Circuits and Systems, 2012, 3, 136-145
http://dx.doi.org/10.4236/cs.2012.32018 Published Online April 2012 (http://www.SciRP.org/journal/cs)

An Analytical Approach for Fast Automatic Sizing of Narrow-Band RF CMOS LNAs*

Jin Young Choi
Electronic & Electrical Engineering Department, Hongik University, Jochiwon, South Korea
Email: jychoi@hongik.ac.kr

Received November 25, 2011; revised December 26, 2011; accepted January 4, 2012

ABSTRACT

We introduce a fast automatic sizing algorithm for a single-ended narrow-band CMOS cascode LNA adopting an inductive source degeneration based on an analytical approach without any optimization procedure. Analytical expressions for principle parameters are derived based on an ac equivalent circuit. Based on the analytical expressions and the power-constrained noise optimization criteria, the automatic sizing algorithm is developed. The algorithm is coded using Matlab, which is shown capable of providing a set of design variable values within seconds. One-time Spectre simulations assuming usage of a commercial 90 nm CMOS process are performed to confirm that the algorithm can provide the aimed first-cut design with a reasonable accuracy for the frequency ranging up to 5 GHz. This work shows one way how accurate automatic synthesis can be done in an analytical approach.

Keywords: Automatic Synthesis; Analytical Approach; CMOS LNA; Narrow Band; Cascode

1. Introduction

In the field of RF transceiver design, there is a strong demand to digitalize even RF analog parts to mount a transceiver on a single chip [1,2] to utilize the capability of automatic synthesis in digital circuit design. However, the low noise amplifier (LNA), which is a critical building block in any RF front-end, is not ready for digitalization yet. Many efforts have been done for design automation of LNA beforehand since the design of LNA is a time-consuming task that typically relies heavily on the experience of RF designers. LNA design automation can significantly simplify the design task, and also opens a possibility towards digitalization.

There are two basic methods for LNA design automation: simulation based or equation based. Although the simulation-based methods [3,4] are more accurate, they are time consuming due to optimization procedures. On the other hand, equation-based methods [5-7] are faster, but are dependent on the accuracy of the models used. To overcome the disadvantages in some extent, advanced methods using both of equation-based and simulation-based approaches [8-10] have been also suggested.

The difficulties in design automation of LNA lie in several aspects. It is topology dependent, and the design itself is difficult involving trade-offs among critical figures of merits such as NF, power gain, impedance matching, power consumption, linearity, and stability. Mentioning the difficulties in a manual design, for example, even only for input and output matching, many iteration steps are needed. It should be also redesigned every time when the fabrication process is changed. Therefore it is desirable if the first-cut design synthesis can be done automatically and fast with an acceptable accuracy.

The purpose of this work is to suggest a methodology for providing a set of first-cut design variables for a narrow-band LNA with a reasonable accuracy once design and process specifications are given.

We introduce a speedy automatic sizing algorithm for a single-ended narrow-band cascode LNA adopting inductive source degeneration based on an analytical approach without any optimization procedure. In Section 2, design assumptions are discussed. In Section 3, analytical expressions for principle parameters are derived based on an ac equivalent circuit assuming a resistive output termination. In Section 4, the developed automatic sizing algorithm is explained in detail. In Section 5, verifications are given to check the accuracy of the automatic sizing results.

2. Design Assumptions

There are many topologies for narrow-band LNAs, however, typical topologies include cascode, common source, and differential configurations, and the cascode

*This work was supported by 2011 Hongik University Research Fund (sponsors).

structure with an inductive source degeneration shown in **Figure 1** is the most attractive one in single-ended topologies since it gives smaller input capacitance and larger in-out isolation [11]. In this work, the cascode LNA topology shown in **Figure 1** is chosen as the objective circuit for automatic sizing even though the same approach can be applied to the other topologies.

There are several assumptions made in this work as follows:

1) Narrow-band LC matching networks are used for input and output as shown in **Figure 1**. R_1 is used to provide capability for adjusting power gain. As the output termination, two cases are considered: resistive or capacitive termination.

2) For sizing of the MOS transistors M_1 and M_2, the power-constrained noise optimization (PCNO) criteria [11] is adopted to trade off noise performance against power consumption.

3) Ideal inductors and capacitors are used by assuming usage of off-chip components. The series resistances of the on-chip inductors can be considered as well, but we choose a simpler case.

4) A current-mirror biasing is adopted as shown in **Figure 1**.

5) The widths of M_1 and M_2 are set as same.

6) The design specifications include operating frequency, input and output terminations, power consumption, power gain, and sufficiently low input and output reflection coefficients S_{11} and S_{22}.

7) The design variables include L_g, L_s, L_1, C_i, C_o, R_1, R_{DB}, and R_B including the widths of M_1, M_2, and M_B in **Figure 1**.

3. Derivation of Analytic Expressions for Principal Parameters

3.1. Input Impedance

Figure 2 is the whole ac equivalent circuit for the cascode LNA shown in **Figure 1** including the input signal source and the output resistive termination. We note that, compared to the complete equivalent circuit of the BSIM4 NMOS transistor in SPICE, only the back-gate transconductance g_{mb} and the gate-body capacitance C_{gb} in the transistor model are ignored to simplify the analysis. The distributed resistances including R_s, R_d, R_g, and R_{sub}, which are included in the BSIM 4 transistor model, are also ignored since they are negligible in large transistors.

In **Figure 2**, g_{m1} and g_{m2} denote the transconductances of M_1 and M_2, respectively. C_{gs}, C_{gd}, and C_{ds} denote the gate-source, gate-drain, and drain-source capacitances of the NMOS transistors, respectively. C_{js} and C_{jd} denote the source-body and drain-body junction capacitances, and C_L is equal to the sum of C_{dg2} and C_{jd2}, which are the

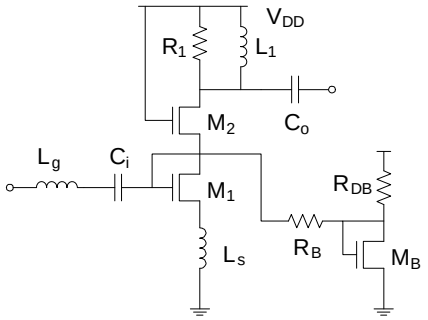


Figure 1. Assumed cascode LNA circuit.

capacitances present at the drain node of M_2 in **Figure 1**.

