THEORY, ALGORITHMS, AND USER'S GUIDE FOR BSIM AND SCALP

by

M.-C. Jeng, P. M. Lee, M. M. Kuo,
P. K. Ko, and C. Hu

Memorandum No. UCB/ERL M87/35

28 May 1987

THEORY, ALGORITHMS, AND USER'S GUIDE FOR BSIM AND SCALP

by

M.-C. Jeng, P. M. Lee, M. M. Kuo, P. K. Ko, and C. Hu

Memorandum No. UCB/ERL M87/35

28 May 1987

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

TITLE PAGE

Theory, Algorithms, and User's Guide for BSIM and SCALP

M.-C. Jeng, P.M. Lee, M.M. Kuo, P.K. Ko, and C. Hu

Department of Electrical Engineering and Computer Sciences
and Electronics Research Laboratory
University of California, Berkeley CA 94720

(Version 2.0)

May 28, 1987

This work was supported by Semiconductor Research Corporation.

PREFACE

The BSIM (Berkeley Short-channel IGFET MODEL) parameter extraction program has been updated substantially since the first time it was released. The BSIM parameter extraction program to date (version 2.0) can extract parameters for the strong-inversion drain current, the subthreshold drain current, and the substrate current. The parameters for the drain current can be read by a special version of SPICE2.G with BSIM models implemented or by SPICE3 (level 4) to simulate MOSFET characteristics. The parameters for the substrate current are processed by SCALP (the Substrate Current And Lifetime Processors) to simulate the degradation and lifetime of MOSFET's. Several ERL memoranda are available for various parts of the parameter extraction program, SPICE and SCALP. However, the connections between these memoranda are loose. The purpose of this memorandum is to provide the users with a unified information source about the BSIM parameter extraction program and SCALP.

This memorandum is divided into five chapters. Chapter 1 deals with the BSIM formulation. It includes drain current model for the strong-inversion region, subthreshold drain current model, capacitance and charge model, substrate current model, and degradation and lifetime model used in BSIM and SCALP, including whenever possible, concise accounts of their derivations. In chapter 2 is a description of the parameter extraction program. It includes the system setup requirements, the measurement procedure, theory and algorithms used in the extraction, and how the final process file is generated. A user's guide for the parameter extraction program is also included in this chapter. Chapter 3 is the BSIM program reference manual for SPICE. It describes how the BSIM model is implemented in SPICE. The user's guide for SPICE2.G with BSIM model implemented is also included. Chapter 4 is the reference manual for SCALP. Examples and results are given in Chapter 5.

To illustrate how these programs are related, a block diagram of the relationship between the BSIM parameter extraction program, SPICE, and SCALP is shown in Fig. 1.

To use BSIM in SPICE and SCALP, the users need parameters extracted with the parameter extraction program. When running the parameter extraction program, the user has the option to choose if the substrate current parameters are to be extracted. All the extracted parameters are stored in a file called process file. If the process file contains no substrate current parameters, it can be read directly by SPICE, otherwise, it has to be processed by the pre-processor of SCALP before being read by SPICE. The output of SPICE is then re-directed to the post-processor of SCALP to calculate the substrate current and lifetime. The detailed operations of these programs are given in later chapters.

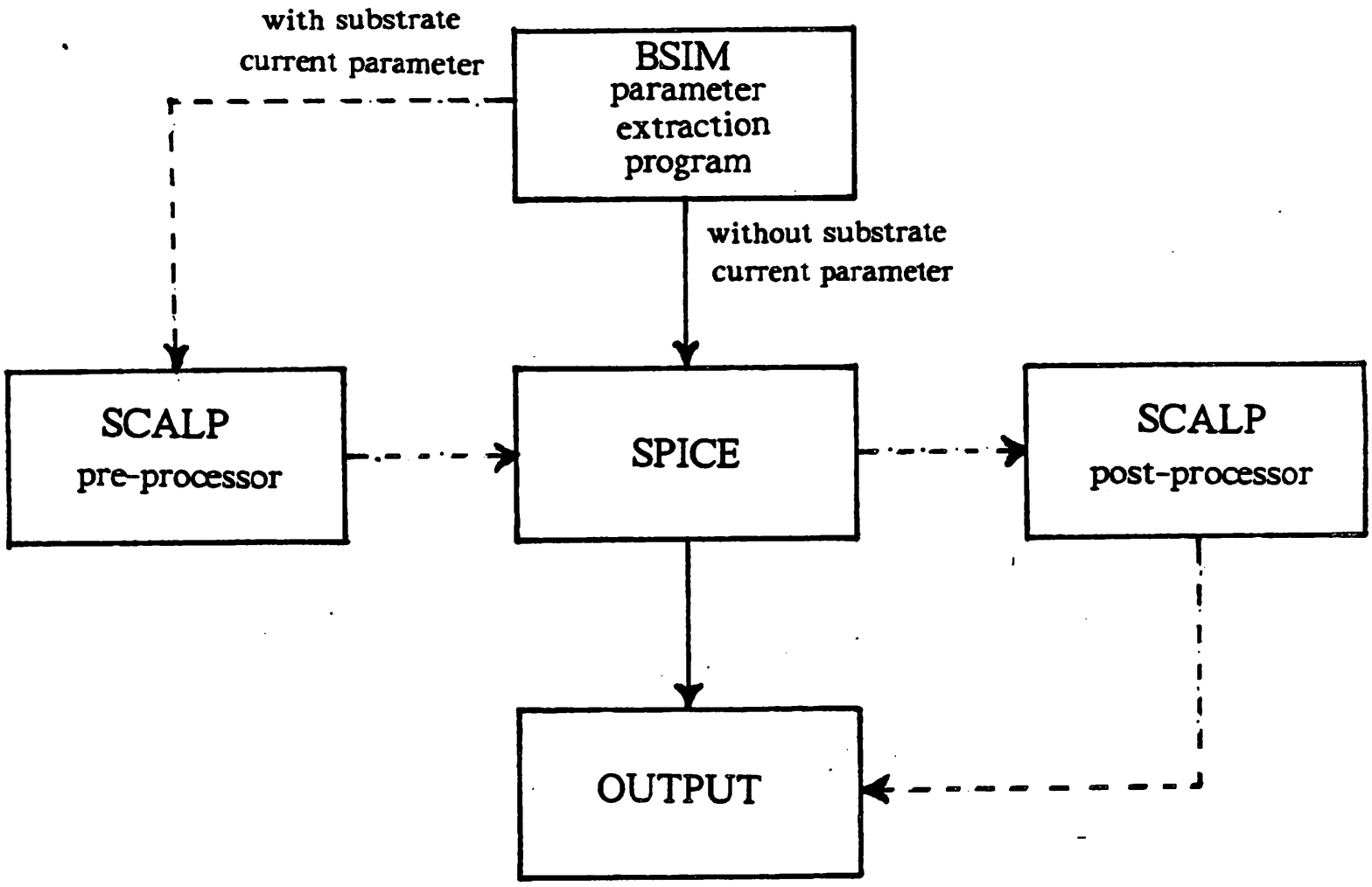


Fig.1 The relationship between BSIM parameter extraction program, SPICE, and SCALP.

With this memorandum, the users should be able to execute the parameter extraction program, SPICE, and SCALP without referring to previous memoranda. However, for users' reference, all previously published ERL memoranda related to SPICE2 with BSIM implementation, BSIM extraction program, and SCALP are categorized below. If the users are looking for the details of these programs, they can refer to the following memos.

Model and SPICE Implementation:

[1] B.J. Sheu, D.L Scharfetter, and H.C. Poon "Compact Short-Channel IGFET Model", Univ. of California, Berkeley, ERL memo. UCB/ERL M84/20.

[2] B.J. Sheu, D.L Scharfetter, and P.K. Ko, "SPICE2 Implementation of BSIM", Univ. of California, Berkeley, ERL memo. UCB/ERL M85/42.

[3] B.J. Sheu, "MOS Transistor Modeling and Characterization for Circuit Simulation", Univ. of California, Berkeley, ERL memo. UCB/ERL M85/85.

Parameter Extraction Program:

[1] B.J. Sheu, D.L Scharfetter, and H.C. Poon "Compact Short-Channel IGFET Model", Univ. of California, Berkeley, ERL memo. UCB/ERL M84/20.

[2] B.S. Messenger, "A Fully Automated MOS Device Characterization System for Process-Oriented Integrated Circuit Design", Univ. of California, Berkeley, ERL memo. UCB/ERL M84/18.

[3] J.R. Pierret, "A MOS Parameter Extraction Program for the BSIM Model", Univ. of California, Berkeley, ERL memo. UCB/ERL M84/99.

[4] A. H.-C. Fung, "A Subthreshold Conduction Model for BSIM", Univ. of California, Berkeley, ERL memo. UCB/ERL M85/22.

[5] M.-C. Jeng, B.J. Sheu, and P.K. Ko, "BSIM Parameter Extraction - Algorithms and User's Guide", Univ. of California, Berkeley, ERL memo. UCB/ERL M85/79.

[6] N. Yuen, "A Fully Automated BSIM Parameter Extraction System Using the HP 4062

Test System", Univ. of California, Berkeley, ERL memo. UCB/ERL M86/41.

[7] P.M. Lee, "BSIM - Substrate Current Modeling", Univ. of California, Berkeley, ERL memo. UCB/ERL M86/49.

SCALP:

[1] M.-C. Jeng, B.J. Sheu, and P.K. Ko, "BSIM Parameter Extraction - Algorithms and User's Guide", Univ. of California, Berkeley, ERL memo. UCB/ERL M85/79.

[2] P.M. Lee, "BSIM - Substrate Current Modeling", Univ. of California, Berkeley, ERL memo. UCB/ERL M86/49.

[3] P.M. Lee, M.M. Kuo, M. Maghsoodnia, P.K. Ko, and C. Hu, "BSIM - Substrate Current Modeling. Appendix C: Implementation of the BSIM Substrate Current and Degradation Models in SCALP", Univ. of California, Berkeley, ERL memo. UCB/ERL M87/8.

ACKNOWLEDGEMENT

The authors wish to thank Prof. D.O Pederson and Prof. D.A. Hodges for their continuous encouragement, Dr. D.L. Scharfetter and Dr. B.J. Sheu for the early stage development of the BSIM project, B.S. Messenger, J.R. Pierret, A.H.-C. Fung, and B. Liu's contributions to various parts of the parameter extraction program.

This memorandum is valid for BSIM parameter extraction program and SCALP released on June 1, 1987 and SPICE2 with BSIM implemented released on Oct. 15, 1985.

Table of Contents

CHAPTER 1: BSIM FORMULATION

With the ever decreasing device dimensions, a fully physical model is impossible to derive because of the 3-D nature of small-geometry effects. Even if it were feasible, the complicated equation forms involved in the physical model would prohibit its use for circuit simulation purposes. Furthermore, a fully device-physics-oriented modeling approach usually makes parameter extraction very difficult. The desire to achieve more accurate modeling and to alleviate the job of parameter extraction created the need of adding empirically-based parameters to the existing physical parameters. These semi-empirical models retain the basic functional forms of fully physical models, while replacing the sophisticated equations by empirical equations to represent small-geometry effects and process variations. The computational efficiency of the model can also be improved by a careful choice of these parameters. Since semi-empirical models have the advantages of simplicity and efficiency, all models in circuit simulation to date are, to a certain extent, semi-empirical models. In this chapter, Concised derivations of BSIM equations for the drain current, substrate current, degradation model and non-reciprocal charge models are presented [1,2].

1.1 Threshold Voltage

A commonly used equation for the threshold voltage can be expressed as:

$$V_{th} = V_{FB} + \phi_s + K_1\sqrt{\phi_s - V_{BS}} - K_2(\phi_s - V_{BS}) - \eta V_{DS} \qquad (1)$$

where V_{FB} is the flat-band voltage, ϕ_s is the surface-inversion potential, K_1 is the body effect coefficient which is equivalent to the parameter γ used in most textbooks, K_2 is the source/drain depletion-charge-sharing effect coefficient, η accounts for the drain-induced-barrier-lowering effect.

Parameters V_{FB}, ϕ_s, K_1 and K_2 are bias-independent, while η are bias-dependent and can further be empirically modeled by three parameters:

$$\eta = \eta_Z + \eta_B V_{BS} + \eta_D (V_{DS} - V_{DD}) \qquad (2)$$

where η_B and η_D account for the effects of substrate and drain biases on η, V_{DS} is the drain voltage, and V_{DD} is the user-specified maximum applied drain voltage during parameter extraction.

