

**A 26 MICROWATT, TWO-STAGE VCO AND MIXER
FOR DIRECT DPSK CONVERSION IN MEDRADIO**

By

ILYA CHIGIREV

A thesis submitted to the
Graduate School—New Brunswick
Rutgers, The State University of New Jersey
in partial fulfillment of the requirements
for the degree of
Master of Science
Graduate Program in Electrical and Computer Engineering

written under the direction of
Predrag Spasojevic
and approved by

New Brunswick, New Jersey

October, 2013

ABSTRACT OF THE THESIS

A 26 Microwatt, Two-Stage VCO and Mixer for Direct DPSK Conversion in MedRadio

By ILYA CHIGIREV

Thesis Director:

Predrag Spasojevic

This thesis proposes a design for a low-power downconverter and demodulator, formed by a voltage controlled oscillator and mixer to create a Vance demodulator. This system is intended to operate in the MedRadio frequency band (401-406 MHz). The MedRadio band is intended by the FCC for ultra low-power radios used for medical applications to carry non-voice data. A combination of gm/ID design methods and MATLAB scripting have been used to efficiently calculate the size of transistors within the VCO circuit. Using the transconductance and bias current ratio (gm/ID) value as a primary design variable efficient operation is ensured with this a geometry independent process. The MATLAB script, combined with simulated device information, allows for automation of device sizing, increasing accuracy of results over conventional square-law hand-calculations.

The result of this work and design method is a 26 microwatt direct down-conversion DPSK demodulator in the MedRadio, designed in an 8RF CMOS process. The circuit utilizes a two-stage VCO design directly coupled to a passive ring mixer. With biasing circuits it requires an active die area of 49 um x 39 um. The circuit is designed to operate

from a 1 V supply and achieves a post-extracted simulated phase noise of -118 dBc/Hz when injection locked. The passive mixer has been designed to achieve a conversion gain of -18.4 dB.

Table of Contents

Abstract	ii
List of Tables	vi
List of Figures	vii
1. Introduction	1
2. Background	3
3. Circuit Design	5
3.1. The Vance Demodulator	6
3.2. Voltage Controlled Oscillator	7
3.3. Passive Mixer	11
4. Design Methods	14
4.1. g_m/I_D Design Method	14
4.2. MATLAB Scripting	16
5. Layout	22
5.1. PVT Variations	22
5.2. Common-Centroid Layout Technique	24
6. Results	29
6.1. VCO	29

6.2. Mixer	32
6.3. System Performance	33
7. Conclusion and Future Work	35
Appendix A. VCO Circuit Analysis	37
Appendix B. Sample OCEAN script	40
Appendix C. Example Parsing Function	43
Appendix D. Example Look-up Function	47
Appendix E. MATLAB Sizing Script	51
References	57

List of Tables

4.1. Script generated devices sizes	20
4.2. Estimated capacitance values	21
6.1. Device sizes for design	30
6.2. Design corner operating conditions	30
6.3. FOM Comparison	32

List of Figures

3.1. Block diagram of proposed system	5
3.2. Vance demodulator	7
3.3. Maneatis delay-cell schematic	8
3.4. Connectivity of Maneatis cells in VCO	8
3.5. VCO tuning range and power consumption	11
3.6. Passive ring mixer schematic	12
3.7. Passive ring mixer visualizes with switches	13
4.1. g_m/I_D test circuit	15
4.2. MATLAB sizing script flow diagram	18
4.3. f_T of NMOS device as a function of g_m/I_D	19
5.1. Comparison of typical and common-centroid layout methods	24
5.2. VCO layout	26
5.3. Mixer layout	27
5.4. Layout for full design	28
6.1. VCO phase noise results	31
6.2. Conversion gain and 1 dB compression point of mixer	33
6.3. Simulated signal demodulation	34
A.1. Common mode realization of Maneatis cell	37
A.2. Small signal realization of Maneatis cell	38

Chapter 1

Introduction

There is a growing need for low power radio frequency (RF) technology for medical implant devices. The original Medical Implant Communication Services (MICS) band has been expanded in recent years to the new Medical Device Radio Communication Service (MedRadio), increasing the available spectrum to 401-406 MHz [1]. MedRadio is specifically designed for medical sensors and radios, primarily intended for implants, however also includes allowable frequencies for body-worn communication devices.

Current technologies for low power radio consume an order of magnitude more power than what is acceptable for implantable operation. Hence, the primary design criteria for MedRadio compliant device is reduced power consumption. The work herein aimed to design a lower power solution to two of the primary components in a radio receiver, the voltage controlled oscillator (VCO) and the mixer. Together these two components are configured to form a Vance demodulator [2], which is able to directly convert differential phase-shift keying (DPSK) signals as part of an overall receiver front-end for a MedRadio transceiver.

The circuit has been designed to operate with minimal power consumption as the primary design goal while maintaining performance and reliability. Additionally the circuit elements have been chosen to minimize size by reducing the need for passive components such as inductors, which dominate die area in comparison to transistors.

Background information about the MedRadio standard is presented in Chapter 2. The implementation and design of the system, with details of individual circuit components are discussed in detail in Chapter 3. Information on design methodologies for low power design

in deeply-scaled CMOS processes presented in Chapter 4. In chapter Chapter 5 details of layout techniques used in this work are given. Simulated, post-extracted design results are discussed in Chapter 6. Finally, conclusions and future directions are discussed in Chapter 7.

Chapter 2

Background

The Medical Implant Communication Service (MICS) band was established by the FCC in 1999 and expanded to the Medical Device Radio Communication Service (MedRadio) in 2009. The MedRadio standard is defined by the published FCC Rules and Regulations as [3]:

An ultra-low power radio service, for the transmission of non-voice data for the purpose of facilitating diagnostic and/or therapeutic functions involving implanted and body-worn medical devices

A key term used in this description is “ultra-low power”. Specifications limit the larger portion of the spectrum (402 MHz-405 MHz) to 100 nW equivalent isotropically radiated power (EIRP). EIRP is the output power of the transmitter adjusted for physical connection (i.e. cable or connection) and antenna losses, or more simply the actual radiated power from the antenna. Portions of the sidebands are similarly limited to EIRP as high as 250 nW to as low as 25 μ W. As a comparison point, a class 3 Bluetooth device has a maximum EIRP of 100 mW [4].

In addition to very tight power restrictions the 402-405 MHz band of MedRadio is limited to a transmit duty cycle of 0.01%, and no more than 10 transmissions per hour, with 1 MHz sidebands allowing for 0.1% transmit duty cycle and 100 transmissions per hour. These restrictions on power and transition duty cycle mean that any device designed for MedRadio standards, particularly for implantable devices, must be power efficient both in transmission as well as sleep states. Implants powered off batteries are of particular concern

as lower power operation points can reduce the need for invasive surgeries, ultimately leading to a smaller risk of infection and complications.

Published work has demonstrated transceivers with transmit and receive power consumption of 0.84 mW and 0.59 mW, respectively [5], with other work achieving transmit and receive power consumption as low as 0.35 mW and 0.4 mW, respectively [6]. Similarly an injection locked oscillator (ILO), a crucial component of the transceiver architecture, has been designed consuming 1.2 mW [7], while other works have demonstrated oscillators consuming less than 200 μ W [8]. Aside from power consumption and physical device size considerations there are other considerations (e.g., reasonable data rates and high reliability [1]), however they are not specifically stressed in this work.

There are many developing implant technologies, including cardiac pacemakers, implantable cardioverter defibrillator, and neurostimulator [9]. All these technologies may benefit from reduced power consumption in the radios as well as reduced device sizes. Work has been ongoing in the design of low power transmitters and receivers in MedRadio, however it is still a developing research area. Motivation for this work stems from the still present need to develop simple and power efficient designs in this power constrained and rapidly developing area of research.

Chapter 3

Circuit Design

The design presented herein is a portion of a low-power receiver front-end for the MedRadio frequency band. The block diagram (Figure 3.1) for the proposed receiver front-end represents a minimal set of circuitry needed to demodulate differential phase-shift key (DPSK) signals. The system presented here forms a Vance demodulator, which can directly down-convert and demodulate DPSK signals [2], which will be described in section 3.1, and is formed primarily with a voltage controlled oscillator (VCO) and mixer, described in sections 3.2 and 3.3 respectively.

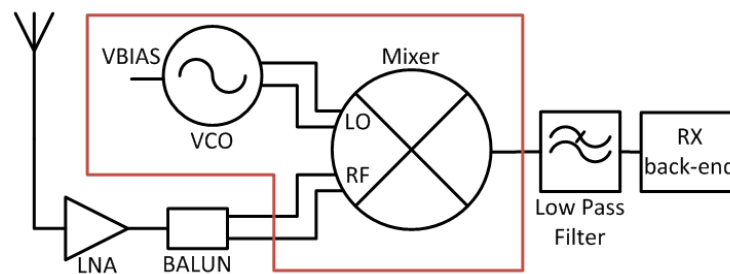


Figure 3.1: Block diagram of the proposed system. The outlined area represents the design for this work.

The Vance demodulator design calls for the output of the VCO, which is tuned to the same frequency and phase as the incoming transmission, to be multiplied with the transmission signal using a mixer. The low noise amplifier (LNA) and balun also act to filter noise from the incoming signal as well as convert it from an unbalanced signal to a balanced one, before it is fed into the mixer for downconversion. The output of the LNA and balun also act as the injection lock signal for the VCO, ensuring that it is at the correct frequency and phase. By injection locking the ring oscillator to the incoming signal, the

quadrature output of the oscillator represents a delayed version of the incoming signal. This delayed replica is fed to one terminal of the mixer, while the LNA directly drives the other terminal. The output of the mixer, the demodulated DPSK signal, is then fed to another LNA for filtering and amplification, after which the signal can be interpreted by the receiver back-end.

In order to reduce power consumption the VCO was designed using the Maneatis delay-cell [2] and a passive mixer. The VCO is designed to be directly coupled to a passive ring mixer, also chosen in order to reduce consumption. This configuration was chosen for its simplicity as well as its power consumption, as it only requires two principle components to demodulate a signal.

Though not included in this work, the literature contains examples of microwatt low noise amplifiers (LNAs) [10]. In the sections to follow detailed information is presented on the Vance demodulator configuration as well as the components which comprise one; namely, the VCO and mixer.

3.1 The Vance Demodulator

The Vance demodulator was proposed as a configuration for demodulating frequency shift key (FSK) signals. The Vance demodulator uses a local oscillator (LO), placed on the same frequency of the incoming signal and generates a signal $\delta\omega$ from an incoming signal $\omega_0 + \delta\omega$. The system diagram is shown in Figure 3.2, as given in the original paper [2].

The Vance demodulator uses the LO, as well as a 90° shifted copy of the LO signal, mixed with the incoming signal to multiply out the carrier frequency and generate two signals. For FSK signals the result is a system which extracts $\sin(\delta\omega t)$ and $\cos(\delta\omega t)$, and then using a combination of more mixers, amplifiers, and integrators generates a signal which is only the $\delta\omega$ portion of the original incoming signal. However for phase shift key (PSK) signals the incoming signal is in the form $\omega_0 + \phi$, and specifically for this implementation binary PSK (BPSK) the signal is in the form of $\omega_0 + n\pi$, where n is either 0 or 1.

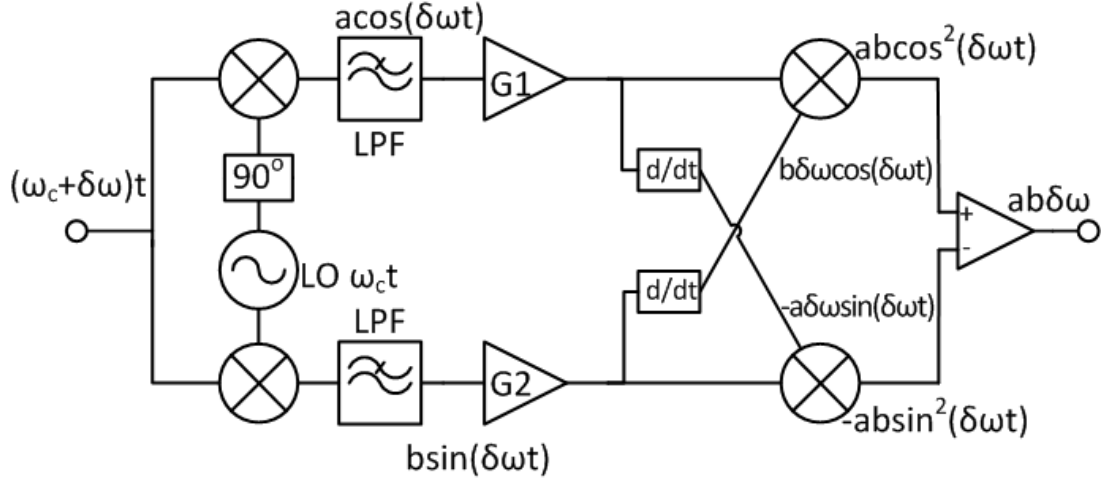


Figure 3.2: Block diagram of Vance demodulator

Using the same first stage of the Vance demodulator the LO, in this design a VCO, is set to the ω_0 and shifted by $\pi/2$, so the incoming transmission is always leading or lagging the LO by $\pi/2$, and when that are mixed the result is $\sin(\pm\pi/2)$. Implementing this modified Vance demodulator system, a BPSK signal can be demodulated using only a LO and a single mixer.