The impedances Z_{in}, Z_{in1}, Z_{in2}, Z_o, Z_{out}, Z_{out1}, and Z_{out2} are self-defined in the circuit. We first consider the resistive output termination case and discuss the capacitive output termination case later in Section 6. We note that C_{gs}, C_{gd}, and C_{ds} are replaced by C_{sg}, C_{dg}, and C_{sd}, respectively, in some part of our derivations for input and output impedances considering the non-reciprocal nature of gate-oxide capacitances in the BSIM4 MOSFET capacitance model [12].

First, we derive Z_{in} by deriving Z_o, Z_{in2}, and Z_{in1} in order. We note that, we use s and $j\omega$ without differentiation since we are dealing with ac response only.

To derive Z_o at the operating frequency, the series C_o and R_{so} in **Figure 2** can be transformed to the parallel equivalents, C_p and R_p [11]. Then $Y_o = 1/Z_o$ is simply expressed as

$$Y_o = \frac{1}{sL_1} + sC_p + \frac{1}{R_p}, \tag{1}$$

where $R_p = R_{so}(Q^2 + 1)$, $C_p = C_oQ^2/(Q^2 + 1)$, and $Q = 1/(\omega R_{so}C_o)$.

Figure 3 shows the ac equivalent circuit to derive an expression for Z_{in2}. Notice that, in the circuit shown in **Figure 3**, the non-reciprocal capacitance C_{sd2} is used instead of C_{ds2}, since we are looking into the source of M_2.

By neglecting the parallel $(C_{sg2} + C_{js2})$ branch, we derive the input admittance Y_{in21} first, and add $s(C_{sg2} + C_{js2})$ to find $Y_{\text{in2}} = 1/Z_{\text{in2}}$. When the $(C_{sg2} + C_{js2})$ branch is neglected, the circuit can be characterized by (2) and (3).

$$v_o = \left[g_{m2}v_{s2} + \left(v_{s2} - v_o\right)\left(g_{ds2} + sC_{sd2}\right)\right] \cdot \left(\frac{1}{sC_L} // R_1 // Z_o \right) \tag{2}$$

$$\begin{aligned} i &= g_{m2}v_{s2} + \left(v_{s2} - v_o\right)\left(g_{ds2} + sC_{sd2}\right) \\ &= \left(g_{m2} + g_{ds2} + sC_{sd2}\right)v_{s2} - \left(g_{ds2} + sC_{sd2}\right)v_o \end{aligned} \tag{3}$$

By eliminating v_o in (2) and (3), we can express Y_{in21} as

$$Y_{\text{in21}} = \frac{i}{v_{s2}} = \frac{\left(g_{m2} + g_{ds2} + sC_{sd2}\right)}{1 + \left(g_{ds2} + sC_{sd2}\right)Z_p}, \tag{4}$$

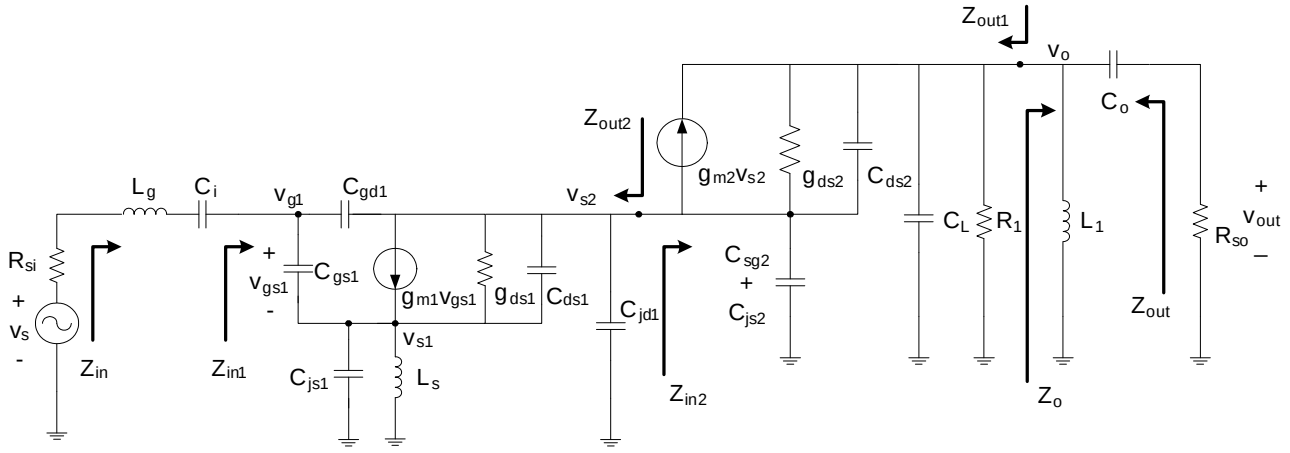


Figure 2. AC equivalent circuit of the cascode LNA in Figure 1.

where $Z_p = (1/sC_L)//R_1//Z_o$.

Then Y_{in2} are expressed as

$$Y_{\text{in2}} = Y_{\text{in21}} + s\left(C_{sg2} + C_{js2}\right). \tag{5}$$

Figure 4 shows the ac equivalent circuit to derive an expression for Z_{in1}. The circuit can be characterized by (6), (7), and (8).

$$\begin{aligned} v_{s1} = &\left[sC_{gs1}\left(v_{g1} - v_{s1}\right) + g_{m1}v_{gs1}\right. \\ &\left.+\left(v_{s2} - v_{s1}\right)\left(g_{ds1} + sC_{ds1}\right)\right]\cdot\left(sL_s // \frac{1}{sC_{js1}}\right) \end{aligned} \tag{6}$$

$$\begin{aligned} v_{s2} = -&\left[sC_{gd1}\left(v_{s2} - v_{g1}\right) + g_{m1}v_{gs1}\right. \\ &\left.+\left(v_{s2} - v_{s1}\right)\left(g_{ds1} + sC_{ds1}\right)\right]\cdot Z_L \end{aligned} \tag{7}$$

$$i = sC_{gs1}\left(v_{g1} - v_{s1}\right) + sC_{gd1}\left(v_{g1} - v_{s2}\right), \tag{8}$$

where $Z_L = (1/(sC_{jd1}))//Z_{\text{in2}}$.

