Dynamic Models for Complex Semiconductor Devices using VHDL-AMS

A thesis submitted to the
Division of Graduate Studies and Research of
the University of Cincinnati
in partial fulfillment of the
requirements for the degree of
MASTER OF SCIENCE
in the Department of
Electrical and Computer Engineering and Computer Science
of the College of Engineering

February, 2001

by
Shivesh Sabnekar
B.E., (ECE), Osmania University(Muffakham Jah College of Engineering and Technology), Hyderabad, India, 1998.

Thesis Advisor and Committee Chair: Dr. Harold W. Carter
Abstract

With intensive research being conducted in the field of VHDL-AMS modeling, the need for a library of simple primitive semiconductor devices has been recognized. These devices can then be used to develop complex devices of higher abstraction level.

VHDL-AMS with its unique features such as representation of ordinary differential-algebraic equations, discontinuity identification and processing, simultaneous representation of continuous-time and discrete-time descriptions presents itself as a powerful language to model analog and mixed signal systems. Previous works on assessing the capabilities of VHDL-AMS to model primitive device models such as diodes and static transistor models yielded cogent results. Hence, to further assess the capabilities of VHDL-AMS, devices that exhibit dynamic characteristics have to be modeled.

The intent of the research described in this thesis is to assess and demonstrate the ability of VHDL-AMS to model the dynamic behavior of complex semiconductor devices. The models thus developed were validated using SPICE.
Dedicated to Baba and Mummy
Acknowledgement

I would like to Thank Dr. Hal Carter for his guidance and support throughout this thesis. Working with him was a pleasure and honor.

My Thanks to Dr. Joseph H. Nevein and Dr. Fred Beyette for readily accepting to be my thesis committee members and their suggestions to improve my work.

Also, I would like to take this opportunity to Thank all the professors at the University of Cincinnati who taught the various courses which I took. I wish to Thank all my employers at the University of Cincinnati without whom I wouldn't have survived here!

My deepest regards to my parents, my brothers and my sister for their constant encouragement and support in my life. Without their good wishes and sacrifices I wouldn't have made it till here. Thanks, for believing in me guys and Thanks for being such a wonderful family.

Well, since any acknowledgement isn't complete mentioning friends, to merely complete the formality I would like to thank my friends, though they aren't responsible for my success.....oh cmon guys cool it I am just kidding!. Well, i think my friends are as much responsible for my successful completion of Masters as anyone else.

First I would like to Thank Santosh, Madhu, Prashant, Tapinder and Abhishek for being family rather than friends.

I would like to Thank the members of our self proclaimed fraternity "Ro-Beta-Ro" for all the fun and good times I had at the University of Cincinnati. Thank you Sohum for 'founding' this fraternity, Thanks Anil(don) for your valuable 'advises', Thanks Vishal for all the 'entertainment', Thanks Bipin for making your point 'loud n clear' and Thanks Dinesh(potta) for all those 'delicious' dishes.

Thanks to my labmates for bearing with me all these years, I am sure they would love to enjoy the silence after I leave the lab!...Thanks to Sanjeev, Kalyan, Murthy, Sudhir, Akber, Nandini, Shravanti, Josh, Ed for the good time I had with them. I would also like to thank my ex-labmates Sathya, Lalit, Vishwa, Sriram, Sarita, Venkat and Wan Fu.

Thanks to my friends Chakri, Vasan, Kasat, Kutty, Kicha, Uday along with whom I had taken the gruelling courses in VLSI.

Thanks to all the other friends of mine whom i might have missed out due to oversight or due to my sheer laziness to not to write further.

A word of thanks to Robert Frost for his lines "The Woods are beautiful.......and miles to go before I sleep" and 'unknown'(to me of course) for his lines "Earth, Wind.............a Mouthful of Sky", they have been an inspiration to me and will remain to inspire me further. Finally, Spl. Thanks go to : God!
Contents

List of Figures iv

List of Tables vi

1 Introduction 1
  1.1 Motivation .................................................. 1
  1.2 Problem Statement ........................................... 2
  1.3 Approach ..................................................... 2
  1.4 Related Work ................................................ 2
  1.5 Summary of Results ......................................... 3
  1.6 Frequently used terms ....................................... 4
  1.7 Organization of the Thesis .................................. 4

2 A Review Of The Analytical Transistor Models 5
  2.1 Bipolar Junction Transistor ................................. 5
    2.1.1 Large Signal Model .................................... 6
  2.2 Junction Field Effect Transistor ........................... 7
    2.2.1 Large Signal Model .................................... 8
  2.3 The MOSFET .................................................. 9
    2.3.1 Large Signal Model .................................... 10
  2.4 The BSIM .................................................... 12
    2.4.1 BSIM static model .................................... 13

3 Simulation Modeling and Results 16
  3.1 The Methodology ............................................. 16
  3.2 The Approach ................................................ 17
    3.2.1 The Experimental approach ............................. 18
    3.2.2 The Quantitative Analysis approach .................... 21
  3.3 The large-Signal BJT ....................................... 22
3.3.1 The I-V characteristics ........................................ 23
3.3.2 The Switching Characteristics assessment ............... 23
3.3.3 Fidelity assessment ............................................. 25
3.3.4 Summary .......................................................... 28
3.4 The JFET model ..................................................... 28
  3.4.1 I-V characteristics ............................................. 29
  3.4.2 The Switching Characteristics assessment ............... 31
  3.4.3 Fidelity assessment ............................................ 32
  3.4.4 Summary .......................................................... 34
3.5 The MOSFET model ................................................ 34
3.6 The I-V characteristics ............................................ 35
  3.6.1 The Switching Characteristics assessment ............... 37
  3.6.2 Fidelity assessment ............................................ 37
  3.6.3 Summary .......................................................... 39
3.7 The BSIM ............................................................. 41
3.8 Timing Results for the Circuit Models ......................... 41
3.9 The Experimental Environment .................................. 44

4 Conclusions and Future Work .................................... 45
  4.1 Summary ............................................................ 45
  4.2 Future Work ....................................................... 46

Bibliography ............................................................. 47

A VHDL-AMS CODE .................................................... 49
  A.1 Signal Generators ................................................ 49
    A.1.1 Ramp Wave ................................................... 49
    A.1.2 Square Wave ................................................ 49
    A.1.3 Sine Wave ..................................................... 50
  A.2 Large Signal BJT ................................................ 51
    A.2.1 Curve-Tracer for BJT ...................................... 54
    A.2.2 Inverter ....................................................... 55
    A.2.3 Amplifier ..................................................... 56
  A.3 Large-Signal JFET ............................................... 58
    A.3.1 Curve-Tracer ................................................ 60
    A.3.2 Inverter ....................................................... 61
    A.3.3 Amplifier ..................................................... 63
  A.4 Large Signal MOSFET .......................................... 64
<table>
<thead>
<tr>
<th>Section</th>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.4.1</td>
<td>Curve-Tracer</td>
<td>67</td>
</tr>
<tr>
<td>A.4.2</td>
<td>Inverter</td>
<td>68</td>
</tr>
<tr>
<td>A.4.3</td>
<td>Amplifier</td>
<td>70</td>
</tr>
<tr>
<td>A.5</td>
<td>BSIM</td>
<td>71</td>
</tr>
<tr>
<td>A.5.1</td>
<td>Curve-Tracer</td>
<td>74</td>
</tr>
</tbody>
</table>
List of Figures

2.1 Large Signal model of a npn BJT. .................................................. 6
2.2 Large Signal model of a JFET. .................................................... 8
2.3 Large signal model of an n-channel MOSFET ............................. 11

3.1 The Curve-trace arrangement. .................................................... 19
3.2 Testbench set-up for Inverter circuit. ........................................... 20
3.3 Testbench set-up for Amplifier circuit. ........................................ 20
3.4 Absolute and Relative Errors. .................................................... 21
3.5 Equations for currents $I_{bc}$ and $I_{bc}$ in VHDL-AMS ................ 22
3.6 I-V characteristics for large-signal npn BJT, lower curve is for $V_{bb}=1.0v$ and the upper curve is for $V_{bb}=2.0v$. ...................... 24
3.7 Distribution of absolute error for the BJT I-V characteristics ....... 25
3.8 Output from Inverter testbench for large-signal npn BJT ............. 26
3.9 Distribution of absolute error for the Inverter circuit set-up for BJT.. 26
3.10 Output from Amplifier testbench for large-signal npn BJT......... 27
3.11 Distribution of absolute error for the Amplifier circuit set-up for BJT. 28
3.12 Equations for current $I_d$ in normal mode, represented in VHDL-AMS 29
3.13 I-V characteristics for large-signal n-channel JFET, the upper curve is for $V_{gg} = -1.0v$ and the lower curve is for $V_{gg} = -2.0v$ .......... 30
3.14 Distribution of absolute error for the for JFET I-V characteristics .. 30
3.15 Output from Inverter set-up for large-signal n-channel JFET. ....... 31
3.16 Distribution of absolute error for the Inverter circuit set-up for JFET. 32
3.17 Output from Amplifier set-up for large-signal n-channel JFET. .... 33
3.18 Distribution of absolute error for the Amplifier circuit set-up. ....... 33
3.19 Equations for current $I_{ds}$ represented in VHDL-AMS ............... 34
3.20 I-V characteristics of a large-signal NMOS, the upper curve is for $V_{io} = 10$ and the lower curve is for $V_{io} = 2.0$ ...................... 36
3.21 Distribution of absolute error for MOSFET I-V characteristics. .... 36
3.22 Output from Inverter set-up for large-signal NMOS. ................. 38
3.23 Distribution of absolute error for the Inverter circuit set-up for MOSFET. 38
3.24 Output from Amplifier set-up for large-signal NMOS. .............. 39
3.25 Distribution of absolute error for the Amplifier circuit set-up for MOS-
FET. ................................................................. 40
3.26 Equations for current $I_{ds}$ represented in VHDL-AMS .............. 42
3.27 Output from Curve-tracer set-up for static BSIM. ......................... 43
3.28 Output from Curve-tracer set-up for static BSIM (Input Characterisitcs). 43
List of Tables

2.1 BJT large-signal model parameters ........................................ 7
2.2 JFET large-signal model parameters ........................................ 9
2.3 MOSFET large-signal model parameters .................................... 12
2.4 List of BSIM parameters ...................................................... 15
3.1 Model parameters of npn BJT ................................................ 24
3.2 Component Values of the Curve-Tracer set-up for npn BJT .............. 24
3.3 Component Values of the Inverter circuit set-up for npn BJT .......... 25
3.4 Component Values of the Amplifier circuit set-up for npn BJT ......... 27
3.5 Summary of results for large-signal npn BJT .............................. 28
3.6 Parameter values for Large-Signal JFET ................................. 29
3.7 Component Values of the Curve-Tracer set-up for n-channel JFET ... 30
3.8 Component Values of the Inverter circuit set-up for n-channel JFET . 31
3.9 Component Values of the Amplifier circuit for n-channel JFET ..... 32
3.10 Summary of results for large-signal n-channel JFET .................... 34
3.11 MOSFET large-signal model parameters ................................. 35
3.12 Component Values of the Curve-Tracer set-up for NMOS ............ 35
3.13 Component Values of the Amplifier circuit set-up for NMOS ......... 37
3.14 Component Values of the Amplifier circuit for NMOS ............... 39
3.15 Summary of results for large-signal NMOS ............................. 40
3.16 List of BSIM parameters ................................................... 42
3.17 Component Values of the Curve-Tracer set-up for static BSIM ...... 43
3.18 Timing results .............................................................. 44
4.1 Summary of Results .......................................................... 46
Chapter 1

Introduction

This thesis presents methods for modeling large-signal transistors in VHDL-AMS\textsuperscript{1} and determining their performance by means of validation and quantitative analysis. All the transistor models developed in this thesis are based on the Level 2 implementation in SPICE\textsuperscript{2}. In the following sections we present the motivation behind this thesis, the problem statement and summary of the results.

1.1 Motivation

The unique features such as representation of general ordinary differential algebraic equations(ODAEs), discontinuity identification and processing and several other features [14] [5] allows VHDL-AMS to model analog and mixed-signal circuits. This has engendered intensive research in modeling of semiconductor devices.

Previous work [6] concentrated on developing simple semiconductor devices, such as diodes and static transistor models which involved only few physical parameters,\textsuperscript{3} using VHDL-AMS. The models and their implementations in various circuits were found to be fairly accurate. Hence this thesis is a follow-on to previous work and concentrates on developing semiconductor device models that exhibit dy-

\textsuperscript{1}VHDL for Analog and Mixed Signals
\textsuperscript{2}Simulation Program with Integrated Circuit Emphasis
\textsuperscript{3}they have a direct and quantifiable physical meaning for example the length and width of a transistor
**1.2 Problem Statement**

The problem we address in this thesis is the degree to which VHDL-AMS is capable of modeling devices with dynamic (i.e. charge-controlled) characteristics such that the accuracy of simulation behavior meets or exceeds similar models in SPICE 3\textsuperscript{4}. In particular we investigate the capability of VHDL-AMS to support the modeling of effective dynamic models for BJT, JFET and MOSFET transistors including the short-channel channel BSIM-1 model.

Besides our interest in the possibility of effective device models in VHDL-AMS we are interested in showing how best to create such models and draw guidelines that VHDL-AMS modelers can use when creating future device models or using the existing device models.

**1.3 Approach**

The approach adopted to investigate our problem is mainly to develop effective dynamic models in VHDL-AMS, execute them on SEAMS\textsuperscript{5} [11] the mixed signal simulator developed in our laboratory (*Distributed Processing Laboratory, University of Cincinnati*) and validate it against similar models in SPICE for accuracy.

**1.4 Related Work**

The work presented in this thesis is a follow-on to research earlier done by Vishwashanth Kasula [6] in the Distributed Processing Laboratory at Univeristy of Cincinnati. His work involved investigating whether semiconductor device models

\textsuperscript{4}here and elsewhere in this thesis SPICE is refered as a general term for SPICE3 simulator and SPICE3 language [13] [12] [15]

\textsuperscript{5}Simulation Environment for Analog and Mixed Signals
such as diodes and static models of transistors in VHDL-AMS which have equivalent performance as SPICE can be modeled in VHDL-AMS, studying how are the features of VHDL-AMS are best suited to describe the models, executing them on SEAMS and validating them against similar models in SPICE for accuracy and execution speed of the SEAMS. The results presented in his thesis were found to be comparable (SEAMS vs SPICE) for accuracy and, with respect to the speed of execution SEAMS was found slower than SPICE, the reason for which are cited in [9]. The development of the dynamic transistor models and their validation was left for future research in his thesis. The effort reported in this thesis is a continuation of his work.

Apart from the research carried out in Distributed Processing Laboratory at University of Cincinnati, a similar research in the area of VHDL-AMS modeling is being conducted in the EDA research group, Univeristy of Southampton, United Kingdom, Duke University, Analogy Inc and many others. For more information on the related work on VHDL-AMS modeling refer http://www.vhdl-ams.org.