The attraction of this simple design is that the implementation requires only a single VCO and mixer pair, provided that the VCO is able to be tuned to the same frequency as the incoming signal and generate a replica signal which is 90° shifted. With only two primary elements it is possible for this design to use little power and occupy a very small area on the die.

3.2 Voltage Controlled Oscillator

The VCO designed in this work is based on the Maneatis delay cell, which uses positive feedback to reduce the required transconductance to achieve oscillation [11]. Using two such cells connected in series, with one of the two connections cross coupled, the VCO is formed. Each of the two cells contributes a 90° shift while the cross coupled connection contributes the remaining 180° needed for stable oscillation. The circuit diagram for the Maneatis cell

is shown in Figure 3.3 and the connections between the two cells to form the full VCO are shown in Figure 3.4.

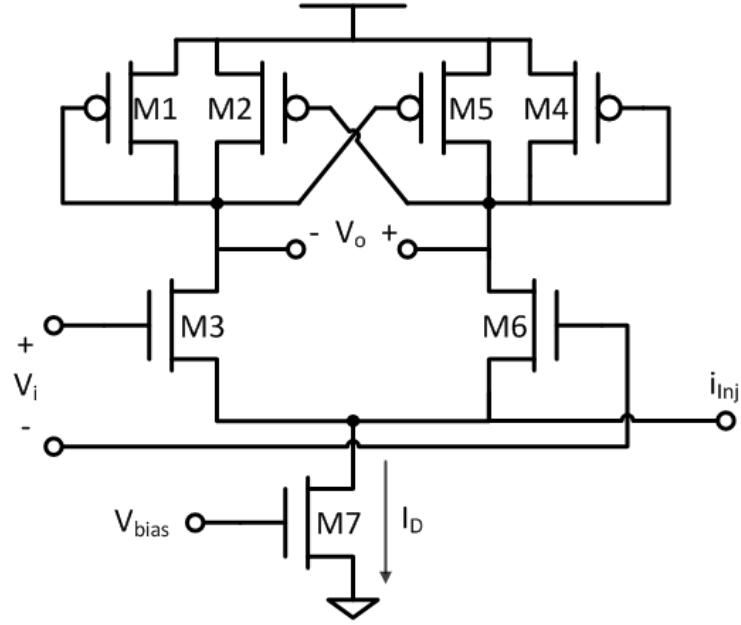


Figure 3.3: Maneatis delay-cell schematic.

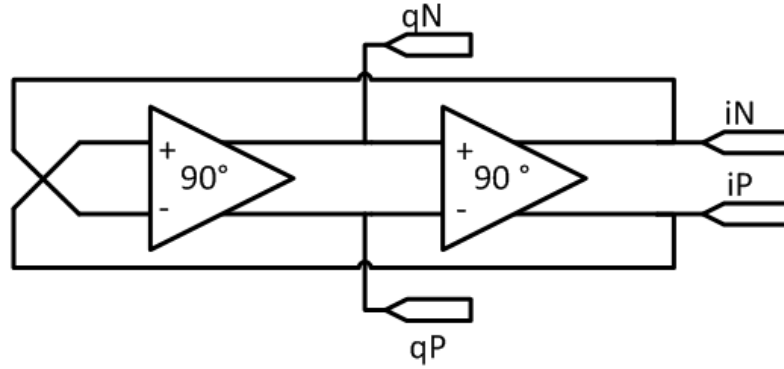


Figure 3.4: Connectivity of Maneatis cells in VCO

By applying a differential analysis on the half-circuit of a single Maneatis cell the following transfer function, $H_c(s)$, is found.

$$H_c(s) = \frac{g_{m3}}{(G_{ds} - g_{m2} + g_{m1}) + sC_L} \quad (3.1)$$

$$G_{ds} = g_{ds_3} + g_{ds_1} + g_{ds_2} \quad (3.2)$$

Looking to the connectivity shown in Figure 3.4, the two cells of the VCO are connected in quadrature to one another, with one connection crossed. The result of this is that the transfer function, $H(s)$, of the total VCO is:

$$H(s) = -(H_c(s))^2 = -\left(\frac{g_{m_3}}{(G_{ds} - g_{m_2} + g_{m_1}) + sC_L}\right)^2 \quad (3.3)$$

Application of the Barkhausen criteria, which state that for stable oscillation the total phase shift of a VCO loop must be $2\pi n$ and the gain of the full loop must be unity, yield a oscillation frequency shown in the equation below.

$$f_{osc} = \left(\frac{1}{2\pi}\right) \cdot \left(\frac{\sqrt{(g_{m_3})^2 - (g_{m_1} - g_{m_2} + G_{ds})^2}}{C_{tot}}\right) \quad (3.4)$$

A full derivation of these equations is found in appendix A

C_{tot} is the total intrinsic capacitance associated with the transistors, including both gate capacitance and drain-to-source capacitance as well as any additional load capacitance from the attached mixer and added layout capacitance. G_{ds} is the total output transconductance in each of the transistors M_1 , M_2 , and M_3 .

The negative transconductance terms result from the cross-coupled transistors, $M_{2,5}$, in parallel with the active load transistors, $M_{1,4}$. This equation allows for the sizing of transistors in such a way as to reduce the $(g_{m_1} - g_{m_2} + G_{ds})$ term to zero and resulting in a VCO whose free-running oscillation frequency is solely dependent on the intrinsic capacitance of the transistors and a single transistor transconductance (M_3), as seen in the equation below.

$$f_{osc} = \left(\frac{1}{2\pi}\right) \cdot \left(\frac{g_{m_3}}{C_{tot}}\right) \quad (3.5)$$

This VCO can be implemented using two Maneatis delay-cells connected in quadrature with no external passive components, reducing the complexity of the design. By using this design the oscillator utilizes only transistors, with small feature sizes compared to passive components, reducing the occupied die area. Additionally, the free-running oscillation frequency of this VCO design is primarily constrained to the sizing and operating point of M_3 , allowing for easy tuning. As a result the oscillator occupies little die area.

Due to the quadrature connection, each of the two cells contributes a 90° phase shift, with the final 180° phase shift necessary for oscillation contributed by a cross-coupled connection from the output of the oscillator back to the input, which satisfies the first Barkhausen criteria.

Frequency control of the VCO can be achieved by controlling either the bias current or supply voltage, with the former preferred for practical applications where the supply voltage rail is likely to be shared among other components as well. This allows the oscillator to operate over a relatively large tuning range, allowing for guaranteed functionality while operating in the low-power moderate inversion region. This region of operation is more susceptible to process, voltage, and temperature (PVT) variations; hence a wide tuning range allows for more robust operation. A more detailed discussion on PVT variations and how they are dealt with will be found in Chapter 5. The first tuning method is to vary the bias current by changing the voltage at the gate of the current source, M_7 . This is used to adjust the free running frequency of the oscillator in implementation, allowing it to be more easily injection locked to the incoming signal.

This can be adjusted via an on-chip generated control voltage, or more likely, by segmenting and digitally controlling a current mirror to increase/decrease bias current as needed. Using this tuning method, the oscillation frequency changes nearly linear with the applied change in the input current as the transconductance varies linearly with the bias current for a given overdrive voltage (e.g., $g_m = 2I_D/V_{ov}$). The free running frequency of the oscillator and VCO power consumption is plotted versus input bias voltage is shown in Figure 3.5.

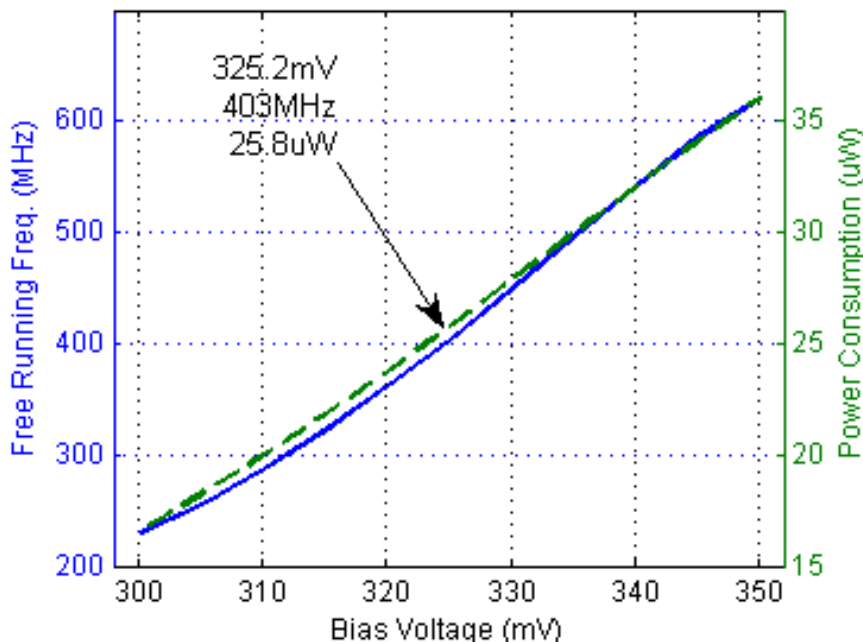


Figure 3.5: VCO free running tuning range (solid line) and power consumption (dashed line) as a function of the biasing voltage.

As can be seen in Figure 3.5, this design allows for a wide range of tunability, covering the entirety of the MedRadio frequencies and allowing for correction due to PVT variations.

The cell can be injection locked by inserting a frequency-dependent current into the I_{inj} port (Figure 3.3). Simulations have validated functionality for injection signals as small as $1\mu\text{A}$. Injection locking is an important component of this design as it is necessary to create the delayed version of the signal needed for demodulation, as described in Chapter 3.1. The added benefit of injection locking also reduces oscillator phase noise compared to its free-running state.

3.3 Passive Mixer

To minimize power consumption in the receiver chain, a passive ring mixer has been selected. The passive ring (Figure 3.6) is implemented using NMOS transistors to allow for the best tradeoff between size and switching resistance.

The mixer consists of a set of alternating switches that are selectively opened and closed

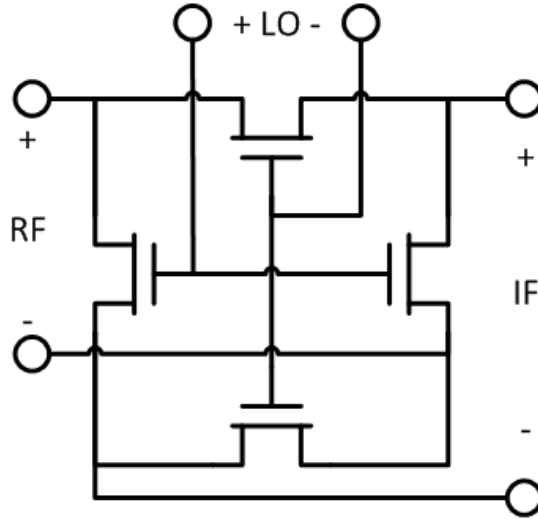


Figure 3.6: Passive ring mixer schematic.

by the injected signal and delayed replica, multiplying the two signals. This can be better visualized in Figure 3.7, where the control lines for each switch are labeled. By mixing a BPSK signal and a frequency matched signal separated in phase by 90° there are only two states in which the mixer will primarily operate; where the LO and RF signal are separated by $\pm\pi/2$. As a result of this constant phase different at any time only one set of alternating switches is open while the other is closed, depending on whether the LO leads or lags the RF signal. This produces a demodulated signal which is completely binary and can be passed to digital circuitry directly after any noise filtering, buffering, and level shifting which may be needed.