1.2 Drain Current in the Linear Region

The drain current in the strong-inversion region is given by

$$I_{DS} = W_{eff}\, v(y)\, q_n(y) \tag{3}$$

$$q_n(y) = C_{ox} \left| V_{GS} - V_{th}(y) - V(y) \right| \tag{4}$$

where W_{eff} is the effective channel width, $q_n(y)$ is the channel-charge density at position y from the source terminal, V(y) and $v(y)$ are the channel potential and carrier velocity at position y, respectively. Note that in (4), the threshold voltage is a function of position too due to the bulk charge effect.

A widely accepted expression for carrier velocity $v(y)$ for MOSFET's is

$$v(y) = \frac{\mu_0 E_y(y)}{\left| 1 + U_0 (V_{GS} - V_{th}) \right| \left(1 + \frac{E_y(y)}{E_c} \right)} \tag{5}$$

where E_c is the critical field given by $E_c = v_{sat}/\mu_0$, v_{sat} is the carrier saturation velocity, $E_y(y)$ is the horizontal channel electric field at position y given by dV(y)/dy, and U_0 is the mobility degradation coefficient due to vertical electric field. Substituting (4) and (5) into (3) and integrating both sides of (3) from source to drain, we have

$$I_{DS} = \frac{\beta}{1 + U_1 V_{DS}} \left| \left(V_{GS} - V_{th} + K_1 \sqrt{\phi_S - V_{BS}} - \frac{1}{2} V_{DS} \right) V_{DS} - K_1 F \right| \tag{6}$$

$$\beta = \frac{\beta_0}{1 + U_0 (V_{GS} - V_{th})} \tag{7}$$

where β_0 is the conductance coefficient given by

$$\beta_0 = \mu_0 C_{ox} \frac{W_{eff}}{L_{eff}} \tag{8}$$

$U_1 = 1/(E_c L_{eff})$, and F is equal to

$$F = \frac{2}{3} \left| (V_{DS} + \phi_S - V_{BS})^{\frac{2}{3}} - (\phi_S - V_{BS})^{\frac{2}{3}} \right| \tag{9}$$

The 3/2 power terms result from the integration of the square-root dependent bulk-doping effect evaluated at the two integration limits. F can be replaced by a numerical approximation to simplify the calculation.

$$F \approx \sqrt{\phi_S - V_{BS}}\, V_{DS} + \frac{0.25\, g\, V_{DS}^2}{\sqrt{\phi_S - V_{BS}}} \tag{10}$$

where

$$g = 1 - \frac{1}{1.744 + 0.8364\,(\phi_S - V_{BS})} \tag{11}$$

The comparison of (9) and (10), is given in Fig. 2.

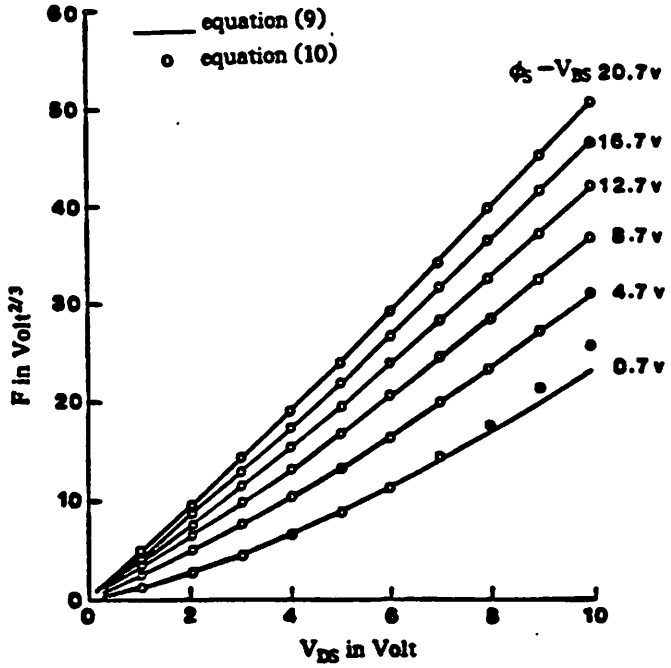


Fig.2 Comparison of F with its approximation.

Substituting (10) into (6), the drain current in the linear region can be obtained

$$I_{DS} = \frac{\beta}{(1 + U_1 V_{DS})} \left| (V_{GS} - V_{th})\, V_{DS} - \frac{a}{2} V_{DS}^2 \right| \tag{12}$$

where

$$a = 1 + \frac{g K_1}{2\sqrt{\phi_s - V_{BS}}} \tag{13}$$

The factor "a" takes care of the bulk-doping effect to the threshold voltage.

Parameters U_0, U_1, and β_0 are all bias-dependent. We have found empirically that U_0 and U_1 can modeled by linear equations of V_{BS} and/or V_{DS}, and β_0 can be modeled by a

linear function of V_{BS} and quadratic function of V_{DS}.

$$U_0 = U_{0Z} + U_{0B} V_{BS} \tag{14}$$

$$U_1 = U_{1Z} + U_{1B} V_{BS} + U_{1D} (V_{DS} - V_{DD}) \tag{15}$$

$$\beta_0 = \beta_1 (\frac{V_{DS}}{V_{DD}} - 1)^2 + \beta_2 (2 - \frac{V_{DS}}{V_{DD}}) \frac{V_{DS}}{V_{DD}} + \beta_{SD} V_{DS} (\frac{V_{DS}}{V_{DD}} - 1) \tag{16}$$

where β_1 and β_2 are the conductance coefficients at $V_{DS} = 0$ and $V_{DS} = V_{DD}$, respectively. β_{SD} is the sensitivity of β_0 to V_{DS} at $V_{DS} = V_{DD}$. Both β_1 and β_2 can further be modeled by

$$\beta_1 = \beta_Z + \beta_{ZB} V_{BS} \tag{17}$$

$$\beta_2 = \beta_S + \beta_{SB} V_{BS} \tag{18}$$

The relationship between β_0, β_1, β_2, and β_{SD} is graphically presented in Fig. 3.

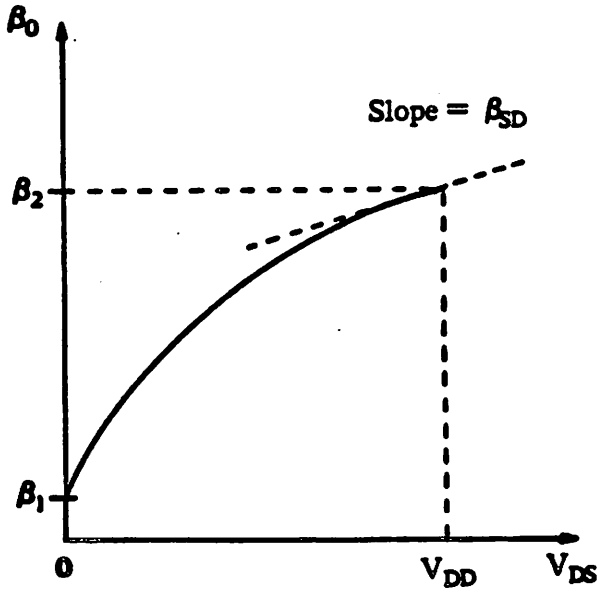


Fig.3 The relationship between β_1, bets$_2$, and β_{SD}.

1.3 Drain Current in the Saturation Region

To facilitate comparison with the usual expression, we define the drain current in the saturation region as

$$I_{DSAT} = \frac{\beta (V_{GS} - V_{th})^2}{2 a K} \tag{19}$$

where the expression of K is to be determined.

Let the drain saturation current and voltage be I_{DSAT} and V_{DSAT},respectively. Then from (12), we have

$$I_{DSAT} = \frac{\beta}{(1 + U_1 V_{DSAT})} \left| (V_{GS} - V_{th}) V_{DSAT} - \frac{a}{2} V_{DSAT}^2 \right| \quad (20)$$

The drain saturation voltage, V_{DSAT}, can be found by setting $\partial I_{DS}/\partial V_{DS} = 0$ in (12), which yields the following equation

$$V_{GS} - V_{th} - a V_{DSAT} - \frac{a}{2} U_1 V_{DSAT}^2 = 0 \quad (21)$$

Solving V_{DSAT} from (21) and substituting it into (20), and equating (20) with (19), we can find the expression for K and V_{DSAT}

$$K = \frac{1 + V_c + \sqrt{1 + 2 V_c}}{2} \quad (22)$$

and

$$V_{DSAT} = \frac{V_{GS} - V_{th}}{a \sqrt{K}} \quad (23)$$

where

$$V_c = \frac{U_1 (V_{GS} - V_{th})}{a} \quad (24)$$

1.4 Drain Current in the Subthreshold Region

In subthreshold region, diffusion current dominates. Therefore, the drain current diminishes exponentially with decreasing gate voltage. In BSIM, the equation for the drain current in the subthreshold region is

$$I_{subth} = \frac{I_{EXP} I_{limit}}{I_{EXP} + I_{limit}} \quad (25)$$

$$I_{EXP} = \beta_0 V_{tm}^2 e^{1.8} e^{\frac{V_{GS} - V_{th}}{n V_{tm}}} \left| 1 - e^{-\frac{V_{DS}}{V_{tm}}} \right| \quad (26)$$

$$I_{limit} = \frac{9 \beta_0 V_{tm}^2}{2} \quad (27)$$

where I_{EXP} is the usual expression for the diffusion current in weak-inversion region except that an empirically determined factor $e^{1.8}$ is used to achieve the best fit in the subthreshold characteristics. I_{limit} is used to set an upper limit on the subthreshold drain current. V_{tm} is the thermal voltage given by $V_{tm} = kT/q$, and n is the subthreshold swing parameter. Parameter n is also bias dependent and can be empirically modeled by

$$n = n_0 + n_B V_{BS} + n_D V_{DS} \tag{28}$$

The total drain current is the sum of the drain currents in the strong-inversion region and in the subthreshold region. There are twenty parameters to model the drain current in BSIM. Seven parameters, V_{FB}, ϕ_S, K_1, K_2, η_Z, η_B, and η_D, are used to model the threshold voltage. Ten parameters, U_{0Z}, U_{0B}, U_{1Z}, U_{1B}, U_{1D}, β_Z, β_{ZB}, β_S, β_{SB}, and β_{SD}, are used to model the drain current in the strong-inversion region. Three parameters, n_0, n_B, and n_D, are used to model the subthreshold drain current. The meaning of each parameter is explained in the following list.

1. V_{FB}: the flat-band voltage.
2. ϕ_s: the surface inversion potential.
3. K_1: the body effect coefficient.
4. K_2: the source/drain charge sharing effect coefficient.
5. η_Z: the value of η at zero substrate bias and $V_{DS} = V_{DD}$.
6. β_Z: the value of β at zero substrate and drain-source biases.
7. U_{0Z}: the value of U_0 at zero substrate bias.
8. U_{1Z}: the value of U_1 at zero substrate bias and $V_{DS} = V_{DD}$.
9. β_{ZB}: the sensitivity of β to substrate bias at $V_{DS} = 0$.
10. η_B: the sensitivity of η to substrate bias.
11. η_D: the sensitivity of η to drain voltage at $V_{DS} = V_{DD}$.
12. U_{0B}: the sensitivity of β to substrate bias at $V_{DS} = 0$.
13. U_{1B}: the sensitivity of U_1 to substrate bias.
14. β_S: the value of β at zero substrate bias and at $V_{DS} = V_{DD}$.
15. β_{SB}: the sensitivity of β to substrate bias at $V_{DS} = V_{DD}$.
16. β_{SD}: the sensitivity of β to drain voltage at $V_{DS} = V_{DD}$.
17. U_{1D}: the sensitivity of U_1 to drain voltage at $V_{DS} = V_{DD}$.
18. n_0: the value of n at zero substrate and drain biases.
19. n_B: the sensitivity of n to substrate bias.
20. n_D: the sensitivity of n to drain voltage.

1.5 Substrate Current Model

As maximum electric field in the drain end increases with shorter channel lengths, more and more hot electrons acquire sufficient kinetic energy not only to damage the silicon-silicon dioxide interface but also to overcome or tunnel through the silicon dioxide barrier and become trapped in the oxide. Both result in threshold voltage shifts that could potentially be fatal in circuit design.

Greater lateral electric field also causes an increase in the number of electrons which possess the required energy to create electron-hole pairs by impact ionization. The excess holes that are produced are swept into the substrate by the vertical field as substrate current. Because both hot-electron-induced degradation and substrate current generation are induced by the same force (namely the channel electric field), it can be shown that a direct correlation exists between the amount of degradation occurring and the substrate current measured [3].