1.5 Summary of Results

The aim of this thesis was to develop models which are accurate when compared to similar models in SPICE with SPICE being considered as benchmark. The important results obtained after using the models in the testbench circuits, are summarized as follows:

1. BJT : The maximum error in dB for this model is -21.36 for the curve-tracer circuit and the minimum error in dB is -58.06 for the inverter circuit
2. JFET: The maximum error in dB for this model is -34.78 for the inverter circuit and the minimum error in dB is -67.13 for the curve-tracer circuit
3. MOSFET: The maximum error in dB for this model is -28.6 for the amplifier circuit and the minimum error in dB is -118.06 for the curve-tracer circuit.

The above results show that the worst-case maximum error considering all models studied is -21.36dB, which is reasonably good.
1.6 Frequently used terms

1. Analytical model: It is the mathematical representation of the device models in terms of conditional equations in different region of operations.

2. Simulation models: It is the representation of the analytical model in VHDL-AMS.

3. Testbench: It is the set-up used by simulation models to verify the Output, switching and fidelity characterisics of the model. It is interchangeably used as the term circuit model in some parts of this thesis.

4. Validation: Whenever this term is used it is implied that the models developed in VHDL-AMS are being compared against similar models in SPICE.

1.7 Organization of the Thesis

- **Chapter 2** is an review of the theory of the analytical transistor models which were developed into respective simulation models. It gives a mathematical description of the transistor models.

- **Chapter 3** discusses the modeling of the transistors described in Chapter 2, the testbenches used for verification, the validation of the models for accuracy and an analysis of the results.

- **Chapter 4** concludes this thesis and discusses the scope for future work.
Chapter 2

A Review Of The Analytical Transistor Models

In this chapter we discuss the different transistor models vis-a-vis the mathematical equations which are used to describe the operation of the transistors in the various regions. This chapter is summary of the transistor models that have been implemented in VHDL-AMS as described in Chapter 3.

2.1 Bipolar Junction Transistor

A typical $npe(pnp)$ BJT\(^1\) consists of a thin layer $p(n)$-type material between two layers of $n(p)$-type material. The modulation of the current-flow in one pn junction due to change of bias on the nearby junction is the basic operation of a BJT. This property is refered to as *bipolar transistor action*. In bipolar transistors, both positive and negative free carriers take part in the device operation. The operation of $npe$ and $pnp$ is similar in every respect except that the roles of electrons and holes are interchanged and polarities of the terminal currents and voltages are reversed. There are several BJT models available such as the *Ebbers-Moll* model and the *Gummel-Poon* model. We present the Large Signal model of the *Ebbers-Moll* BJT.

\(^1\)Bipolar Junction Transistor
2.1.1 Large Signal Model

Charge storage effects in a BJT can be modeled with help of three types of capacitances: two linear junction capacitances ($C_{jE}$, $C_{je}$), two nonlinear capacitances ($C_{de}$, $C_{dc}$) and a constant substrate capacitance ($C_{js}$). The junction and diffusion capacitances can be grouped together to form the capacitances $C_{be}(=C_{de}+C_{je})$, $C_{bc}(=C_{dc}+C_{jc})$ and $C_{cs}(=C_{js})$, as shown in Figure 2.1. These capacitances can be equivalently represented by voltage-controlled equations as follows

\[
C_{be} = \begin{cases} 
\tau_F \frac{dI_{be}}{dV_{be}} + C_{je}(0)(1 - \frac{V_{be}}{\phi_E})^{-m_E} & : V_{be} \leq FC \times \phi_E \\
\tau_F \frac{dI_{be}}{dV_{be}} + \frac{C_{je}(0)}{F_2}(F_3 + \frac{m_E V_{be}}{\phi_E}) & : V_{be} \geq FC \times \phi_E 
\end{cases}
\]  

(2.1)

\[
C_{bc} = \begin{cases} 
\tau_R \frac{dI_{bc}}{dV_{bc}} + C_{jc}(0)(1 - \frac{V_{bc}}{\phi_C})^{-m_C} & : V_{bc} \leq FC \times \phi_C \\
\tau_R \frac{dI_{bc}}{dV_{bc}} + \frac{C_{jc}(0)}{F_2}(F_3 + \frac{m_C V_{bc}}{\phi_C}) & : V_{bc} \geq FC \times \phi_C 
\end{cases}
\]  

(2.2)

Figure 2.1: Large Signal model of a npn BJT.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CJE</td>
<td>Base-emitter zero-bias depletion capacitance ($C_{je}$)</td>
</tr>
<tr>
<td>CJC</td>
<td>Base-collector zero-bias depletion capacitance ($C_{jc}$)</td>
</tr>
<tr>
<td>CJX</td>
<td>Collector-substrate zero-bias depletion capacitance ($C_{js}$)</td>
</tr>
<tr>
<td>VJE</td>
<td>Base-emitter built-in potential $\phi_E$</td>
</tr>
<tr>
<td>VJC</td>
<td>Base-collector built-in potential $\phi_C$</td>
</tr>
<tr>
<td>VJS</td>
<td>Substrate junction built-in potential $\phi_S$</td>
</tr>
<tr>
<td>TF</td>
<td>Ideal total forward transit time $\tau_F$</td>
</tr>
<tr>
<td>TR</td>
<td>Ideal total reverse transit time $\tau_R$</td>
</tr>
<tr>
<td>FC</td>
<td>Coefficient for forward-bias depletion capacitance (FC)</td>
</tr>
<tr>
<td>MJE</td>
<td>Base-emitter junction grading factor ($m_E$)</td>
</tr>
<tr>
<td>MJC</td>
<td>Base-collector junction grading factor ($m_C$)</td>
</tr>
<tr>
<td>MJS</td>
<td>Collector-substrate junction grading factor ($m_S$)</td>
</tr>
</tbody>
</table>

Table 2.1: BJT large-signal model parameters

$$C_{cs} = \begin{cases} 
C_{js}(0)(1 - \frac{V_c}{\phi_S})^{-m_S} & : V_{cs} \leq 0 \\
C_{js}(0)(1 + \frac{m_S V_c}{\phi_S}) & : V_{cs} \geq 0 
\end{cases} \quad (2.3)$$

where the first sum terms account for the diffusion capacitance and the second for junction. All the parameters in the above equation that are required to characterize the large-signal model of a BJT are described in the Table 2.1. F_2 and F_3 are the model constants given by the following equations.

$$F_2 = (1 - FC)^{1+m} \quad (2.4)$$

$$F_3 = 1 - FC(1 + m) \quad (2.5)$$

where $m$ is $m_E$ for base-emitter junction and $m_C$ for base-collector junction.

### 2.2 Junction Field Effect Transistor

The JFET\(^2\) is a semiconductor device whose operation depends on the control of current by an electrical field. Since the conduction involves predominantly one kind of carrier, the JFET is also called a *unipolar* transistor. It consists of a conductive

\(^2\)Junction Field Effect transistor
channel that has two ohmic contacts, one acting as the cathode (source) and the other acting as the anode (drain). The conduction in this channel is facilitated by applying an appropriate voltage between source and drain. The third electrode (gate) forms the rectifying junction with the channel. In accordance with the channel formed, we have two types of JFET configurations: n-channel (majority carriers are electrons) and p-channel (majority carriers are holes) JFETs. Here we present the large-signal models of n-channel JFET and the p-channel JFET models can be appropriately obtained by reversing the junction voltages and currents.

### 2.2.1 Large Signal Model

Charge storage in a JFET occurs in the two gate junctions. The capacitances involved are just those due to ionic space charge in the depletion regions, as neither of the gate junctions normally is forward-biased. These two capacitances are modeled as shown in the large-signal model of a n-channel JFET in Figure 2.2. These capacitances can be expressed by the following voltage-dependent equations.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CGS</td>
<td>Zero-bias gate-source junction capacitance (C_{gs}(0))</td>
</tr>
<tr>
<td>CGD</td>
<td>Zero-bias gate-drain junction capacitance (C_{gd}(0))</td>
</tr>
<tr>
<td>PB</td>
<td>Gate junction potential ( \phi_O )</td>
</tr>
<tr>
<td>Fc</td>
<td>Forward-bias depletion capacitance coefficient</td>
</tr>
<tr>
<td>m</td>
<td>Junction grading coefficient (m)</td>
</tr>
</tbody>
</table>

Table 2.2: JFET large-signal model parameters

\[
C_{gs} = \begin{cases} 
C_{gs}(0)(1 - \frac{V_{gs}}{\phi_O})^{-m} & : V_{gs} \leq FC \times \phi_O \\
\frac{C_{gs}(0)}{F_2}(F_3 + \frac{mV_{gs}}{\phi_O}) & : V_{gs} \geq FC \times \phi_O 
\end{cases} \quad (2.6)
\]

\[
C_{gd} = \begin{cases} 
C_{gd}(0)(1 - \frac{V_{gd}}{\phi_O})^{-m} & : V_{gd} \leq FC \times \phi_O \\
\frac{C_{gd}(0)}{F_2}(F_3 + \frac{mV_{gd}}{\phi_O}) & : V_{gd} \geq FC \times \phi_O 
\end{cases} \quad (2.7)
\]

where \( F_2 \) and \( F_3 \) are model constants given by equations 2.4 and 2.5 respectively. The parameters required to characterize the JFET are listed in the Table 2.2.

### 2.3 The MOSFET\(^3\)

Metal Oxide Semiconductor Field Effect Transistor (MOSFET) is a four terminal device consisting of source(S), drain(D), gate(G) and the substrate(B). Unlike the JFET where the gate is electrically connected to the source and drain, the MOSFET gate is insulated from source and the drain, hence MOSFET is sometimes referred to as Insulated-Gate Field-Effect transistor (IGFET). MOSFET structure has two possible configurations: \textit{n-channel} and \textit{p-channel}, which is determined by the substrate over which the device is fabricated. A \textit{n-channel} MOSFET (n-MOS) is made of a \textit{p-type substrate} on which a layer of thermal-oxide is built. A high concentration of \textit{n-type} dopant is thermally diffused inside the silicon to obtain so-called \textit{source} and \textit{drain}. On the thin-oxide layer a conducting material is deposited, which acts as a \textit{gate}. The region covered by the thin-oxide layer and by the gate represents the \textit{chan-\(^3\)Metal Oxide Semiconductor Field Effect Transistor
nel. A p-channel MOSFET (p-MOS) is also fabricated in a similar fashion, by using
n-type substrate and a p-type dopant.

A MOS transistor is termed a majority-carrier device, in which the current
in a conducting channel between source and drain is modulated by a voltage applied
to the gate. The normal conduction characteristics of an MOS transistor can be
divided into three regions of operation based on the voltages as shown below:

- **Cut-off region**: where the current flow is essentially zero.

- **Non-Saturated (Linear) region**: weak inversion region where the drain cur-
rent is dependent on the gate and the drain voltage, with respect to the substrate.

- **Saturated region**: channel is strongly inverted and the drain current flow is
ideally independent of the drain-source voltage (strong inversion region)

Here we present the large-signal model of a n-channel MOSFET, the p-
channel MOSFET model can be obtained by appropriately reversing the direction of
junction voltages and currents.

### 2.3.1 Large Signal Model

Figure 2.3 shows the large-signal model for a n-channel MOSFET, to account
for the charge-storage effects in MOSFET we use a model similar to the one proposed
by Meyer []. The charge-storage effects in the large-signal model are represented by
three non-linear capacitances: $C_{gb}$, $C_{gs}$ and $C_{gd}$. The equations

for these capacitances are as follows:

**Accumulation region**

For $V_{gs} \leq V_{on} - 2\phi_P$

$$C_{gb} = C_{ox} + C_{gbo}L_{eff}$$
$$C_{gs} = C_{gso}W$$
$$C_{gd} = C_{gdo}W$$

**Depletion region**
Figure 2.3: Large signal model of an n-channel MOSFET

For $V_{on} - 2\phi_P \leq V_{gs} \leq V_{on}$

$$
C_{gb} = C_{ox} \frac{V_{on} - V_{gs}}{2\phi_P} + C_{gbo}L_{eff} \\
C_{gs} = \frac{2}{3} C_{ox} \left( \frac{V_{on} - V_{gs}}{2\phi_P} + 1 \right) + C_{gso}W \\
C_{gd} = C_{gdo}W
$$

(2.9)

Saturation region

For $V_{on} < V_{gs} < V_{on} + V_{ds}$

$$
C_{gb} = C_{gbo}L_{eff} \\
C_{gs} = \frac{2}{3} C_{ox} + C_{gso}W \\
C_{gd} = C_{gdo}W
$$

(2.10)

Linear region

For $V_{gs} > V_{on} + V_{ds}$

$$
C_{gb} = C_{gbo}L_{eff} \\
C_{gs} = C_{ox} \left\{ 1 - \left[ \frac{V_{gs} - V_{on} - V_{ds}}{2(V_{gs} - V_{on} - V_{ds})} \right]^2 \right\} + C_{gso}W \\
C_{gd} = C_{ox} \left\{ 1 - \left[ \frac{V_{gs} - V_{on}}{2(V_{gs} - V_{on})} \right]^2 \right\} + C_{gdo}W
$$

(2.11)
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CGSO</td>
<td>Zero-bias gate-source junction capacitance ($C_{gso}$)</td>
</tr>
<tr>
<td>CGDO</td>
<td>Zero-bias gate-drain junction capacitance ($C_{gdo}$)</td>
</tr>
<tr>
<td>CGBO</td>
<td>Zero-bias gate-substrate junction capacitance ($C_{gbo}$)</td>
</tr>
<tr>
<td>PHI</td>
<td>Gate junction potential $\phi_p$</td>
</tr>
<tr>
<td>COX</td>
<td>The channel capacitance ($C_{ox}$)</td>
</tr>
</tbody>
</table>

Table 2.3: MOSFET large-signal model parameters

where $V_{on} = V_{TH} + \frac{nKT}{q}$ and $n = 1 + \frac{qN_{FS}}{C_{oz}} + \frac{C_B}{C_{oz}}$ if $N_{FS}$ is provided else $V_{on} = V_{TH}$ and $L_{eff} = L - 2X_{ji} ; C_{ox} = C_{ox}' W L_{eff}$

The parameters required represent the large signal-model of the MOSFET are listed in the Table 2.3.

2.4 The BSIM\(^4\)

The BSIM(Berkeley Short-Channel IGFET\(^5\) Model) is based on the device physics of small-geometry of MOSTs(Metal Oxide Semiconductor Transistors). The special effects included in a BSIM are (1) the vertical field dependence of carrier mobility, (2) the carrier-velocity saturation, (3) the drain induced barrier lowering, (4) the depletion charge-sharing by the source and drain, (5) the nonuniform doping profile for the ion-implanted devices, (6) the channel-length modualtion, (7) the sub-threshold conduction, and (8) geometric dependencies.

With so many effects included the number of parameters which influence the operation of a BSIM increase. The parameters involved in a BSIM model can be classified into (a) Physical parameters and (b) Electrical parameters.

(a) Physical parameters: They are treated as having direct and quantifiable physical meaning, an example is the gate oxide thickness

(b) Electrical parameters: They are the parameters defined from parameter extraction\(^6\), and thus may actually have little direct physical meaning. As the

\(^4\)Berkeley Short-Channel IGFET Model
\(^5\)Insulated Gate Field-Effect Transistor
\(^6\)It is the method using which values of parameters are determined from electrical data.
basic model formulation evolves into more complex forms, the number of electrical parameters become large. The value of an electrical parameter is obtained by

\[
P = P_O + \frac{P_L}{L_{eff}} + \frac{P_W}{W_{eff}}
\]

(2.12)

where \( P_O, P_L, P_W \) are the process parameters associated with the electrical parameter.