While decreasing transistor size for the mixer yields a better conversion gain, it results in more noise at the output of the mixer, creating the need for better filtering before a demodulated signal can be passed to the digital circuitry. The theoretical limit on conversion gain for a passive mixer is given as [12]:

$$20 \log \left(\frac{2}{\pi} \right) = -3.9dB \quad (3.6)$$

This theoretical limit is achieved under the assumption of the gates being driven by

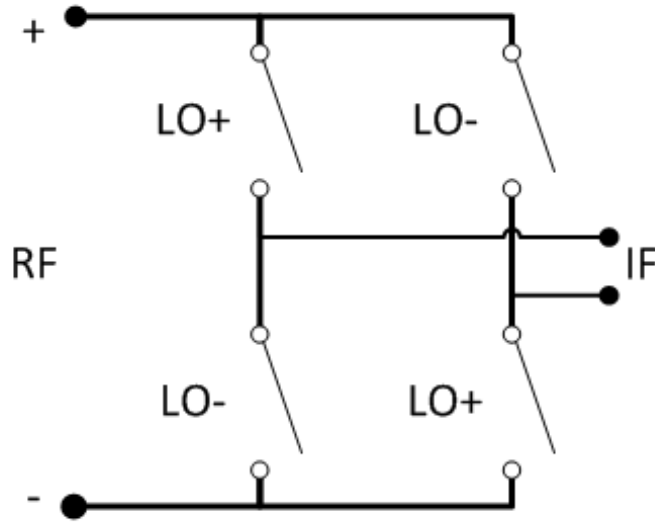


Figure 3.7: Passive ring mixer visualizes as alternating switches

square waves, to minimize transition times as well as a negligible on-state resistance. The equation below shows the relationship of the on state resistance, r_{ds} , to relevant device properties [13].

$$r_{ds} \propto \frac{L}{C_{ox}W} \quad (3.7)$$

By having a device with a short channel length and large width, r_{ds} is minimized. However, the gate oxide capacitance, C_{ox} is linearly dependent on the area of the device. A larger area creates a slower device than a smaller one would for a given drive current, but contributes to less noise at the output, with the oxide capacitance acting as a filter. In this design a minimal L was chosen with a channel width which ensured that the output signal would be sufficiently large that a signal could be reliably read, and further amplified after the mixer, without contributing significant noise.

In choosing the passive mixer, a tradeoff was made between gain and power consumption. For applications where more gain is necessary, an active mixer can be employed consuming similar power to the VCO delay cells [12].

Chapter 4

Design Methods

As the trend toward transistor miniaturization continues, the conventional equations that describe long-channel devices are becoming less accurate. Using conventional hand calculations for short channel devices results in longer time designing and simulating circuits. The g_m/I_D method of circuit design allows for increased accuracy in the design of circuits that utilize short channel devices by creating look-up tables where the device operating criteria, current consumption and operating region are defined without specifying a geometry. The following chapter describes the g_m/I_D methodology as well as how it is used in this design.

4.1 g_m/I_D Design Method

With the focus on low power design for this work, and in mobile devices in general, it is crucial to be able to design transistors in a manner which places them in ideal states for power efficiency. In CMOS circuits a transistors operates with highest energy efficiency in weak inversion, but it is slow. Hence, the region of moderate inversion provides a good trade off between efficiency and speed [14].

Traditional square-law hand calculations are typically used under the assumption that the devices are operating in strong inversion; These equations are not accurate in weak and moderate inversion. This is because standard square-law hand calculations do not take into account short-channel effects that start to become dominant in deep-submicron CMOS. Under weak and moderate inversion the device operates under an exponential relationship rather than one that is easily modeled with the square-law relationship, which assume the

device is completely off until the threshold voltage reached. For these reasons square-law hand calculation methods of design yield inaccurate results under these operating conditions and require significant amounts of simulation and adjustment before a suitable solution is reached [15].

The g_m/I_D design methodology makes use of the relationship between g_m/I_D and the normalized drain current $\hat{I} = I_D/(W/L)$, both of which are geometry independent. The relationship of these two quantities can be expressed as the following equation [15].

$$\frac{g_m}{I_D} = \frac{1}{I_D} \frac{\partial I_D}{\partial V_G} = \frac{\partial(\ln I_D)}{\partial V_G} = \frac{\partial \ln \left[\frac{I_D}{(W/L)} \right]}{\partial V_G} \quad (4.1)$$

At weak inversion the value of g_m/I_D is maximized, and lower values indicate increasingly stronger inversion. Using this single value as a primary design variable the operating point of a transistor can be determined, ensuring efficient operation. The g_m/I_D and \hat{I} relationship can be determined by extracting the values through software simulation. Utilizing a simple circuit configuration, shown in Figure 4.1, SPICE can be used to extract variables of interest at different bias conditions.

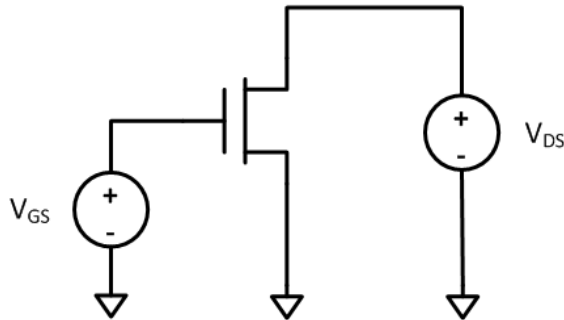


Figure 4.1: g_m/I_D test circuit for device information extraction

OCEAN is a scripting language specific to circuit simulation which allows for a circuit to be simulated over a range of values and store desired circuit conditions and device information to a file. The OCEAN script is used to simulate and extract any information about the device, such as intrinsic capacitance, gain-bandwidth products (f_t), and most

importantly g_m/I_D and g_{ds}/I_D values. These values are automatically generated during the simulation, extracted using the OCEAN script, and stored in a file specific to the technology being used. By varying V_{GS} , V_{DS} , and L , for a fixed width a single device can be characterized completely, and as g_m/I_D is geometry independent it can be applied to any transistor in the same technology. The OCEAN script used for this work is presented in appendix B

This method is a powerful tool for design as it allows for scripting of designs where changes can be readily made. Additionally, the same circuit can be designed in different fabrication processes, provided devices in those processes have complete simulation profiles. The method of combining the g_m/I_D design method with the process of extracting relevant data from simulation models, and applying them into a script is detailed in the following section.

4.2 MATLAB Scripting

One of the most powerful benefits of the g_m/I_D method is combining it with sizing scripts. The OCEAN script used in simulation software extracts any value which can be obtained from the simulation model. With this data readily available precise scripts can be used to design the device size based on a few key parameters.

For this work MATLAB was used to write a script for the sizing of transistors in the Maneatis cell. Using equations as described in section 3.2, a full VCO can be designed using this combination of the g_m/I_D method and a MATLAB script, taking significantly less time than hand calculations and with increased accuracy.

Before a full sizing script could be written, two important other pieces of code were developed. The first of which is a technology file parsing script, and the second is a set of look-up functions. The parsing script is used to neatly organize the desired information that has been extracted and saved using OCEAN script. In addition to organizing the data in such a way that MATLAB scripts can easily look up the information many of the values

have either been scaled and some data points combined to make more relevant values. As an example the intrinsic drain capacitance, C_d , is the combination of several values from the simulation data (C_{dd} , C_{jd} , C_{cgs} , C_{ovljd}). Since none of these values alone are used in this design, but C_d is desired, it is preferable to be able to extract the single value in the sizing script rather than all four and combine them afterward.

The contents of this resultant technology data file are geometry independent parameters (e.g., current density, I_D/W ; transconductance efficiency, g_m/I_D). A sample of this script used is shown in appendix C, showing how C_d and g_m/I_D values are added to the technology data file. All values extracted from the OCEAN script are added to the technology file similarly.

The look-up functions provide a means to take data from the simulated technology file based only on relevant values needed to find that information. As an example the look-up function for the value of g_m/I_D required knowing the device type, channel length, V_{ds} , and the desired oscillating frequency. Using this information two new arrays are then created, containing g_m/I_D and the desired frequency values corresponding to given L and V_{ds} values. The intersection of these two matrices is where a single g_m/I_D value exists for the given valuables. An example of a look-up function is presented in appendix D. This function finds a g_m/I_D value for a given technology file, device type, channel length, v_{ds} , and frequency of operation. Like the parsing script, functions for different desired values are written similarly.

Using the look-up functions a MATLAB script that is written to automate the sizing of the VCO and the mixer transistors. This script also takes into account the mixers loading effect on the VCO as well as the added capacitive load of the layout traces. The script uses desired properties of the delay cell (e.g., total power consumption, desired oscillation frequency, and the desired operating points of the transistors in both the mixer and VCO) to look up information in the tables to satisfy such conditions. For optimal power efficiency the g_m/I_D values of the VCO were chosen in a region of moderate inversion, 23 V^{-1} .

Layout capacitance was estimated based on the cell size and the cell geometries were

iterated in the script until a satisfactory solution was achieved that optimized the design for low power at the desired operating frequency.

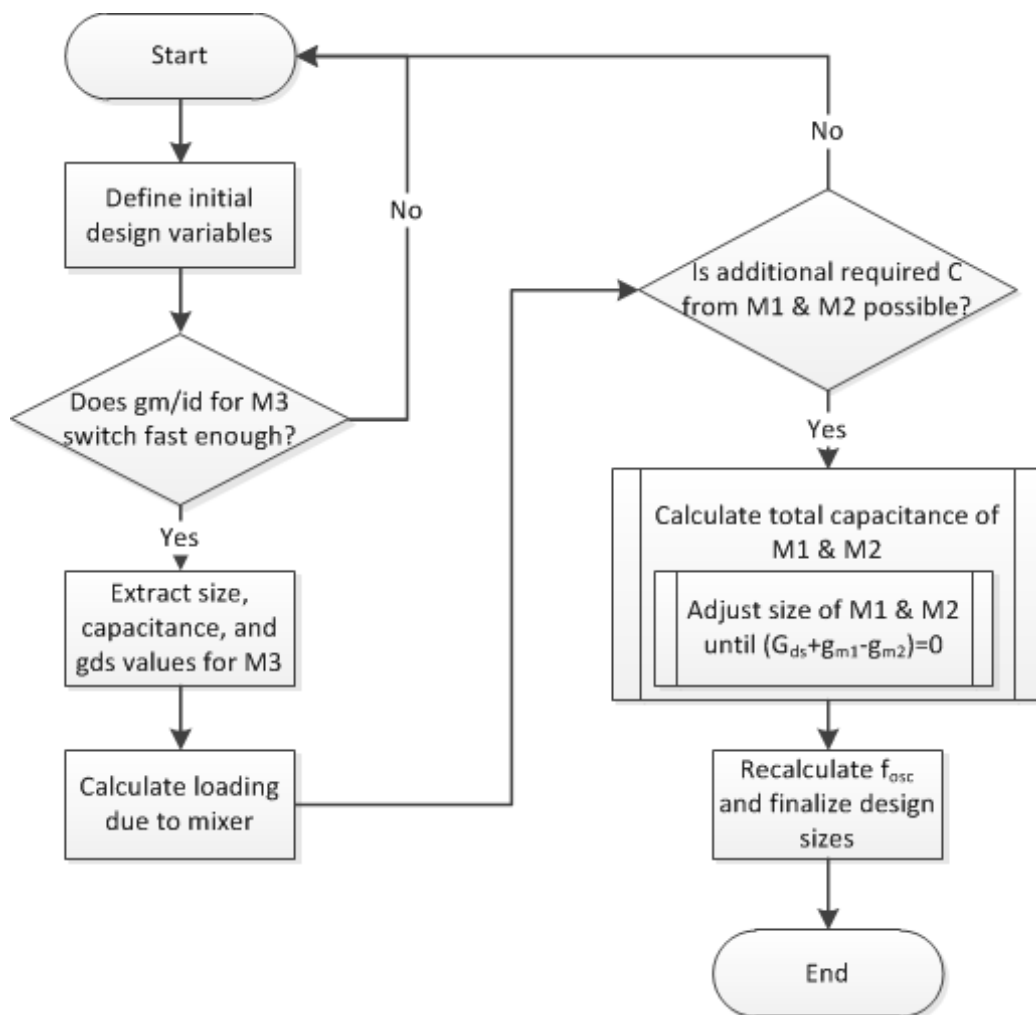


Figure 4.2: MATLAB sizing script flow diagram

Figure 4.2 shows the general flow chart of how the sizing script functions. After deciding the initial operation conditions, i.e. current consumption, desired oscillation frequency, device lengths, and g_m/I_D values for the switching transistors M_3 and M_6 , the first check is if the desired operation point allows for a fast enough transistor. The transition frequency, f_T , of a transistor is the frequency at which the short-circuit current gain is unity, and goes down as the device is closer to weak inversion. The goal being to find a device with as high a g_m/I_D value possible that will allow for the design to operate, ensuring optimal power efficiency. The test condition is placed such that the device is not operating at f_T , as

this would potentially cause the design to be unstable. Figure 4.3 shows the relationship of g_m/I_D to f_T for the NMOS device in this technology. Looking at this figure for a 120 nm device 25 V^{-1} is approximately the boundary between weak and moderate inversion where the device is able to start switching at speeds suitable for this design. As such a value of 23 V^{-1} is chosen. Ensuring this the f_T condition is met, the I_D/W value for this device can be found and since I_D is explicitly chosen for the width of the device can be calculated. This also gives the precise g_{m3} value which is a crucial design value, as seen in equation (3.5).