Because of this fact, knowledge of the substrate current of the devices used in a circuit can provide valuable information on how reliable the circuit will be. It is therefore important to develop an accurate model of the substrate current and to incorporate it into circuit simulation.

The substrate current model used in BSIM is based on work done by El-Mansy [4,5] and Ko [6]. El Mansy derived an exponential relationship of the channel electric field in the saturated region of the channel using quasi-two-dimensional concepts, and using this electric field model he derived a simplified model for the substrate current. This model was subsequently improved by Ko to include the effect of junction depth and channel doping. The following is an outline of the derivations done to obtain the substrate current expression.

Because the main contribution to the substrate current is from electron impact ionization, the derivation is begun by integrating the electron impact ionization coefficient $\alpha_n = A_i e^{-\frac{B_i}{E}}$ in the velocity-saturated region of the channel.

$$I_{BS} = I_{DS} A_i \int_{y=0}^{\Delta L} e^{-\frac{B_i}{E_s(y)}} dy \tag{29}$$

$y = 0$ is at the edge of the saturated region in the channel, $y = \Delta L$ is at the drain, and $E_s(y)$ is the electric field in the channel direction. To find $E_s(y)$, a pseudo-two-dimensional analysis is performed of a Gaussian box enclosing the saturated region. This results in an exponential relationship of $E_s(y)$ versus distance.

$$E_s(y) = E_{crit} \cosh(\frac{y}{l_c}) \tag{30}$$

E_{crit} is the critical field for velocity saturation, and l_c can be termed as the effective width of the "pinch-off" region of the channel. $E_s(y)$ can also be expressed in terms of voltage within the saturated region.

$$E_s(y) = [\frac{(V_s(y) - V_{DSAT})^2}{l_c^2} + E_{crit}^2]^{\frac{1}{2}} \tag{31}$$

After an appropriate change in variables, equation (29) can then be rewritten as

$$I_{BS} = I_{DS} l_c A_i \int_{E_s=E_{crit}}^{E_d} \frac{e^{-\frac{B_i}{E_s}}}{(E_s^2 - E_{crit}^2)^{1/2}} dE_s \tag{32}$$

$$\approx \frac{I_{DS} A_i l_c E_d}{B_i} e^{-\frac{B_i}{E_d}} \tag{33}$$

where E_d is the electric field at the drain end.

$$E_d = [\frac{(V_{DS} - V_{DSAT})^2}{l_c^2} + E_{crit}^2]^{\frac{1}{2}} \tag{34}$$

In saturation, $E_d >> E_{crit}$, so that equation (34) can be approximated by

$$E_d \approx \frac{V_{DS} - V_{DSAT}}{l_c} \tag{35}$$

Inserting equation (35) into equation (33), we obtain the final expression for I_{bs}.

$$I_{BS} = \frac{A_i}{B_i} I_{DS} (V_{DS} - V_{DSAT}) e^{-\frac{B_i l_c}{V_{DS} - V_{DSAT}}} \tag{36}$$

Leaving A_i and B_i fixed at commonly used values of 2×10^6 cm^{-1} and 1.7×10^6 V/cm, respectively, for n-channel devices (for p-channel devices, A_i and B_i are 1×10^7 cm^{-1} and

3.7×10^6 V/cm, respectively), two parameters remain to be determined, V_{DSAT} and l_c.

For short-channel devices, V_{DSAT} departs from the well-known relationship $V_{DSAT} = V_{GS} - V_{th}$ for long-channel devices because electrons in the channel region become velocity-saturated before V_{DS} reaches $V_{GS} - V_{th}$. The model used in BSIM to account for this behavior was derived by Sodini and Ko [7].

$$V_{DSAT} = \frac{E_{crit}L(V_{GS} - V_{th})}{E_{crit}L + (V_{GS} - V_{th})} \tag{37}$$

where L is the channel length. E_{crit} is then extracted as a parameter from measured V_{DSAT} values.

Several approximate analytical forms for l_c have been published, including those of El-Mansy et al [4,5] and Ko [6] with a $t_{ox}^{1/2}$ dependency, but none can comprehensively account for dependencies on bias and size. In this work, a semi-empirical approach is established in which these dependencies are extracted directly from measured data.

Again, E_{crit} and l_c are empirically fitted to some equations of bias voltages.

$$E_{crit} = E_{crit0} + E_{critG} V_{GS} + E_{critB} V_{BS} \tag{38}$$

$$l_c = \sqrt{t_{ox}} \left[l_1 + l_2 \left(\frac{1}{V_{GS} + 2} \right) \right] \tag{39}$$

where

$$l_1 = l_{c0} + l_{c1} \left(\frac{1}{V_{BS} - 4} \right) + \left[l_{c2} + l_{c3} \left(\frac{1}{V_{BS} - 4} \right) \right] V_{DS} \tag{40}$$

$$l_2 = l_{c4} + l_{c5} \left(\frac{1}{V_{BS} - 4} \right) + \left[l_{c6} + l_{c7} \left(\frac{1}{V_{BS} - 4} \right) \right] V_{DS} \tag{41}$$

t_{ox} is the gate oxide thickness. The factors "2" and "4" in (39), (40), and (41) were determined so that the expressions would be valid for V_{GS} and $V_{BS} = 0$ while giving a good fit to data measured from a wide variety of MOSFET's.

1.6 Degradation and Lifetime Model

Device degradation is typically measured by the amount of threshold voltage shift (ΔV_{th}) that occurs. Other parameters, e.g. $\Delta I_{DS}/I_{DS0}$, may be used to characterize device degradation. In that case, ΔV_{th} in the equations below should be replaced by, for example,

$\Delta I_{DS}/I_{DS0}$.In DC static stressing, the device lifetime can be modeled by two basic equations dependent on I_{DS}, I_{BS}, and other parameters [3]:

$$\Delta V_{th} = A\, t^n \tag{42}$$

$$\Delta V_{thf} = A\, \tau^n \tag{43}$$

$$\tau = W_{eff}\, B\, I_{BS}^{-m}\, I_{DS}^{m-1} \tag{44}$$

$$m = \frac{\phi_{it}}{\phi_i} \tag{45}$$

$$B = H\, \Delta V_{thf}^{1/n} \tag{46}$$

where the values of n , m and H are determined from stress experiments and H is dependent on device processing technology. ΔV_{thf} is the amount of shift in threshold voltage defined at device failure, t is the stressing time, τ is the device lifetime, and $q\phi_i$ and $q\phi_{it}$ are the critical energies required for impact ionization and the creation of interface traps, respectively. These equations can then be combined with numerical calculations to find device lifetime. A typical log - log plot of τ versus I_{BS} is shown in Fig. 4.

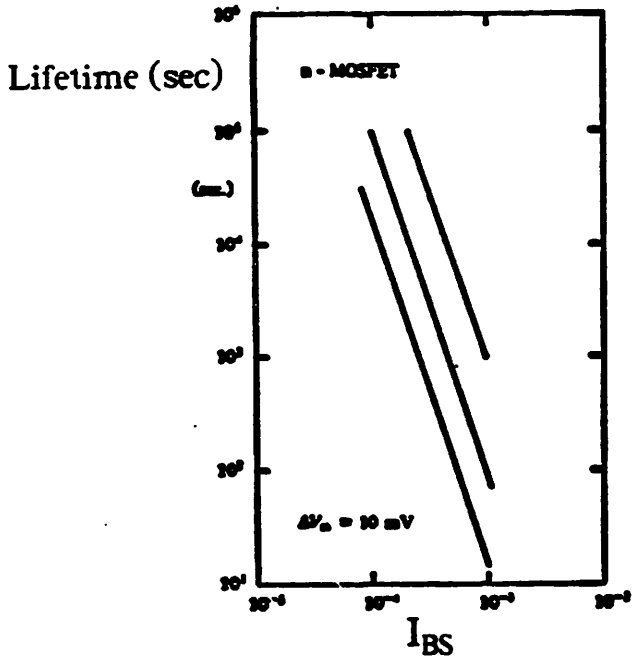


Fig.4 Typical log - log plots of device lifetime τ versus I_{BS} for various device technologies (From Hu et al [3]).

Using a quasi-static analysis, equations (42), (43), and (44) can be extended to model the dynamic stressing behavior in a circuit for which the substrate current of a device is a function of time. Let $\Delta V_{thf} = \Delta V_{th}\,|_{t=\tau}$ be the threshold voltage shift defined at device failure. By solving for A in equation (42) using equations (43) and (44), we get:

$$\Delta V_{thf} = A\,\tau^n = A\,[\,W_{eff}\,B\,I_{BS}^{-m}\,I_{DS}^{m-1}]^n \tag{47}$$

$$A = \Delta V_{thf}\,[\,W_{eff}\,B\,I_{BS}^{-m}\,I_{DS}^{m-1}]^{-n} \tag{48}$$

Thus equation (42) becomes

$$\Delta V_{th} = \Delta V_{thf}\,(\,W_{eff}\,B\,)^{-n}\,I_{BS}^{mn}\,I_{DS}^{n(1-m)}\,t^n \tag{49}$$

or

$$\Delta V_{th}^{1/n} = \Delta V_{thf}^{1/n}\,(\,W_{eff}\,B\,)^{-1}\,I_{BS}^{m}\,I_{DS}^{(1-m)}\,t \tag{50}$$

$$= (\,W_{eff}\,H\,)^{-1}\,I_{BS}^{m}\,I_{DS}^{(1-m)}\,t \tag{51}$$

Since $\Delta V_{th}^{1/n}$ is a linear function of time, this quantity can simply be summed over the time period of the SPICE analysis. If $t_0, \ldots, t_P$ are the individual time points of SPICE, then

$$[\Delta V_{th(tot)}]^{1/n} = [\Delta V_{th}(t_1 - t_0)]^{1/n} + \cdots + [\Delta V_{th}(t_P - t_{P-1})]^{1/n} \tag{52}$$

Moreover, to find the device lifetime assuming a periodic signal, a simple linear extrapolation in terms of $\Delta V_{th}^{1/n}$ is all that is necessary. The number of time intervals of the SPICE analysis needed so that $\Delta V_{th} = \Delta V_{thf}$ is simply

$$N = \left[\frac{\Delta V_{thf}}{\Delta V_{th(tot)}}\right]^{1/n} \tag{53}$$

Thus, if the length of the SPICE analysis is $t_P = T$ and is equal to the period of the signal, the lifetime is found from

$$\tau = N\,T \tag{54}$$

$$= T\left[\frac{\Delta V_{thf}}{\Delta V_{th(tot)}}\right]^{1/n} \tag{55}$$

or

$$\tau = \frac{T\,W_{eff} \cdot B}{\sum_{h=1}^{p} [I_{BS}(t_h)]^m\,[I_{DS}(t_h)]^{1-m}\,(t_h - t_{h-1})}$$

$$\tau = \frac{T\,W_{eff}\,H\,\Delta V_{thf}^{1/n}}{\sum_{h=1}^{p} [I_{BS}(t_h)]^m\,[I_{DS}(t_h)]^{1-m}\,(t_h - t_{h-1})} \tag{56}$$

1.7 Charge and Capacitance Model

Both conductive and capacitive current components exist due to the same charge

storage in an MOS transistor. The charge and capacitance model presented here was intimately derived with its associated dc model. Charge conservation, channel-charge partition, and body-bias effects are properly incorporated and the device physics are explained and properly modeled.

Terminal charges instead of terminal voltages are chosen as state variables to avoid the charge non-conservation problem. The expressions for the charge densities are similar to those of [8]. However, due to different ways of treating the physics behind the transistor operation, the functional dependence of α_x and channel charge partition are quite different. The equations for the charge in different operation regions are given below.

1.7.1 Accumulation region: ($V_{GS} < V_{FB}$)

$$Q_G = W_{eff} L_{eff} C_{ox} (V_{GS} - V_{FB} - V_{BS}) \tag{57}$$

$$Q_B = -Q_G \tag{58}$$

$$Q_S = 0 \tag{59}$$

$$Q_D = 0 \tag{60}$$

$$Q_C = Q_S + Q_D = 0 \tag{61}$$

where Q_G, Q_B, Q_C, Q_S, and Q_D are terminal charges associated with gate, bulk, channel, source, and drain, respectively.