By eliminating v_{s1} and v_{s2} in (6), (7) and (8), $Y_{\text{in1}} = 1/Z_{\text{in1}}$ is expressed as

$$Y_{\text{in1}} \equiv \frac{i}{v_{g1}} = Y_{\text{in11}} + Y_{\text{in12}} + Y_{\text{in13}}, \tag{9}$$

where $Y_{\text{in11}} = \left(sC_{gs1} + sC_{gd1}\right)$,

$$Y_{\text{in12}} = \frac{\left[\left(sC_{gs1} + g_{m1}\right)\left(g_{m1} + g_{ds1} + sC_{ds1}\right) + \left(sC_{gd1} - g_{m1}\right)e_1\right]\cdot sC_{gd1}}{\left(g_{ds1} + sC_{ds1}\right)\left(g_{m1} + g_{ds1} + sC_{ds1}\right) - e_1e_2},$$

$$Y_{\text{in13}} = \frac{\left[\left(sC_{gs1} + g_{m1}\right)e_2 + \left(sC_{gd1} - g_{m1}\right)\left(g_{ds1} + sC_{ds1}\right)\right]\cdot sC_{gs1}}{\left(g_{ds1} + sC_{ds1}\right)\left(g_{m1} + g_{ds1} + sC_{ds1}\right) - e_1e_2},$$

$$e_1 = \frac{1}{sL_s // \dfrac{1}{sC_{js1}}} + sC_{gs1} + g_{m1} + g_{ds1} + sC_{ds1}$$

and $e_2 = \dfrac{1}{Z_L} + sC_{gd1} + g_{ds1} + sC_{ds1}$.

Then Z_{in} is expressed as

$$Z_{\text{in}} = Z_{\text{in1}} + sL_g + \frac{1}{sC_i}. \tag{10}$$

3.2. Output Impedance

Z_{out} derivation can be done similarly as the Z_{in} derivation using the equivalent circuit in **Figure 2** assuming R_{si} input termination. We present the results only here.

$Y_{\text{out2}} = 1/Z_{\text{out2}}$ is expressed as

$$Y_{\text{out2}} \equiv \frac{i}{v_{s2}} = Y_{\text{out21}} + Y_{\text{out22}} + Y_{\text{out23}} + Y_{\text{out24}}, \tag{11}$$

where $Y_{\text{out21}} = g_{ds1} + sC_{ds1} + sC_{dg1}$,

$$Y_{\text{out22}} = \frac{\left[sC_{sg1}\left(g_{ds1} + sC_{ds1}\right) + sC_{dg1}d_1\right]\cdot\left(g_{m1} - sC_{dg1}\right)}{d_1d_2 - sC_{sg1}\left(sC_{sg1} + g_{m1}\right)},$$

$$Y_{\text{out23}} = -\frac{\left[\left(g_{ds1} + sC_{ds1}\right)d_2 + sC_{dg1}\left(sC_{sg1} + g_{m1}\right)\right]\left(g_{m1} + g_{ds1} + sC_{ds1}\right)}{d_1d_2 - sC_{sg1}\left(sC_{sg1} + g_{m1}\right)},$$

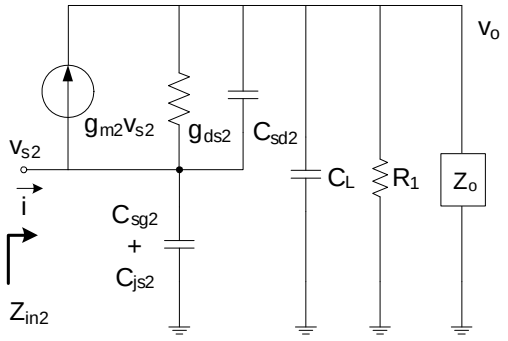


Figure 3. AC equivalent circuit to find Z_{in2}.

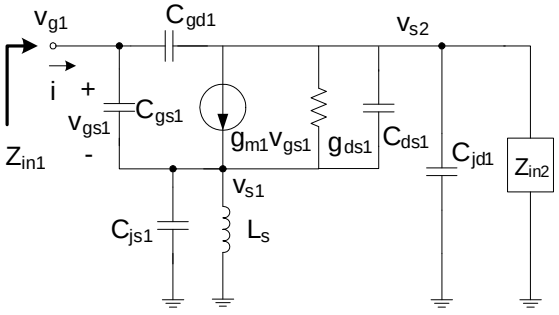


Figure 4. AC equivalent circuit to find Z_{in1}.

$$Y_{\text{out24}} = sC_{jd1},$$

$$d_1 = \frac{1}{\frac{1}{sC_{js1}} // sL_s} + sC_{sg1} + g_{m1} + \left(g_{ds1} + sC_{ds1}\right),$$

$$d_2 = \frac{1}{Z_i} + sC_{dg1} + sC_{sg1},$$

and $Z_i = R_{si} + sL_g + \frac{1}{sC_i}$.

$Y_{\text{out1}} = 1/Z_{\text{out1}}$ is expressed as

$$Y_{\text{out1}} = \frac{1}{Z_2} + sC_L + \frac{1}{R_1}, \tag{12}$$

where $Z_2 = \frac{1}{g_{ds2} + sC_{ds2}} + Z_1\left(1 + \frac{g_{m2}}{g_{ds2} + sC_{ds2}}\right)$ and

$Z_1 = Z_{\text{out2}} // \frac{1}{s\left(C_{sg2} + C_{js2}\right)}$.

Then Z_{out} is expressed as

$$Z_{\text{out}} = \frac{1}{sC_o} + Z_{\text{out1}} // sL_1. \tag{13}$$

3.3. Power Gain

To derive the LNA voltage gain, the equivalent circuit in **Figure 2** is simplified into the one shown in **Figure 5**,

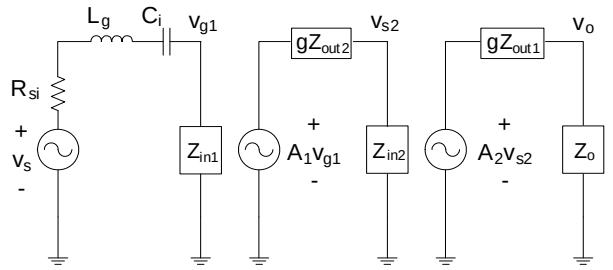


Figure 5. Equivalent circuit to find the voltage gain.

where the whole circuit is expressed as a 3-stage cascaded amplifier.