In this thesis we present the static model of a BSIM.

2.4.1 BSIM static model

In this section the BSIM equations for the static model [2] are summarized. The various parameters in the BSIM are used to model the threshold voltage and the drain current.

a. Threshold Voltage.

\[
V_{th} = V_{FB} + \phi_S + K_1\sqrt{\phi_S - V_{BS}} - K_2(\phi_S - V_{BS}) - \eta V_{DS}
\]

(2.13)

where \( \eta \) can be further modeled as

\[
\eta(V_{DS}, V_{BS}) = \eta_0 + \eta_B V_{DS} + \eta_D(V_{DS} - V_{DD})
\]

b. Drain Current.

Cutoff region \( V_{GS} \leq V_{TH} \)

\[
I_{DS} = 0
\]

(2.14)

Linear region \( V_{GS} > V_{th} \) and \( V_{DS} > V_{GS} - V_{th} \)

\[
I_{DS} = \frac{\mu O}{(1 + U_O(V_{GS} - V_{th}))} \frac{C'_{ox} W_{eff}}{(L_{eff} + U_1 V_{DS})} [(V_{GS} - V_{th})V_{DS} - \frac{\alpha}{2} V_{DS}^2]
\]

(2.15)

where \( \alpha = 1 + \frac{g K_1}{2 \sqrt{T \phi_S - V_{BS}}} \) and \( g = 1 - \frac{1}{1.744 + 0.8364(\phi_S - V_{BS})} \)
Saturation region $V_{GS} > V_{th}$ and $V_{DS} > V_{GS} - V_{th}$

$$I_{DS} = \frac{\mu O}{(1 + U_O(V_{GS} - V_{th}))} \frac{C'_{ox} W_{eff} (V_{GS} - V_{TH})^2}{2L_{eff} \alpha K}$$

(2.16)

where $K = \frac{1 + v_c + \sqrt{1 + 2v_c}}{2}$ and $v_c = \frac{v_1 (V_{GS} - V_{TH})}{\alpha}$

The above equations are determined by the carrier mobility $\mu O$ and the parameters $U_O$ and $U_1$ which account for mobility degradation effects due to vertical and horizontal electrical fields, respectively. The factor $\alpha$ represents the bulk(body) doping effect, while the factor $g$ is the coefficient of average body effect on drain current. The parameters $U_O$ and $U_1$ are bias dependent and are modeled as follows:

$$U_O(V_{DS}, V_{BS}) = U_{OZ} + U_{OB} V_{BS}$$

(2.17)

$$U_1(V_{DS}, V_{BS}) = U_{1Z} + U_{O1} V_{BS} + U_{1D} (V_{DS} - V_{DD})$$

(2.18)

The parameter $\mu O$ is obtained by quadratic interpolation through three data points: $\mu O$ at $V_{DS} = 0$, $\mu O$ at $V_{DS} = V_{DD}$, and the sensitivity of $\mu O$ to the drain bias at $V_{DS} = V_{DD}$, with

$$\mu O(V_{DS} = 0) = \mu Z + \mu Z_B V_{BS}$$

$$\mu O(V_{DS} = V_{DD}) = \mu S + \mu S_B V_{BS}$$

The parameters required to model a BSIM have been listed in the Table 1.4
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOX</td>
<td>Gate Oxide Thickness</td>
</tr>
<tr>
<td>LD</td>
<td>Source/Drain Underdiffusion of gate</td>
</tr>
<tr>
<td>WD</td>
<td>Isolation reduction of Channel width</td>
</tr>
<tr>
<td>VDD</td>
<td>Maximum applied supply voltage</td>
</tr>
<tr>
<td>MUZ</td>
<td>Zero bias Low field Mobility</td>
</tr>
<tr>
<td>X2MZ</td>
<td>Substrate bias effect on low field mobility</td>
</tr>
<tr>
<td>MUS</td>
<td>High drain bias mobility</td>
</tr>
<tr>
<td>X2MS</td>
<td>Substrate bias effect on high drain bias mobility</td>
</tr>
<tr>
<td>X3MS</td>
<td>Slope of mobility at Vds = Vdd</td>
</tr>
<tr>
<td>UO</td>
<td>Gate field induced mobility reduction parameter</td>
</tr>
<tr>
<td>X2UO</td>
<td>Substrate bias effect on Gate field induced mobility reduction</td>
</tr>
<tr>
<td>VFB</td>
<td>Flatband Voltage</td>
</tr>
<tr>
<td>PHI</td>
<td>Surface Potential</td>
</tr>
<tr>
<td>K1</td>
<td>Body effect on threshold voltage(first order)</td>
</tr>
<tr>
<td>K2</td>
<td>Body effect on threshold voltage(second order)</td>
</tr>
<tr>
<td>ETA</td>
<td>DIBL coeff when Vds = Vdd and Vbs = 0</td>
</tr>
<tr>
<td>X2E</td>
<td>Substrate bias effect on DIBL coefficient</td>
</tr>
<tr>
<td>X3E</td>
<td>Drain bias effect on DIBL coefficient</td>
</tr>
<tr>
<td>U1</td>
<td>High drain field mobility reduction parameter</td>
</tr>
<tr>
<td>X2U1</td>
<td>Substrate bias effect on High drain field mobility reduction</td>
</tr>
<tr>
<td>X3U1</td>
<td>Drain bias effect on High drain field mobility reduction</td>
</tr>
<tr>
<td>NO</td>
<td>Low field subthreshold ideality factor</td>
</tr>
<tr>
<td>NB</td>
<td>Substrate bias effect on the subthreshold ideality factor</td>
</tr>
<tr>
<td>ND</td>
<td>Drain bias effect on the subthreshold ideality factor</td>
</tr>
</tbody>
</table>

Table 2.4: List of BSIM parameters
Chapter 3

Simulation Modeling and Results

In this chapter we present the methodology adopted to model the various transistor models described in Chapter 2. We also show the implementation of the same in different testbench circuits to verify. A significant part of this chapter illustrates the validation of results(plots) obtained from the testbench circuits with their respective SPICE models\(^1\). The accuracy of the model is determined by calculating the error in dB.

3.1 The Methodology

Throughout the development of the simulation models presented in this chapter, we followed a clearly defined set of steps i.e. a methodology for creating and evaluating the correctness of the models. The specific steps are:

- Create a simulation model of the transistor based on the analytical.

- Identify the testbench circuit in which the model is to be implemented. Select the component values for the circuit and parameters values for the model.

- Verify the correctness of the parameter and component values by doing a mathematical analysis\([3]\) of the circuit.

\(^1\)SPICE has been considered as the benchmark to validate the models
• After confirming the component values and parameter values, use circuit analysis techniques such as Load Line Analysis [3] to determine the quiescent point. For instance, if transistor is to be used as an amplifier, design the circuit component values such that the transistor operates in a linear region. Determining the quiescent point helps in assigning the initial conditions to the transistor model.

• While developing the circuit model in VHDL-AMS make sure that all the modeling rules (syntactic and semantic) given in the LRM [?] are being obeyed.

• Run the circuit model on a suitable simulator (in this thesis the simulator is SEAMS). The results obtained from the simulation should be verified with those obtained from the Load Line analysis.

• Finally, validate the circuit model results against a standard benchmark simulator (in this thesis the benchmark simulator is SPICE) and use quantitative analysis techniques [1] to calculate absolute error between the two waveforms [1] [4].

We adhered to the process outlined above but in an incremental fashion to enhance the robustness of the model. Thus, we started with a simple model, defined its basic functionality and incrementally incorporated new properties of the model to produce the final model at the desired level of abstraction. We checked the model for solvability and the correct functionality and then proceeded to next stage.

3.2 The Approach

We exploited two approaches in this thesis for verification and validation of the simulation models. They are:

1. The Experimenteral Approach: The purpose of this approach was to verify the VHDL-AMS models by using them in testbench circuits and then executing these models on SEAMS. The results from SEAMS are then validated with results from SPICE.
2. The Quantitative Analysis Approach: The purpose of this approach was to calculate the accuracy of the VHDL-AMS models using quantitative analysis technique. This approach is adopted to determine if the VHDL-AMS models developed are accurate or not.

### 3.2.1 The Experimental approach

The approach adopted to develop the simulation models from the analytical models in was to first write an abstract model and then refine for high fidelity validating after each developmental stage. Thus, the model so developed was then implemented in three different testbench circuits for verification. The characteristics [10] for which the transistor models can be verified are enumerated below:

1. Switching Characterisites:-ability of the transistor to make transitions from one region of operation to another due to change in the input conditions.

2. Fidelity:-ability of the transistor to produce an output signal which is just like the input signal in all respects except amplitude.

3. Linearity:- ability of a transistor to amplify the fundamental frequency of an input signal while minimizing the generation of undesirable harmonics

For this research we only consider the first two characteristics to verify the transistor models developed. The analysis for linearity is left for future research.

Apart from verifying the aforementioned characteristics we also verify the operation of the models in common three regions: cut-off, active and saturation.

#### The Testbench

Three different testbench circuits were used to verify the models’ switching, fidelity and output characteristics. Their description is given below:

1. The Curve-Tracer circuit: Figure 3.1 shows a curve-tracer circuit that is used to obtain the output characteristics of a transistor model for a constant value of
input current and varying the output voltage using a Ramp source signal generator. Thus, using this circuit a family of curves for the output characteristics was obtained by changing the value of the input current.

2. The Inverter Circuit: The circuit shown in Figure 3.2 is used to obtain the switching characteristics of a transistor model. This was done by implementing the Model Under Test (MUT) as an inverter. A square wave signal generator is used to feed the input signal to the circuit. The Square wave generator is a mixed-signal system described in VHDL-AMS. This system consists of a digital process that generates a clock, another digital process which converts the discrete logic levels (here only '0' and '1') produced by clock into discrete levels of real values and an analog system which converts the values into an analog waveform.

3. The Amplifier circuit: The circuit shown in Figure 3.3 is used to verify the fidelity of the MUT\(^2\). A sine wave signal generator is used to feed the input signal to the circuit. The output expected for this circuit is an amplified sine wave with or without phase shift. Thus, any distortion of the output is undesirable if the MUT has to be declared to exhibit fidelity.

\(^2\)Model Under Test

The VHDL-AMS code for the signal generators in the aforementioned test-bench circuits can be found in the Appendix. It should be noted that all the test-
Figure 3.2: Testbench set-up for Inverter circuit.

Figure 3.3: Testbench set-up for Amplifier circuit.
3.2.2 The Quantitative Analysis approach

In this section we quantitatively analyze the results obtained by executing the VHDL-AMS transistor models in their implementation in the testbench circuits as described in the previous section, with the results obtained for the same models in SPICE to estimate accuracy. Since, the results are in the form of either waveforms or curves, hence, certain metric [4] is needed to make the comparisons and thus estimate the accuracy. Also, a methodology has to be adopted for calculation of the metric.

The Metric

As we are interested in knowing the accuracy of our models, we choose the error between two waveforms/curves(from SEAMS and SPICE) as the basis of comparison. There are are two types of errors that can be calculated:

**Absolute Error:** The difference between the measured waveform and a reference waveform.

**Relative Error:** The ratio of the absolute error to the reference waveform.

We use absolute error measured in dB, throughout this chapter.
\begin{verbatim}
IF (Vbe > -5.0 * Vt) use
  lbe == (I_{sat}*(exp(Vbe/Vt) - 1.0) + Vbe*Gm)/BF;
ELSIF (Vbe <= -5.0 * Vt) use
  lbe == (-1.0*I_{sat} + Vbe*Gm)/BF;
end use;
IF (Vbc > -5.0 * Vt) use
  lbc == (I_{sat}*(exp(Vbc/Vt) - 1.0) + Vbc*Gm)/BR;
ELSIF (Vbc <= -5.0 * Vt) use
  lbc == (-1.0*I_{sat} + Vbc*Gm)/BR;
end use;
\end{verbatim}

Figure 3.5: Equations for currents $I_{be}$ and $I_{bc}$ in VHDL-AMS

**Calculation of Metric**

To compare two waveforms with asynchronous sampling we first interpolate one waveform with the other, and take the sample-by-sample difference to obtain the absolute error curve. Since we prefer to state the error in dB, we then convert the maximum absolute error into error using

$$error\ dB = 20 \log \frac{e}{ref}$$

where $e$ is the maximum absolute error and $ref$ is the peak-peak value of the reference waveform

**3.3 The large-Signal BJT**

The VHDL-AMS representation of the large-signal model of the BJT is based on the Eber-Molls model which is the standard Level-2 implementation of the BJT in SPICE. The mathematical equations which describe the large-signal BJT model have been described in the chapter 3. The implementation of these equations in VHDL-AMS is given below. It should be noted that only the equations for base-emitter($I_{be}$) and base-collector($I_{bc}$) currents are given in Figure 3.5. Refer to the Appendix for the representation of the entire model in VHDL-AMS.
The results obtained by implementing the large-signal BJT model in the testbench circuits and executing them on SEAMS are given below. These plots are compared against the results obtained from SPICE. Also a quantitative analysis (as described in previous section) of the results is performed to check the accuracy of the model. The list of parameter values used to obtain the results is shown in Table 3.1

### 3.3.1 The I-V characteristic

Figure 3.6 shows a family of curves obtained from SEAMS and SPICE. The top curve is for base voltage of \( V_{bb} = 2.0V \) and the bottom curve is for a base bias of \( V_{bb} = 1.0V \). We can notice that traces for both VHDL-AMS and SPICE are almost identical. Figure 3.7 shows the distribution of the absolute error between the waveforms from SEAMS and SPICE, we see that the absolute error is \( 20.0 \mu \text{V} \) for most of the sample points. The spikes which are seen in the absolute error distribution plot is due to error introduced by interpolation.

It should also be noted that SEAMS is an analog and mixed signal simulator [11] whereas SPICE is an analog simulator [7] and hence there is a possibility that there can be considerable difference between the values evaluated at the same sample points by the simulators contributing to the inconstancy of the absolute error distribution.