1.7.2 Subthreshold region: ($V_{FB} < V_{GS} < V_{th}$)

$$Q_G = W_{eff} L_{eff} C_{ox} \frac{K_1^2}{2} \left| -1 + \sqrt{1 + \frac{4 (V_{GS} - V_{FB} - V_{BS})}{K_1^2}} \right| \tag{62}$$

$$Q_B = -Q_G \tag{63}$$

$$Q_S = 0 \tag{64}$$

$$Q_D = 0 \tag{65}$$

$$Q_C = 0 \tag{66}$$

1.7.3 Inversion region: ($V_{th} > V_{th}$)

Rewriting the drain current equation in the linear region in a form similar to that of a long-channel device, we have

$$I_{DS} = \beta [(V_{GS} - V_{th}) V_{DS} - \frac{\alpha_x}{2} V_{DS}^2] \tag{67}$$

Then the drain saturation voltage is given by

$$V_{DSAT} = \frac{V_{GS} - V_{th}}{\alpha_x} \tag{68}$$

Comparing (68) with (23), we have

$$\alpha_x = a\sqrt{K} \tag{69}$$

where a and K are given in (13) and (22), respectively. For small value of $V_{GS} - V_{th}$, α_x can be approximated by

$$\alpha_x = a\,[\,1 + U_1\,(\,V_{GS} - V_{th}\,)\,] \tag{70}$$

The distributed charge density of the gate, channel, and bulk can, therefore, be expressed as

$$q_g = C_{ox}\,(\,V_{GS} - V_{FB} - \phi_S - V_y\,) \tag{71}$$

$$q_c = -C_{ox}\,(\,V_{GS} - V_{th} - \alpha_x\,V_y\,) \tag{72}$$

$$q_b = -C_{ox}\,[\,V_{th} - V_{FB} - \phi_S - (\,1 - \alpha_x\,)\,V_y\,] \tag{73}$$

where V_y is the electron quasi-Fermi potential with respect to the source. Note that

$$q_g + q_c + q_b = 0 \tag{74}$$

This relationship follows the constrain of charge neutrality in a one-dimensional MOS capacitor structure.

Expressions for the total charge stored in the gate, bulk, and channel regions can be obtained by integrating the distributed charge densities over the area of the active gate region (from $y = 0$ to $y = L_{eff}$ and $z = 0$ to $z = W_{eff}$). That is

$$Q_G = W_{eff} \int_0^{L_{eff}} q_g(y)\,dy \tag{75}$$

$$Q_B = W_{eff} \int_0^{L_{eff}} q_b(y)\,dy \tag{76}$$

$$Q_C = W_{eff} \int_0^{L_{eff}} q_c(y)\,dy \tag{77}$$

Substituting (71)-(73) into (75)-(77) and replacing the differential channel length "dy" with the corresponding differential potential drop "dV", we obtain the following expressions for total gate, bulk, and channel charges in the static equilibrium.

$$Q_G = W_{eff} L_{eff} C_{ox} \left| V_{GS} - V_{FB} \phi_S - \frac{V_{DS}}{2} + \frac{\alpha_x V_{DS}^2}{12 (V_{GS} - V_{th} - \frac{\alpha_x}{2} V_{DS})} \right| \tag{78}$$

$$Q_B = W_{eff} L_{eff} C_{ox} \left| -V_{th} + V_{FB} + \phi_S + \frac{(1-\alpha_x)}{2} V_{DS} - \frac{(1-\alpha_x) V_{DS}}{12} \frac{\alpha_x V_{DS}}{(V_{GS} - V_{th} - \frac{\alpha_x}{2} V_{DS})} \right| \tag{79}$$

$$Q_C = -W_{eff} L_{eff} C_{ox} \left| V_{GS} - V_{th} - \frac{\alpha_x}{2} V_{DS} + \frac{\alpha_x^2 V_{DS}^2}{12 (V_{GS} - V_{th} - \frac{\alpha_x}{2} V_{DS})} \right| \tag{80}$$

Note that the charge-neutrality relationship still holds.

$$Q_G + Q_B + Q_C = 0 \tag{81}$$

In BSIM, the source/drain partitioning of the channel charge smoothly changes from 40/60 (or 0/100, or 50/50) in the saturation region to 50/50 distribution in the triode region.

(A) 40/60 Channel-Charge Partitioning

For 40/60 partitioning, the method proposed by Oh [9] is used.

$$Q_S = -W_{eff} \int_0^{L_{eff}} (1 - \frac{y}{L_{eff}}) q_c(y) \, dy \tag{82}$$

$$Q_D = -W_{eff} \int_0^{L_{eff}} \frac{y}{L_{eff}} q_c(y) \, dy = Q_C - Q_S \tag{83}$$

Carrying out the integration in (79) and (80), with q_c replaced by (69), we obtain the total charge associated with the source and drain terminals.

(A.1) Triode region:

The expressions for Q_G and Q_B are the same as those in (78) and (79), and

$$Q_S = -W_{eff} L_{eff} C_{ox} \left| \frac{V_{GS} - V_{th}}{2} + \frac{\alpha_x^2 V_{DS}^2}{12 (V_{GS} - V_{th} \frac{\alpha_x}{2} V_{DS})} - \frac{\alpha_x V_{DS}}{(V_{GS} - V_{th} - \frac{\alpha_x}{2} V_{DS})^2} \left[\frac{(V_{GS} - V_{th})^2}{6} - \frac{\alpha_x V_{DS} (V_{GS} - V_{th})}{8} \right. \right.$$

$$+ \frac{\alpha_x^2 V_{DS}^2}{40}] \tag{84}$$

$$Q_D = -W_{eff} L_{eff} C_{ox} \left[\frac{V_{GS} - V_{th}}{2} - \frac{\alpha_x V_{DS}}{2} + \frac{\alpha_x V_{DS}}{(V_{GS} - V_{th} - \frac{\alpha_x}{2} V_{DS})^2} \right.$$

$$\left. [\frac{(V_{GS} - V_{th})^2}{6} - \frac{\alpha_x V_{DS} (V_{GS} - V_{th})}{8} + \frac{\alpha_x^2 V_{DS}^2}{40}] \right. \tag{85}$$

(A.2) Saturation region:

$$Q_G = W_{eff} L_{eff} C_{ox} (V_{GS} - V_{FB} - \phi_S - \frac{V_{GS} - V_{th}}{3\alpha_x}) \tag{86}$$

$$Q_B = W_{eff} L_{eff} C_{ox} [V_{FB} + \phi_S - V_{th} + \frac{(1 - \alpha_x) (V_{GS} - V_{th})}{3\alpha_x}] \tag{87}$$

$$Q_S = -\frac{2}{5} W_{eff} L_{eff} C_{ox} (V_{GS} - V_{th}) \tag{88}$$

$$Q_D = -\frac{4}{15} W_{eff} L_{eff} C_{ox} (V_{GS} - V_{th}) \tag{89}$$

(B) 0/100 Channel-Charge Partitioning

(B.1) Linear region

Expressions for Q_G and Q_B are the same as those in (78) and (79), and

$$Q_S = -W_{eff} L_{eff} C_{ox} \left| \frac{V_{GS} - V_{th}}{2} + \frac{\alpha_x V_{DS}}{4} - \frac{(\alpha_x V_{DS})^2}{24 (V_{GS} - V_{th} - \frac{\alpha_x}{2} V_{DS})} \right| \tag{90}$$

$$Q_D = -W_{eff} L_{eff} C_{ox} \left| \frac{V_{GS} - V_{th}}{2} - \frac{3 \alpha_x V_{DS}}{4} + \frac{(\alpha_2 V_{DS})^2}{8 (V_{GS} - V_{th} \frac{\alpha_x}{2} V_{DS})} \right| \tag{91}$$

(B.2) Saturation region:

Expressions for Q_G and Q_B are the same as those in (86) and (87), and

$$Q_S = -\frac{2}{3} W_{eff} L_{eff} C_{ox} (V_{GS} - V_{th}) \tag{92}$$

$$Q_D = 0 \tag{93}$$

(C) 50/50 Channel-Charge Partition

The charge expressions for the 50/50 partitioning method are listed below.

(C.1) Linear Region

Expressions for Q_G and Q_B are the same as those in (78) and (79).

$$Q_S = -\frac{1}{2} W_{eff} L_{eff} C_{ox} \left| V_{GS} - V_{th} - \frac{\alpha_x V_{DS}}{2} + \frac{\alpha_x^2 V_{DS}^2}{12 (V_{GS} - V_{th} - \frac{\alpha_x}{2} V_{DS})} \right. \quad (94)$$

$$Q_D = Q_S \quad (95)$$

(C.2) Saturation Region

Expressions for Q_G and Q_B are the same as those in (86) and (87).

$$Q_D = -\frac{1}{3} W_{eff} L_{eff} C_{ox} (V_{GS} - V_{th}) \quad (96)$$

$$Q_S = Q_D \quad (97)$$

The non-reciprocal capacitances can be obtained from charge derivatives

$$C_{ij} = \left| \frac{\partial Q_i}{\partial V_j} \right| \quad (98)$$

where i and j stand for g, b, d, and s. For example, C_{gd} describes the gate current produced by a change in the drain voltage, and C_{dg} gives the drain current which results from a change in the gate voltage. C_{gd} and C_{dg} differ both in value and in physical interpretation.

These sixteen capacitances have the following properties.

$$\sum_{j=g,b,d,s} C_{ij} = 0 \qquad \text{for } i = g,b,d,s \quad (99)$$

and

$$\sum_{i=g,b,d,s} C_{ij} = 0 \qquad \text{for } j = g,b,d,s \quad (100)$$

The above two conditions put seven constrains on the capacitances. Nine of the capacitances are independent and must be calculated explicitly.

Fig. 5a shows plots of the four normalized terminal charges with the 40/60 channel-charge partitioning method. Selected plots of MOS transistor capacitances normalized to the total gate-oxide capacitance, $C_{ox} W_{eff} L_{eff}$, are shown in Fig. 5b and 5c. Similar plots of terminal charges and capacitances for the 0/100 and 50/50 channel-charge partitioning

methods can be easily obtained, too. Notice that the charges and capacitances are all continuous at the boundary of the linear and saturation regions. When a transistor is biased in the saturation region, the channel is decoupled from the drain as predicted by long-channel device theories. This decoupling of the drain from the conduction channel results in all partial derivatives with respect to the drain voltage (C_{gd},C_{bd},C_{dd}, and C_{sd}) to be zero.

As the gate bias increases and reaches the threshold voltage, the transistor channel charge starts to appear and all the 16 capacitances abruptly change in value. The discontinuity in capacitances are caused by the strong-inversion approximation used in charge formulation. In transient analysis, the discontinuity in capacitances poses no problem because they are multiplied by terms of voltage difference which vanish as convergence is reached [10]. However, for small-signal ac analysis. Smooth capacitance characteristics are highly desired. A common remedy for the problem of discontinuity is to smooth out the capacitance characteristics. This proves to be a non-trivial task because MOS transistor capacitances are function of more than one voltage. In general, smoothing has to be done on a multi-dimensional basis.

Figure 5d compares calculated results with the measured data given in [11] for a long-channel transistor. Good agreement is found.

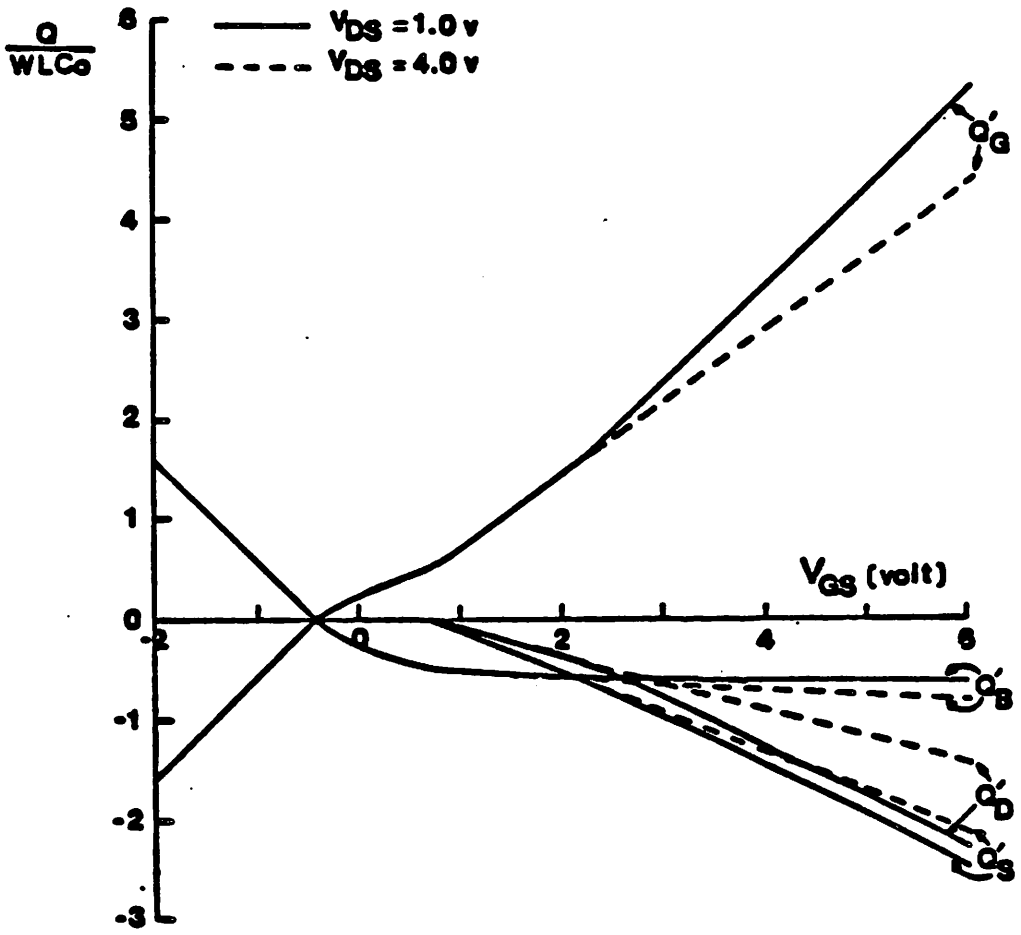


Fig.5aPlots of normalized terminal charges versus gate bias for two drain voltages. The parameters are V_{BS} = 0 V, V_{DS} = 1 and 4 V, a = 1.224, V_{th} = 0.7 V, K_1 = 0.633, ϕ_S = 0.625 V.