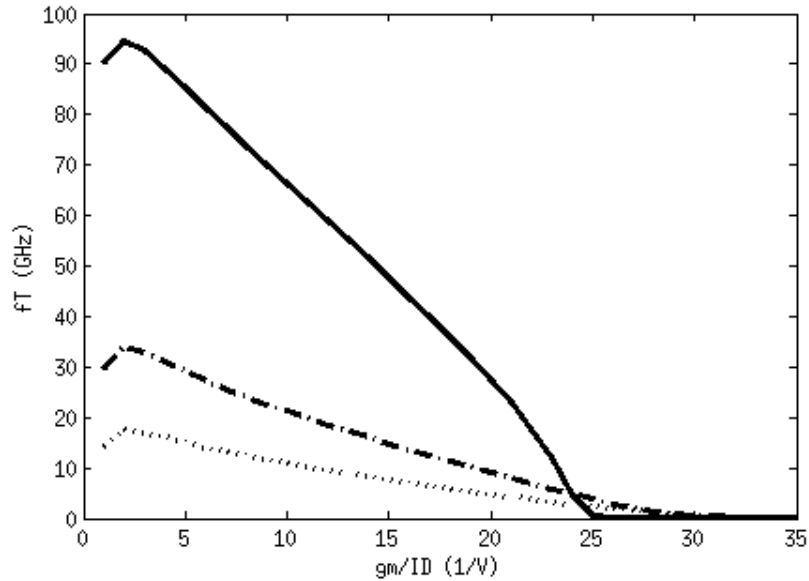


Figure 4.3: This plot shows the relationship of g_m/I_D to f_T for a NMOS device in the technology being used. The three lines are for device lengths of 120 nm (solid), 240 nm (dashed), and 360 nm (dotted).

The mixer is designed to operate at a g_m/I_D value of 37 V^{-1} to ensure that the devices operate efficiently at low power, and as switching speed is not a large concern they can operate in weak inversion as opposed to moderate inversion. Using this value the capacitive load to the VCO added by the mixer, C_{mixer} , can be calculated.

Knowing g_{m3} and the intrinsic capacitance associated with M_3 , also calculated using look-up scripts, the required total capacitance of the remaining capacitors, M_1 and M_2 , is calculated.

$$C_p = \frac{g_{m3}}{2\pi f_{osc}} - C_{fixed}$$

C_{fixed} is the total loading capacitance of the now fixed design values, transistor M_3 and the mixer and C_p is the required capacitance needed from the transistors M_1 and M_2 to satisfy oscillation conditions at the given frequency.

$$C_p = C_{dM1} + C_{dM2} + C_{ggM1} + C_{ggM2}$$

To calculate the capacitive contribution of M_1 and M_2 the g_m/I_D for both is considered as single design variable, and then the ratio of sizing M_1 to M_2 is considered to verify that the term $(g_{m_{p2}} - g_{m_{p1}} + G_{ds})$ goes towards 0. In implementation the condition was placed that the term go toward -1 to ensure that the design oscillated under PVT variations.

The final devices obtained from the script are shown in Table 4.1. Table 4.2 lists the capacitance values found and used in the MATLAB script as well as the layout capacitance estimated for this design. Operating current, I_D , for a single delay cell was used as $10.63 \mu\text{A}$. The full MATLAB script can be seen in appendix E

Table 4.1: Device sizes for design, generated using MATLAB sizing script

Device	Width (μm)	Length (μm)
VCO		
M1,M4	0.66	0.24
M2,M5	0.93	0.24
M3,M6	2.10	0.12
M7	2.58	0.36
Mixer		
M1-4	9.6	0.12

Table 4.2: Capacitance values found and estimated using the MATLAB design script

Device	$C_g(fF)$	$C_d(fF)$
VCO		
M1,M4	1.25	0.95
M2,M5	1.75	1.34
M3,M6	2.03	2.55
C_{mixer}		13.36 fF
C_{Layout}		22.4 fF

Chapter 5

Layout

The layout of both the VCO and mixer was guided by two main concerns, robustness against manufacturing and operating variations as well as the parasitic capacitance as a result of layout. Manufacturing and operating variations are most commonly referred to as process-voltage-temperature (PVT) variations. The largest concern for this design is maintaining efficient operation of the deicing under typical conditions while being able to function within the PVT variations of the 130 nm IBM process, the chosen process technology for this design. The following sections provide detail on PVT variations and how they were designed for using layout techniques. Layout was performed in the Cadence Allegro design suite.

5.1 PVT Variations

Voltage and temperature variations are associated with the supplied voltages and the operating temperature of both the environment and the circuit itself. In compliance with the FCC regulations for the MedRadio frequency band implantable radio devices must operate within the range of 25°C to 45°C, while body worn devices within 0°C to 55°C. For the purposes of this design we are considering an implantable device. Although this range is still large typical body temperature variation, including severe fevers and chills is limited to within approximately $\pm 3^\circ\text{C}$ of 37°C. For these reasons an implantable device is considered to be in a much more stable temperature environment than body worn and other commercial devices. As was shown in Figure 3.5, this VCO design is capable of a wide range of tuning to compensate for temperature variations.

Variations in the supply voltages can be caused by several factors, including temperature variations and noise in the supply circuit, but also by resistance of the supply rails and connection buses. The resistance of metal buses is typically low, but non-zero, and because of this across a whole circuit there may be small voltage drops between elements connected on different points of the bus, where in simulation ideal connections were considered. The circuit presented in this work is quite small and as a result such factors are not heavily considered. Additionally all simulations were performed post-extraction to ensure that any parasitic effects of the layout are taken into account. To help protect against noise in the supply circuit decoupling capacitors were placed between the supply and ground buses.

Layout extraction is a tool which analyzes the physical layout of the circuit for parasitic effects. Most important in this design is the added effect capacitance, as a result of connection buses and vias which connect different metal layers. Using this extracted information the VCO devices can be sized to compensate for the added capacitive load of the layout. Resistance added by metal connection buses is also part of post-extracted information which is used, but not explicitly compensated for in the design scripts, as it is assumed to be negligible in a small design such as this. Post-extracted simulations yield a more accurate results by taking these added layout effects into consideration over the ideal connection used in schematic entry.

Process variations are cause for more concern and thus, were more heavily considered during the design process. Process variations is the term given to general variations within the semiconductor fabrication process. These variations can come in the form of wafer inconsistencies such as doping concentrations and oxide thicknesses, as well as manufacturing inconsistencies (e.g., optical shadows, anisotropic etching, etc.) which can result in variations in the exact size of devices on the die. The 130 nm IBM design kit supplies process corner information, representing the typical expected characteristics of a fabricated die as well as the extreme deviation from this normal. The typical corner (TT) is what the design is optimized for, however it is important to ensure that operation is still possible in the

outlying corners for a PVT variation robust design. These corners are designated with two letters, the first for NMOS devices and the second for PMOS, and whether the device operates slower (S) or faster (F) than the typical conditions. Having these corners as simulation profiles allows for testing under PVT variations to ensure the design still operates, even if not under optimized conditions.

5.2 Common-Centroid Layout Technique

In order to ensure minimal impact from process variations a common-centroid layout (CCL) technique was used wherever possible. The result of this design method is that PVT variations across a die have a smaller impact on any one transistor, as any shift from the expected perimeters is re-distributed across the paired transistors in layout. This is specifically relevant to physical variation in device size and die properties, such as doping concentrations and oxide thickness. The physical differences in layout between normal and CCL techniques are shown in Figure 5.1.

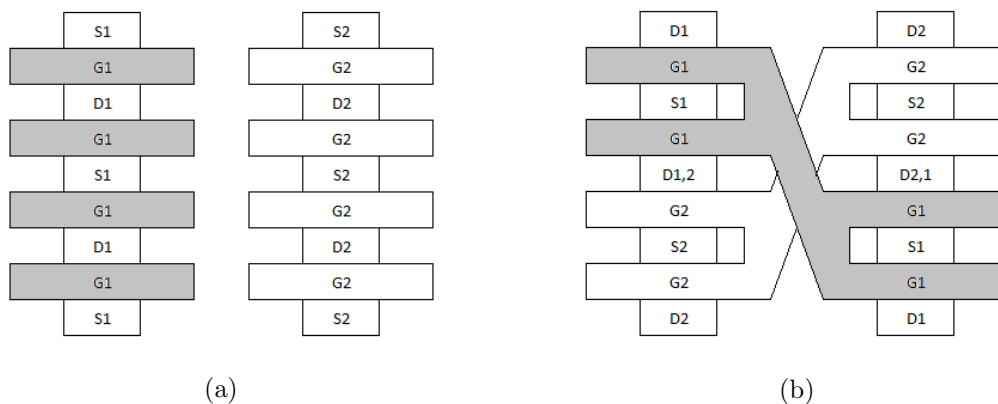


Figure 5.1: Two side by side transistors, with normal layout connections 5.1a and with common-centroid layout techniques 5.1b. The gate of transistor 1 is shown in gray.

As an example, looking at Figure 5.1, if there is a change in doping concentration from left to right in 5.1a the change in concentration will cause the two transistors to operate under slightly different bias points, and even though they should be matched one may be faster or slower than the other. Using CCL the terminals of each transistor appear on both

the left and right side of the layout and so even though the change in doping concentration will still effect the overall design, the impact on any single transistor is minimized as one half of a transistor will appear with higher dopant and the other with a lower concentration. The drawback of CCL is that the added crossing metal connections as well as added vias between metal layers results in additional parasitic capacitance. The capacitance of this design plays a critical factor in the overall circuit performance; however, the inclusion of this capacitance in the sizing script allows for compensation.

As shown in Figure 5.1b for CCL to be implemented the drain of the transistors must share a connection. This created a limitation such that it cannot be used in all situations. However where applicable this technique allows for more balanced circuit layouts that are more resistant to PVT variations. An added benefit of this layout method creates connection points on both sides of the pair, rather than forcing connection to M_1 on the left, and M_2 on the right, as would be the case for 5.1a. Additionally CCL requires that the transistors be implemented with a multi-finger layout, in which a single wide transistor is divided into a number of gate fingers, which add up to the total width.

Layout for the VCO can be seen in Figure 5.2. This layout consists of two separate Maneatis cells connected with buses running left to right in the middle of the cell. Connections between the different transistors within a single cell run up and down. A separate ground guard ring has been added to M_7 on each transistor as it is the transistor that dictates the current draw of the cell and the ring adds an extra level of noise isolation. CCL is used for transistors M_3 and M_6 as the drains are connected together, however for the load transistors ($M_{1,2}$ and $M_{4,5}$) do not share drain contacts so the CCL is modified by stacking the two loading transistors, but splitting the devices so they do not share drains when the cross connection is made. This still makes the two paths more resistant against variations, while not taking full advantage of CCL. The other reason for this modified CCL layout for the PMOS devices is that the devices are too small to be implemented with a multi-finger layout.