Z_{in1}, Z_{in2} and Z_o in **Figure 5** are already derived in (9), (5) and (1), respectively. Notice that A_1v_{g1}, gZ_{out2}, A_2v_{s2}, and gZ_{out1} are the Thevenin equivalent voltages and impedances of the 2nd and 3rd gain stages in **Figure 2**. Therefore gZ_{out2} and gZ_{out1} differ from Z_{out2} and Z_{out1} in (11) and (12), respectively, and can be derived as follows.

By definition, gZ_{out2} corresponds to the impedance seen to the left of the v_{s2} node when $v_{g1} = 0$ in **Figure 2**, and can be derived using the equivalent circuit shown in **Figure 6**.

The circuit can be characterized by the Equations (14) and (15).

$$v_{s1} = \left[-g_{m1}v_{s1} + \left(g_{ds1} + sC_{ds1}\right)\left(v_{s2} - v_{s1}\right)\right] \cdot \left(\frac{1}{s\left(C_{sg1} + C_{js1}\right)} // sL_s\right) \tag{14}$$

$$i = -g_{m1}v_{s1} + \left(g_{ds1} + sC_{ds1}\right)\left(v_{s2} - v_{s1}\right) \tag{15}$$

By eliminating v_{s1} in (14) and (15), gY_{out21} is expressed as

$$gY_{\text{out21}} \equiv \frac{i}{v_{s2}} = \frac{\left(g_{ds1} + sC_{ds1}\right)\left[s\left(C_{sg1} + C_{js1}\right) + \frac{1}{sL_s}\right]}{s\left(C_{sg1} + C_{js1}\right) + \frac{1}{sL_s} + g_{m1} + g_{ds1} + sC_{ds1}}. \tag{16}$$

Then $gY_{\text{out2}} = 1/gZ_{\text{out2}}$ is expressed as

$$gY_{\text{out2}} \equiv \frac{i_{s2}}{v_{s2}} = gY_{\text{out21}} + s\left(C_{dg1} + C_{jd1}\right). \tag{17}$$

By definition, A_1 corresponds to the voltage gain v_{s2o}/v_{g1}, where v_{s2o} is the v_{s2} node voltage when open, and can be derived using the equivalent circuit shown in **Figure 7**. The circuit can be characterized by the Equations (18) and (19).

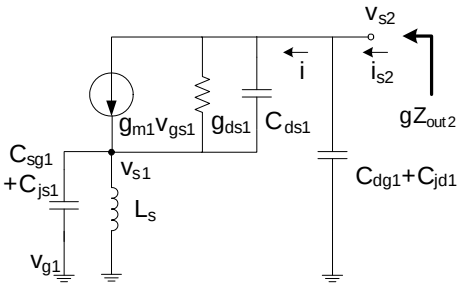


Figure 6. AC equivalent circuit to find gZ_{out2}.

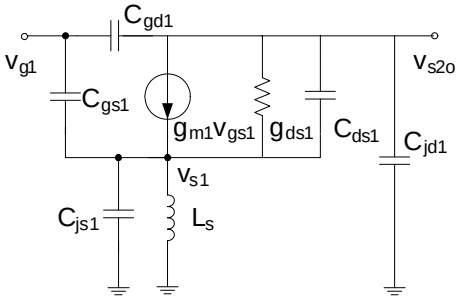


Figure 7. AC equivalent circuit to find A_1.

$$v_{s1} = \Big[sC_{gs1}\left(v_{g1} - v_{s1}\right) + g_{m1}v_{gs1} + \left(v_{s2o} - v_{s1}\right)\left(g_{ds1} + sC_{ds1}\right)\Big] \cdot \left(sL_s // \frac{1}{sC_{js1}} \right) \tag{18}$$

$$v_{s2o} = -\Big[sC_{gd1}\left(v_{s2o} - v_{g1}\right) + g_{m1}v_{gs1} + \left(v_{s2o} - v_{s1}\right)\left(g_{ds1} + sC_{ds1}\right)\Big] \cdot \frac{1}{sC_{jd1}} \tag{19}$$

By eliminating v_{s1} in (18) and (19), we get

$$A_1 \equiv \frac{v_{s2o}}{v_{g1}} = -\frac{\left(sC_{gs1} + g_{m1}\right)\left(g_{m1} + g_{ds1} + sC_{ds1}\right) + \left(sC_{gd1} - g_{m1}\right) f_1}{\left(g_{ds1} + sC_{ds1}\right)\left(g_{m1} + g_{ds1} + sC_{ds1}\right) - f_1 f_2}, \tag{20}$$

where $f_1 = \dfrac{1}{sL_s // \dfrac{1}{sC_{js1}}} + sC_{gs1} + g_{m1} + g_{ds1} + sC_{ds1}$ and

$f_2 = sC_{jd1} + sC_{gd1} + g_{ds1} + sC_{ds1}$.

gZ_{out1} corresponds to the impedance seen to the left of the v_o node with $v_{s2} = 0$ in **Figure 2**. Since $g_{m2}v_{s2}$ and $(C_{gs2} + C_{js2})$ do not function when $v_{s2} = 0$, $gY_{out1} = 1/gZ_{out1}$ is simply expressed as

$$gZ_{out1} = g_{ds2} + sC_{ds2} + sC_L + \frac{1}{R_1}. \tag{21}$$

A_2 corresponds to the voltage gain v_{oo}/v_{s2}, where v_{oo} is the v_o node voltage when open, and A_2 derivation can be done in the similar fashion to the one for A_1 derivation. The resulting A_2 is expressed as

$$A_2 \equiv \frac{v_{oo}}{v_{s2}} = \frac{g_{m2} + g_{ds2} + sC_{sd2}}{g_{ds2} + sC_{sd2} + 1 \Big/ \left(\dfrac{1}{sC_L} // R_1 \right)}. \tag{22}$$

In **Figure 2**, the available input power P_i, which is supplied to the LNA when impedance matched, is defined as

$$P_i = \frac{v_s^2}{4R_{si}}. \tag{23}$$

The maximum output power P_o, which is supplied to the resistive load R_{so} when impedance matched, is expressed as

$$P_o = \frac{v_o^2}{R_p} = \frac{v_{out}^2}{R_{so}}, \tag{24}$$

where v_o and v_{out} are defined in **Figure 2**, and R_p is the transformed parallel resistance of R_{so}, which is already defined relating (1).