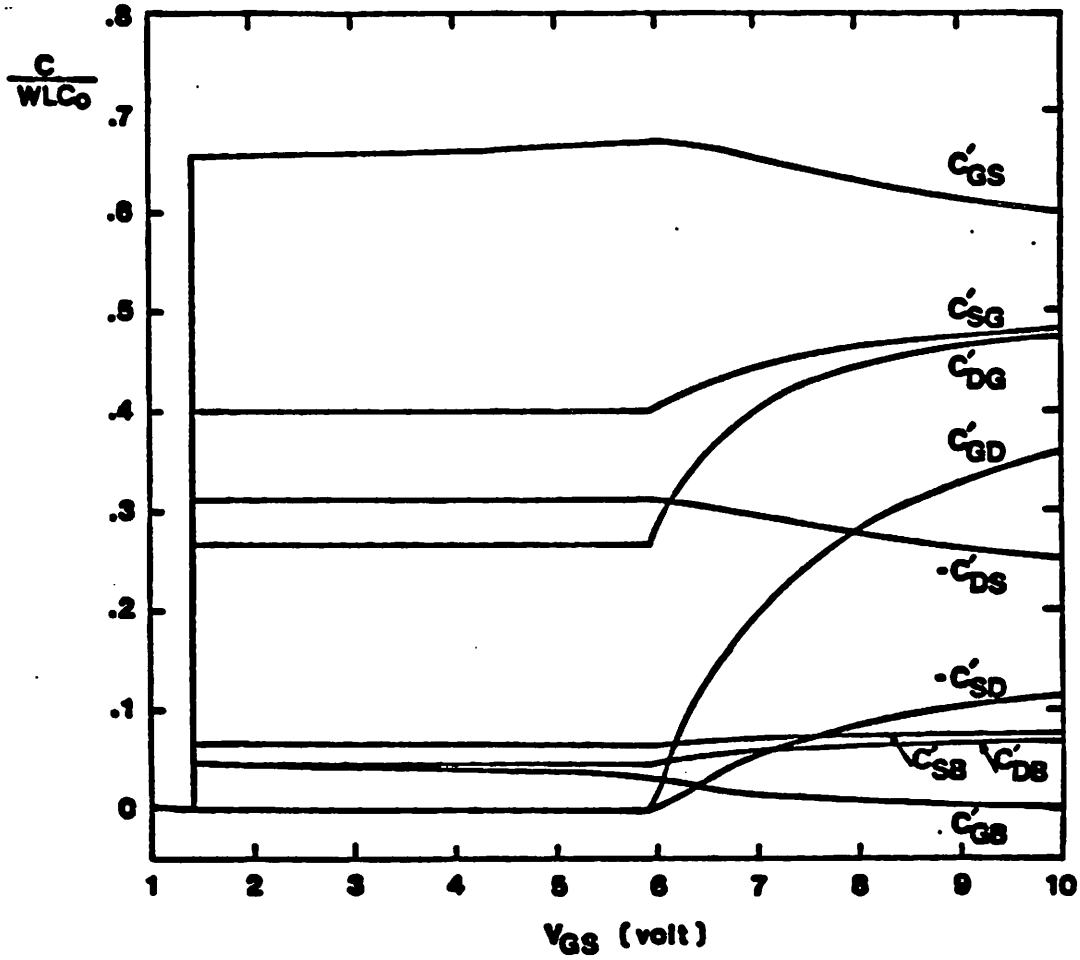


Fig.5bSelected plots of normalized capacitances versus the gate bias. The parameters are V_{BS} = -3.0 V. V_{DS} = 4 V. a = 1.131. V_{th} (at V_{BS} = -3.0 V) = 1.4 V. K_1 = 0.63. ϕ_S = 0.62.

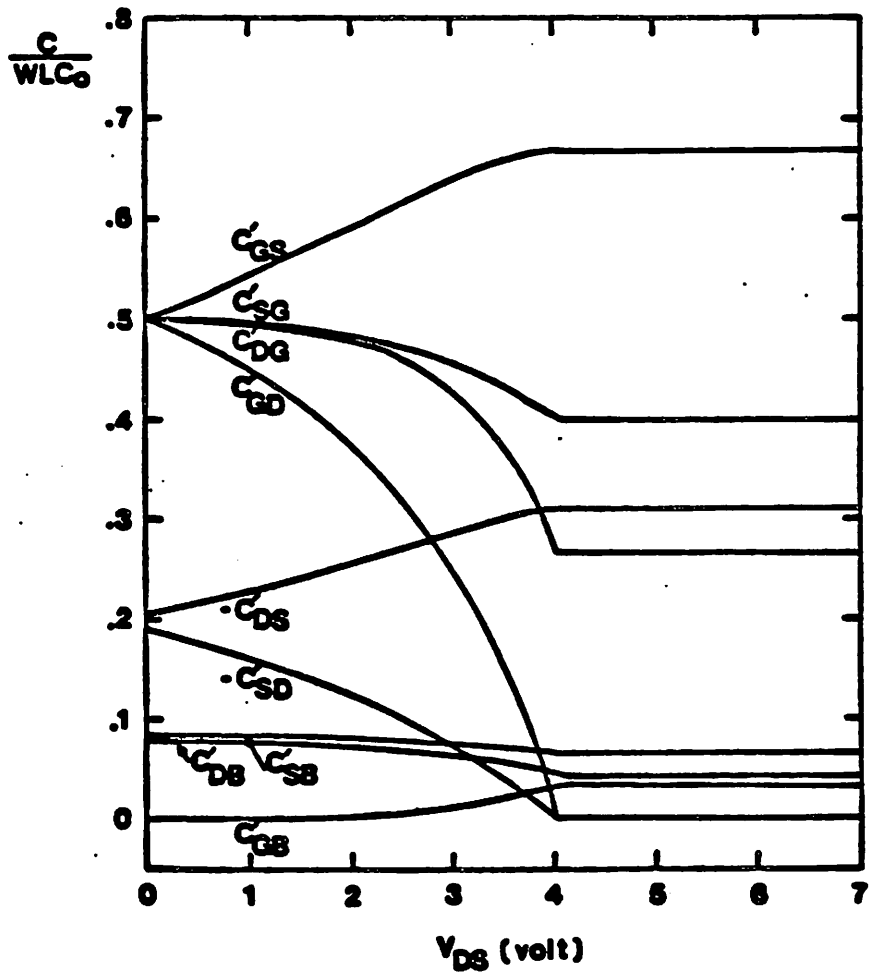


Fig.5cSelected plots of normalized capacitance versus the drain bias. The parameters are V_{BS} = -3.0 V, V_{DS} = 4 V, a = 1.131, V_{th} (at V_{BS} = -3.0 V) = 1.4 V, K_1 = 0.63, ϕ_S = 0.62.

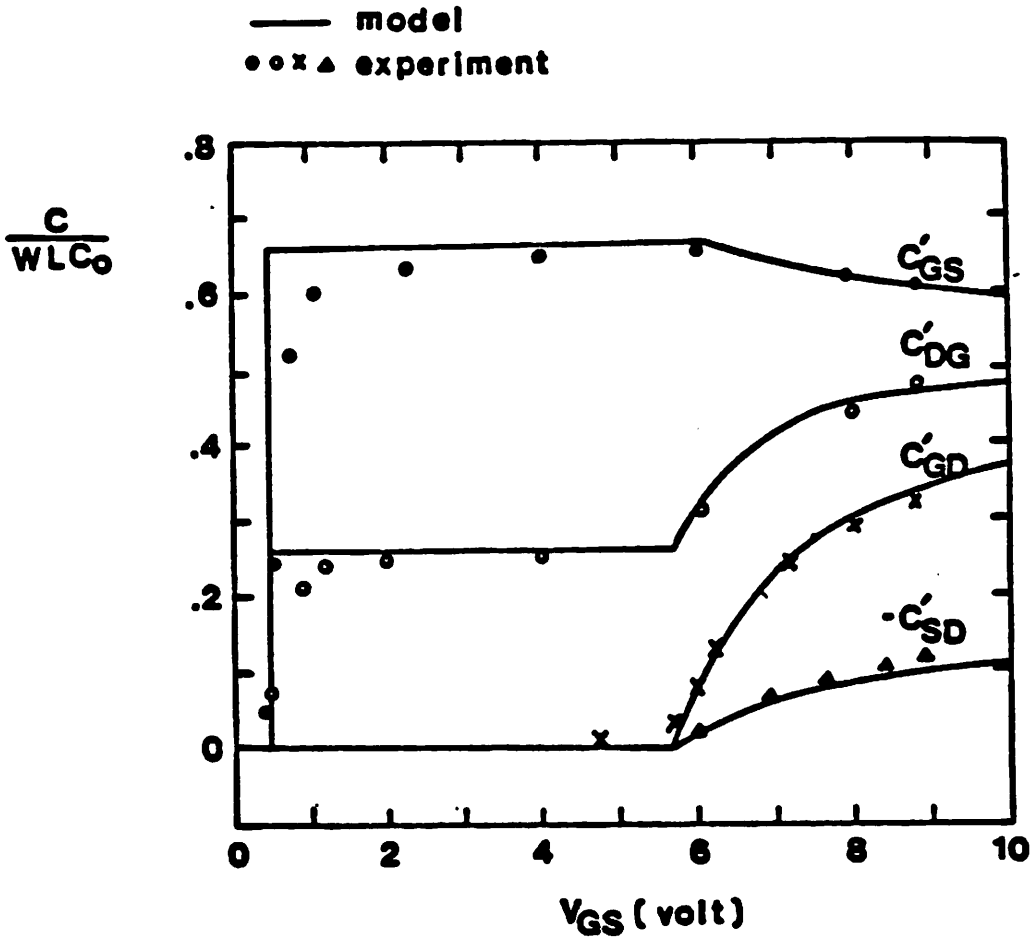


Fig.5dComparison of several calculated and measured capacitances versus the gate bias with $V_{BS} = 0$ V and $V_{DS} = 4$ V.

Chapter 2: Parameter Extraction Program

This chapter consists of three sections. Section 1 is the user's guide of the parameter extraction program. Section 2 describes the measurement procedures. Section 3 describes the theory and algorithms used in the extraction program.

2.1 BSIM Parameter Extraction Program User's Guide

This section describes the system setup required by the parameter extraction program and shows the users how to run the program in a step-by-step approach.

2.1.1 PARAMETER EXTRACTION SYSTEM

An indispensable tool in parameter extraction is a fully automated characterization system, because large amounts of measurement are required. An HP9836 based integrated system was developed for automated extraction of BSIM parameters. The schematic diagram of the system hardware is shown in Fig. 6. This system consists of three major parts: a HP-200 series computer (HP9836, for example) with at least 1.5 mega byte of memory, an HP4145 parameter analyzer and a probe station (A version of BSIM parameter extraction program is also available for the HP4062 system [12]). The probe station can be either automatic or manual. In the case of an automatic probe station (eg. Electroglas 2001X), the computer can also drive the probe station to the position of those devices specified in a probe file which will be explained in a later section. The system communicates with a SPICE-resident VAX computer via the RS232 bus. Good electrical and optical shielding is ensured by placing the probe station inside a light-shielded aluminum box. The system has the capability of measuring current down to below 1pA, limited by the measurement instruments. The software is written in HP PASCAL. The program is modular and menu-driven for easy operation and future model modification. The only required user-supplied inputs are the device dimensions in the test structure and the SMU (Stimulus/Measurement Unit) connections for HP4145 parametric analyzer. This information is specified in a file called "probe file" which is read by the program.

