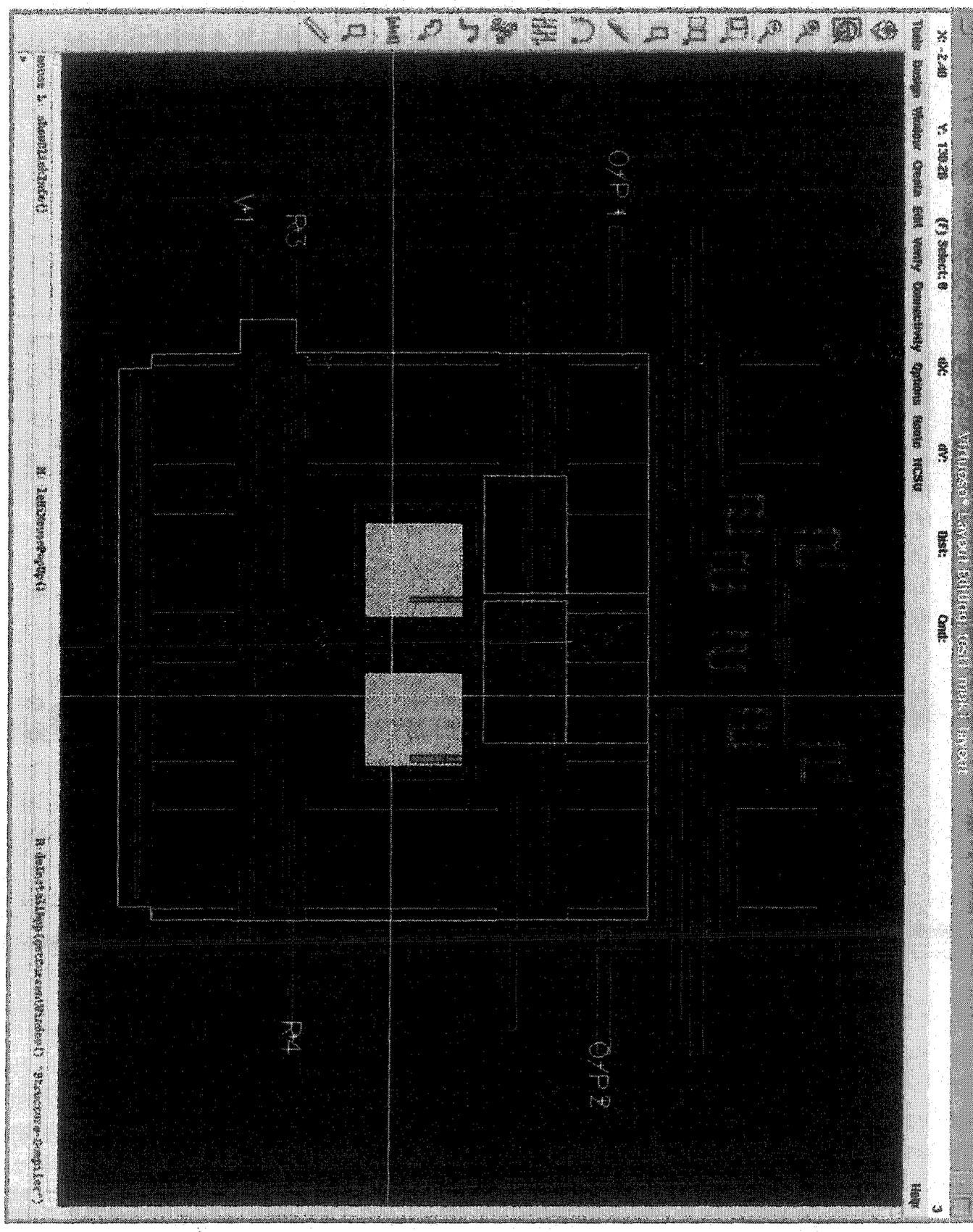
APPENDIX D



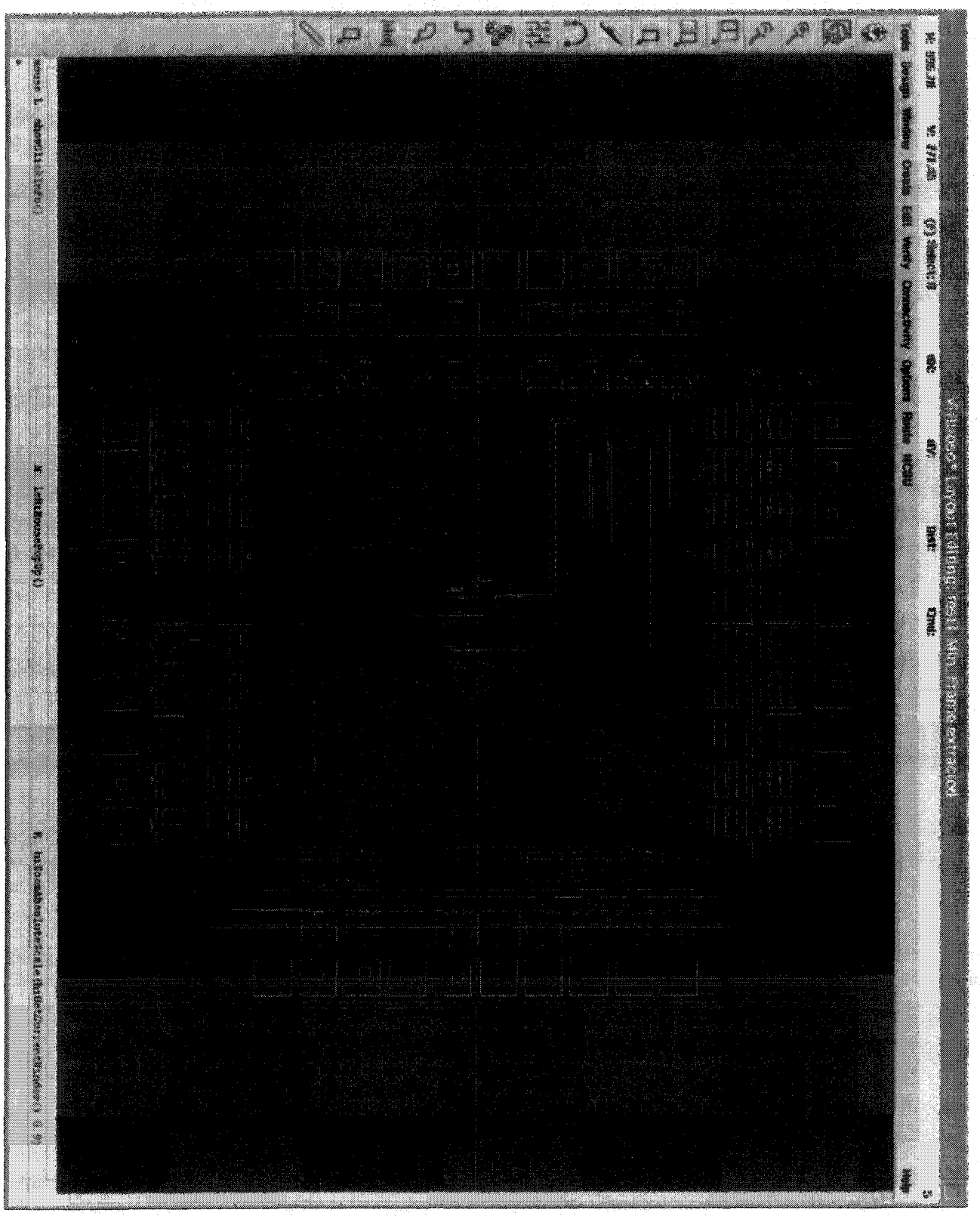
Transistor Layout using 10 fingers



Portion of the circuit involving multiple transistors



Self Oscillating Gilbert Cell Mixer



Pad layout

VITA

Radovan Kuzet was born in Zadar, Croatia, on January 5, 1978, the sun of Zivko Kuzet and Dusanka Kuzet. After completing prestigious Nikola Tesla High School he moved to Shreveport, Louisiana where he attended Bossier Parish College majoring in Industrial Control Systems, and played NJCAA Division I basketball in '97-'99. Upon graduation in 1999 he moved to Edinburg, Texas where he earned a Bachelor of Science in Electrical Engineering in 2002 and played NCAA Division I basketball during '99 - '01 seasons. He graduated with magna cum laude status and many honors including, but not limited to, Lou Hassell Award for outstanding athletic and academic performance, Dean's Least, Who's Who, National Honor Society and All American Scholar. He is currently working as the research assistant at UTPA as well as High School math and science teacher at Weslaco ISD. He will be graduating from Engineering Master's Program with an electrical major in December 2004.