Then the available power gain G is expressed as

$$G = \frac{P_o}{P_i} = \frac{4R_{si}}{R_p}\left(\frac{v_o}{v_s}\right)^2 = \frac{4R_{si}}{R_p}\left(\frac{v_{g1}}{v_s}\frac{v_{s2}}{v_{g1}}\frac{v_o}{v_{s2}}\right)^2 \equiv \frac{4R_{si}}{R_p} A_{v1}^2 A_{v2}^2 A_{v3}^2, \tag{25}$$

where A_{v1}, A_{v2}, and A_{v3} can be easily derived from **Figure 5** as follows.

$$A_{v1} \equiv \frac{v_{g1}}{v_s} = Z_{in1} \Big/ \left(R_{si} + sL_g + \frac{1}{sC_i} + Z_{in1} \right) \tag{26}$$

$$A_{v2} \equiv \frac{v_{s2}}{v_{g1}} = A_1 Z_{in2} \Big/ \left(gZ_{out2} + Z_{in2} \right) \tag{27}$$

$$A_{v3} = \frac{v_o}{v_{s2}} = A_2 Z_o \Big/ \left(gZ_{out1} + Z_o \right) \tag{28}$$

4. Automatic Sizing Algorithm

Figure 8 shows the automatic sizing algorithm developed in this work. The inputs to the algorithm include design and process specifications, and the outputs include synthesized design variable values are for R_{DB}, W, nfb, L_s, L_g, C_i, R_1, L_1, C_o. Here, we explain the procedures from top to bottom in accordance with each step, which is explicitly indicated in **Figure 8**.

4.1. 1st Step: Entering Design and Process Specifications

The 1st step in the automatic sizing is to enter the design

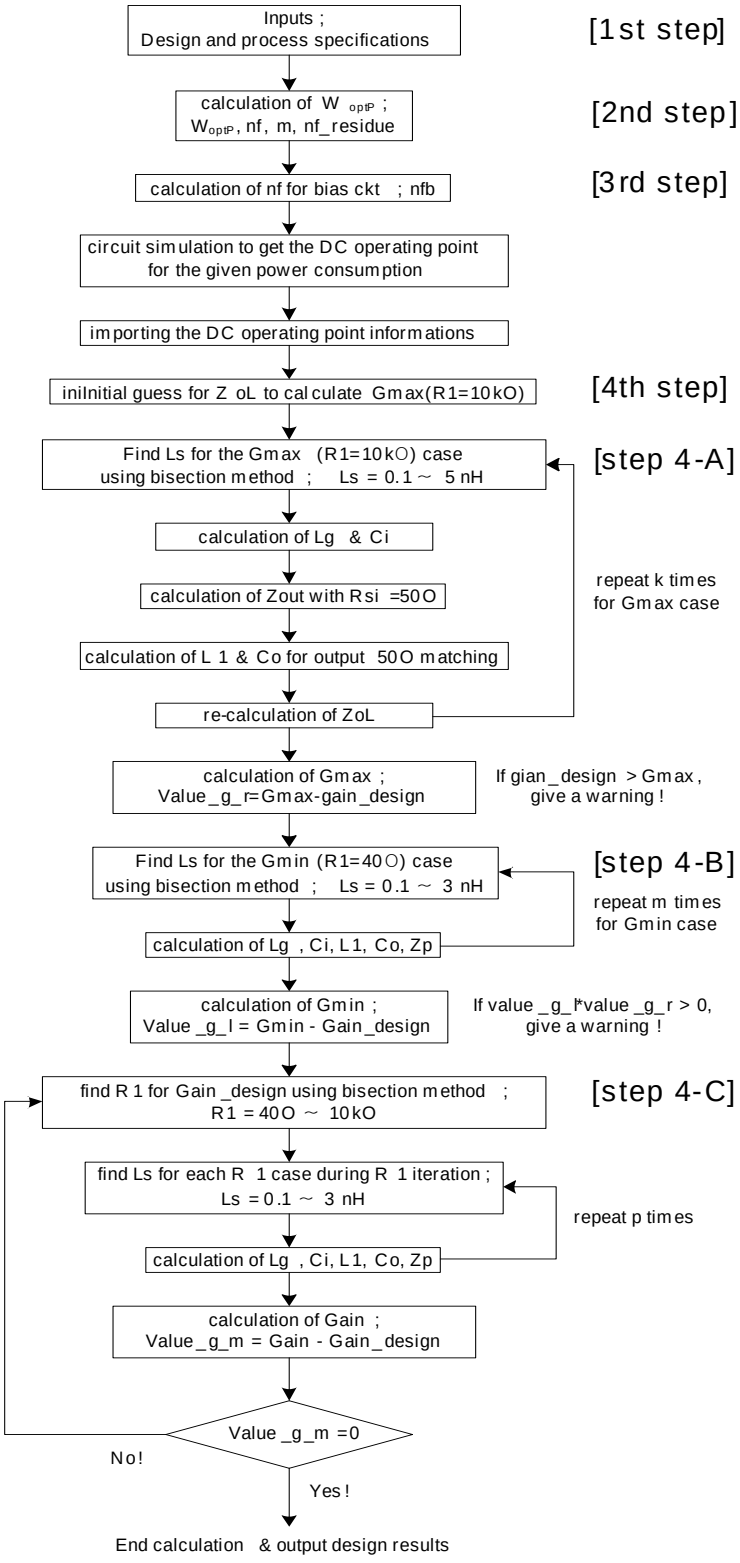


Figure 8. Automatic sizing algorithm.

and process specifications. The design specifications include the operating frequency f, the input output terminations R_{si} and R_{so}, the supply current I_{DD}, the desired power gain Gain_design. Instead of I_{DD}, the power consumption PWR and the supply voltage V_{DD} can be entered to calculate I_{DD} by PWR/V_{DD}. The process specifi-

cations include the transistor channel length L, the transistor channel width per finger WF, and the maximum finger number nf_max defined for one unit of transistors.

4.2. 2nd Step: Calculation of Optimum Transistor Width

The next step is to calculate the transistor channel width W for optimum noise performance. The width for optimum noise performance is usually too large for practical use, and therefore the power-constrained noise optimization (PCNO) device width W_{optP} [11] is adopted as W in this work. W_{optP} is calculated according to the last rough equation in (29).

$$W_{optP} = \frac{3}{2}\frac{1}{\omega L C_{ox} R_{si} Q_{sp}} \approx \frac{1}{3\omega L C_{ox} R_{si}} \tag{29}$$

As shown in (29), W_{optP} increases continuously as the frequency decreases. Therefore it may be necessary to define a maximum value for W considering lower frequency design. We suggest to limit W below 1000 μm.