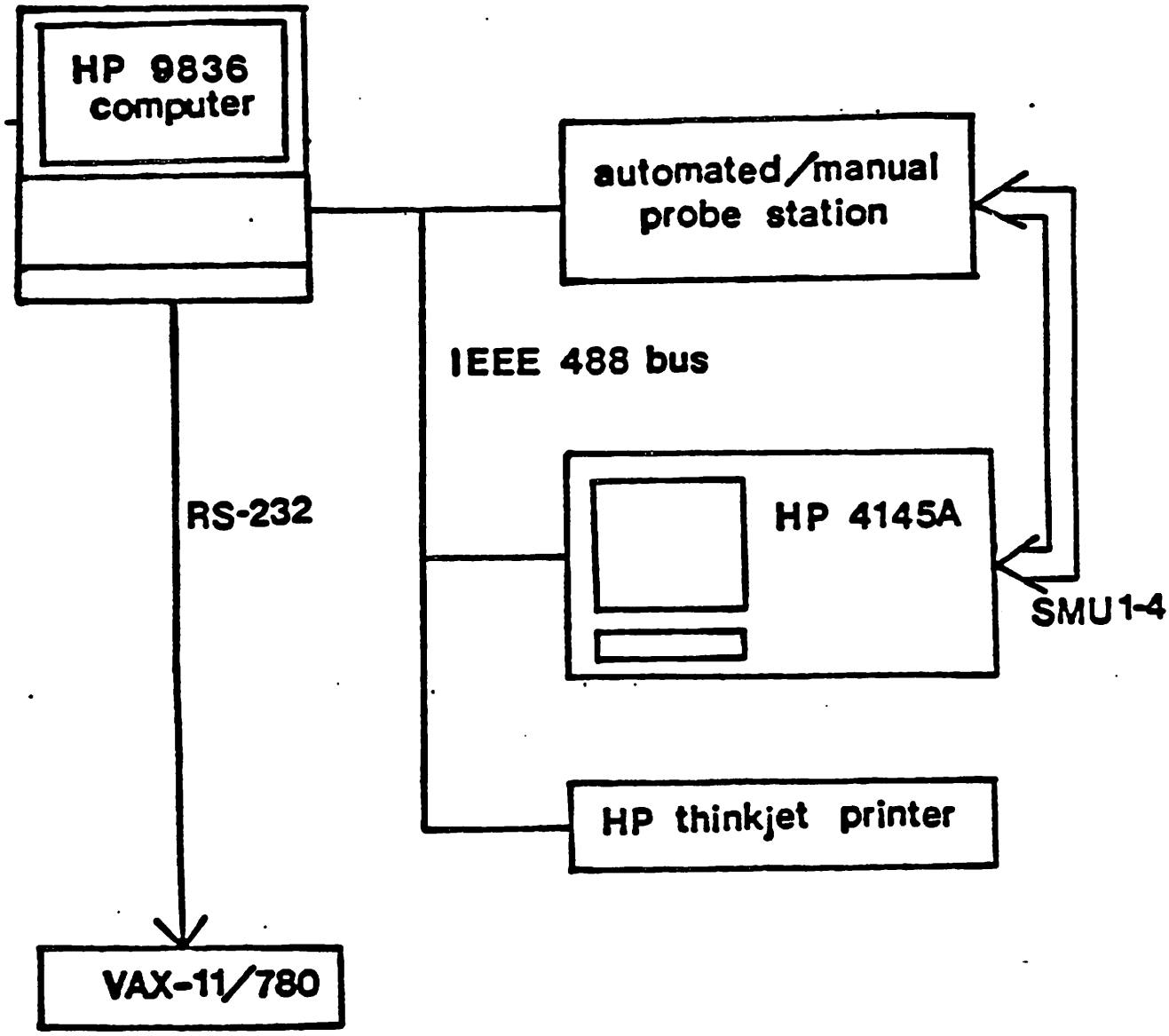


Fig.6 The parameter extraction system.

Before the formal extraction procedures begin, device functionality tests are performed to determine the device type and ensure the proper connections. After the extraction is completed, a verification procedure is executed. If the parameter values are not within the physical ranges, that device will be discarded or re-measured as the user desires. Since parameters are extracted separately from each device, every parameter can be plotted against the channel length or channel width to check the effects of device dimension to the parameters. This information is very useful when interpolation or extrapolation of device characteristics is required. Higher accuracy of simulation results can be achieved by deleting the device with abnormal parameter values from the test structure.

All the parameters of each device are stored in a temporary file. After completion of extracting all devices specified in the probe file, this temporary is processed to generated a final size-independent parameter set called "process file". This process file is read by SPICE to calculate parameters of devices with any channel length and channel width. The user's guide of SPICE2 with BSIM implemented can be found in Chap. 3. A more detailed description and implementation of the extraction system is explained in later sections of this chapter.

In the following descriptions, user actions are in BOLD print and program prompts are in *italic* letters.

2.1.2 4145 SETUP INSTRUCTIONS

A) **Place the Software Diskett Revision A5 in the floppy disc drive slot on the lower left side of the 4145. The disc label should be up and towards you. Close the floppy disc drive door.**

B) **Press the "ON" switch on the left side of the 4145, and the machine should display a menu after it calibrates itself.**

2.1.3 2001X PROBER SETUP INSTRUCTIONS

There are four operation modes in the parameter extraction program. Each mode is explained in the display menu of the program. If you are using the automatic mode (Mode 1) or semi-automatic mode with automatic prober (Mode 2)*, the following instructions should be followed. For Mode 3 and Mode 4 operations, which use manual probe stations, this section can be skipped.

A) Make sure that no foreign objects are on the prober stage, or on the wafer chuck.

B) If the probe card you want to use is not in the probe card rack, put the probe card you want to use in the rack and press it back into the edge connector.. Make sure that all four screws which hold the probe card are securely tightened down.

C) Press the "ON" switch on the lower right front of the prober front panel.

D) Look at the Prober Video Display and answer the following questions as shown below. If you do not respond fast enough, a default response will be chosen by the prober, and it will move on to the next question.

Type Message Plus Enter=> **ENTER**

Wait for Pattern Rec I/O Test... **Wait about 30 seconds**

Rom Test? **Y**

Repeat Test? **ENTER**

E) When the standard display comes up it should say

XY MOTOR BLANK*

at the bottom of the screen. This means that the stage is floating on the platform and can be moved around. Pull the stage so that it is touching the front cushion on the prober platform. Now slide the stage along the front cushion to the right until it is

* The instruction listed here are specifically for the 2001 automatic prober.

contacting both the front and the right cushions. To hold the stage in place, hit the button inside the left side of the joystick control panel. (This button is recessed, and is in a cutout hole on the left vertical side of the joystick box)

F) On the front panel of the prober, above the label saying Model 2001X, is a small vacuum lever with a black handle. Pull the lever out so that it is perpendicular to the panel, and you should here a hissing noise as the vacuum turns on.

G) At this point, the I/O Mode should be set for the Prober. Turn to the control panel with the video monitor, and perform the bold actions to the *italic* video monitor requests

Press the blue 'Set Mode' key.

Select Line?= **7 and ENTER**

IOMODE? 0=OFF, 1=SERIAL, 2=GPIB **2 and ENTER**

If the line 9 GPIB (IEEE-488) address is not equal to 14, then set it to 14 as follows
Select Line?= **9 and ENTER**
GPIB ADDRESS=? 1 TO 15 **14 and ENTER**

Press the Yellow *ON LINE* Key on the monitor panel, which sets the prober up to receive signals from the 9836.

H) The stage up and down limits must now be either set or verified depending upon whether a new probe card will be used. Press the blue *SET PRMTR* key on the monitor panel, and observe the Z UP LIMIT and the Z DOWN LIMIT. The Z UP LIMIT should be about 30MILS above the Z DOWN LIMIT. Typical values might be Z UP LIMIT=370MILS and Z DOWN LIMIT=340MILS. If the probe card has not been changed, these values should have been previously setup, and require only verification. Using a new probe card requires that the LIMITS be lowered significantly, and then adjusted by raising the LIMITS incrementally until the probes barely touch the wafer when the chuck is up. This should be done by an experienced person. After the probes just touch the wafer, the Z UP LIMIT should be raised by 2.5 to 3.0 MILS to provide sufficient overdrive.

I) Place the wafer on the stage, and press both the *VAC* and the *LAMP* buttons on the joystick control panel. The video dislay should show that the wafer and chuck vacuum are on.

J) Press the *Align Scan* button on the panel with the joystick, and the stage should move under the probes. The index, jog and scan modes are selected by twisting the joystick. Faster movement is provided by pressing the red button on the joystick. The wafer is aligned by pressing the *Pause* key twice so that the stage is moving back and forth under the probes. The twist knob on the joystick control box is a theta adjust, and is used to align the wafer. Alignment is done by watching the wafer pass under a probe and using the theta adjust until patterns are tracking the probe across the wafer.

K) Once the wafer is aligned, the wafer should be moved so that the probes are over the device to be tested, and then the stage is raised with the *Z* button. The stage may have to be lowered and moved so that the probes will contact the center of the pads.

L) The Prober is setup for Automatic Operation

2.1.4 HP 9836 BOOTING INSTRUCTIONS

Four operating system disks are needed to set up HP 9836: Pascal 2.0 BOOT Disc, Pascal 2.0 SYSVOL Disc, Pascal 2.0 ACCESS Disc and Pascal 2.0 CMPASM Disc. These disks should come with the machine (HP 9836). To make HP 9836 setup procedure easier to follow, a specially modified version of PASCAL 2.0 SYSVOL (provided in the BSIM package) is used. The setup procedures described below refer to the modified version.

Note 1: For users who use Pascal 3.0 system, the booting procedures are the same except that the "LIBRARY" file in SYSVOL Disc has to be replaced by another "LIBRARY" file suitable for 3.0 system. The procedures regarding to how to create such a file can be found in "HP Pascal 3.0 Workstation".

Note 2: If you are using a color monitor (eg. HP9836C), you have to use Pascal 3.0 system.

Note 3: If you have a hard disc, you can store all the files which are in the Booting Discs into the hard disc and modify the "AUTOSTART" file, so that the machine can boot itself. This decreases the booting time dramatically.

(A) **INSERT** Pascal 2.0 SYSVOL in the left disc drive(unit#4).

INSERT Pascal 2.0 BOOT in right disc drive(unit#3).

PRESS the switch located on the front bottom right of the keyboard in to turn on the HP 9836.

(B) The operating system is now being loaded. Messages indicating the loading of OS are flashed on the screen.

DO NOTHING until the following message appears on the top of the screen.

Replace BOOT with ACCESS and Press 'ENTER'

TAKE Pascal 2.0 BOOT out of the right disc drive(unit#3).

INSERT Pascal 2.0 ACCESS in the right disc drive(unit#3).

PRESS [ENTER] key.

Note: If an error message appeared on the screen " not enough memory ", You have to modify the "AUTOSTART" file in the SYSVOL Disc using "EDITOR" in the ACCESS Disc. In "AUTOSTART" file, the fourth line is a number "2500", you should change it to a smaller number suitable to the memory size of your computer and re-boot the system again.

(D) **IGNORE** messages flashed on the screen until the following messages appear.

Replace ACCESS with CMPASM and Press 'ENTER'

TAKE Pascal 2.0 ACCESS out of the right disc drive(unit#3).

INSERT Pascal 2.0 CMPASM in the right disc drive(unit#3).

PRESS [ENTER] key.

(E) **DO NOTHING** until the following appears on the screen.

Replace CMPASM & SYSVOL with A.CODE & B.CODE, and Press 'ENTER'

TAKE Pascal 2.0 SYSVOL out of the right disc drive(unit#3).

INSERT A.CODE in the right disc drive(unit#3).

TAKE Pascal 2.0 CMPASM out of the right disc drive(unit#4).

INSERT B.CODE in the right disc drive(unit#4).

PRESS [ENTER] key.

Go to section "LOAD the BSIM MACHINE CODE FILES".