The component values are listed in the Table 3.2

### 3.3.2 The Switching Characteristics assessment

The large-signal BJT, it is implemented in the testbench described in the earlier section. The component values have been listed in Table 3.3. The input signal generated by the square wave signal generator has an amplitude of 1V(p-p) and the frequency is 1KHz. The plots obtained from SEAMS and SPICE are shown in figure 3.8. It should be noted that there is a slight difference in the p-p amplitude of both SPICE and SEAMS, the reason for this is can be attributed to the way
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>1.8104e-15</td>
</tr>
<tr>
<td>BF</td>
<td>100.0</td>
</tr>
<tr>
<td>BR</td>
<td>1.0</td>
</tr>
<tr>
<td>CJJE</td>
<td>1.0pF</td>
</tr>
<tr>
<td>CJC</td>
<td>1.0pF</td>
</tr>
<tr>
<td>VJE</td>
<td>0.75</td>
</tr>
<tr>
<td>VJC</td>
<td>0.75</td>
</tr>
<tr>
<td>TF</td>
<td>408.8ps</td>
</tr>
<tr>
<td>TR</td>
<td>58.98ns</td>
</tr>
<tr>
<td>FC</td>
<td>0.5</td>
</tr>
<tr>
<td>MJE</td>
<td>0.377</td>
</tr>
<tr>
<td>MJC</td>
<td>0.3416</td>
</tr>
</tbody>
</table>

Table 3.1: Model parameters of npn BJT

<table>
<thead>
<tr>
<th>Component</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rb</td>
<td>1K</td>
</tr>
<tr>
<td>Rc</td>
<td>1K</td>
</tr>
<tr>
<td>Re</td>
<td>1K</td>
</tr>
<tr>
<td>Vcc</td>
<td>1-10V(linearly)</td>
</tr>
</tbody>
</table>

Table 3.2: Component Values of the Curve-Tracer set-up for npn BJT

Figure 3.6: I-V characteristics for large-signal npn BJT, lower curve is for $V_{bb}=1.0v$ and the upper curve is for $V_{bb}=2.0v$. 
Figure 3.7: Distribution of absolute error for the BJT I-V characteristics

<table>
<thead>
<tr>
<th>Component</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rb</td>
<td>100K</td>
</tr>
<tr>
<td>Rc</td>
<td>1.8K</td>
</tr>
<tr>
<td>Re</td>
<td>1K</td>
</tr>
<tr>
<td>Vbb</td>
<td>1.0V</td>
</tr>
<tr>
<td>Vcc</td>
<td>10V</td>
</tr>
</tbody>
</table>

Table 3.3: Component Values of the Inverter circuit set-up for npn BJT

discontinuities are handled in SEAMS and SPICE. For details regarding this please refer [9] Figure 3.9 shows the absolute error distribution for the results obtained from the Inverter circuit. From this plot we see that the absolute error is around 35.0mv at most of the sample points. The reasons for the inconstancy can be attributed to the reasons mentioned in the Subsection 3.3.1

3.3.3 Fidelity assessment

The plot shown in Figure 3.10 is the result obtained by implementing the large-signal BJT in the amplifier testbench. We notice that the SEAMS and the SPICE plots are found to be in close agreement. This is because the operating point for the BJT amplifier is maintained in the active region and from the plots of curve-tracer set-up of the large-signal BJT model we observe that SEAMS and SPICE
Figure 3.8: Output from Inverter testbench for large-signal npn BJT

Figure 3.9: Distribution of absolute error for the Inverter circuit set-up for BJT.
results in close agreement in the active region. From the Figure 3.10 we can see that the amplitude of output waveform is around 1.5v for a input sinusoidal signal of amplitude 0.4v(p-p) and frequency 1kHz. From, this we can conclude that our model exhibits fidelity of The component values are listed in the Table 3.4. The absolute error distribution obtained by performing quantitative analysis over the SEAMS and SPICE waveforms is shown in Figure 3.11. The figure indicates that the absolute error varies about a mean value of 15.0mv. Again the inconstancy can be attributed to the reasons cited in Subsection 3.3.1.
Figure 3.11: Distribution of absolute error for the Amplifier circuit set-up for BJT.

<table>
<thead>
<tr>
<th>Type of Testbench</th>
<th>Max. Absol error(e)</th>
<th>p-p value of reference waveform(ref)</th>
<th>Error dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Curve-tracer</td>
<td>2.0e-5</td>
<td>2.324e-4</td>
<td>-21.36</td>
</tr>
<tr>
<td>Inverter</td>
<td>0.001</td>
<td>0.8</td>
<td>-58.06</td>
</tr>
<tr>
<td>Amplifier</td>
<td>0.03</td>
<td>1.488</td>
<td>-33.9</td>
</tr>
</tbody>
</table>

Table 3.5: Summary of results for large-signal npn BJT

### 3.3.4 Summary

The final results for the large-signal BJT model are shown in Table 3.5. From the table we observe that the error in dB is maximum for the curve-tracer and is minimum for the amplifier, from this we conclude that the BJT simulation model shows high fidelity characteristics and very good switching characteristics and the Output characteristics are reasonable.

### 3.4 The JFET model

The VHDL-AMS representation of the JFET model is based on the Shichman and Hodges quadratic FET model [8] described in Chapter 2. The mathematical equations which describe the model are given in Chapter 2. The VHDL-AMS code for the drain current for the model is given in Figure 3.12. For the entire representation
if((vgs <= vto) and (vds >= 0.0)) use -cut-off
id == 0.0;
elseif((vds < (vgs-vto)) and (vgs > vto) and (vds >= 0.0)) use -linear
id == vds * beta * ((2.0 * (vgs_free - vto)) - vds_free) * (1.0 + lambda * vds);
elseif((vds >= vgs-vto) and (vgs > vto) and (vds >= 0.0)) use -saturation
id == beta * (pow((vgs_free - vto), 2.0)) * (1.0 + lambda * vds_free);
end use;

Figure 3.12: Equations for current \( I_d \) in normal mode, represented in VHDL-AMS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CGS</td>
<td>1.0pF</td>
</tr>
<tr>
<td>CGD</td>
<td>5.0pF</td>
</tr>
<tr>
<td>PB</td>
<td>1.0</td>
</tr>
<tr>
<td>FC</td>
<td>0.5</td>
</tr>
<tr>
<td>m</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Table 3.6: Parameter values for Large-Signal JFET

of the model in VHDL-AMS please refer the Appendix.

The parameter values for the JFET have been listed in the Table 3.6.

### 3.4.1 I-V characteristics

Figure 3.13 shows the plots from SEAMS and SPICE for the large-signal JFET model implemented in the curve-tracer set-up. The plots from SEAMS and SPICE show that they are in close agreement with each other. The upper curve is for gate-bias \( V_{gg} = -1.0 \) and the lower curve is for the gate-bias \( V_{gg} = -2.0 \). Figure 3.14 shows the absolute error distribution. From the figure for absolute error distribution for the JFET I-V characteristics we observe that the absolute error is around 0.5\( \mu \)V formost of the sample points and the reasons for the overshoots in the absolute error can be attributed to the reasons mentioned in the Subsection ?? The component values have been listed in the Table 3.7.
<table>
<thead>
<tr>
<th>Component</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rg</td>
<td>100K</td>
</tr>
<tr>
<td>Rs</td>
<td>1K</td>
</tr>
<tr>
<td>Rd</td>
<td>1K</td>
</tr>
<tr>
<td>Vcc</td>
<td>1-10V(linear)</td>
</tr>
</tbody>
</table>

Table 3.7: Component Values of the Curve-Tracer set-up for n-channel JFET

Figure 3.13: I-V characteristics for large-signal n-channel JFET, the upper curve is for $V_{gg} = -1.0v$ and the lower curve is for $V_{gg} = -2.0v$

Figure 3.14: Distribution of absolute error for the for JFET I-V characteristics
<table>
<thead>
<tr>
<th>Component</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rg</td>
<td>100K</td>
</tr>
<tr>
<td>Rd</td>
<td>1K</td>
</tr>
<tr>
<td>Rs</td>
<td>1K</td>
</tr>
<tr>
<td>Vgg</td>
<td>-2.0V</td>
</tr>
<tr>
<td>Vdd</td>
<td>5V</td>
</tr>
</tbody>
</table>

Table 3.8: Component Values of the Inverter circuit set-up for n-channel JFET

Figure 3.15: Output from Inverter set-up for large-signal n-channel JFET.

### 3.4.2 The Switching Characterisitcs assessment

Figure 3.15 shows the plots from SEAMS and SPICE for the large-signal JFET model implemented as an inverter. We notice that the output from SEAMS and SPICE are almost identical. The output waveform is the inverted waveform of an input square-wave signal of amplitude 1v(p-p) and frequency 1kHz. Hence we can conclude that the model shows switching characteristics. Figure 3.16 shows the absolute error distribution. From the Figure 3.16 we observe that the absolute error is in the order of micro volts for most of the sample points. The reasons for the inconstancies in the error distribution are same as cited in Subsection 3.3.1. The component values for this set-up have been listed in Table 3.8.
Figure 3.16: Distribution of absolute error for the Inverter circuit set-up for JFET.

<table>
<thead>
<tr>
<th>Component</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rg</td>
<td>100K</td>
</tr>
<tr>
<td>Rs</td>
<td>1.8K</td>
</tr>
<tr>
<td>Rd</td>
<td>4.7K</td>
</tr>
<tr>
<td>Vbb</td>
<td>-2.0V</td>
</tr>
<tr>
<td>Vdd</td>
<td>20V</td>
</tr>
</tbody>
</table>

Table 3.9: Component Values of the Amplifier circuit for n-channel JFET

3.4.3 Fidelity assessment

Figure 3.17 shows the plots from SEAMS and SPICE for the large-signal JFET model implemented as an Amplifier. We observe that the plots from both SPICE and SEAMS are in close agreement with each other and have almost the same peak to peak voltage. The input signal was a sinusoidal wave of 1v(p-p) amplitude and frequency 1kHz. Figure 3.18 shows the distribution of the absolute error. From the plot for absolute error we observe that the absolute error is about a mean value of 80mv. The component values and the parameter values have been listed in the Table 3.9.
Figure 3.17: Output from Amplifier set-up for large-signal n-channel JFET.

Figure 3.18: Distribution of absolute error for the Amplifier circuit set-up.
<table>
<thead>
<tr>
<th>Type of Testbench</th>
<th>Max. Absol error(e)</th>
<th>p-p value of reference waveform(ref)</th>
<th>Error dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Curve-tracer</td>
<td>4.4e-7</td>
<td>0.001</td>
<td>-67.13</td>
</tr>
<tr>
<td>Inverter</td>
<td>0.0007</td>
<td>0.55</td>
<td>-57.91</td>
</tr>
<tr>
<td>Amplifier</td>
<td>0.155</td>
<td>8.5</td>
<td>-34.78</td>
</tr>
</tbody>
</table>

Table 3.10: Summary of results for large-signal n-channel JFET

IF (Vds >= 0.0) and (Vgs < Vth) use -cut-off
Id_s == 0.0;
ELSIF (Vgs > Vth) and (Vds <= Vgs - Vth) use -linear region
Id_s == KP*(W/L)*(((Vgs-Vth)*Vds) - (POW(Vds,2.0)/2.0))*(1.0+lambda*Vds);
ELSIF (Vgs > Vth) and (Vds > Vgs - Vth) use -Saturation region
Id_s == (KP/2.0)*(W/L)*POW((Vgs - Vth,2.0)*(1.0+lambda*Vds);
end use;

Figure 3.19: Equations for current \( I_{ds} \) represented in VHDL-AMS

3.4.4 Summary

The summary of the final results after quantitative analysis are shown in Table 3.10. From the table we can conclude that the JFET model high output characteristics, reasonably good fidelity characteristics and good switching characteristics.

3.5 The MOSFET model

The basic large-signal model of the MOSFET represented in VHDL-AMS is similar to the one described in Chapter 3. The VHDL-AMS representation of the drain-source current is given in Figure 3.19 (please refer the Appendix for the VHDL-AMS code for the entire model). The MOSFET model represented in VHDL-AMS is the implementation of the Level3 large-signal model in SPICE2. There are lot of parameters associated with this model of MOSFET, hence we have reduced the parameters used to represent this model without affecting its functionality.

The parameter values for the large-signal MOSFET have been listed in the
Table 3.11: MOSFET large-signal model parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CGSO</td>
<td>0.2pF</td>
</tr>
<tr>
<td>CGBO</td>
<td>0.2pF</td>
</tr>
<tr>
<td>PHI</td>
<td>0.7</td>
</tr>
<tr>
<td>COX</td>
<td>1.65e-13</td>
</tr>
</tbody>
</table>

Table 3.12: Component Values of the Curve-Tracer set-up for NMOS

<table>
<thead>
<tr>
<th>Component</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rg</td>
<td>1K</td>
</tr>
<tr>
<td>Rs</td>
<td>1K</td>
</tr>
<tr>
<td>Rd</td>
<td>1K</td>
</tr>
<tr>
<td>Vcc</td>
<td>1-10V(linear)</td>
</tr>
</tbody>
</table>

Table 3.11. The results obtained by implementing this model in our testbench circuits are as follows:

3.6 The I-V characteristics

The plots obtained by implementing the MOSFET model in the curve-tracer arrangement are shown in the Figure 3.20 We observe that the results obtained from both SEAMS and SPICE are in close agreement with each other. It should be noted that the two curves shown in the plots are obtained by changing the value of the MOSFET parameter $V_{to}$. The upper curve is for $V_{to} = 1.0$ and the lower curve is for $V_{to} = 2.0$, indicating that output characteristics of a MOSFET are appreciably sensitive to its parameters. The absolute error distribution is shown in the Figure 3.21 indicating an absolute error of 0.02μv for most of the sample points. The component values are listed in the Table 3.12
Figure 3.20: I-V characteristics of a large-signal NMOS, the upper curve is for $V_{to} = 10$ and the lower curve is for $V_{to} = 2.0$.

Figure 3.21: Distribution of absolute error for MOSFET I-V characteristics.
<table>
<thead>
<tr>
<th>Component</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rg</td>
<td>1K</td>
</tr>
<tr>
<td>Rs</td>
<td>1.0e-3K</td>
</tr>
<tr>
<td>Rd</td>
<td>10K</td>
</tr>
<tr>
<td>Vgg</td>
<td>1.0V</td>
</tr>
<tr>
<td>Vdd</td>
<td>5V</td>
</tr>
</tbody>
</table>

Table 3.13: Component Values of the Amplifier circuit set-up for NMOS

3.6.1 The Switching Characteristics assessment

The plot in Figure 3.22 shows the switching characteristics of the MOSFET. We can see that the inverter output from both SEAMS and SPICE are in close agreement. A square wave of amplitude of 5V(p-p) and frequency 1KHz was given as the input to the inverter testbench circuit. Figure 3.9 shows the absolute error distribution. From the plot for error distribution we notice that the absolute error is around 10.0mv for most of the sample points and also that the number of sample points for which there is an overshoot is also more than the number of overshoots for the BJT and JFET model. The reason for this can be attributed to the large number of parameters in MOSFET which increases the complexity of the VHDL-AMS model leading to marked difference in the values evaluated by SEAMS and SPICE at some sample points. The reason for such marked difference is explained in the Subsection 3.3.1. The component values have been listed in the Table 3.13.

3.6.2 Fidelity assessment

The plot in the Figure 3.24 is obtained by implementing the MOSFET model as an amplifier to test the fidelity characteristics of the transistor model. From the plots we observe that the results from SPICE and SEAMs are in close agreement with each other. The input to this testbench was a sine wave whose amplitude was 1V(p-p) and the resulting output is a sine wave (same as the input) with amplitude of 2.1V(p-p), from which we can conclude that our model satisfies the fidelity criterion. Figure 3.25 shows the absolute error distribution. From the plot for absolute error
Figure 3.22: Output from Inverter set-up for large-signal NMOS.

Figure 3.23: Distribution of absolute error for the Inverter circuit set-up for MOSFET.
<table>
<thead>
<tr>
<th>Component</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rg</td>
<td>1K</td>
</tr>
<tr>
<td>Rs</td>
<td>1K</td>
</tr>
<tr>
<td>Rd</td>
<td>1K</td>
</tr>
<tr>
<td>Vgg</td>
<td>8V</td>
</tr>
<tr>
<td>Vdd</td>
<td>9V</td>
</tr>
</tbody>
</table>

Table 3.14: Component Values of the Amplifier circuit for NMOS

Figure 3.24: Output from Amplifier set-up for large-signal NMOS.

distribution we observe that the error is about a mean of 40mV. The component values for the testbench have been listed in the Table 3.14.