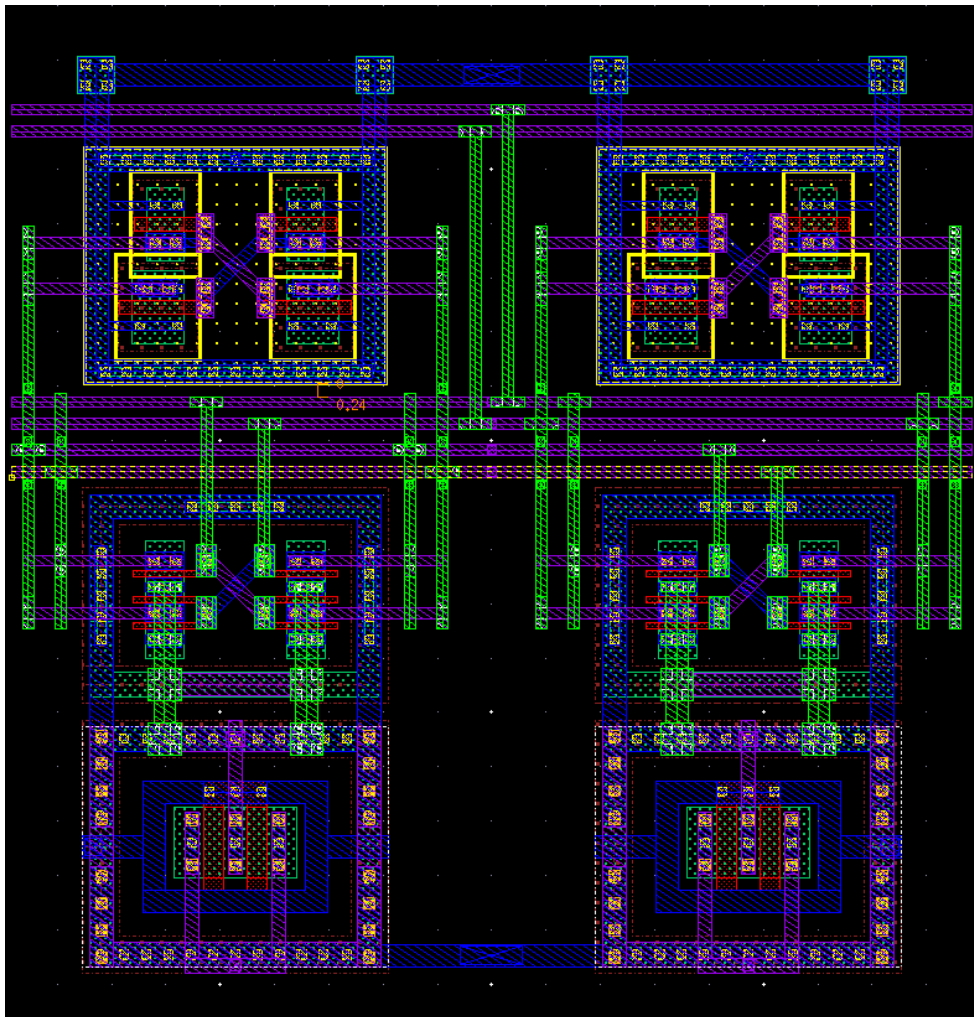


Figure 5.2: VCO layout, not including bypass and added capacitance

The layout for the mixer is shown in Figure 5.3. Similar to the VCO layout, CCL could not be used for the entire design, as only the bottom bottom pair of switches as seen in Figure 3.7 share a drain connection. As a result the mixer is more susceptible to mismatch due to PVT variations. Mismatch in the mixer is not desirable, however it only contributes to slight changes in conversion gain and output bias levels, but does not effect the ability to demodulate incoming signals. The mixer output requires a low pass filter, amplification, as well as DC level shifting even under ideal operating conditions, thus these variations can be accounted for in subsequent stages of the receiver. In the case of the VCO, mismatch can potentially cause two branches of the Maneatis cell to be different enough from each

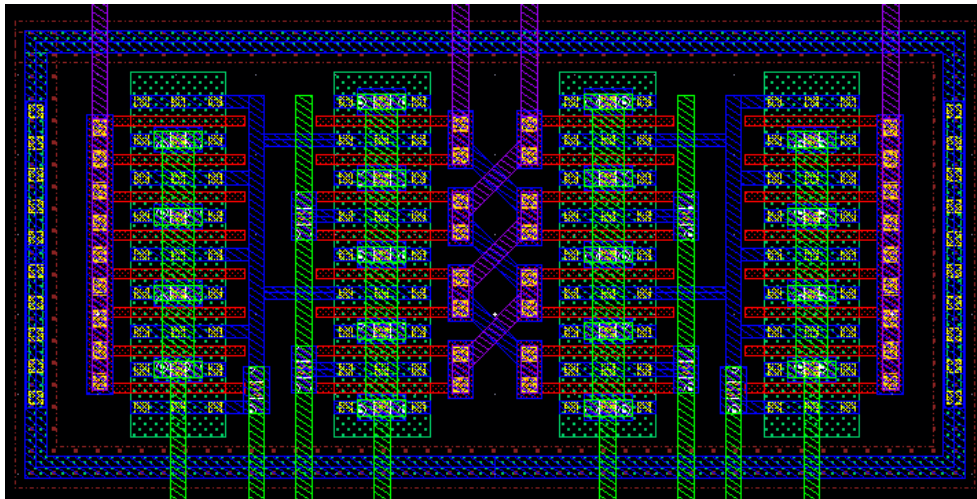


Figure 5.3: Mixer layout not including coupling capacitors and biasing resistors.

other as to prevent oscillation and so it is more crucial the CCL be used in that layout.

Only one branch of the VCO is connected to the mixer, the loading effect of the layout capacitance is different on the interconnections between the Maneatis cells. To compensate for this additional capacitors were added to the unconnected lines so as to more closely match the layout capacitance of the other pair. Additionally decoupling capacitors between the supply and ground rails were added for protection against noise in supply lines. The overall layout for the final design is shown in Figure 5.4. The VCO and mixer pair has been designed and extracted using an 8RF CMOS technology, in the 130 nm IBM process. The overall design is small and largely dominated by the added passive components. Total it occupies a die area of $49 \mu\text{m} \times 39 \mu\text{m}$, not including bonding pads.

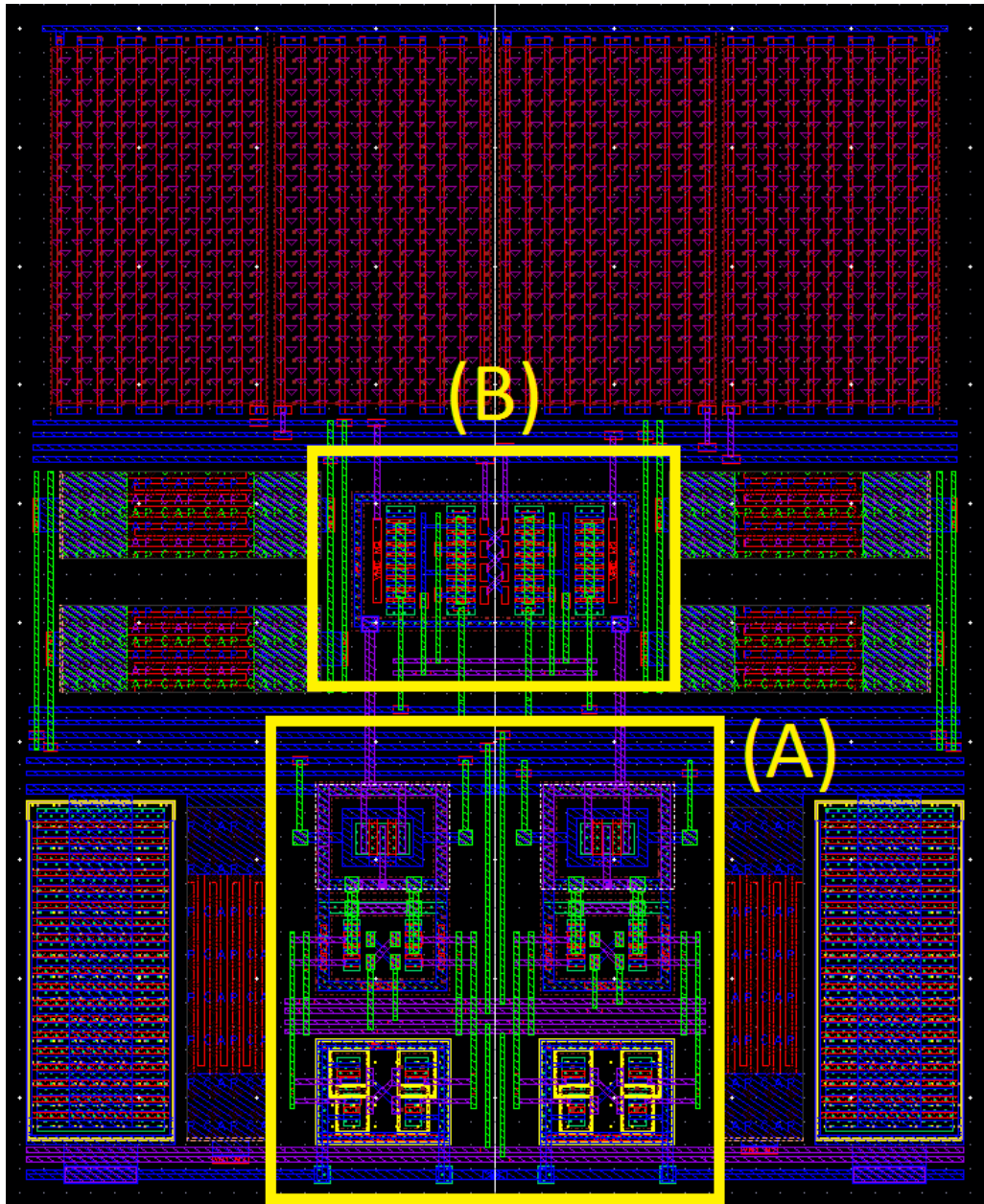


Figure 5.4: The layout for the (A) VCO and (B) mixer. The surrounding area contains bias, bypass, and coupling circuits.

Chapter 6

Results

This design has been extracted and simulated using the IBM 130 nm design process, and all simulations have been performed at 37°C as the intended use is for a medical implant. After transistors were initially sized using the MATLAB script, the layout was designed and extracted. Small changes to device sizes were implemented to meet the required layout rules. Those changes are presented in Table 6.1. The extracted layout information added layout capacitance and any resistance added by the connection buses. With the post-extracted information adjustments were made to the DC current consumption of the VCO sizing and of the two circuits to ensure functionality in the three process corners (TT, SS, FF).

6.1 VCO

The post-extracted simulation results of the circuit show that the overall design consumes 25.8 μA with a supply voltage of 1 V, at the TT design corner. The design script called for a g_m/I_D value of 23 V^{-1} , however after adjusting for the included parasitic effects found in posted extracted simulation M_3 and M_6 operate at 21.7 V^{-1} .

The circuit has also been tested in fast and slow fabrication corners. An adjustment of the biasing current and supply voltage has been shown to allow the design to operate across expected PVT variations. By adjusting the two tuning parameters, the supply voltage and current consumption of the delay cells the VCO has been demonstrated to oscillate at the SS corner with a post consumption of 50 μW . In the SS corner M_3 and M_6 are operating at a g_m/I_D value of 20.8 V^{-1} in order to satisfy conditions of oscillation at the same size

Table 6.1: Device sizes for design used in layout.

Device	Width (μm)	Length (μm)
VCO		
M1,M4	0.6	0.24
M2,M5	0.96	0.24
M3,M6	2.13	0.12
M7	2.6	0.36
Mixer		
M1-4	9.6	0.12

device. To start oscillation at the SS design corner the supply voltage must be 1.8 V and supply current for the VCO is $27.8 \mu\text{A}$.

At the FF design corner the design skews in the other direction. The supply voltage must be adjusted to 0.825 V and the supply current is changed to $19.7 \mu\text{A}$. Under these conditions transistors M_3 and M_6 operate at a g_m/I_D value of 22.43 V^{-1} . Although at this corner the g_m/I_D point shows the device working in weaker inversion than the TT corner, the reduction in current and supply voltage generates a output signal half the strength of than in the TT corner. This has a significant effect in the mixer stage, as will be explained in the section 6.2.

Both the FF and SS design corners can be made to satisfy oscillation conditions by adjusting the current and supply voltage to the VCO, however in both corners the transistors will be operating in heavier inversion than the TT corner, for which the sizing was optimized. Operating conditions have been summarized in Table 6.2.