If W_F and nf_max are defined, the finger number nf is first calculated as W/W_F, and the number of the maximum-fingered units m is calculated as the integer value of nf/nf_max, and the residual finger number nf_residue is determined as the residue to give an information for the transistor layout. Then the final W is determined by $W = W_F \times (m \times \text{nf\_max} + \text{nf\_residue})$. We note that W_F and nf_max are usually defined in most of recent processes.

4.3. 3rd Step: Calculation of Bias Circuit Design Variables and Getting DC Operating Point Information

The next step is to determine the bias circuit variable values and to get the dc operating point information.

The finger number for the bias transistor nfb and the drain bias resistance R_{DB} in **Figure 1** should be determined. By limiting the bias circuit current around 100 μA, for example, we can determine nfb by nfb = (100 μA/I_{DD}) × nf. For the decoupling resistor R_B, we can simply use 5 kΩ, which is a reasonable value.

The next procedure is to determine R_{DB}, which, however, is very difficult to determine by calculation. Since I_{DD} is sensitive to the value of R_{DB}, it should be manually determined to give the specified I_{DD} value by dc circuit simulations. This procedure is one obstacle against full design automation in this work. However, it is an essential procedure since it provides the accurate operating point information to proceed with the remaining part of the design automation. The needed operating point information include the values of g_m, g_{ds}, C_{gs}, C_{sg}, C_{gd}, C_{dg}, C_{ds}, C_{sd}, C_{js}, and C_{jd} of M_1 and M_2 in **Figure 1**, which should be imported into the automatic sizing algorithm.

4.4. 4th Step: Iterations to Determine Design Variable Values

There are three main iteration loops in the automatic sizing algorithm as shown in **Figure 8**. The 1st loop finds G_{max}, which corresponds to the case with the upper limit of R_1, which is chosen arbitrarily large enough as 10 kΩ in this work. To find G_{max}, we need to find all the design variable values for the G_{max} case simultaneously. Iteration is needed since the input and output matching designs affect each other. The 2nd loop finds G_{min}, which corresponds to the case with the lower limit of R_1, which is arbitrarily chosen small as 40 Ω in this work to allow a larger allowable gain range. This iteration is also needed for the same reason explained for the G_{max} case. The 3rd loop finds the proper R_1 value for the desired gain Gain_ design by the bisection method, which lies within the lower and upper boundaries G_{min} and G_{max}, and its inner loop finds the corresponding design variable values for the present gain value during iteration similarly as in the 1st and 2nd iteration loops.

4.4.1. Iterations to Solve for the G_{max} Case

As explained above, Z_{in1} is affected by output matching design, and Z_{out} is affected by input matching design. Therefore we need some iteration to determine L_s. Since Z_{in2} is affected by Z_o, which is unknown yet, we need an initial guess for Z_o to find the 1st L_s value. As shown in **Figure 8**, an initial guess for $Z_{oL} = Z_o//(1/sC_L)$ is given as 50/g·m^2, which is shown to be large enough for all possible situations in the procedure, to solve for Z_{in2} by (5).

The impedance seen at the gate of M_1 is equal to Z_{in1}, which is derived in (9). By setting the real part of Z_{in1} Re(Z_{in1}) equal to R_{si} for input impedance matching, we can find L_s. However this equation Re(Z_{in1}) = R_{si} is too complicated to get the solution directly with the other present design variables values given, and therefore L_s is solicited numerically within the lower and upper boundaries of 0.1 nH and 5 nH. We use the bisection method for this purpose.

The next procedure is to calculate L_g and C_i, which nullify the imaginary part of Z_{in1} Im(Z_{in1}) in **Figure 2**. Z_{in1} is usually capacitive to give a negative value for Im(Z_{in1}), and therefore L_g can be calculated using the equation Im(Z_{in1}) – 1/(ωC_i) + ωL_g = 0, where C_i is simply a large dc blocking capacitor. We first calculate L_{g1}, which nullifies Im(Z_{in1}) using Im(Z_{in1}) + ωL_{g1} = 0. Although C_i is larger the better, considering the layout size, 1/(ωC_i) = ωL_{g1}/10 is used to determine C_i. L_g is then recalculated using Im(Z_{in1}) – 1/(ωC_i) + ωL_g = 0.

Depending on to the operating frequency and the desired gain, Z_{in1} may happen to be inductive, or this situation can happen in the middle of the iterations. For this case, a nominal single bond wire inductance of 1 nH is

assumed for L_g and $\mathrm{Im}(Z_{\mathrm{in1}}) - 1/\omega C_i + \omega L_g = 0$ is used to calculate the required C_i value.

In the next procedure, the design variables L_1 and C_o are determined using the equations $\mathrm{Re}(Z_{\mathrm{out}}) = R_{so}$ and $\mathrm{Im}(Z_{\mathrm{out}}) = 0$ for output impedance matching to R_{so}, where $\mathrm{Re}(Z_{\mathrm{out}})$ is the real part of Z_{out} expressed in (13).

If we let Z_{out1} in (12) equal to $A + jB$, the real and imaginary parts of $Z_{\mathrm{out1}}//j\omega L_1$ in (13) are expressed as

$$\begin{aligned}\mathrm{Re}\left(Z_{\mathrm{out1}}//j\omega L_1\right) &= \frac{A\omega^2 L_1^2}{A^2 + \left(B + \omega L_1\right)^2} \text{ and} \\ \mathrm{Im}\left(Z_{\mathrm{out1}}//j\omega L_1\right) &= \frac{\left(A^2 + B^2\right)\omega L_1 + B\omega^2 L_1^2}{A^2 + \left(B + \omega L_1\right)^2}.\end{aligned} \tag{30}$$

Then by letting $\mathrm{Re}(Z_{\mathrm{out}}) = \mathrm{Re}(Z_{\mathrm{out1}}//j\omega L_1) = R_{so}$, L_1 is expressed as

$$L_1 = \frac{R_{so}B + \sqrt{R_{so}^2 B^2 + \left(A - R_{so}\right)\left(A^2 + B^2\right)R_{so}}}{\omega\left(A - R_{so}\right)} \tag{31}$$

By letting $\mathrm{Im}(Z_{\mathrm{out}}) = \mathrm{Im}(Z_{\mathrm{out1}}//j\omega L_1) - 1/(\omega C_o) = 0$, C_o is expressed as

$$C_o = \frac{1}{\omega \cdot \mathrm{Im}\left(Z_{\mathrm{out1}}//j\omega L_1\right)}. \tag{32}$$

Using (31) and (32), L_1 and C_o can be simply calculated.