3.6.3 Summary

The final summary of the results after performing a quantitative analysis of the results obtained from the MOSFET transistor model using the testbenches is shown in Table 3.15. From the results we can conclude that the MOSFET exhibits a good output characteristics and reasonably good switching characteristics and fidelity.
Figure 3.25: Distribution of absolute error for the Amplifier circuit set-up for MOS-FET.

<table>
<thead>
<tr>
<th>Type of Testbench</th>
<th>Max. Absol error(e)</th>
<th>p-p value of reference waveform(ref)</th>
<th>Error dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Curve-tracer</td>
<td>4.0e-8</td>
<td>3.2e-4</td>
<td>-118.06</td>
</tr>
<tr>
<td>Inverter</td>
<td>0.01</td>
<td>4.9</td>
<td>-53.8</td>
</tr>
<tr>
<td>Amplifier</td>
<td>0.078</td>
<td>2.1</td>
<td>-28.6</td>
</tr>
</tbody>
</table>

Table 3.15: Summary of results for large-signal NMOS
3.7 The BSIM

The representation of the BSIM model in VHDL-AMS is based on the structure described in Chapter 3. A BSIM model is characterised by lots of parameters which are classified as physical and electrical. There are lots of representations of a BSIM model, they are: the static model, the charge model, the capacitance model, the resistance model. For this thesis we are studying the static model because the estimation of the values of the parameters (physical and electrical) is very complex and beyond the scope of this thesis. It should also be noted that the version of SPICE simulator that has been used in this thesis to validate the transistor model does not support the BSIM models. Hence only the results obtained from SEAMS have been presented. Also, since the results for the BSIM model could not be validated by our benchmark simulator we do not perform quantitative analysis of the results. We present only the results obtained by implementing the BSIM model in the curve-tracer arrangement to study output characteristics. We also do not

The plots shown in Figure 3.27 is obtained by implementing the BSIM model in the curve-tracer set-up. The component and the parameter values have been listed in the Table 3.17 and Table 3.16 respectively. Also the input characteristics of the BSIM have been shown in Figure 3.28 this plot was obtained by varying the input current and the output voltage was fixed-biased to 1.0V. The results obtained for the drain-source currents for both the input and output characteristics have been verified by doing the manual analysis of the testbench circuit.

The representation of the drain current equations in VHDL-AMS for a static BSIM is shown in Figure 3.26.

3.8 Timing Results for the Circuit Models

The execution speeds of all the circuit models studied in this thesis are presented in the Table 3.18 where:

- ASPs : Analog Solution Points
\[
\text{IF} \ (V_{gs} \leq V_{th}) \ \text{and} \ (V_{ds} \neq 0.0) \ \text{use -Cutoff region}
\]
\[I_{ds} = 0.0;\]

\[
\text{ELSIF} \ (V_{gs} > V_{th}) \ \text{and} \ (V_{ds} \leq (V_{gs-Vth})/\text{ALPHA}) \ \text{use -Linear region}
\]
\[I_{ds} = \frac{M_{U0}}{(1.0+U0*(V_{gs-Vth}))} \left( \frac{C_{ox} \cdot W}{L+U1*V_{ds}} \right) \left( (V_{gs-Vth})^*V_{ds} - (\text{ALPHA} \cdot \text{POW}(V_{ds}, 2.0)/2.0) \right);\]

\[
\text{ELSIF} \ (V_{gs} > V_{th}) \ \text{and} \ (V_{ds} > (V_{gs-Vth})/\text{ALPHA}) \ \text{use -Saturation region}
\]
\[I_{ds} = \frac{M_{U0}}{(1.0+U0*(V_{gs-Vth}))} \left( \frac{C_{ox} \cdot W}{(2.0*K*\text{ALPHA}*L)} \right) \left( \text{POW}((V_{gs-Vth}), 2.0) \right);\]

end use;

Figure 3.26: Equations for current $I_{ds}$ represented in VHDL-AMS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOX</td>
<td>13.0e-9</td>
</tr>
<tr>
<td>LD</td>
<td>0.24e-6</td>
</tr>
<tr>
<td>WD</td>
<td>0.4e-6</td>
</tr>
<tr>
<td>VDD</td>
<td>5.0V</td>
</tr>
<tr>
<td>MUZ</td>
<td>476</td>
</tr>
<tr>
<td>X2MZ</td>
<td>9.69</td>
</tr>
<tr>
<td>MUS</td>
<td>586</td>
</tr>
<tr>
<td>X2MS</td>
<td>-26.7</td>
</tr>
<tr>
<td>X3MS</td>
<td>1.56</td>
</tr>
<tr>
<td>U0</td>
<td>0.117</td>
</tr>
<tr>
<td>X2U0</td>
<td>1.09e-3</td>
</tr>
<tr>
<td>VFB</td>
<td>-0.747</td>
</tr>
<tr>
<td>PHI</td>
<td>0.75</td>
</tr>
<tr>
<td>K1</td>
<td>1.12</td>
</tr>
<tr>
<td>K2</td>
<td>0.156</td>
</tr>
<tr>
<td>ETA</td>
<td>0</td>
</tr>
<tr>
<td>X3E</td>
<td>-46.8e-6</td>
</tr>
<tr>
<td>X2U1</td>
<td>500e-6</td>
</tr>
<tr>
<td>X3U1</td>
<td>-18.1e-6</td>
</tr>
<tr>
<td>NO</td>
<td>1.57</td>
</tr>
<tr>
<td>NB</td>
<td>18.1e-3</td>
</tr>
<tr>
<td>ND</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 3.16: List of BSIM parameters
<table>
<thead>
<tr>
<th>Component</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rg</td>
<td>1K</td>
</tr>
<tr>
<td>Rs</td>
<td>1K</td>
</tr>
<tr>
<td>Rd</td>
<td>1K</td>
</tr>
<tr>
<td>Vdd</td>
<td>1-10V(linearly)</td>
</tr>
</tbody>
</table>

Table 3.17: Component Values of the Curve-Tracer set-up for static BSIM

Figure 3.27: Output from Curve-tracer set-up for static BSIM.

Figure 3.28: Output from Curve-tracer set-up for static BSIM (Input Characteristics).
<table>
<thead>
<tr>
<th>Type of Model</th>
<th>Type of circuit</th>
<th>SEAMS execution time</th>
<th>ASPs</th>
<th>NIT</th>
<th>TST</th>
<th>Total time</th>
</tr>
</thead>
<tbody>
<tr>
<td>BJT</td>
<td>Curve-Tracer</td>
<td>14.46s</td>
<td>1042</td>
<td>3012</td>
<td>10s</td>
<td>245.0s</td>
</tr>
<tr>
<td></td>
<td>Inverter</td>
<td>19.22s</td>
<td>1</td>
<td>2908</td>
<td>10ms</td>
<td>332.16s</td>
</tr>
<tr>
<td></td>
<td>Amplifier</td>
<td>12.54s</td>
<td>1032</td>
<td>2360</td>
<td>10ms</td>
<td>234.4s</td>
</tr>
<tr>
<td>JFET</td>
<td>Curve-Tracer</td>
<td>13.31s</td>
<td>1042</td>
<td>2701</td>
<td>10s</td>
<td>177.39s</td>
</tr>
<tr>
<td></td>
<td>Inverter</td>
<td>17.99s</td>
<td>1</td>
<td>3007</td>
<td>10ms</td>
<td>332.6s</td>
</tr>
<tr>
<td></td>
<td>Amplifier</td>
<td>11.85s</td>
<td>1032</td>
<td>2071</td>
<td>10ms</td>
<td>205.4s</td>
</tr>
<tr>
<td>MOSFET</td>
<td>Curve-Tracer</td>
<td>11.45s</td>
<td>1042</td>
<td>2085</td>
<td>10s</td>
<td>176.34s</td>
</tr>
<tr>
<td></td>
<td>Inverter</td>
<td>24.57s</td>
<td>1</td>
<td>3253</td>
<td>10ms</td>
<td>309.51s</td>
</tr>
<tr>
<td></td>
<td>Amplifier</td>
<td>12.98s</td>
<td>1042</td>
<td>2697</td>
<td>10ms</td>
<td>178.3s</td>
</tr>
</tbody>
</table>

Table 3.18: Timing results

- NIT: Number of Iterations
- TST: Transient simulation time
- Total time: time taken for the entire process of simulation i.e. scram + g++ + seams [11]

When we compared a particular case of speed of execution of only SEAMS and SPICE for the MOSFET curve-tracer circuit it was found that the execution speed of SEAMS was slower than SPICE by a single order whereas when the execution speed was compared for total time for SEAMS and SPICE then SEAMS was found to be slower by double order. For reasons for such slow execution speed of SEAMS compared to SPICE please refer [7]

### 3.9 The Experimental Environment

SEAMS version 1.1C was used in the experiments performed. It was run in a Unix environment on a Sun Enterprise machine having Sun Solaris 7 as the operating system. The C++ compiler used to run SEAMS was gcc version 2.7.2.1. spice3f4 was used to run the spice models on the same machine.
Chapter 4

Conclusions and Future Work

4.1 Summary

The aim of this thesis was to develop complex dynamic transistor models as described by Level-2 implementations in SPICE using the VHDL-AMS as the language for modeling. The models thus developed were then executed on SEAMS using testbenches which were designed to verify the switching characteristics and fidelity of the transistor models. The results obtained from SEAMS were then compared with equivalent models in SPICE using a quantitative analysis technique. The comparisons were done to determine the accuracy of the models using the Absolute Error between the waveforms as the metric. The final results were presented in form of Error dB. From, the final results it can thus be concluded that the models developed using VHDL-AMS are fairly accurate. It should be understood that the results collected for comparison were from two different types of simulators and also the quantitative analysis technique used to calculate the absolute error metric itself may introduce some errors.

The results are summarized in the Table 4.1:

From the table we can conclude that all the models that have been studied have acceptable levels of accuracy and hence we can conclude that the effective complex dynamic transistor models can be developed using VHDL-AMS.
<table>
<thead>
<tr>
<th>Type of model</th>
<th>Error in dB for curve-tracer</th>
<th>Error in dB for inverter</th>
<th>Error in dB for amplifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>BJT</td>
<td>-21.36</td>
<td>-58.06</td>
<td>-33.9</td>
</tr>
<tr>
<td>JFET</td>
<td>-67.13</td>
<td>-57.91</td>
<td>-34.78</td>
</tr>
<tr>
<td>MOSFET</td>
<td>-118.06</td>
<td>-53.8</td>
<td>-28.6</td>
</tr>
</tbody>
</table>

Table 4.1: Summary of Results

4.2 Future Work

Some directions and suggestions for further work in this field are given below:

- The transistor models were verified only for switching characteristics and fidelity. A further study to verify the linearity of the models can be done.

- All the models were executed on SEAMS and compared with equivalent SPICE models. Instead, a comparison with another VHDL-AMS simulator can be done [1].

- Instead of the absolute error metric other metrics can be used for comparisons [4].

- The BSIM model was only verified, it needs to be validated. Further, higher levels of BSIM models can be developed.
Bibliography


Appendix A

VHDL-AMS CODE

A.1 Signal Generators

A.1.1 Ramp Wave

PACKAGE electricalSystem IS
  NATURE electrical IS real ACROSS real THROUGH;
  FUNCTION SIN(X : real) RETURN real;
  FUNCTION EXP(X : real) RETURN real;
  FUNCTION POW(X,Y : real) RETURN real;
  FUNCTION SQRT(X : real) RETURN real;
END PACKAGE electricalSystem;
use work.electricalSystem.all;
ENTITY rampSource IS
  GENERIC( amp : real := 1.0);
  PORT( TERMINAL ta2, tb2 : electrical);--Interface ports.
END rampSource;

ARCHITECTURE rampbehavior OF rampSource IS
quantity Vramp across Iramp through ta2 to tb2;

BEGIN
  vsource: Vramp == (amp * real(time'pos(now)) * 1.0e-15);
END ARCHITECTURE rampbehavior;

A.1.2 Square Wave

PACKAGE electricalSystem IS
NATURE electrical IS real ACROSS real THROUGH;
FUNCTION SIN(X : real) RETURN real;
FUNCTION EXP(X : real) RETURN real;
FUNCTION POW(X,Y : real) RETURN real;
FUNCTION SQRT(X : real) RETURN real;
END PACKAGE electricalSystem;
USE work.electricalsystem.all;
ENTITY square_wave is
generic ( mag : real := 2.0;
       delay : time := 1 us
     );
port(terminal ta, tb:electrical);
end ENTITY ;
ARCHITECTURE behav of square_wave IS
-- terminal ta, tb:electrical;
quantity vsqr across isqr through ta to tb;
signal vsig : real :=0.0;
signal clk : bit :='0';
begin
v sqr := vsig;
brk4 : break on vsig;
clock : process
begin
  clk <= '1';
  wait for delay ;
  clk <= '0';
  wait for delay ;
end process; --- clock
generator : process
variable xv : real := 0.0;
begin
  if(clk = '1') then
    xv := mag*1.0;
  elsif(clk = '0') then
    xv := mag*0.0;
  end if;
  vsig <= xv;
  wait on clk;
end process; --- generator;
end ARCHITECTURE behav;

A.1.3 Sine Wave

PACKAGE electricalSystem IS
  NATURE electrical IS real ACROSS real THROUGH;
FUNCTION SIN(X : real) RETURN real;
FUNCTION EXP(X : real) RETURN real;
FUNCTION POW(X,Y : real) RETURN real;
FUNCTION SQRT(X : real) RETURN real;
END PACKAGE electricalSystem;

use work.electricalsystem.all;
entity sine_wave is
generic ( mag : real := 1.0;
 f : real := 1.0e3
);
end entity ;
architecture behav of sine_wave is
terminal t1,t2 :electrical;
quantity Vsine across Isine through time from electrical'reference;
BEGIN
Vsine == mag*sin(2.0*(22.0/7.0)+f*real(time*pos(now)))*1.0e-16;
END architecture behav;

A.2 Large Signal BJT

PACKAGE electricalSystem IS
 NATURE electrical IS real ACROSS real THROUGH;
FUNCTION SIN(X : real) RETURN real;
FUNCTION EXP(X : real) RETURN real;
FUNCTION POW(X,Y : real) RETURN real;
END PACKAGE electricalSystem;

use work.electricalsystem.all;

-----------Large Signal BJT static model-----------------------------

entity bjt_large is
generic (Isat : real := 1.0e-16; --saturation current
 Br : real := 1.0; --ideal max reverse current gain
 Bf : real := 100.0; --ideal max forward current gain
 Rb : real := 0.0; --ohmic resistance at Base
 Rc : real := 0.0; --ohmic resistance at Collector
 Re : real := 0.0; --ohmic resistance at Emitter
 Vje: real := 0.76; --Base emitter built-in potential
 Vjc: real := 0.76; --Base collector built-in potential
 Vjs: real := 0.76; --Substrate jn built-in potential
 Mjc: real := 0.33; --Base collector jn grading coeff
 Mje: real := 0.33; --Base emitter jn grading coeff
Mjs: real := 0.0; -- Substrate junction exponential factor
FC : real := 0.5; -- Coeff for forward bias depletion
TF : real := 0.0; -- Ideal forward transit time
TR : real := 0.0; -- Ideal reverse transit time
Ce0:real := 0.0; -- zero bias collector junction capacitance
Ce0:real := 0.0; -- zero bias emitter junction capacitance
Ce0:real := 0.0; -- zero bias capacitance
}

don terminal (E,B,C : electrical);

architecture bjt_behav of bjt_large is

terminal b1,c1,e1 : electrical;