Table 6.2: Operating conditions in the three primary design corners for free running oscillation of 403 MHz

Corner	Supply Voltage	Current Consumption	$M_{3,6} g_m/I_D$	Output Voltage Amplitude
TT	1 V	$25.8 \mu\text{A}$	21.7 V^{-1}	110 mV
SS	1.8 V	$27.8 \mu\text{A}$	20.8 V^{-1}	120 mV
FF	0.825 V	$19.7 \mu\text{A}$	22.4 V^{-1}	56 mV

The VCO achieves a free running oscillation frequency of 403.5 MHz at a typical process corner, with an associated phase noise of -51.8 dBc/Hz using a 1 MHz offset. When the

VCO is under injection lock conditions, using a 1 μA IL signal, the phase noise improves to -118 dBc/Hz, using an ideal injection lock signal. The free running and injection locked VCO phase noise characteristics are shown in Figure 6.1.

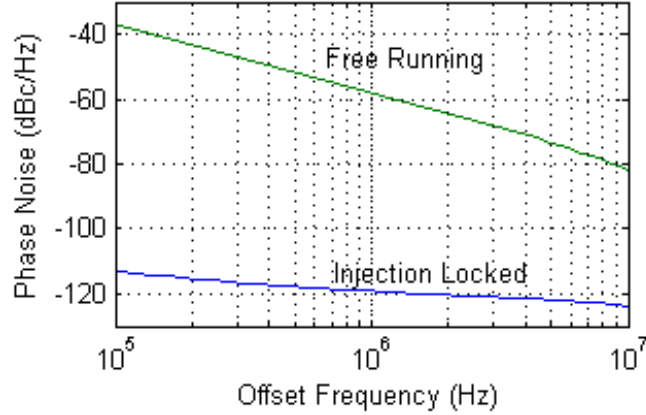


Figure 6.1: VCO phase noise characteristic under free running and injection locked conditions.

To compare this work with other VCOs, the figure of merit (FOM) by which different VCO designs can be compared is given below as equation (6.1). This FOM captures the two most critical design criteria for a VCO, power consumption and phase noise.

$$FOM = 10 \log \left[\left(\frac{F_{osc}}{F_m} \right)^2 \frac{1}{P} \right] - L \{F_m\} \quad (6.1)$$

F_{osc} and F_m are the oscillation frequency and offset frequency, respectively, P is power consumed, in mW, and L is the phase noise at the offset frequency, in dBc/Hz, most commonly taken at a 1 MHz offset [16]. This FOM is the general formula applied to VCOs, however heavily leans towards showcasing improvements in phase noise rather than power consumption. The aim of this design was to implement a VCO with as small a power consumption as possible and phase noise was not as heavily considered. As a result the FOM for this design is higher than other works listed, while operating at a power point at least an order of magnitude lower.

As can be seen in Table 6.3, this design exhibits a good phase noise while operating at

Table 6.3: Figure of merit comparison between this work and other published works. ¹ is operating as a free running VCO, while ² is injection locked. ³ results are simulation based while others are measurement based.

	Frequency (MHz)	Phase Noise (dBc/Hz)	Freq. Off-set (MHz)	Power (mW)	FOM
This Work ^{1,3}	403.5	-51.8	1	0.0258	119.8
This Work ^{2,3}	403.5	-118.9	1	0.0258	186.9
[5]	403.5	-95.6	1	0.85	148.4
[7]	403.5	-107	0.1	1.2	178.3
[16]	5650	-88	1	5	156.1
[17] ³	400	-93.5	0.16	1.05	161.3

a low power point. [16] has been include as a comparison point as the design outlined in it also uses a two-stage Maneatis cell VCO, while the other papers utilize different delay cells and configurations. This work as well as [17] use simulation based results, while the other works listed in Table 6.3 are measured results.

6.2 Mixer

The passive mixer achieved an overall voltage conversion gain of -18.4 dB and 1 dB compression point of -13.5 dBm, as shown in Figure 6.2. This conversion gain is far below the theoretical limit (\approx -4 dB) obtained when square-wave drive signals are used. To obtain a conversion gain closer to the theoretical limit a VCO output signal closer to 10 dB would need to be generated. Such a signal would result in a significant power consumption increase in the VCO. As was shown in Table 6.2, even though under the FF design corner the VCO operates in lower inversion, making it more efficient, the output is nearly half of the TT corner. Looking at Figure 6.2 this would translate to a mixer conversion gain likely close to -21dB. This decrease in power would contribute to added noise in the output signal, necessitating for the need of more filtering and amplification at that output.

The 1dB compression point is used as a reference point where increased RF signal power no longer produces a larger output signal, and is used for non-linear circuits. For MedRadio

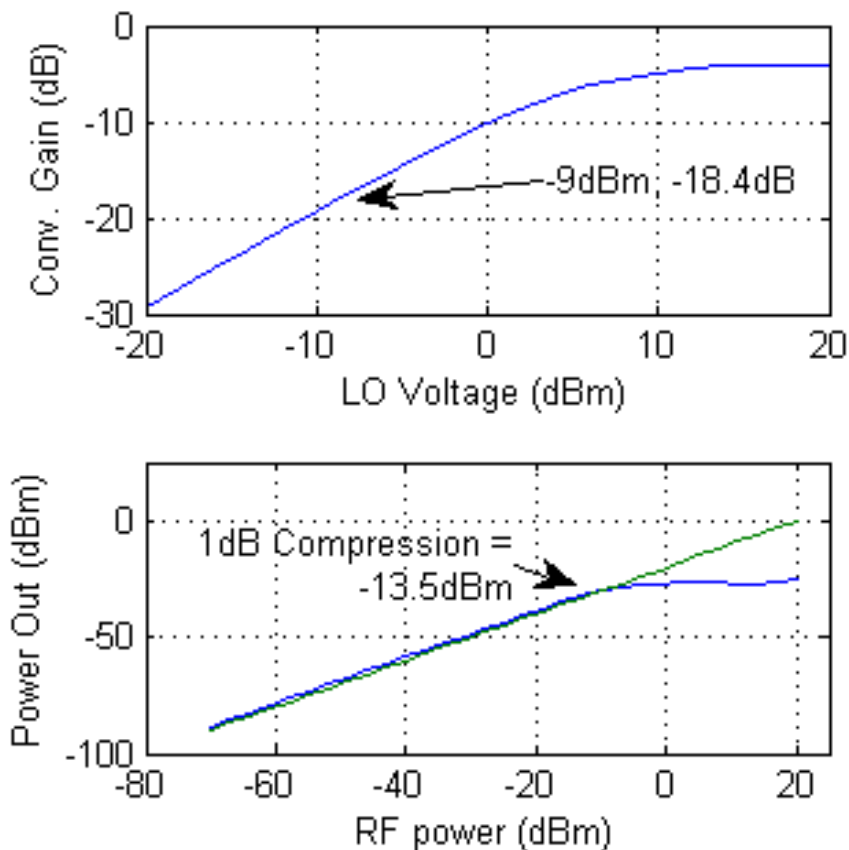


Figure 6.2: Simulated mixer conversion gain as a function of VCO input power (top), and 1 dB compression point (bottom).

the 402-405 MHz band transmission is limited to $25 \mu\text{W}$ EIRP, or -16 dBm, and it is expected that a received signal will be below this after transmission loss. With a 1 dB compression point of -13.5 dB amplification of the incoming signal is required for a higher demodulated output signal. The choice of a passive mixer was a tradeoff between power consumption and conversion gain. The effects of this tradeoff can be mitigated by using a high-gain LNA, or by using an active mixer.

6.3 System Performance

Post-extracted simulations have demonstrated that this system is capable of demodulating a PSK signal. MATLAB is used to generate a piecewise linear voltage signal carrying the

DPSK at the desired injection locking frequency. The signal is used as the injection locking frequency for the oscillator and to drive the RF port of the designed mixer. Representative input/output waveforms are shown in Figure 6.3.

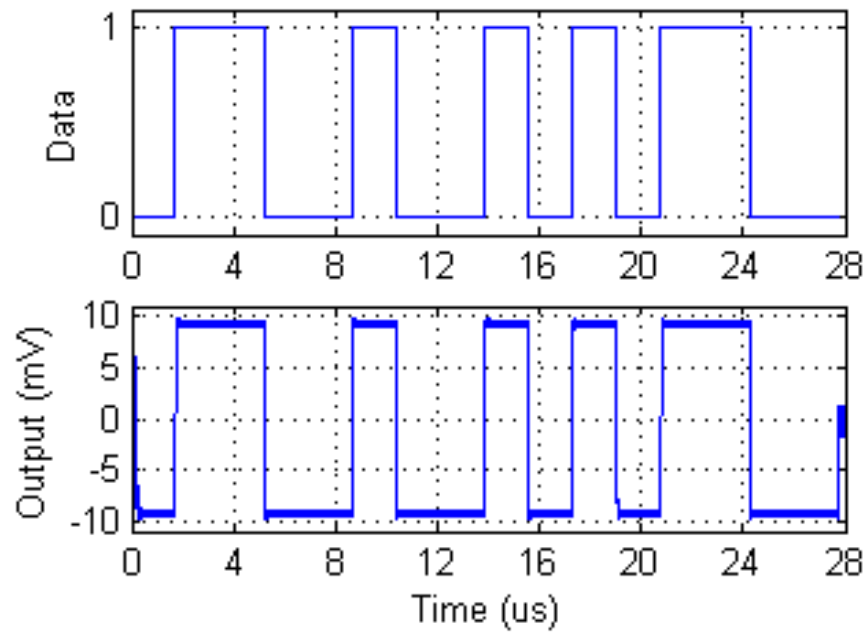


Figure 6.3: Simulated mixer output with the injection-locked VCO.

Chapter 7

Conclusion and Future Work

This thesis has shown the design of a low power VCO and mixer used to implement a MedRadio band DPSK demodulator using the 130 nm IBM device library. Using the geometry independent g_m/I_D design methodology, combined with MATLAB scripts a design process was developed which allows for the fast and accurate sizing of device sizes. This process is additionally advantageous over traditional hand-calculations as it can be adopted for similar implementations of the same design under different perimeters. Similarly the design can be easily transferred to a different fabrication technology using these scripts with minor modifications.

Layout for the design was implemented with the goal of system reliability against fabrication variations. To realize this, common-centroid layout techniques were used whenever possible to ensure even distribution of process variations, also contributing to even distribution of connections and parasitic layout non-idealities. The overall $49 \mu\text{m} \times 39 \mu\text{m}$ of die area required for this design include the circuits, complete with supply voltage decoupling capacitors, and easily accessible connection buses, for test pads as well as future connectivity to the remaining elements of the receiver not included in this work.

The VCO was designed as a two-stage ring, based on a Maneatis delay cell, has a phase noise of -118 dBc/Hz at 1 MHz offset when injection locked, and requires no external passive components, making it ideal for MedBand where size is a key component. With the goal of reduced power consumption a passive ring mixer has been chosen in the design, and has been shown to have a conversion gain of -18.4 dB. The VCO and mixer configuration is used as a Vance demodulator, and is a simple and power efficient solution to DPSK signal

demodulation. Targeted for low power applications, the design consumes $26 \mu\text{W}$ of power from a 1 V supply. Future work will verify this design in fabrication as well as expand the current work to a full receiver with the integration of a Balun, LNA and receiver back-end.

Appendix A

VCO Circuit Analysis

Looking at Figure 3.3 and starting with a difference-mode analysis, the following circuit diagram can be drawn.

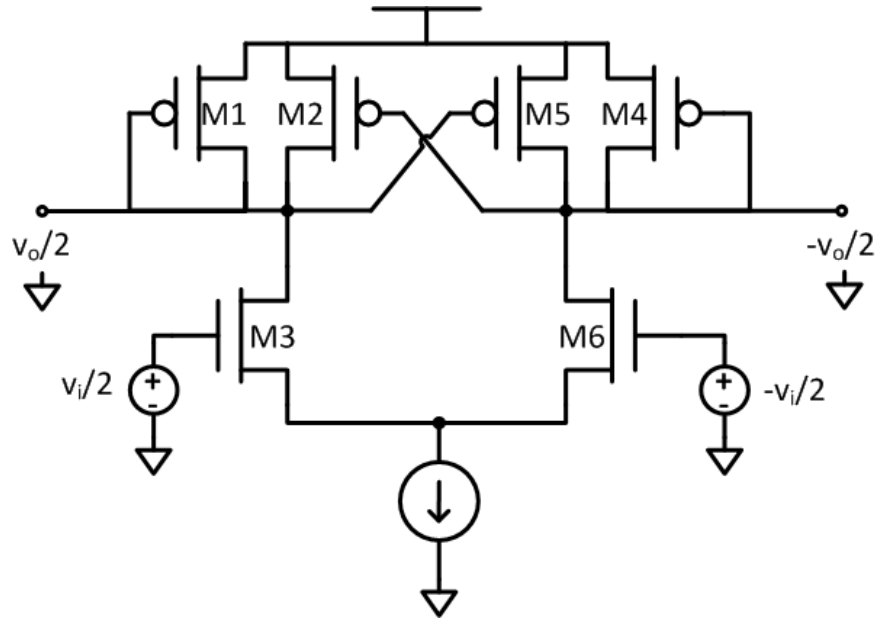


Figure A.1: Common mode realization of Maneatis cell

Analyzing the half circuit, the circuit can then be re-drawn using small signal model resulting in a small signal diagram as shown in Figure A.2.