Now the 1st set of the design variable values are ready to update Z_{oL} and the remaining iterations are performed to find the final design variable values for the $G_{\max}$ case. It was found that the iteration number for this loop should be larger than 10.

Right after the iteration loop, A_1, gZ_{out2}, A_2, and gZ_{out1} are calculated using (20), (17), (22), and (21), respectively, and $G_{\max}$ is calculated using (25).

If the $G_{\max}$ value is smaller than the desired gain, the routine gives a warning and stops.

4.4.2. Iterations to Solve for the $G_{\min}$ Case

The 2nd loop finds the design variable values for the $G_{\min}$ case. The same iteration as above with the last Z_{oL} value as an initial guess is performed to find $G_{\min}$ using (25) again.

4.4.3. Iterations to Solve for the Gain_Design Case

The 3rd loop finds the proper R_1 value for the desired gain Gain_design using the bisection method while the inner loop finds the corresponding design variable values for the present gain value. This inner iteration loop is exactly same as the 1st and 2nd loops. After all the design variables are determined for the present gain value, the gain is calculated using (25) again. If the calculated gain is equal to Gain_design within the allowed tolerance, the calculation stops to output the final set of the design variable values, which include W, nf, m, nf_residue, nfb, L_s, L_g, C_i, R_1, L_1, and C_o.

5. Verifications

The automatic sizing algorithm explained in Section 4 was coded using Matlab (Version 7.9.0.529) assuming usage of a 90 nm commercial CMOS process. The design variable sets for seven different operating frequencies ranging from 0.7 GHz to 5 GHz were synthesized, and verifications were done by one-time Spectre circuit simulations with the corresponding BSIM4.5.0 MOSFET model [12] for the assumed process.

The design specifications include I_D = 5 mA, V_{DD} = 1.2 V, Gain_design = 21 dB, and $R_{si} = R_{so}$ = 50 Ω. The process specifications include L = 75 nm, W_F = 3 μm, and nf_max = 64, where 75 nm for L is the effective channel length in this process. The maximum transistor width was set as $W_{\max}$ = nf_max × m × W_F = 64 × 5 × 3 μm = 960 μm, which is below 1000 μm as we suggested.

As examples of the verifications, **Figures 9** and **10** show the simulated LNA characteristics without any tuning for the operating frequency of 1 GHz and 5 GHz, respectively, when the corresponding sets of the design variable values obtained using the automatic sizing algorithm are used for the simulations. The synthesized design variable values are as follows;

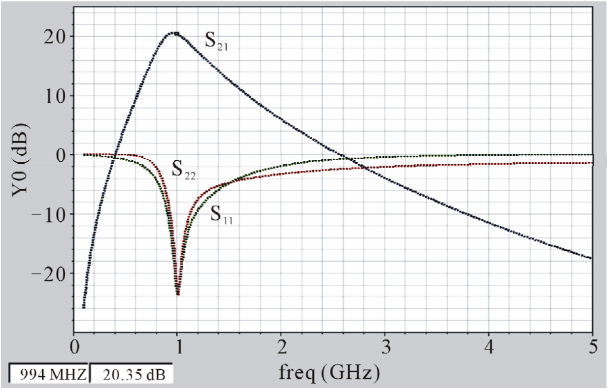
For 1 GHz design, R_{DB} = 12.7 kΩ, W = 960 μm (m = 5, nf_residue = 0), nfb = 6, L_s = 1.382 nH, L_g = 19.557 nH, C_i = 14.25 pF, R_1 = 497.1 Ω, L_1 = 11.904 nH, C_o = 1.447 pF.

For 5 GHz design, R_{DB} = 5.96 kΩ, W = 231 μm (m = 1, nf_residue = 13), nfb = 2, L_s = 0.5383 nH, L_g = 2.690 nH, C_i = 4.142 pF, R_1 = 1.752 kΩ, L_1 = 2.813 nH, C_o = 0.190 pF.

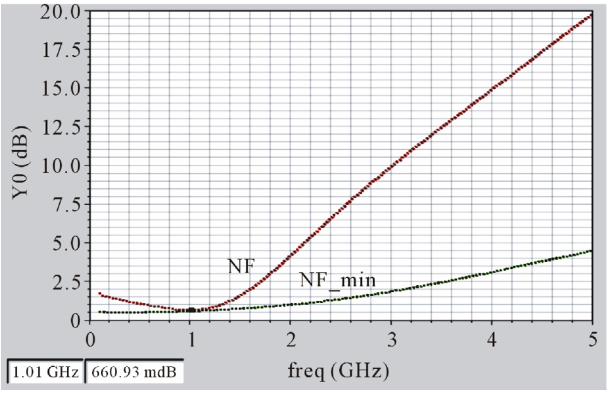
Table 1 summarizes the simulated results of the seven designs, which reside in the frequency range, where the automatic sizing program could provide the design variable set for Gain_design of 21 dB. Notice that, for the operating frequencies below 1 GHz, the synthesized W values are restricted to below 960 μm, which is equal to the value for $W_{\max}$.

In **Table 1**, we can see that the input and output matchings (S_{11} and S_{22}) are pretty good for all the designs, and the noise figure is pretty close to the noise figure minimum, which demonstrates the adequacy of the designs.

We note that power gain values are about the same with S_{21} values. The S_{21} values in **Table 1** are smaller than the desired gain of 21 dB. This seems to be caused by neglecting g_{mb}, C_{gb}, R_s, R_d, R_g, and R_{sub} in the equivalent circuit in **Figure 2**. However we believe that the result is pretty good for the first-cut quick design.



(a)



(b)

Figure 9. Simulated (a) *s* parameter and (b) noise characteristics for f = 1 GHz: S_{21} = 20.31 dB, NF = 0.660 dB, NF_{min} = 0.585, S_{11} = –23.6 dB, S_{22} = –23.0 dB.