CONSTANT Gm :real := 1.0e-12; -- conductance
CONSTANT Vt :real := 0.026; -- thermal voltage

-- Quantity Declarations start

total quantity Vb across Ib through b to b1;
total quantity Ve across Ie through e to e1;
total quantity Vc across Ic through c to c1;
total quantity Vbc across Ibc through b1 to c1;
total quantity Vbe across Ibe through b1 to e1;
total quantity Vct across Ict through c1 to e1;
total quantity Ibc_c through b1 to c1;
total quantity Ibe_c through b1 to e1;
total quantity Vcs across Ics_c through c1 to electrical reference;
total quantity f1,f2,f3,f4,f5,f6 : real := 0.0;

-- End of Quantity declarations

BEGIN

Vb := Ib * Rc;
Vc := Ic * Rb;
Ve := Ie * Re;

-------- Diode Equations

brk : break Vbe => 1.0, Vbc => -2.0;

IF (Vbe > -5.0 * Vt) use
Ibe := (Isat*(exp(Vbe/Vt) - 1.0) + Vbe*Gm)/BF;
ELSIF (Vbe <= -5.0 * Vt) use
Ibe := (-1.0*Isat + Vbe*Gm)/BF;
end use;

IF (Vbc > -5.0 * Vt) use
Ibc == (Isat*(exp(Vbc/Vt) - 1.0) + Vbe*Gm)/BR;
ELSIF (Vbc <= -5.0 * Vt) use
Ibc == (-1.0*Isat + Vbc*Gm)/BR;
end use;

Ict == Ibe*BF - Ibc*BR;

-----------Free Quantities

const1 : f1 == (Vje/(1.0-Mje)) * (pow((1.0-(1.0-FC)), (1.0-Mje)));
const2 : f2 == pow((1.0-FC),(1.0*Mje));
const3 : f3 == 1.0 - FC*(1.0*Mje);
const4 : f4 == (Vjc/(1.0-Mjc)) * (pow((1.0-(1.0-FC)), (1.0-Mjc)));
const5 : f5 == pow((1.0-FC),(1.0*Mjc));
const6 : f6 == 1.0 - FC*(1.0*Mjc);
-----------CapConditions:
--break2 : break Vbe => 0.0, Vbc => 0.0, Vcs => 0.0;

brk4 : break Vbe => 0.0;
IF (Vbe < FC * Vje) use
    Ibe_c == ((TF*Isat*(exp(Vbe/Vt))/Vt)+Ce0*pow((1.0 -
                (Vbe/Vje)), -1.0*Mje))*Vbe'dot;
ELSIF (Vbe >= FC * Vje) use
    Ibe_c == ((TF*Isat*(exp(Vbe/Vt))/Vt)*Ce0*(f2 +
                (Mje*Vbe/Vje)))*Vbe'dot;
end use;

brk3: break Vbc => 0.0;
IF (Vbc < FC * Vjco) use
    Ibc_c == ((TR*Isat*(exp(Vbc/Vt))/Vt)+Ce0*pow((1.0 - (Vbc/Vjco)),
                -1.0*Mjco))*Vbc'dot;
ELSIF (Vbc >= FC * Vjco) use
    Ibc_c == ((TR*Isat*(exp(Vbc/Vt))/Vt) +Ce0*f5)*(f6 +
                (Mjc*Vbc/Vjco)))*Vbc'dot;
end use;

brk2 : break Vcs => 0.0;
IF (Vcs < 0.0 ) use
    Ics_c == Cm0 * pow((1.0 - (Vcs/Vjco)), -1.0*Mjco)*Vcs'dot;
ELSIF (Vcs > 0.0) use
    Ics_c == Cm0 * (1.0 + (Mjco*Vcs/Vjco))*Vcs'dot;
end use;

end architecture bjtbehav;
A.2.1 Curve-Tracer for BJT

use std.textio.all;
use work.electricalsystem.all;

entity bjt_testbench is
end bjt_testbench;

architecture structure of bjt_testbench is

terminal t0, t1, t2, t3, t4 : electrical;

COMPONENT bjt_large is
  generic (Isat: real := 1.0e-16; --saturation current
  BR : real := 1.0; --ideal max reverse current gain
  BF : real := 100.0; --ideal max forward current gain
  Rb : real := 0.0; --ohmic resistance at Base
  Re : real := 0.0; --ohmic resistance at Collector
  Re : real := 0.0; --ohmic resistance at Emitter
  Vje: real := 0.75; --Base emitter built-in potential
  Vjc: real := 0.75; --Base collector built-in potential
  Vjs: real := 0.75; --Substrate jn built in potential
  Njc: real := 0.33; --Base collector jn grading coeff
  Nje: real := 0.33; --Base emitter jn grading coeff
  Njs: real := 0.0; -- Substrate jn exponential factor
  FC : real := 0.5; --Coeff for forward bias depletion
  TF : real := 0.0; --Ideal forward transit time
  TR : real := 0.0; --Ideal reverse transit time
  Cc0:real := 0.0; --zero bias collector jn capacitance
  Cc0:real := 0.0; --zero bias emitter jn capacitance
  Cc0:real := 0.0; --zero bias capacitance
);

  port (terminal E,B,C : electrical);
end COMPONENT;

for all : bjt_large use entity work.bjt_large(bjtbhv);

COMPONENT rampSource IS
  GENERIC( amp : real := 1.0e7);
  PORT( TERMINAL ta2,tb2 : electrical);--Interface ports.
end COMPONENT;

for all : rampsource use entity work.rampsource(rampbehavior);

quantity vbb across t0 to electrical’reference;
quantity vbo across ibb through t0 to t1;
quantity vrc across irc through t3 to t2;
quantity vre across ire through t4 to electrical’reference;
begin

vcc : rampsourcing generic map(1.0)  
   port map(t3, electrical'reference);  
inp_vol : vbb := 1.0; 
inpcur : vb0 := ibb + 1.0e3; 
em : vrc := irc + 1.0e3; 
vol_e: vre := ire + 1.0e3;

bjt : bjt_large  
generic map(Isat => 1.8104e-15,rbb=>1.0e-3,rc=>1.0e-3,re=>1.0e-3,  
   CcO=>1.0e-3,CsO=>1.0e-3,CsC=>1.0e-3,MJE=>0.377,MJO=>0.3416,TFe=+408.8ps,TR=+59.98)  
   port map(t4,t1,t2);

end architecture structure;

A.2.2 Inverter

use std.textio.all;  
use work.electricalsystem.all;

entity bjt_large_switch is  
end bjt_large_switch;

architecture structure of bjt_large_switch is

terinal t1, t2, t3, t4, t5 : electrical;

COMPONENT bjt_large is

generic (Isat: real := 1.0e-16; --saturation current  
   Bn: real := 1.0;  --ideal max reverse current gain  
   BF: real := 100.0;  --ideal max forward current gain  
   Rb: real := 0.0;  --ohmic resistance at Base  
   Re: real := 0.0 ;  --ohmic resistance at Emitter  
   Vje: real := 0.75;  --Base emitter built-in potential  
   Vjc: real := 0.75;  --Base collector built-in potential  
   Vjs: real := 0.75;  --Substrate jn built in potential  
   Mjc: real := 0.33;  --Base collector jn grading coeff  
   Mje: real := 0.33;  --Base emitter jn grading coeff  
   Mjs: real := 0.0; -- Substrate jn exponential factor  
   FC: real := 0.5; --Coeff for forward bias depletion  
   TF: real := 0.0; --Ideal forward transit time  
   TR: real := 0.0; --Ideal reverse transit time)
C0: real := 0.0; -- zero bias collector junction capacitance
C0: real := 0.0; -- zero bias emitter junction capacitance
C0: real := 0.0; -- zero bias capacitor

port (terminal E, B, C : electrical);
end COMPONENT;

for all : bjt_large use entity work.bjt_large(bjt_behav);

component square_wave is
generic (mag : real := 2.0;
delay : time := 1000us
);
port (terminal ta, tb: electrical);
end component;

for all : square_wave use entity work.square_wave(behav);

quantity Vbb across t1 to electrical reference;
quantity V1 across I1 through t1 to t2;
quantity V2 across I2 through t5 to t4;
quantity Vcc across t5 to electrical reference;

BEGIN

R1: V1 = I1 * 100.0e3;
R1: V2 = I2 * 1.8e3;
Base_bias : Vbb = 1.0;
Collector_bias : Vcc = 10.0;

-- COMPONENT INSTANTIATIONS

bjt : bjt_large generic map (Isat =>
1.8104e-13,Rbe=>1.0,Rc=>1.0,BE=>1.0,CE=>0.0,Co=>0.0,Ce=>0.0)
port map (electrical reference, t3, t4);

input: square_wave generic map (mag => 1.0, delay => 1000 us)
port map (t2, t3);

end architecture structure;

----------------------------------------End of Modeling-----------------------------

A.2.3 Amplifier

use std.textio.all;
use work.electricalsystem.all;

entity bjt_large_ampr is
end bjt_large_ampr;

architecture structure of bjt_large_ampr is

terminal t1, t2, t3, t4, t5 : electrical;

COMPONENT bjt_large is

   generic (Isat: real := 1.0e-16; --saturation current
   Bn: real := 1.0; --ideal max reverse current gain
   Bf: real := 100.0; --ideal max forward current gain
   Rb: real := 0.0; --ohmic resistance at Base
   Rc: real := 0.0; --ohmic resistance at Collector
   Re: real := 0.0; --ohmic resistance at Emitter
   Vje: real := 0.75; --Base emitter built-in potential
   Vjc: real := 0.75; --Base collector built-in potential
   Vjs: real := 0.75; --Substrate jn built in potential
   Mjc: real := 0.33; --Base collector jn grading coeff
   Mje: real := 0.33; --Base emitter jn grading coeff
   Mjs: real := 0.0; -- Substrate jn exponential factor
   FC: real := 0.5; --Coeff for forward bias depletion
   TF: real := 0.0; --Ideal forward transit time
   TR: real := 0.0; --Ideal reverse transit time
   Cc0: real := 0.0; --zero bias collector jn capacitance
   Ce0: real := 0.0; --zero bias emitter jn capacitance
   Cc0: real := 0.0 --zero bias capacitance
   );

   port (terminal E, B, C : electrical);
end COMPONENT;

for all : bjt_large use entity work.bjt_large(bjtbehav);

quantity Vs across t2 to t1;
quantity Vbb across t1 to electrical'reference;
quantity V1 across I1 through t2 to t3;
quantity V2 across I2 through t5 to t4;
quantity Vcc across t5 to electrical'reference;

BEGIN

imp_vol : Vs => 0.4*sin(2.0*(22.0/7.0)*1000.0*real(time('pos(now)))/1.0e-15);
Rs: V1 => I1 = 100.0e3;
Rc: V2 => I2 = 2.0e3;
Base_bias : Vbb => 1.0;
Collector_bias : Vcc => 10.0;

--COMPONENT INSTANTIATIONS

bjt : bjt_large generic map(isat =>
1.8104e-15,Rbe=>1.0,Rce=>1.0,Re=>1.0,BF=>100.0)
port map(electrical'reference,t3,t4);
end architecture structure;

A.3 Large-Signal JFET

PACKAGE electricalSystem IS
  NATURE electrical IS real ACROSS real THROUGH;
FUNCTION SIN(X : real) RETURN real;
FUNCTION EXP(X : real) RETURN real;
FUNCTION POW(X,Y : real) RETURN real;
END PACKAGE electricalSystem;

use work.electricalsystem.all;

entity njfet_large is
  generic(T : real := 300.0;
    vto : real := -2.0; -- Zero-bias threshold voltage
    beta : real := 1.0e-4; -- transconductance parameter
    lambda : real := 0.0; -- channel length modulation
    af : real := 1.0; -- flicker noise exponent
    kf : real := 0.0; -- flicker noise coefficient
    iss : real := 1.0e-14; -- gate junction saturation current
    pb : real := 1.0; -- gate junction potential
    fc : real := 0.5; -- forward-bias depletion capacitance coeff
    m : real := 0.5; -- grading coeff
    cgd0 : real := 0.0; -- zero-bias gate-drain junction cap
    cgs0 : real := 0.0; -- zero-bias gate-source junction cap
    rd : real := 1.0e-6; -- drain ohmic resistance
    rs : real := 1.0e-6); -- source ohmic resistance
  port (terminal g,s,d : electrical);
end entity njfet_large;

architecture behav of njfet_large is
  terminal d1, s1 : electrical;
  quantity vds across id through d1 to s1;
  quantity vrdd through id through d to d1;
  quantity vrs across irs through s1 to s;
  quantity vgs across igs through g to s1;
  quantity igsc through g to s1;
  quantity vgd across igd through g to d1;
quantity igd_c through g to d1;
constant gmin : real := 1.0e-12;
quantity kTq : real := 2.586e-2; -- (kT/q) thermal voltage at T=300K
quantity vds_free : real := 2.0;
quantity vgs_free : real := 0.0;
quantity vgd_free : real := 2.0;
quantity f1,f2,f3 : real;
begin

--------- Setting initial conditions
initreg : break vgs => 0.0, vds => 0.0, vgd => 0.0;
therm_volt : kTq := 2.586e-2 * (T/300.0);
dres : vrd := ird * rd;
oup_res : vds_free = vds;
inp_res : vgs_free = vgs;
vgdf : vgd_free = vgd;
sres : vrs = irs * rs;

-------- Current is in Amps.
-- Normal mode
-------- Cut off Region
regions : if((vgs <= vto) and (vds >= 0.0)) use
        gncn : id = 1.0e-9 * vds;
-------- Linear Region
elseif((vds < (vgs-vto)) and (vgs > vto) and (vds >= 0.0)) use
        gln : id = vds*beta*(2.0*(vgs_free-vto) - vds_free)*(1.0 + lambda*vd);
-------- Saturation Region
elseif((vds >= vgs-vto) and (vgs > vto) and (vds >= 0.0)) use
        gsn : id = beta*(pow((vgs_free-vto),2.0))*(1.0 + lambda*vd);