The formula which comes from the small signal diagram can be written as

$$\frac{v_i}{2} = g_{m3} \cdot \frac{v_o}{2} \cdot \left(\frac{1}{g_{ds3} + g_{ds1} + g_{ds2} - g_{m2} + g_{m1} + Z_L} \right) \quad (\text{A.1})$$

For simplicity all g_{ds} terms will be grouped into a single term, G_{ds} . Z_L represents the load of the cell, which in this design is only the intrinsic capacitance associated with each

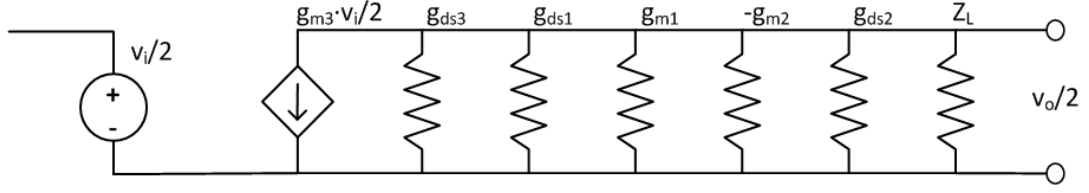


Figure A.2: Small signal realization of Maneatis cell

transistors.

$$Z_L = sC_L = s(C_{gs3} + C_{ds3} + C_{ds2} + C_{gs5} + C_{gs1} + C_{ds1}) \quad (\text{A.2})$$

Since either side of the cell is sized to be symmetric C_{gs5} can be substituted with C_{gs2} . Substituting in and simplifying we get the following.

$$H_c(s) = \frac{g_{m3}}{(G_{ds} - g_{m2} + g_{m1}) + sC_L} \quad (\text{A.3})$$

However because of how the full VCO is connected, shown in Figure 3.4, the full transfer function is shown below. The negative term is a result of the crossed connections from the second stage back to the first.

$$H(s) = -(H_c(s)^2) = -\left(\frac{g_{m3}}{(G_{ds} - g_{m2} + g_{m1}) + sC_L}\right)^2 \quad (\text{A.4})$$

The Barkhausen criteria state the two necessary conditions for a ring oscillator are the the total loop gain muse be unity, and that total phase shift must be $2\pi n$ where n is a whole integer. By applying the first criteria we get that:

$$|H(s)| = \text{Rel}[H(s)] = \text{Rel} \left[- \left(\frac{g_{m_3}}{(G_{ds} - g_{m_2} + g_{m_1}) + sC_L} \right)^2 \right] \quad (\text{A.5})$$

$$= \text{Rel} \left[\frac{-g_{m_3}^2}{(G_{ds} - g_{m_2} + g_{m_1})^2 - (\omega C_L)^2 + 2sC_L(G_{ds} - g_{m_2} + g_{m_1})} \right] \quad (\text{A.6})$$

$$= \frac{-g_{m_3}^2}{(G_{ds} - g_{m_2} + g_{m_1})^2 - (\omega C_L)^2} \quad (\text{A.7})$$

$$= \frac{g_{m_3}^2}{(\omega C_L)^2 - (G_{ds} - g_{m_2} + g_{m_1})^2} = 1 \quad (\text{A.8})$$

Solving for ω we find that

$$\frac{g_{m_3}^2}{(\omega C_L)^2 - (G_{ds} - g_{m_2} + g_{m_1})^2} = 1 \quad (\text{A.9})$$

$$\omega = \sqrt{\frac{g_{m_3}^2 + (G_{ds} - g_{m_2} + g_{m_1})^2}{C_L^2}} \quad (\text{A.10})$$

Appendix B

Sample OCEAN script

```

simulator( 'spectre )
design(  "/scratch/ilyac/simulation/ocean_tutorial/spectre/
        schematic/netlist/netlist" )
resultsDir(  "/scratch/ilyac/simulation/ocean_tutorial/spectre/
            schematic" )
path(  "/vlsipdks/ibm/ibm8rfcmos/IBM_PDK/cmrf8sf/relDM/Spectre/
        models" )
modelFile(
    '( "/vlsipdks/ibm/ibm8rfcmos/IBM_PDK/cmrf8sf/relDM/Spectre/
        models/design.scs" "" )
    '( "/vlsipdks/ibm/ibm8rfcmos/IBM_PDK/cmrf8sf/relDM/Spectre/
        models/allModels.scs" "tt" )
)
stimulusFile( ?xlate nil
    "/home/ilyac/simulation/ocean_tutorial/spectre/schematic/
        netlist/stimuli/mosDC.scs"
)
analysis( 'dc ?saveOppoint t ?param "vgs" ?start "0.05"
        ?stop "1.5" ?step "0.01" )
desVar( "vgs" 1)
desVar( "L" 0.12u)

```

```

save( 'i "/T0/D" "/V0/MINUS" )
temp( 27 )
paramAnalysis( "L" ?start 0.12u ?stop 1u ?step 0.02u
)
paramRun()
gdsoverid = getData("T0:gds" ?result "dc")/getData("T0:ids" ?
    result "dc")
ft = ((1 / 6.28) * (getData("T0:gm" ?result "dc") / (getData("T0:
    cgg" ?result "dc") + getData("T0:covlgb" ?result "dc") +
    getData("T0:covlgs" ?result "dc"))))
avo = (getData("T0:gm" ?result "dc") / getData("T0:gds" ?result "
    dc"))
ftavo = ft*avo
vov = 2*(getData("T0:ids" ?result "dc") / getData("T0:gm" ?result
    "dc"))

ocnPrint(?output "mosDCChar/nfet_ft.dat" ?numberNotation '
    scientific ft)
ocnPrint(?output "mosDCChar/nfet_avo.dat" ?numberNotation '
    scientific avo)
ocnPrint(?output "mosDCChar/nfet_gdsoverid.dat" ?numberNotation '
    scientific gdsoverid)
ocnPrint(?output "mosDCChar/nfet_gmoverid.dat" ?numberNotation '
    scientific getData("T0:gmoverid" ?result "dc"))
ocnPrint(?output "mosDCChar/nfet_vov.dat" ?numberNotation '
    scientific vov)

```



```
ocnPrint(?output "mosDCChar/nfet_ftavo.dat" ?numberNotation '
    scientific ftavo)
ocnPrint(?output "mosDCChar/nfet_cgg.dat" ?numberNotation '
    scientific getData("T0:cgg" ?result "dc"))
ocnPrint(?output "mosDCChar/nfet_cgd.dat" ?numberNotation '
    scientific getData("T0:cgd" ?result "dc"))
ocnPrint(?output "mosDCChar/nfet_cdd.dat" ?numberNotation '
    scientific getData("T0:cdd" ?result "dc"))
ocnPrint(?output "mosDCChar/nfet_idw.dat" ?numberNotation '
    scientific getData("T0:ids" ?result "dc"))
```

Appendix C

Example Parsing Function

```
% This version of the parser condenses all capacitors down to
    drain and
% gate total capacitance. A future version will add total
    capacitance
% looking into the source.

% 2/22/2012 — Version notes changed capacitances and changed
    voltage
% sweeps to courser sweep for vgs and finer sweep for vds.

clear all
close all
clc

filePath = '/home/noyade/cadence_work/8RF/mosDCChar/';

% deviceList is a list by name of the devices simulated using
    OCEAN scripts
% to create parametric lookup tables.
deviceList = {'nfet' 'lvtnfet' 'lpnfet' 'zvtnfet' 'dgnfet' '
    zvtDgnfet' 'nfet33' ...
```

```

        'pfet' 'lvtpfet' 'lppfet' 'dgpfet' 'pfet33'};
n = length(deviceList);

% In the section below we will set the sweep ranges for the
    components
% that we swept with our OCEAN scripts so that they have the same
% dimensions. This will make parsing of the data easier.
for p = 1:n
    device = deviceList(p);
    deviceName = sprintf('%s', char(device));
    if strcmp(char(device), 'dgnfet') || strcmp(char(device), '
        dgpfet')
        ibm.(deviceName).L = (0.24e-6:0.08e-6:2e-6)*1e9;
        ibm.(deviceName).vds = 0.02:0.02:2.5;
        ibm.(deviceName).vgs = 0.05:0.05:2.5;
    elseif strcmp(char(device), 'zvtnfet')
        ibm.(deviceName).L = (0.42e-6:0.08e-6:2.26e-6)*1e9;
        ibm.(deviceName).vds = 0.02:0.02:1.5;
        ibm.(deviceName).vgs = 0.05:0.05:1.5;
    elseif strcmp(char(device), 'zvt dgnfet')
        ibm.(deviceName).L = (0.56e-6:0.08e-6:2.4e-6)*1e9;
        ibm.(deviceName).vds = 0.02:0.02:2.5;
        ibm.(deviceName).vgs = 0.05:0.05:2.5;
    elseif strcmp(char(device), 'nfet33') || strcmp(char(device), '
        pfet33')
        ibm.(deviceName).L = (0.4e-6:0.08e-6:2.24e-6)*1e9;
        ibm.(deviceName).vds = 0.02:0.02:3.3;

```

```

        ibm.(deviceName).vgs = 0.05:0.05:3.3;
    else
        ibm.(deviceName).L = (0.12e-6:0.04e-6:1e-6)*1e9;
        ibm.(deviceName).vds = 0.02:0.02:1.5;
        ibm.(deviceName).vgs = 0.05:0.05:1.5;
    end
end

end

% gm/iD
for p = 1:n
    device = deviceList(p);
    parameter = 'gmoverid';
    deviceName = sprintf('%s',char(device));
    parameterName = sprintf('%s',parameter);
    fileName = strcat(device, '_3d_', parameter);
    file = strcat(filePath, fileName, '.dat');
    fid = fopen(char(file));
    gmoverid = zeros(length(ibm.(deviceName).vgs),length(ibm.(
        deviceName).vds),length(ibm.(deviceName).L));
    for x = 1:length(ibm.(deviceName).L)
        fscanf(fid, '%*s %*s %*s %*s %*s %*s\n\nL =%f',1);
        fscanf(fid, '%s',1);
        fscanf(fid, '%f',length(ibm.(deviceName).vds));
        t = fscanf(fid, '%f',[length(ibm.(deviceName).vds)+1,
            length(ibm.(deviceName).vgs)]);
        gmoverid(:, :, x) = t(1:end,2:end);
    end
    clear t
end

```

```

    end

    fclose(fid);

    ibm.(deviceName).(parameterName) = gmoverid;

    clear gmoverid
end

% Cd

for p = 1:n
    device = deviceList(p);
    parameter = 'cd';
    deviceName = sprintf('%s',char(device));
    parameterName = sprintf('%s',parameter);
    ibm.(deviceName).(parameterName) = ibm.(deviceName).cdd + ibm
        .(deviceName).cjd + ...
        ibm.(deviceName).cde + ibm.(deviceName)
            .covlgd;
end

clear fid file fileName filePath m n p x parameter parameterName
clear device deviceList deviceName
save /home/noyade/cadence_work/8RF/mosDCChar/matlab/
    ibm8rfTechfile.mat ibm