Note: The operating system has been successfully loaded. If you are running the program first time, there will be an error message like the following displayed at the bottom of the screen.

cannot open '#3:measure.CODE', ioresult=34

This is because that the program you have in hand has not been compiled yet. You should go through the procedures in section "LOAD and COMPILE the BSIM MODULAR SOURCE CODE FILES".

2.1.5 LOADING AND COMPILING BSIM MODULAR SOURCE CODE FILES

This part is for user who has not generated the machine code files of the BSIM parameter extraction program. In order to proceed, user has to have the BSIM modular source code files which can be found in two floppy discs labeled as "bsim2.1.TEXT" and "bsim2.2.TEXT" in the BSIM package. The BSIM parameter extraction program is separated into six modules: *lib.TEXT*, *measure.TEXT*, *po_graph.TEXT*, *iv_graph.TEXT*, *extract.TEXT*, and *main.TEXT*. The first three .TEXT files are stored in the bsim2.1.TEXT, while the last three are in the bsim2.2.TEXT. Two previously initialized blank floppy dises labeled "A.CODE" and "B.CODE" are needed to store the compiled machine code file corresponding to each text file.

(A) **IGNORE** the instruction shown on the screen.

REMOVE the Pascal 2.0 CMPASM from the right disc drive(unit#3).

LEAVE the Pascal 2.0 SYSVOL in the left disc drive(unit#4).

HIT [S] key.

(B) The system prompts you for the following:

Stream what file?

TYPE SYSVOL:AUTO4

HIT [ENTER] key.

(C) The following message should appear on the screen.

Stream what file? SYSVOL:AUTO4

Replace CMPASM & SYSVOL with BSIM.A & BSIM.B and Press 'ENTER'

INSERT BSIM.A in the right disc drive(unit#3).

INSERT BSIM.B in the left disc drive(unit#4).

CLOSE both disc drive doors.

HIT [ENTER] key.

(D) The system is automatically compiling the source codes and storing the machine codes into the RAM. The whole process should take about eight minutes. User should wait until the whole process has been finished and then proceed. After the compilation has done and the following line is shown.

Command: Compiler Editor Filer Initialize Librarian Run eXecute Version?

do the following procedures.

REMOVE BSIM.A from the right disc drive(unit#3).

REMOVE BSIM.B from the left disc drive(unit#4).

(E) INSERT the Pascal 2.0 SYSVOL in the left disc drive(unit#4).

HIT [S] key and you will the see the following line.

Stream what file?

TYPE SYSVOL:AUTO5

(F) User should have two initialized floppy discs labeled "A.CODE" and "B.CODE" available to copy all the machine code files from the RAM into the floppy discs. On the screen, a message is shown as the following.

Stream what file? SYSVOL:AUTO5

Remove SYSVOL, Insert Two Blank Discs In, and Press 'ENTER'

REMOVE SYSVOL from the left disc drive(unit#4).

INSERT disc "A.CODE into right disc drives (unit#3).

INSERT disc "B.CODE into left disc drives (unit#4).

PRESS [ENTER] key.

(G) The system is copying three specified code files into each floppy disc. It takes about a minute to complete.

DO NOTHING until the system has finished compiling and has shown the following message on the top of the screen.

Command: Compiler Editor Filer Initialize Librarian Run eXecute Version?

TAKE the disc out of the right disc drive(unit#3).

TAKE the disc out of the left disc drive (unit#4).

(H) Now, user should have two floppy discs storing the generated machine codes.

INSERT Pascal 2.0 SYSVOL in the left disc drive(unit#4).

HIT [S] key and the system prompts you as follows:

Stream what file?

TYPE SYSVOL:AUTO3

(I) The following message should display on the top of the screen.

Stream what file? SYSVOL:AUTO3

2.1.6 EXECUTING BSIM PARAMETER EXTRACTION PROGRAM

Replace CMPASM & SYSVOL with A.CODE & B.CODE, and Press 'ENTER'

NOTE that the message shown in here is the same one shown in the last section part (E). Since the A.CODE and the B.CODE are available now, user can execute the procedures in section #3.

2.1.7 LOADING BSIM MACHINE CODE FILES

User who has skipped section #2 should have the A.CODE and the B.CODE discs available. And, user who has gone through the procedures in section #2 should also have the discs by now. If not, repeat the procedures in section #2 starting from part (b).

(A) INSERT A.CODE file in the right disc drive(unit#3).

INSERT B.CODE file in the left disc drive(unit#4).

HIT [ENTER] key.

DO NOTHING while the system is loading all the machine codes into the RAM:LIBRARY and then go to section #4.

2.1.8 EXECUTING BSIM MACHINE CODE FILES

The BSIM parameter extraction program is ready to be executed at this stage. The following describes the procedures to execute the extraction program.

(A) PRESS [X] key.

(B) The following message should appear on the screen.

Execute what file?

TYPE #4:main.CODE

HIT [ENTER] key.

(A) The following message will appear on the screen

loading 'main.CODE'

followed by a screenful of BSIM MENU PAGE which is shown in Fig. 7a.

(B) **READ** the menu page. (if you wish)

ENTER the number of desired operation mode.

```
            BSIM AUTOMATIC MOS DEVICE CHARACTERIZATION PROGRAM
                    UC BERKELEY FALL 1986 VERSION 1.0
This Program can be used in any of the following modes:
[1] Fully Automatic, [2] Semi Automatic--with an automatic prober,
[3] Semi Automatic--with a manual prober, and [4] Single Device Operation.

FULLY AUTOMATIC OPERATION requires a prober file, and tests all devices
in the file without interuption. This mode requires an automatic prober.

SEMI AUTOMATIC--[AUTOMATIC PROBER] OPERATION requires a prober file and auto-,
matically moves to each device in the file.  This mode stops at each device to
allow the user to switch connections. This mode requires an automatic prober.

SEMI AUTOMATIC--[MANUAL PROBER] OPERATION is similar to SEMI AUTOMATIC--
[AUTOMATIC PROBER], but does not require an automatic prober.

SINGLE DEVICE OPERATION allows the user to analyze an individual device,
extract BSIM parameters, and compare simulated versus measured data.
                                [1]:FULLY AUTOMATIC
                                [2]:SEMI AUTOMATIC--[AUTOMATIC PROBER]
                                [3]:SEMI AUTOMATIC--[MANUAL PROBER]
Select a Mode of Operation  >   [4]:SINGLE DEVICE
                                [5]:SIMULATION USING EXISTING PROCESS FILE
                                [6]:EXIT BSIM
```

Fig.7aThe initial menu page of the extraction program.

(C) Five different operation modes are provided. Operation selected depends on whether an automatic prober or a manual prober is used. The fifth operation mode gets you back to the main command level. From this point on, different operation mode selected will prompt you differently. The following will describe different modes of operation separately.

1)

Mode 1: Fully Automatic

(see below)

Mode 2: Semi Automatic — [AUTOMATIC PROBER]

(see below)

Mode 3: Semi Automatic — [MANUAL PROBER]

Mode 1, 2, 3 have similar prompts from the program. If you are running "SINGLE DEVICE MODE" (Mode 4) or "PLAYBACK ONLY" (Mode 5), go to step (8).

Fig. 7b is displayed after the number key [1], [2] or [3] is pressed.

```
            Please select one of the following options:2

[1]: NORMAL OPERATION
[2]: STORE MEASURED DATA INTO MASS STORAGE UNIT
[3]: ONLY STORE MEASURED DATA, NO CALCULATION IS PERFORMED
[4]: READ DATA FROM MASS STORAGE UNIT


Please indicate your mass storage unit: >

[F]: FLOPPY DISC
[H]: HARD DISC
[R]: RAM
F
Want to extract subthreshold parameters?  (Y/N)  >Y

Want to extract substrate current parameters?  (Y/N)  >Y
     *** Enter the substrate biases desired for substrate current extraction:
          MINIMUM BODY BIAS > 0
          MAXIMUM BODY BIAS > 5
```

Fig.7bThe second menu page of the extraction program.

The program first prompts you 4 options. [1]: NORMAL OPERATION. If you choose this option, the extraction program will measure data from HP4145, extract all the parameters and store these

parameters in the output file which will be specified in next page. But the measured data will not be stored. [2]: STORE MESURED DATA INTO MASS STORAGE UNIT. This option is identical to option 1 except that the measured data will be stored. If you choose this option you will be asked to choose the mass storage unit. The floppy disc must be in the right drive (unit #3:). For option 3, the measured data will be stored, but no parameters are extracted. You can save the data for later calculation (using option 4). If you choose option 4, the computer do not measure data, but read data from the mass storage unit you specified (FLOPPY DISC, HARD DISC, or RAM). You should have the data file ready in the mass storage unit by this time. If you choose option 4, the program will prompt you the device type. This is because no measurement is made for this option, the program has no mean to determine the device type. The user can choose if subthreshold and substrate current parameters are to be extracted. If the substrate current parameters are to be extracted, the user has to input the maximum and minimum substrate biases.

Note: For options 2, 3, and 4, the program will used the following convention for the file names of the data file. The user does not need to supply a file name for the data stored. If the device has a drawn channel width W and drawn channel length L, then the file name of the data file will be "W/L.TEXT". For example, if W = 20 μm and L = 1.5 μm, the file name will be "20/1.5.TEXT".

2) When you finish the second menu page, the screen will display the third

page as shown in Fig. 7c.

```
            ***AUTOMATIC OR SEMI-AUTOMATIC OPERATION***

Process Name=?  >
Lot=?  >
Wafer=?  >
Date=?  >
Operator=?  >
Output File=?  >
VDD(volts)=?  >          VGG(volts)=?  >          VBB(volts)=?  >
TEMPERATURE(deg. C)=?  >
TOX(angstroms)=?  >

Prober File=?  >

Probing Instructions
     The prober should be on, and the probes should be down
on the starting die, starting position.  (see prober instructions)


HIT a "C" for changes, or any other key to start.  >
```

Fig.7cThe third menu page of the extraction program for multiple devices extraction.

INPUT all information requested. If you hit "ENTER" only, default values will be used. Output file is defaulted to "bsimout.TEXT". V_{DD}, V_{GG}, and V_{BB}, the maximum applied voltages for drain, gate, and substrate during parameter extraction, are all defaulted to 5 V. Temperature is defaulted to 27 °C. T_{ox} is defaulted to 300 Å. Probe file is defaulted to "probe.TEXT". Note that the prober File **HAS TO BE** in RAM. The procedures as to how to load Prober File into RAM and how to create prober files are explained in section 2.6.

3) **HIT** any key except [C] key to start measurements and extractions.

(4) Fig. 7d is now displayed.

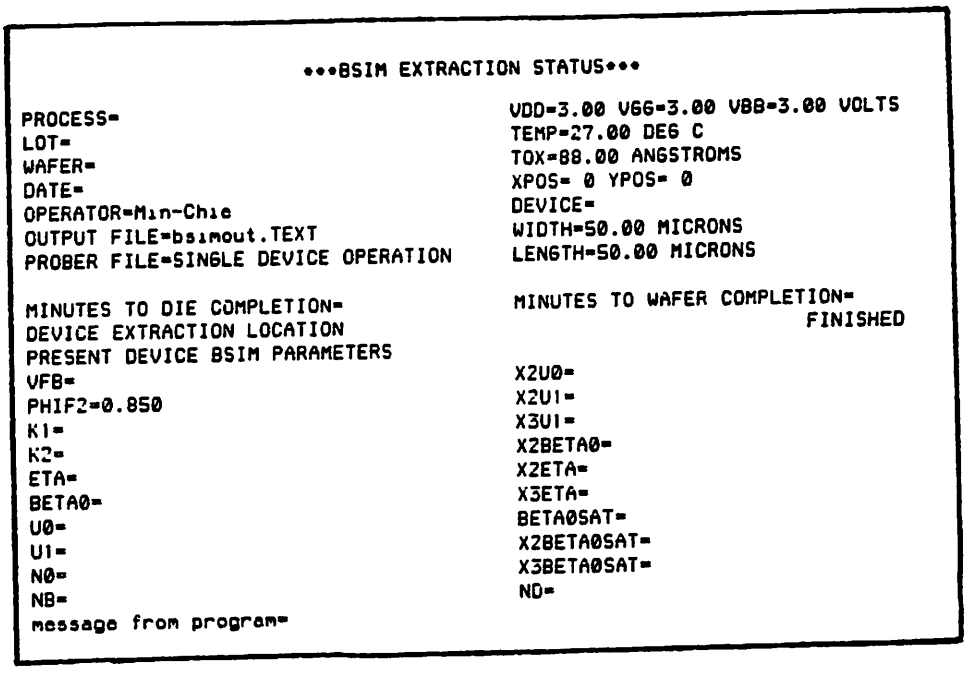


Fig.7d The display on the computer screen during measurement.