-- Inverted mode
-------- Cut off Region
elseif((vgs <= vto) and (vds <= 0.0)) use
        gcc : id = 1.0e-9 * vds;
-------- Linear Region
elseif((-1.0*vds) < (vgs-vto)) and (vgs > vto) and (vds <= 0.0)) use
        gnl : id = vds*beta*((2.0*(vgs_free-vto)) + vds_free)*(1.0 - lambda*vd);
-------- Saturation Region
elseif((-1.0*vds) >= vgs-vto) and (vgs > vto) and (vds <= 0.0)) use
        gns : id = -1.0*(beta)*(pow((vgs_free-vto),2.0))*(1.0 - lambda*vd)
end use;
-------- Gate diode equations
initsub : break vgd => 0.0, vgs => 0.0;
----- Gate to source
subcond1 : if (vgs > -5.0*ktq) use
    gsf : igs = ((iss*(exp(vgs/ktq) - 1.0)) + (gmin*vgs));
elseif(vgs <= -5.0*ktq) use
    gsr : igs = -1.0*iss + (gmin*vgs);
end use;
----- Gate to drain
subcond2 : if (vgd > -5.0*ktq) use
    gdf : igd = ((iss*(exp(vgd/ktq) - 1.0)) + (gmin*vgd));
elseif(vgd <= -5.0*ktq) use
    gdr : igd = -1.0*iss + (gmin*vgd);
end use;
const1 : f1 = (pb/(1.0-m)) * (pow((1.0-(1.0-fc)), (1.0-m)));
const2 : f2 = pow((1.0-fc),(1.0+m));
const3 : f3 = 1.0 - fc*(1.0+m);

--- CAPACITANCE EQUATIONS

break3 : break vgs=>0.0,vgd=>0.0;

IF (vgs < fc * pb) use
    igs_c = (cgs0*pow((1.0 -(vgs/pb)),-1.0*m))*vgs'dot;
ELSIF (vgs >= fc * pb) use
    igs_c = ((cgs0/f2)*(f3 *(m*vgs/pb)))*vgs'dot;
end use;

IF (vgd < fc * pb) use
    igd_c = (cgd0*pow((1.0 -(vgd/pb)),-1.0*m))*vgd'dot;
ELSIF (vgd >= fc * pb) use
    igd_c =((cgd0/f2)*(f3 *(m*vgd/pb)))*vgd'dot;
end use;

end architecture behav; --- of njfet_large;

A.3.1 Curve-Tracer

Use work.electricalsystem.all;
entity testbench is
end testbench;

architecture structure of testbench is

terminal t0,t1,t2, t3, t4 : electrical; --t0,t1
quantity V_in across I_{in} through t_0 to electrical reference;
quantity V_1 across I_1 through t_0 to t_1;
quantity V_2 across I_2 through t_4 to t_2;
quantity V_{ramp} across I_{ramp} through t_4 to electrical reference;
quantity V_3 across I_3 through t_3 to electrical reference;

cOMPONENT mjfet_large IS
  GENERIC (T : REAL := 300.0;
    VTO : REAL := -2.0; -- Zero-bias threshold voltage
    BETA : REAL := 1.0e-4; -- transconductance parameter
    LAMBDA : REAL := 0.0; -- channel length modulation
    AF : REAL := 1.0; -- flicker noise exponent
    KF : REAL := 0.0; -- flicker noise coefficient
    IS : REAL := 1.0e-14; -- gate junction saturation current
    PB : REAL := 1.0; -- gate junction potential
    FC : REAL := 0.5; -- forward-bias depletion capacitance coeff
    M : REAL := 0.5; -- grading coeff
    CGD0 : REAL := 0.0; -- zero-bias gate-drain junction cap
    CGS0 : REAL := 0.0; -- zero-bias gate-source junction cap
    RD : REAL := 0.0; -- drain ohmic resistance
    RS : REAL := 0.0); -- source ohmic resistance
  PORT (terminal g, s, d : electrical);
END COMPONENT;

FOR ALL mjfet_large USE ENTITY WORK.mjfet_large(behav);
BEGIN

  mjfet_large
    GENERIC MAP (VTO => -4.0, BETA => 1.0e-3, LAMBDA => 1.0e-15)
    PORT MAP (T1, T2);

  IN_VOLT : V_in = -1.0;
  RGL : V_1 = I1*100.0e3;
  RD1 : V_2 = I2*1.0e3;
  RS1 : V_3 = I3*1.0e3;
  SRC : V_{ramp} = (1.0 * REAL(time('pos(now)) * 1.0e-15);

END ARCHITECTURE STRUCTURE;

*************End of Modeling************

A.3.2 Inverter

USE WORK.electricalsystem.all;
entity ampr is
generic (f : real := 1.0e3);
end ampr;

architecture structure of ampr is
terminal t1,t2,t3,t4,t5,t6 : electrical;

component njfet_large is
  generic (T : real := 300.0;
    vto : real := -2.0; -- Zero-bias threshold voltage
    beta : real := 1.0e-4; -- transconductance parameter
    lambda : real := 0.0; -- channel length modulation
    af : real := 1.0; -- flicker noise exponent
    kf : real := 0.0; -- flicker noise coefficient
    iss : real := 1.0e-14; -- gate junction saturation current
    pb : real := 1.0; -- gate junction potential
    fc : real := 0.5; -- forward-bias depletion capacitance coeff
    m : real := 0.5; -- grading coeff
    cg0 : real := 0.0; -- zero-bias gate-drain junction cap
    cgs0 : real := 0.0; -- zero-bias gate-source junction cap
    rd : real := 1.0e-6; -- drain ohmic resistance
    rs : real := 1.0e-6); -- source ohmic resistance
  port (terminal g,s,d : electrical);
end component;

for all : njfet_large use entity work.njfet_large(behav);

component square_wave is
  generic(mag : real := 2.0;
    delay : time := 1000us
  );
  port (terminal ta, tb : electrical);
end component;

for all : square_wave use entity work.square_wave(behav);

quantity Vgg across t1 to electrical'reference;
--quantity Vc1 across Ic1 through t3 to electrical'reference; --cap added for --amp
quantity V1 across I1 through t2 to t3;
quantity V2 across I2 through t5 to t4;
quantity V3 across I3 through t6 to electrical'reference;
--quantity Vc2 across Ic2 through t4 to electrical'reference; --cap added for --amp
quantity Vdd across t5 to electrical'reference;
BEGIN

Vin: V1 => I1 * 100.0e3;
R1: V2 => I2 * 1.0e3;
R2: V3 => I3 * 1.0e3;
--C1: Ic1 => 10.0e-9 * Vc1'dot;
--C2: Ic2 => 10.0e-9 * Vc2'dot;
Base_bias : Vgg => -2.0;
Collector_bias : Vdd => 5.0;

--COMPONENT INSTANTIATIONS

input: square_wave generic map(mag=>0.4,delay=>1 ns)
port map(t2,t1);

jnl : njfet_large
  generic map(vto => -4.0, beta => 1.0e-3, lambda => 0.01,
               cgso=>1.0e-15,cgdo=>1.0e-15)
  port map(t3,t6,t4);

END architecture;

A.3.3 Amplifier

use work.electricalsystem.all;

entity ampr is
  generic (f : real :=1.0e3);
end ampr;

architecture structure of ampr is
  terminal t1,t2,t3,t4,t5,t6 : electrical;

quantity V1 across I1 through t2 to t3;
quantity V2 across I2 through t5 to electrical'reference;
quantity V3 across I3 through t4 to t6;
quantity Vdd across Idd through t6 to electrical'reference;
quantity Vin across Iin through t1 to t2;
quantity Vgg across Igg through t1 to electrical'reference;

component njfet_large is
  generic(T : real := 300.0;
           vto : real := -2.0; -- Zero-bais threshold voltage
           beta : real := 1.0e-4; -- transconductance parameter
           lambda : real := 0.0; -- channel length modulation
af : real := 1.0; -- flicker noise exponent
kf : real := 0.0; -- flicker noise coefficient
ims : real := 1.0e-14; -- gate junction saturation current
pb : real := 1.0; -- gate junction potential
fc : real := 0.5; -- forward-bias depletion capacitance coeff
m : real := 0.5; -- grading coeff
cgd0 : real := 0.0; -- zero-bias gate-drain junction cap
cgs0 : real := 0.0; -- zero-bias gate-source junction cap
rd : real := 1.0e-6; -- drain ohmic resistance
rs : real := 1.0e-6; -- source ohmic resistance

port (terminal g,s,d : electrical);
end component;
for all njfet_large use entity work.njfet_large(benv);

BEGIN

jnl : njfet_large
    generic map(vto => -4.0, beta => 1.0e-3, lambda => 0.0, RD=>100.0,
               RS => 100.0, pb => 1.0)
    port map(t3,t5,t4);

vol1: V1 = I1*100.0e3;
vol2: V2 = I2*1.0e3;
vol3: V3 = I3*4.7e3;
Vol_bias: Vdd = 20.0;
Vol_bias2: Vgg = -2.0;
input: Vin = 1.0*sin(2.0 * (22.0/7.0) * f * real(time*pos(nov)) +1.0e-15);

end structure;

A.4 Large Signal MOSFET

PACKAGE electricalSystem IS
    NATURE electrical IS real ACROSS real THROUGH;
    FUNCTION SIN(X : real) RETURN real;
    FUNCTION EXP(X : real) RETURN real;
    FUNCTION POW(X,Y : real) RETURN real;
    FUNCTION SQR(X : real) RETURN real;
END PACKAGE electricalSystem;

use work.electricalsystem.all;

entity nmos_large is
    generic (vto : real := -2.0; -- Zero-bias threshold voltage
KP : real := 2.0e-5; --Transconusy vductance parameter
Xj1 : real := 0.0; --Lateral Diffusion
lambda : real := 0.0; --channel length modulation
gamma : real := 0.0; --body effect parameter
ijs : real := 1.0e-14; -- gate junction saturation current
CGB0 : real := 0.0; --gate bulk overlap capacitance per
meter
CGD0 : real := 0.0; --gate drain overlap capacitance per
meter
CGS0 : real := 0.0; --gate source overlap capacitance per
meter
W : real := 100.0e-6; --width of the channel
L : real := 5.0e-6; --length of the channel
Cox : real := 1.66e-13; --the channel capacitance
PHI : real := 0.6; --surface inversion potential
Rd : real := 0.0;
Rs : real := 0.0;
Rg : real := 0.0;
Rb : real := 0.0
end ( terminal g,s,d,b : electrical);
architecture behavior of nmos_large is
terminal g1,s1,d1,b1 : electrical;
CONSTANT KP : real := 2.0e-5;
CONSTANT Xj1 : real := 0.0;
CONSTANT CGB0 : real := 0.0;
CONSTANT CGD0 : real := 0.0;
CONSTANT CGS0 : real := 0.0;
CONSTANT Rd : real := 1.0;
CONSTANT Rs : real := 1.0;
CONSTANT Rg : real := 0.0;
CONSTANT Rb : real := 1.0;
CONSTANT PHI : real := 0.6;
CONSTANT Cox : real := 0.0; --1.66e-13;
CONSTANT Vt : real := 0.026; --Vt = kT/q
CONSTANT GMIN : real := 1.0e-12;
CONSTANT n : real := 1.0;
Quantity Vao across Iso through s to s1;
quantity Vgo across Igo through g to g1;
quantity Vdo across Ido through d to d1;
quantity Vbo across Ibo through b to b1;
Quantity Vgs across Igs through g1 to s1;
-- Quantity Ig1 through g1 to s1;
quantity Vgd across Igd through g1 to d1;
-- quantity Ig1 through g1 to d1;
quantity Vbs across Ibs through s1 to b1;
quantity Vbd across Ibd through d1 to b1;
quantity Vds across Ids through d1 to s1;
quantity Vgb across Igbd through gi to b1;
quantity Igbi through gi to b1;
quantity Icb through s1 to b1;
quantity Vth: real := 0.0;
quantity Von: real := 0.0;

BEGIN

Vth := vto + gamma*(SQRT(PHI - Vbs) - SQRT(PHI));
Von := Vth + n*Vt;
Vso := Iso*Rs;
Vdo := Ido*rd;
Vgo := Igo*rg;
Vbo := Ibo*Rb;

--CAPACITANCE EQUATIONS

brk1: break Vgs => 0.0, Vgd => 0.0, Vds => 0.0;

IF (Vgs < Von - PHI) use --Accumulation region
	Igb = (Cox + CBQ*L) + Vgb'dot;
	Igs = (CQSO*W)*Vgs'dot;
	Igd = (CQDD*W)*Vgd'dot;
ELSIF (Vgs > Von - PHI) and (Vgs < Von) use -- Depletion region
	Igb = (Cox*(Von - Vgs)/PHI + CBQ*L) + Vgb'dot;
	Igs = (0.666+Cox*((Von - Vgs)/PHI + 1.0) + CQSO*W)*Vgs'dot;
	Igd = (CQDD*W)*Vgd'dot;
ELSIF (Vgs > Von) and (Vgs < Von + Vds) use -- Saturation region
	Igb = (CBQ*L) + Vgb'dot;
	Igs = ((0.666+Cox) + CQSO*W)*Vgs'dot;
	Igd = (CQDD*W)*Vgd'dot;
ELSIF (Vgs > (Von + Vds)) use --Linear region
	Igb = (CBQ*L) + Vgb'dot;
	Igs = (Cox*(1.0-PW((Vgs-Von)/(2.0*(Vgs-Von)-Vds),2.0)) + CQSO*W)*Vgs'dot;
	Igd = (Cox*(1.0-PW((Vgs-Von)/(2.0*(Vgs-Von)-Vds),2.0)) + CQDD*W)*Vgd'dot;
end use;

--DIODE EQUATIONS

brk2: break Vgs => 0.0, Vds => 0.0;
IF (Vds >= 0.0) and (Vgs < Vth) use
	Ids = 0.0;
ELSIF (Vgs > Vth) and (Vds <= Vgs - Vth) use --linear region
	Ids = kf*(W/L)*(((Vgs-Vth)*Vds) - (PW(Vds,2.0)/(2.0))*(1.0+lambda*Vds);
ELSIF (Vgs > Vth) and (Vds > Vgs - Vth) use --Saturation region

\[ I_{ds} = \frac{(Kp/2.0)\times(Vg/L)\times P0W((Vgs - Vth),2.0)}{(1.0+\lambda \times \text{mds} \times Vds)}; \]

end use;

brk3: break Vbs => 0.0, Vbd => 0.0;

-- Substrate to source

IF (Vbs < 0.0) use

\[ I_{bs} = ((\text{iss} \times \exp(Vbs/Vt) - 1.0)) + (\text{Gmin} \times Vbs)); \]

ELSIF (Vbs >= 0.0) use

\[ I_{bs} = ((\text{iss} \times Vbs/Vt) + (\text{Gmin} \times Vbs)); \]

end use;

-- Substrate to drain

IF (Vbd < 0.0) use

\[ I_{bd} = \text{iss} \times \exp(Vbd/Vt) - 1.0) + \text{Gmin} \times Vbd; \]

ELSIF (Vbd >= 0.0) use

\[ I_{bd} = \text{iss} \times Vbd/Vt + \text{Gmin} \times Vbd; \]

end use;

end architecture behavior;