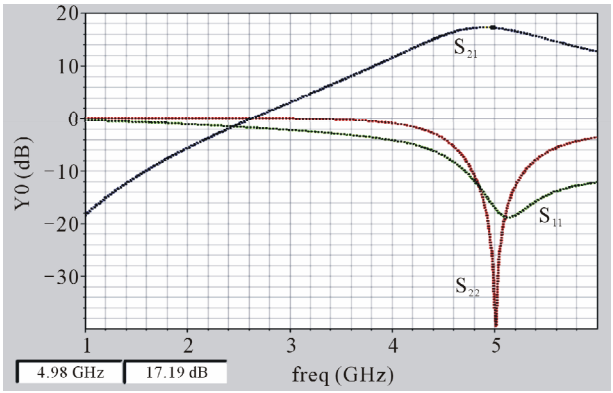


Figure 10. Simulated *s* parameters for f = 5 GHz: S_{21} = 17.16 dB, S_{11} = –16.9 dB, S_{22} = –34.8 dB.

Table 2 summarizes the synthesized available gain ranges with the corresponding R_1 values for each design. We can see that a wide range of power gain can be obtained by varying the R_1 values as expected.

6. Conclusions

The analytical expressions for the principle parameters were derived using the ac equivalent circuit of the single-ended narrow-band cascode CMOS LNA adopting the inductive source degeneration. Based on the expressions, the automatic sizing algorithm was developed by adopting the power-constrained noise optimization criteria. The algorithm was coded using Matlab, and could provide a set of design variable values within seconds. One-time Spectre simulations without any tuning assuming usage of a commercial 90 nm CMOS process were performed to confirm that the automatic sizing program can synthesize the aimed first-cut design with a reasonable accuracy for the frequency range reaching up to 5GHz.

This work showed in detail how the accurate automatic sizing can be done in an analytical approach. The approach can be applied to a common source LNA more easily since the derivation of principal parameters will be simpler with a fewer gain stages. It can be also applied to a differential LNA easily since the derivation will be basically same. The approach seems applicable to more complicated designs even though the derivation procedures will contain enhanced complexity.

The automatic sizing program may be utilized efficiently for additional tuning purpose. For example, after

Table 1. Simulation summary for the desired gain Gain_design of 21 dB.

f [GHz]	W [μm]	S_{21} [dB]	S_{11} [dB]	S_{22} [dB]	NF [dB]	NF_{min} [dB]
0.7	960	20.29	–24.0	–23.2	0.826	0.556
0.8	960	20.42	–24.8	–22.9	0.734	0.562
1	960	20.31	–23.6	–23.0	0.660	0.585
2	576	19.10	–20.3	–22.7	0.783	0.728
3	384	18.41	–19.0	–24.9	0.933	0.856
4	291	17.78	–17.5	–29.8	1.032	0.948
5	231	17.16	–17.0	–34.8	1.183	1.073

Table 2. Synthesis summary for the available gain ranges with the corresponding R_1 values.

f [GHz]	W [μm]	S_{21} [dB]	R_1 [Ω]
0.7	960	17.4 - 22.0	55.9 - 218
0.8	960	16.8 - 23.3	59.8 - 1.2 k
1	960	14.0 - 23.2	54.6 - 6.5 k
2	576	12.7 - 23.3	55.8 - 9.1 k
3	384	12.0 - 23.2	55.8 - 8.4 k
4	291	12.0 - 23.0	68.0 - 9.8 k
5	231	11.0 - 22.9	55.8 - 9.1 k

examining the first-cut synthesis result with verifying circuit simulations, a smaller value for W_{M2} compared to the synthesized one for W_{M1} can be entered into the automatic sizing program to obtain another design variable set for better linearity.

REFERENCES

[1] K. Muhammad, R. B. Staszewski and D. Leipold, "Digital RF Processing: Toward Low-Cost Reconfigurable Radios," *IEEE Communications Magazine*, Vol. 43, No. 8, 2005, pp. 105-113. doi:10.1109/MCOM.2005.1497564

[2] A. A. Abidi, "The Path to the Software-Defined Radio Receiver," *IEEE Journal of Solid-State Circuits*, Vol. 42, No. 5, 2007, pp. 954-966. doi:10.1109/JSSC.2007.894307

[3] G. Zhang, A. Dengi and L. R. Carley, "Automatic Synthesis of a 2.1 GHz SiGe Low Noise Amplifier," *Proceedings of IEEE Radio Frequency Integrated Circuits Symposium*, Seattle, 2-4 June 2002, pp. 125-128.

[4] M. Chu and D. J. Allstot, "An Elitist Distributed Particle Swarm Algorithm for RFIC Optimization," *Proceedings of Asia and South Pacific Design Automation Conference*, Shanghai, 18-21 January 2005, pp. 671-674.

[5] P. Vancorenland, C. De Ranter, M. Steyaert and G. Gielen, "Optimal RF Design Using Smart Evolutionary Algorithms," *Proceedings of Design Automation Conference*, Los Angeles, 4-8 June 2000, pp. 7-10.

[6] G. Tulunay and S. Balkır, "A Compact Optimization Methodology for Single-Ended LNA," *Proceedings of IEEE International Symposium Circuits and Systems*, Geneva, 23-26 May 2004, pp. V-273-V-276.

[7] T.-K. Nguyen, C.-H. Kim, G.-J. Ihm, M.-S. Yang and S.-G. Lee, "CMOS Low-Noise Amplifier Design Optimization Techniques," *IEEE Transactions on Microwave Theory and Technique*, Vol. 52, No. 5, 2004, pp. 1433-1442. doi:10.1109/TMTT.2004.827014

[8] G. Tulunay and S. Balkir, "Automatic Synthesis of CMOS RF Front-Ends," *Proceedings of IEEE International Symposium Circuits and Systems*, Island of Kos, 21-24 May 2006, pp. 625- 628.

[9] A. Nieuwoudt, T. Ragheb and Y. Massoud, "SOC-NLNA: Synthesis and Optimization for Fully Integrated Narrow-Band CMOS Low Noise Amplifiers," *Proceedings of Design Automation Conference*, San Francisco, 24-28 July 2006, pp. 879-884.

[10] W. Cheng, A. J. Annema and B. Nauta, "A Multi-Step P-Cell for LNA Design Automation," *Proceedings of IEEE International Symposium Circuits and Systems*, Seattle, 18-21 May 2008, pp. 2550-2553.

[11] T. H. Lee, "The Design of CMOS Radio-Frequency Integrated Circuits," 2nd Edition, Cambridge University Press, Cambridge, 2004.

[12] BSIM4.5.0 MOSFET Model, User's Manual, University of California, Berkeley, 2004.