```

Appendix D

Example Look-up Function

```
function gmid = lookup_gmid_3d(varargin)

if nargin > 5
    error('You used too many input arguments');
end

t = varargin{1};
if ~ischar(t)
    error('Your first argument must be a string corresponding to
        your technology file. ');
else
    load(t)
end

d = varargin{2};
if ~ischar(d)
    error('Your second argument must be a string corresponding to
        the device you are trying to design (e.g. nfet, pfet, etc
        ...). ');
elseif ~(strcmp(d, 'nfet') || strcmp(d, 'lpnfet') || strcmp(d, '
    lvtnfet') || ...
```

```

    strcmp(d, 'zvtnfet') || strcmp(d, 'dgnfet') || strcmp(d, '
        nfet33') || ...
    strcmp(d, 'hvtinfet33') || strcmp(d, 'pfet') || strcmp(d, '
        lppfet') || ...
    strcmp(d, 'lvtpfet') || strcmp(d, 'dgpfet') || strcmp(d, '
        pfet33') || ...
    strcmp(d, 'hvtpfet33'))
    error('The only legitimate values for this parameter
        correspond to IBM 8RF CMOS transistors (e.g. nfet, dgnfet,
        etc...).');
end

l = varargin{3};
if ~isreal(l)
    error('Your third argument must be an integer corresponding
        to the length of the device you are designing.');
```

```

elseif l < min(ibm.(d).L) || l > max(ibm.(d).L)
    error('Data must in units of nm between %u nm and %u nm',
        int16(min(ibm.(d).L)), int16(max(ibm.(d).L)));
end

vd = varargin{4};
if isempty(vd)
    vd = max(ibm.(d).vds)/2;
end
if ~isreal(vd)
```

```

    error('Your fourth argument must be a real number
           corresponding to the vds for which you are trying to
           lookup ft. ');
elseif vd < 0 || vd > max(ibm.(d).vds)
    error('Data must be a real number between 0 and Vdd,
           corresponding to the Vdd of the transistor you are trying
           to lookup. ');
end

f = varargin{5};
if ~isreal(f)
    error('Your fourth argument must be an integer corresponding
           to the frequency for which you are trying to lookup gm/id.
           ');
elseif f < 0 || f > 120
    error('Data must be an integer in units of GHz between 0 and
           120 ');
else
    fT = f;%*1e9;
end

[X,Y] = meshgrid(ibm.(d).vds,ibm.(d).L);
gmVsVgs = zeros(length(ibm.(d).vgs),1);
ftVsVgs = zeros(length(ibm.(d).vgs),1);
for p = 1:length(ibm.(d).vgs)
    gm = reshape(ibm.(d).gmoverid(p, :, :), length(ibm.(d).vds),
                 length(ibm.(d).L));

```



```
ft = reshape(ibm.(d).ft(p, :, :), length(ibm.(d).vds), length(ibm
    .(d).L));
gmVsVgs(p) = interp2(X, Y, gm', vd, l);
ftVsVgs(p) = interp2(X, Y, ft', vd, l);
end

gmid = interp1(ftVsVgs, gmVsVgs, fT);
```

Appendix E

MATLAB Sizing Script

```
techfile    ='ibm8rfTechfile.mat';
```

```
nDevice     ='nfet';
```

```
pDevice     ='pfet';
```

```
%pick a tail curretail
```

```
iTail       =10.625E-6;
```

```
iHalf       =iTail/2;
```

```
% frequency at which we watail osc.
```

```
fosc        = 425E6;
```

```
n.Vds       = 1/3;
```

```
p1.Vds      = 1/3;
```

```
p2.Vds      = 1/3;
```

```
tail.Vds    = 1/3;
```

```
n.L         = 120;
```

```
p1.L        = 240;
```

```
p2.L        = 240;
```

```
tail.L      = 360;
```

```
n.gmid      = 23;
```

```

dPM      = 1.0; % Drain parasitic multiplier (try 1.25)
gPM      = 1.0; % Gate parasitic multiplier (try 1.16)
mPM      = 1.0; % Mixer parasitic multiplier (try 1.2)

% check that the Nmos device can switch fast enough
n.ft     =lookup_ft_3d(techfile ,nDevice ,n.L,n.Vds,n.gmid);

% now we can design the bottom pair for the curretail and fcalc
n.idw   =lookup_idw_3d(techfile ,nDevice ,n.L,n.Vds,n.gmid);
n.w     =1E6*(iHalf/n.idw);

% MIXER STUFF
wMixer = 9.6;
lMixer = 120;
vdsMixer = 0.1;
gmidMixer = 37;
mixerDevice = 'nfet';
cggMixer = (wMixer/20)*lookup_cggw_3d(techfile ,mixerDevice ,lMixer
    ,vdsMixer ,gmidMixer);
cMixer = 2*cggMixer;

% now that we have a good standardized size , the rest of the info
n.gm    = n.gmid*iHalf;
n.cdd   = dPM*(n.w/20)*(lookup_cdw_3d(techfile ,nDevice ,n.L,n.Vds,n
    .gmid));

```

```

n.cgg = gPM*(n.w/20)*(lookup_cgwg_3d(techfile ,nDevice ,n.L,n.Vds,
    n.gmid));

ctotn = n.cgg+n.cdd;
ctot = ctotn + cMixer;
ctotp = 1e15*(n.gm/(2*pi*fosc) - ctot*1e-15);

if ctotp < 0
    fprintf('The gm and capacitance from the NMOS do not allow a
        solution , please try using more curretail or a smaller gm/
        id')
    stop
end

n.va =lookup_va_3d(techfile ,nDevice ,n.L,n.Vds,n.gmid);
n.gds =(1/n.va)*iHalf*1e6;

% now we define a 'total' gm for the top two devices , which we
    will split
% using zeta
gmidpt = 10; p1.cdd = 0; p2.cdd = 0; p1.cgg = 0; p2.cgg = 0;
    cLayout = 22.4;
if ctotp < cLayout
    fprintf('This design necessitates less capacitance than the
        parasitic ,
        please try a smaller gm/id or more bias curretail')

```

```

    stop
end

while (p1.cdd+p2.cdd+p1.cgg+p2.cgg+cLayout) < ctotp
    zeta = 1.0; gtot = 10; vap = lookup_va_3d(techfile ,pDevice ,p1
        .L,p1.Vds,gmidpt);
    while gtot > -1
        p2.iD = 1E6*iHalf/(1+zeta);
        p1.iD = p2.iD*zeta;

        p2.gm = gmidpt*p2.iD;
        p1.gm = gmidpt*p1.iD;

        p2.gds = -(1/vap)*p2.iD*2;
        p1.gds = -(1/vap)*p1.iD*2;

        gtot = n.gds + p1.gds + p2.gds + p2.gm - p1.gm;

        fprintf('GT=%2.2f uS, zeta=%2.2f\n', gtot, zeta)

        zeta = zeta + 0.05;
    end

    idwp = lookup_idw_3d(techfile , pDevice , p1.L, p1.Vds, gmidpt);

    p2.w = -(p2.iD/idwp);
    p1.w = -(p1.iD/idwp);

```

```

Wt = p2.w + p1.w;

p2.cdd = dPM*(p2.w/20)*(lookup_cdw_3d(techfile,pDevice,p2.L,p2
    .Vds,gmidpt));
p1.cdd = dPM*(p1.w/20)*(lookup_cdw_3d(techfile,pDevice,p1.L,p1
    .Vds,gmidpt));
p2.cgg = gPM*(p2.w/20)*lookup_cgg_3d(techfile,pDevice,p2.L,p2.
    Vds,gmidpt);
p1.cgg = gPM*(p1.w/20)*lookup_cgg_3d(techfile,pDevice,p1.L,p1.
    Vds,gmidpt);

fprintf('|- Wp=%2.2f um, Ctotp=%3.2f fF, gmidp=%2.2f\n', Wt,p1
    .cgg+p1.cdd+p2.cgg+p2.cdd+cLayout, gmidpt)

gmidpt = gmidpt + .1;
end

% Re-estimate oscillation frequency
foscp = 1e-6*(1/(2*pi))*(sqrt(n.gm^2 - ((gtot)*1E-6)^2)/((p1.cdd+
    p2.cdd+p1.cgg+p2.cgg+ctot+cLayout)*1E-15));

% Size tail curretail source
tail.gmid = 20;
tail.idw = lookup_idw_3d(techfile,nDevice,tail.L,tail.Vds,tail.
    gmid);
tail.w = (iTail/tail.idw)*1e6;

```

```
fprintf( '|-- Mn: \t W=%3.1f um\t L=%3.0f nm\n', n.w, n.L)
fprintf( '|-- Mp2:\t W=%3.1f um\t L=%3.0f nm\n', p2.w, p1.L)
fprintf( '|-- Mp1:\t W=%3.1f um\t L=%3.0f nm\n', p1.w, p1.L)
fprintf( '|-- Mc: \t W=%3.1f um\t L=%3.0f nm\n', tail.w, tail.L)
fprintf( '|-- Estimated Frequency of Oscillation is: fosc=%3.0f
      MHz\n\n', fosc)
```

References

- [1] P. Bradley, "Wireless medical implant technology - recent advances and future developments," in *Solid-State Device Research Conference (ESSDERC), 2011 Proceedings of the European*, 2011, pp. 54–58.
- [2] I. A. W. Vance, "An integrated circuit v.h.f. radio receiver," *Radio and Electronic Engineer*, vol. 50, no. 4, pp. 158–164, 1980.
- [3] , "Title 47 CFR 95.401: What are the citizens band radio services?" Code of Federal Regulations, Tech. Rep., October 2009.
- [4] , "Title 47 CFR 95.628 - MedRadio transmitters," Code of Federal Regulations, Tech. Rep., October 2011.
- [5] L.-C. Liu, M.-H. Ho, and C.-Y. Wu, "A medradio-band low-energy-per-bit CMOS OOK transceiver for implantable medical devices," in *Biomedical Circuits and Systems Conference (BioCAS), 2011 IEEE*, 2011, pp. 153–156.
- [6] J. Bohorquez, A. Chandrakasan, and J. Dawson, "A 350uW CMOS MSK transmitter and 400uW OOK super-regenerative receiver for medical implant communications," *Solid-State Circuits, IEEE Journal of*, vol. 44, no. 4, pp. 1248–1259, 2009.
- [7] K. W. Li, L. Leung, and K. N. Leung, "Low power injection locked oscillators for MICS standard," in *Biomedical Circuits and Systems Conference, 2009. BioCAS 2009. IEEE*, 2009, pp. 1–4.
- [8] N. Van Helleputte and G. Gielen, "An ultra-low-power quadrature PLL in 130nm CMOS for impulse radio receivers," in *Biomedical Circuits and Systems Conference, 2007. BIOCAS 2007. IEEE*, 2007, pp. 63–66.
- [9] H. Savci, A. Sula, Z. Wang, N. Dogan, and E. Arvas, "MICS transceivers: regulatory standards and applications [medical implant communications service]," in *Southeast-Con, 2005. Proceedings. IEEE*, 2005, pp. 179–182.
- [10] H.-K. Cha, M. Raja, X. Yuan, and M. Je, "A CMOS MedRadio receiver RF front-end with a complementary current-reuse LNA," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 59, no. 7, pp. 1846–1854, 2011.
- [11] J. Maneatis and M. Horowitz, "Precise delay generation using coupled oscillators," *Solid-State Circuits, IEEE Journal of*, vol. 28, no. 12, pp. 1273–1282, 1993.
- [12] M. Voltti, T. Koivi, and E. Tiiliharju, "Comparison of active and passive mixers," in *Circuit Theory and Design, 2007. ECCTD 2007. 18th European Conference on*, 2007, pp. 890–893.

- [13] V. Geffroy, G. De Astis, and E. Bergeault, "RF mixers using standard digital CMOS 0.35 μm process," in *Microwave Symposium Digest, 2001 IEEE MTT-S International*, vol. 1, 2001, pp. 83–86 vol.1.
- [14] E. Vittoz and J. Fellrath, "CMOS analog integrated circuits based on weak inversion operations," *Solid-State Circuits, IEEE Journal of*, vol. 12, no. 3, pp. 224–231, 1977.
- [15] F. Silveira, D. Flandre, and P. Jespers, "A gm/ID based methodology for the design of CMOS analog circuits and its application to the synthesis of a silicon-on-insulator micropower OTA," *Solid-State Circuits, IEEE Journal of*, vol. 31, no. 9, pp. 1314–1319, 1996.
- [16] B. Fahs, W. Ali-Ahmad, and P. Gamand, "A two-stage ring oscillator in 0.13- μm CMOS for UWB impulse radio," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 57, no. 5, pp. 1074–1082, 2009.
- [17] N. Haghighi and K. Raahemifar, "A novel low phase noise quadrature VCO for medical implantable transceivers," in *Electrical and Computer Engineering, 2008. CCECE 2008. Canadian Conference on*, 2008, pp. 001 487–001 490.