A.4.1 Curve-Tracer

Use work.electricalsystem.all;
use std.textio.all;

entity testbench is
end entity testbench;

architecture structure of testbench is

terminal t0, t1, t2, t3, t4 : electrical;

quantity Vin across Iin through t0 to electrical'reference;
quantity V1 across I1 through t0 to t1;
quantity V4 across I4 through t1 to electrical'reference;
quantity V2 across I2 through t3 to electrical'reference;
quantity V3 across I3 through t4 to t2;
quantity Vamp across Iramp through t4 to electrical'reference;
signal vds_sig: real := 0.0;

component nmos_large is

generic (vto : real := -2.0; -- Zero-bias threshold voltage

-- KP : real := 2.0e-5; --Transconuse resistance parameter

end component nmos_large;
-- Xj1 : real := 0.0; --Lateral Diffusion
lambda :real := 0.0; --channel length modulation
gamma : real := 0.0; --body effect parameter
iws : real := 1.0e-14; -- gate junction saturation current
-- CGBO : real := 0.0; --gate bulk overlap capacitance perimeter
-- CGDO : real := 0.0; --gate drain overlap capacitance perimeter
-- CGSO : real := 0.0; --gate source overlap capacitance perimeter
W : real := 100.0e-6; --width of the channel
L : real := 100.0e-6 -- length of the channel
-- Cox : real := 1.65e-13; --the channel capacitace
-- PHI : real := 0.6; -- surface inversion potential
-- Rd : real := 0.0;
-- Rs : real := 0.0;
-- Rg : real := 0.0;
-- Rb : real := 0.0
);
port ( terminal g,s,d,b : electrical);
end component;

for all : nmos_large use entity work.nmos_large(behavior);

BEGIN

transl : nmos_large generic map(lambda=>0.03,vto=> -1.0)
port map(t1,t3,t2,electrical'reference);
vol_in : VIn = 5.0; --(1.0 * real(time'pos(now)) * 1.0e-15);
vol1 : VI = I1*1.0e3;
vol2 : V2 = I2*1.0e3;
vol3 : V3 = I3*1.0e3;
vol4 : V4 = I4*1.0e3;
src : Vramp = (1.0 * real(time'pos(now)) * 1.0e-15);

end architecture structure;

A.4.2 Inverter

use work.electricalsystem.all;
use std.textio.all;

entity testbench is
end entity testbench;

architecture structure of testbench is
terminal t1,t2,t3,t4,t5 : electrical;
component sqwav is
  generic ( mag : real :=2.0;  
            td : time :=1000.0 us );
  port (terminal ta,tb:electrical);
end component ;

for all : sqwav use entity work.sqwav(behav);

component nmos_large is
  generic ( vto : real := -2.0; -- Zero-bais threshold voltage  
            KP : real := 2.0e-5; --Transcon use w/distance parameter  
            -- Xj1 : real := 0.0; --Lateral Diffusion  
            lambda :real := 0.0; --channel length modulation  
            gamma : real := 0.0; --body effect parameter  
            iss : real := 1.0e-14; -- gate junction saturationcurrent  
            CDBO : real := 0.0; --gate bulk overlap capacitanceparameter  
            CDBO : real := 0.0; --gate drainoverlapcapacitanceparameter  
            COSO : real := 0.0; --GateSourceoverlapcapacitanceparameter  
            W : real := 30.0e-6; --width of the channel  
            L : real := 10.0e-6 -- length of the channel  
            -- Cox : real := 1.66e-13; --the channel capacitance  
            -- PHI : real := 0.6; -- surface inversion potential  
            -- Rd : real := 0.0;  
            -- Rs : real := 0.0;  
            -- Rg : real := 0.0;  
            -- Rb : real := 0.0  
            );
  port ( terminal g,s,d,b : electrical);
end component ;

for all : nmos_large use entity work.nmos_large(behavior);

quantity V1 across t1 through t3 to t5;
quantity Vdd across t5 to electrical'reference;
quantity V2 across t2 through t4 to electrical'reference;
quantity Vgg across t1 to electrical'reference;

BEGIN
  sqw_in : sqwav generic map (mag⇒6.0,td⇒1.0 ms)  
            port map (t2,t1);

  trans1 : nmos_large generic map (lambda⇒0.0,vto⇒  
                                1.0,KP⇒2.0e-3,iss⇒100.0e-6,L⇒100.0e-6)  
            port map (t2,t4,t3,t4);

  Rd : V1 == I1 * 10.0e3;
$V_{bias} : V_{dd} = 5.0;$
$R_s : V_{2} = 1.0 \times 10^{-3};$
$\text{gate\_bias} : V_{gg} = 1.0;\$

end architecture structure;

\textbf{A.4.3 Amplifier}

use work.electricalsystem.all;
use std.textio.all;

entity testbench is
end entity testbench;

architecture structure of testbench is
terminal t0,t1,t2,t3,t4,t5 : electrical;

\text{quantity} V_{gg} \text{ across } I_{gg} \text{ through } t0 \text{ to electrical'reference};
\text{quantity} V_{a_{in}} \text{ across } I_{a_{in}} \text{ through } t0 \text{ to } t1;
\text{quantity} V_{1} \text{ across } I_{1} \text{ through } t1 \text{ to } t2;
\text{quantity} V_{2} \text{ across } I_{2} \text{ through } t2 \text{ to } t3;
\text{quantity} V_{3} \text{ across } I_{3} \text{ through } t3 \text{ to } t4;
\text{quantity} V_{dd} \text{ across } I_{dd} \text{ through } t4 \text{ to electrical'reference};

component nmos_large is
  generic (vto : real := -2.0; -- Zero-bias threshold voltage
    KP : real := 2.0 \times 10^{-3}; -- Transconuse width parameter
        -- Xji : real := 0.0; -- Lateral Diffusion
    lambda : real := 0.0; -- channel length modulation
    gamma : real := 0.0; -- body effect parameter
    iss : real := 1.0 \times 10^{-14}; -- gate junction saturation current
    OBB : real := 0.0; -- gate bulk overlap capacitance parameter
    OBB : real := 0.0; -- gate drain overlap capacitance parameter
    OBB : real := 0.0; -- gate source overlap capacitance parameter
    \text{W} : real := 30.0 \times 10^{-6}; -- width of the channel
    \text{L} : real := 10.0 \times 10^{-6}; -- length of the channel
        -- Cox : real := 1.65e-13; -- the channel capacitance
        -- PHI : real := 0.6; -- surface inversion potential
        \text{Rd} : real := 0.0
        -- Rs : real := 0.0;
        -- Rg : real := 0.0;
        -- Rb : real := 0.0
    );
  port ( terminal g,s,d,b : electrical);
end component;

for all : nmos_large use entity work.nmos_large(behavior);
BEGIN

trans1 : mmos_large generic map(lambda=>0.0,vtc=>
2.0,KP=>1.0e-3,WM=>100.0e-6,LM=>100.0e-6,CCSR=>1.0e-12,DE60S=>1.0e-12,DE10S=>1.0)
port map(t2,t4,t3,t4);

vol_bias : Vgg == 8.0;
vol_in : Vsin = 1.0 * sin(2.0 * (22.0/7.0) * 1000.0 *
real(time'pos(nov)) * 1.0e-15);
vol1 : V1 == I1*100.0e3;
vol2 : V2 == I2*1.0e3;
vol3 : V3 == I3*1.0e3;
---- cap curr : Ics = 1.0e-9*v3'dot;
src1 : V4d == 9.0;
----src2 : Vss == -5.0;
-- out_vol : Vout = Iout*1.0e3;

end architecture structure;

A.5 BSIM

PACKAGE electricalSystem IS

NATURE electrical IS real ACROSS real THROUGH;
FUNCTION SIN(X : real) RETURN real;
FUNCTION EXP(X : real) RETURN real;
FUNCTION POW(X,Y : real) RETURN real;
FUNCTION SQRT(X : real) RETURN real;
END PACKAGE electricalSystem;

use work.electricalsystem.all;

entity bsim is

generic(Cox : real := 1.05e-13;
W : real := 20.0e-6;
L : real := 20.0e-6
--VFB : real := -0.747;
--PHI : real := 0.75;
--K1 : real := 1.12;
--K2 : real := 0.1566;
--MUZ : real := 476.0;
--UOZ : real := 0.117;
--U1Z : real := 0.0611;
--X2MZ : real := 9.69;
--X290 : real := 1.09e-3;
--X291 : real := 500.0e-6;
--MUS : real := 586.0;
--X216 : real := -26.7;
--X314 : real := -13.1e-6
);
port( terminal g,s,d,b : electrical);
end bsim1;

architecture behavior of bsim1 is

terminal s1,d1 : electrical;
quantity Vgs across g to s;
quantity Vbd across b to d;
quantity Gd across g to d;
quantity Vds across g to s;
quantity Vdsat : real := 0.0;
quantity Vth : real := 0.0;
--CONSTANT Vth : real := 1.0;
quantity N : real := 0.0;
quantity N0 : real := 0.0;
quantity U1 : real := 0.0;
quantity M0 : real := 0.0;
quantity alpha : real;
quantity k : real := 0.0;
quantity G : real := 0.0;
quantity Vc : real := 0.0;
--quantity sqrt_k : real := 0.0;

----quantity Vdsat : real := 0.0;
--quantity dummy : real := 0.0;
CONSTANT No : real := 1.6229;
CONSTANT Nb : real := 0.0186;
--CONSTANT N0 : real := 0.0;
CONSTANT GMIN : real := 1.0e-12;
CONSTANT iss : real := 1.0e-14;
CONSTANT Vt : real := 0.026; -- Vt = kT/q
CONSTANT Vdd : real := 5.0;
--CONSTANT C0x : real := 1.666e-13;
CONSTANT VFB : real := -0.55;
CONSTANT PHI : real := 0.75;
CONSTANT K1 : real := 1.1691;
CONSTANT K2 : real := 0.18596;
CONSTANT MUS : real := 476.0;
CONSTANT \( V_{OZ} \) : real := 0.3244;
CONSTANT \( V_{IZ} \) : real := 0.00631;
CONSTANT \( X_{MIZ} \) : real := 18.714;
CONSTANT \( X_{MISO} \) : real := -0.000662;
CONSTANT \( X_{SU1} \) : real := 0.0212;
CONSTANT \( M_{US} \) : real := 549.2;
CONSTANT \( X_{MUS} \) : real := 24.02;
CONSTANT \( X_{SU1} \) : real := -2.897e-5;

begin

\[ \text{----Threshold Voltage Equation} \]

\[ V_{th} = (V_{FB} + PHI + K_{1}*SQR(T(PHI-V_{bs})) - K_{2}*(PHI-V_{bs}) - N*V_{ds}); \]
\[ X = N_{w} + \text{Nd}V_{bs}; \]

\[ \text{----other drain,source,gate eqns} \]

\[ V_{s1} = I_{s1}*1.0e9; \]
\[ V_{d1} = I_{d1}*1.0e9; \]

\[ \text{----Drain Current Equations} \]

\[ \text{brk1 : break } V_{gs} \Rightarrow 0.0, \text{ } V_{ds} \Rightarrow 0.0; \]

\[ \text{IF } (V_{gs} \leq V_{th}) \text{ and } (V_{ds} \geq 0.0) \text{ use } \text{--Cutoff region} \]
\[ \text{Ids} = 0.0; \]
\[ \text{ELSIF } (V_{gs} > V_{th}) \text{ and } (V_{ds} \leq (V_{gs}-V_{th})/\text{ALPHA}) \text{ use } \text{--Linear region} \]
\[ \text{Ids} = \]
\[ \left( \frac{1}{\text{MUO}} \cdot \text{MUO} \cdot (V_{gs}-V_{th}) \right) \cdot \left( \text{Cox} \cdot \frac{W}{L+U_{s}+V_{ds}} \right) \cdot \left( (V_{gs}-V_{th}) \cdot V_{ds} - \text{ALPHA} \cdot \text{POW}(V_{ds},2.0)/2.0) \right); \]
\[ \text{ELSIF } (V_{gs} > V_{th}) \text{ and } (V_{ds} > (V_{gs}-V_{th})/\text{ALPHA}) \text{ use } \text{--Saturation region} \]
\[ \text{Ids} = \left( \frac{1}{\text{MUO}} \cdot \text{MUO} \cdot (V_{gs}-V_{th}) \right) \cdot \left( \text{Cox} \cdot \frac{W}{(2.0+K \cdot \text{ALPHA} \cdot L)} \cdot \text{POW}((V_{gs}-V_{th}),2.0) \right); \]

end Use;

\[ \text{----substrate to source eqn} \]

\[ \text{brk2 : break } V_{bs} \Rightarrow 0.0, \text{ } V_{bd} \Rightarrow 0.0; \]

\[ \text{IF } (V_{bs} < 0.0) \text{ use} \]
\[ I_{bs} = (I_{ss}*(\exp(V_{bs}/V_{t}) - 1.0)) + (\text{GMIN} \cdot V_{bs}); \]
\[ \text{ELSIF } (V_{bs} \geq 0.0) \text{ use} \]
\[ I_{bs} = (I_{ss} \cdot V_{bs}/V_{t}) + (\text{GMIN} \cdot V_{bs}); \]
end use;
-- Substrate to drain

IF (Vbd < 0.0) use
  Ibd = iss*(exp(Vbd/Vt) - 1.0) + GMIN*Vbd;
ELSIF (Vbd >= 0.0) use
  Ibd = iss*Vbd/Vt + GMIN*Vbd;
end use;

---- Free Quantity Equations

ALPHA = 1.0 + G*K1/(2.0*SQRT(PHI-Vbs));
G = 1.0 - 1.0/(1.744+0.8364*(PHI-Vbs));
K = (1.0+Vc+SQRT(1.0+2.0*Vc))/2.0;
Vc = U1*(Vgs-Vth)/(L+ALPHA);
U0 = U0Z + X200*Vbm;
U1 = U1Z + X2U1*Vbm + X3U1*(Vds-Vdd);

-- sqrt_k = SQRT(K);
-- Vdsat = (Vgs-Vth)/(ALPHA*SQRT(K));

-- break Vds > 0.0;

IF (Vds<=0.0) use
  MUO = MUS + X2M2*Vbm;
ELSIF (Vds > 0.0) and (Vds <=Vdd) use
  MUO = MUS + X2MS*Vbm;
end use;

end architecture behavior;

A.5.1 Curve-Tracer

use work.electricalsystem.all;

class testbench is
  end class testbench;

architecture structure of testbench is

terminal t0,t1,t2,t3,t4 : electrical;

quantity Vin across Iin through t0 to electrical-reference;
quantity V1 across I1 through t0 to t1;
quantity V2 across I2 through t3 to electrical-reference;
quantity V3 across I3 through t4 to t2;
quantity Vramp across Iramp through t4 to electrical-reference;
component bsim1 is

    generic(Cox : real := 1.66e-13;
        W : real := 20.0e-6;
        L : real := 20.0e-6
        --VF : real := -0.747;
        --PMI : real := 0.75;
        --K1 : real := 1.12;
        --K2 : real := 0.156;
        --M10 : real := 476.0;
        --U10 : real := 0.117;
        --U11 : real := 0.0611;
        --X13 : real := 9.69;
        --X15 : real := 1.09e-3;
        --X11 : real := 500.0e-6;
        --M13 : real := 686.0;
        --X15 : real := -26.7;
        --X15 : real := -18.1e-6
    );
    port ( terminal g,s,d,b : electrical);

end component;

for all : bsim1 use entity work.bsim1(behavior);

BEGIN

trans1 : bsim1 generic map(Cox=1.66e-13,W=200.0e-6,L=0.1e-6)
    port map(t1,t3,t2,t3);
vol_in : Vin = 20.0; --(1.0 * real(time('pos(now)) * 1.0e-15);
vol1 : V1 = I1+1.0e3;
vol2 : V2 = I2+1.0e3;
vol3 : V3 = I3+1.0e3;
src : Vramp = (1.0 * real(time('pos(now)) * 1.0e-15); --1.0;