OBSERVE CLOSELY the measurements displayed on HP 4145A screen.

Before the parameters are filled with values, you will be prompted with

Are the measurements satisfactory enough to proceed?(Y/N) >

HIT [Y] key to extract parameter values.

HIT [N] key if measurements are bad. You will be prompted with

Would you like to remeasure this device?(Y/N) >

HIT [Y] key, step 4 is repeated.

5) For Mode 1 of operation, this step is skipped.

For modes 2, you should be prompted with

Are probes on next device? If so, Press "ENTER" >

For modes 3, you should be prompted with

Move the probes to the next device and Press "ENTER"

MOVE the probes to the next device

HIT [ENTER] key.

6) Steps 4-6 are repeated until there is no more device on the die to be tested.

7) For Mode 1 of operation, this step is skipped.

For modes 2 and 3, if there is no more die on the wafer to be characterized, skip this step. If there are still dies on the wafer to be characterized, you will be prompted with either

Are probes on first device of next die? If so, Hit "ENTER" >

or

Move the probes to the first device of next die then hit 'ENTER'

MOVE the probes to the first device of next die.

HIT [ENTER] key. Steps starting at 4 are repeated.

Go to step D).

8)

Mode 4

SINGLE DEVICE

Mode 5

SIMULATION USING EXISTING PROCESS FILE

For operation mode 4 and 5, Fig. 7e is displayed. Fig. 7e is similar to Fig. 7b except more information has to be provided by the user, because the probe file is not used.

```
                        ***SINGLE DEVICE OPERATION***

Process Name=?  >
Lot=?  >
Wafer=?  >          XPOSITION=?  >          YPOSITION=?  >
Date=?  >
Operator=?  >
Output File=?  >
VDD(volts)=?  >         VGG(volts)=?  >         VBB(volts)=?  >
TEMPERATURE(deg. C)=?  >
TOX(angstroms)=?  >
PHIF2 or NSUB=?  >
drawn width (microns)=?  >
drawn length (microns)=?  >
Device type=?  >    [1] enhancement, [2] zero-threshold, [3] depletion

SMU connected to DRAIN=?  >
SMU connected to GATE=?  >
SMU connected to SOURCE=?  >
SMU connected to BODY=?  >

Hit a "C" for changes or any other key to start.  >
```

Fig.7e The third menu page of the extraction program for single device extraction.

9) INPUT all information requested. Default values are the same as those for Mode 1, 2, and 3. Some more default values are: PHIF2 = 0.7, drawn width = 20, drawn length = 20, device type = 1. SMU connections are defaulted to 1, 3, 2, 4 for drain, gate, source, and substrate, respectively.

10) For mode 4

HIT any key except [C] key to start measurements and extractions. Fig. 7d is displayed on the screen. After extraction, all parameters are displayed on the screen.

For Mode 5, no measurement is made, but the program will prompt you to

input the device type as follows.

Please select device type: n-type=1 p-type=-1 >

(D) After all the devices are measured, you will now be prompted with

Would you like to view IV curves?(Y/N) >

HIT [N] key, BSIM MENU PAGE (Fig. 7a) will appear. Skip the rest of the steps.

HIT [Y] key will bring you to the graphics mode. Fig. 7f will be displayed on the screen.

```
                    ***PREPARATION FOR I-V GRAPHICS***

ENTER X DIE POSITION OF DEVICE TO BE GRAPHED= >

ENTER Y DIE POSITION OF DEVICE TO BE GRAPHED= >

                                          [1] NMOS enhancement
SELECT THE NUMBER CORRESPONDING           [2] NMOS depletion
TO THE DEVICE TYPE WHICH                  [3] NMOS zero-threshold
YOU WOULD LIKE TO GRAPH= >                [4] PMOS enhancement
                                          [5] PMOS depletion
                                          [6] PMOS zero-threshold

DEVICE WIDTH (microns) = >

DEVICE LENGTH (microns) = >
```

Fig.7f

INPUT all information requested for your desired graph. X and Y die positions of the device should be one of the "1"s or "X" specified in the probe file.

(E) Fig. 7g should now appear. If you choose "SUBSTRATE CURRENT GRAPHICS", Fig. 7h will appear.

```
                    ***BSIM I-V GRAPHICS MENU***

The BSIM I-V graphics routines will draw measured and/or simulated I-V data.
If the program is operating in the "SINGLE" mode, the 31 ELECTRICAL parameters
just extracted will be used. In the "AUTOMATIC" or "SEMI-AUTOMATIC" mode, the
31 ELECTRICAL parameters will be generated from the 96 parameter process file.

            SELECT A NUMBER FOR A GIVEN DISPLAY MODE= >
                  1)Measured Data Only
                  2)Simulated Data Only
                  3)Measured and Simulated Data

        SELECT A NUMBER FOR A GIVEN GRAPH TYPE= >
   1)IDS versus VDS     VBS=?   VGS_start=?   VGS_end=?
   2)IDS versus VGS     VDS=?   VBS_start=?   VBS_end=?
   3)log(IDS) versus VGS  VDS=?  VBS_start=?  VBS_end=?
   4)gm or rd versus VDS  VBS=?  VGS_start=?  VGS_end=?
   5)gm versus VGS      VBS=?   VDS_start=?   VDS_end=?
   6)SUBSTRATE CURRENT GRAPHICS
```

Fig.7g

```
      ***BSIM SUBSTRATE CURRENT GRAPHICS***


SELECT A NUMBER FOR A GIVEN GRAPH TYPE= >
    1)IBODY versus VGS          VBS=? >
    2)log(IBODY) versus VGS     VBS=? >
    3)log(IBODY/IDS) versus VDS VBS=? >
```

Fig.7h

After selecting your desired graph, you will be prompted with

New SMU connections?(Y/N)

HIT [Y] key will give you a chance to specify the SMU connection.

HIT [N] key, you will be given the connections you made last and you have a chance to make the right connections if they are not what the program thought they are.

(F) You should now be prompted with

PLEASE SELECT AN OPTION: >

[1]: NORMAL OPERATION

[2]: STORE I-V DATA INTO STORAGE DEVICE

[3]: READ I-V DATA FROM STORAGE DEVICE

For option 1 (NORMAL OPERATION), the program will measure the device and calculate the device characteristics. The measured playback data are not stored. For option 2, the measured playback data will be stored. The mass storage unit is fixed to the floppy disc at the left drive (unit #4:). For option 3, measured data are read from the floppy disc at the left drive.

After you have selected an option, you will be prompted

Place probes on device and Press "ENTER"

PROBE the device you want to graph.

HIT [ENTER] key when prompted by

Press "ENTER" to continue >

BE PATIENT at this point. Measurements and calculations take time.

DO NOTHING until desired graph is displayed on the screen along with a selection menu of selections that may be made about the graph. Five selections may be made about the graph selected on the screen.

(1) **Zoom Using Knob and Keys:** Activated by hitting the number key can be moved horizontally by turning the knob which is located at the upper left corner of the HP 9836 keyboard. It can also be moved vertically by pressing the [SHIFT] key and turning the knob simultaneously. To zoom a portion of the current graph, a box which encloses the portion has to be defined. Move the cross hair to a point where one of the four corners is there to be, then hit [ENTER] key to define that corner. Move the cross hair horizontally and vertically to define the box and hit [ENTER] key again to zoom the portion contained in the box. Selection menu is then displayed again along with zoomed graph.

(2) **Redraw Full Graph:** Activated by hitting the number key '2'. When a portion of a graph has been zoomed, this option can get the full scaled graph back onto the screen again, as if no zooming has ever been done.

(3) **Select New Graph for Current Device:** Activated by hitting the number key '3'. Fig. 7g is displayed again. Step E is repeated for the new graph.

(4) **Select New Device:** Activated by hitting the number key '4'. Fig. A4 is then displayed. This option is only meaningful when more than one device has been tested, i.e. when automatic or semi-automatic mode has been selected. Input all information requested by Fig. 7f about the dev-

```
                    ***BSIM PARAMETER vs. W or L GRAPH***

This graphics mode allows one to compare extracted, size-DEPENDENT parameters
from the 31-parameter ELECTRICAL file, to size-INDEPENDENT values, approximated
from the 96-parameter PROCESS file.

If you plot W on the x-axis, then L becomes the 3rd variable, and vice versa.
You may choose to plot only one third-variable value, or you may plot all of
them.  Choosing only one allows finer details to be analyzed.  The x-axis
values are scaled linear with respect to 1/EFFECTIVE SIZE.

You will choose:  1) the type of device to plot
                  2) the BSIM parameter to plot on the y-axis
                  3) whether W or L will be plotted on the x-axis
                  4) and whether all sizes or one size device will
                     be plotted for the third parameter

SELECT THE DEVICE TYPE YOU WANT TO PLOT= >
         [1] NMOS enhancement
```

Fig.7i

```
W/L ratios of devices successfully tested are listed here:
W         50.0   10.0    5.0    5.0    2.0    2.0
L          2.0    1.0    1.5    0.9    1.2    1.0









SELECT DESIRED GRAPH=? >
          [1] BSIM PARAMETER vs. W --- for all values of L
          [2] BSIM PARAMETER vs. L --- for all values of W
          [3] BSIM PARAMETER vs. W --- for single value of L.  L=? >
          [4] BSIM PARAMETER vs. L --- for single value of W.  W=? >
```

Fig.7j

```
SELECT THE PARAMETER TO BE GRAPHED= >
  [1] VFB: Flat-band voltage
  [2] 2PHIF: Surface-inversion potential
  [3] K1: Body effect coefficient
  [4] K2: Charge-sharing coefficient
  [5] ETA: Drain-induced-barrier-lowering (DIBL) effect
           at Vbs=0 & Vds=Vdd
  [6] BETA0: Low field electron mobility at Vbs=0 & Vds=0
  [7] U0: Vertical-field mobility degradation effect at Vbs=0
  [8] U1: Horizontal-field mobility degradation effect
          at Vbs=0
  [9] X2MU0
 [10] X2ETA: Sensitivity of DIBL effect to Vbs
 [11] X3ETA: Sensitivity of DIBL effect to Vds at Vds=Vdd
 [12] X2U0
 [13] X2U1
 [14] MU0SAT: High field electron mobility at Vbs=0 & Vds=Vdd
 [15] X2MU0SAT
 [16] X3MU0SAT
 [17] X3U1
 [18] N0
 [19] ND
 [20] NB
 [21] ECRIT0
 [22] ECRIT6
 [23] ECRITB
 [24] LC0
 [25] LC1
 [26] LC2
 [27] LC3
 [28] LC4
 [29] LC5
 [30] LC6
 [31] LC7
```

Fig.7k

ice to be graphed. Steps F and G are repeated to select desired graph.

(5) **Select New Character:** The character used to represent data is defaulted to "x". You can change it to any character you like by hitting the number key '4'

(6) **Exit I-V Graphics Menu:** Activated by hitting the number key displayed.

(G) After you exit IV graphics routine, you will be prompted with (except for operation mode 4 - Single Device Mode)

Would you like to view plots of BSIM PARAMETER vs W or L?(Y/N) >

If you have answered YES to the above question, you will now be prompted with Fig. 7i.

READ instructions.

HIT the number key corresponding to desired device type.

(H) Fig. 7j should appear on the screen.

HIT the number key corresponding to desired graph.

ENTER length or width value if number 3 or 4 is selected.

(I) Fig. 7k should appear.

HIT the number key corresponding to the desired parameter to be graphed.

(J) You should be prompted with

Press 'c' to make change or press 'ENTER' >

HIT [C] key to change.

HIT [ENTER] key to continue.

(K) Requested graph is now displayed along with the selection menu shown in Fig. A9. Selections '1', '2' are explained when I-V Graphics is explained in SINGLE DEVICE operation. Selection '3' brings Fig. 7k back onto the screen and steps H through K are repeated.

2.1.9 STORING PROCESS FILES ONTO A DISK

When the execution of the extraction program is completed, a process file is created in the RAM under the name you input for the prompt "output file >" when Fig. 7c or 7e was displayed. This process file has to be stored onto a disk before it can be transferred to VAX and used as input for SPICE simulation. This section describes the procedure to be followed.

(A) INSERT an initialized blank disk into the right disc drive (unit#3).

(B) MAKE SURE the main command line is displayed at the top of the screen.

HIT [F] key to invoke FILER.

HIT [F] key again to invoke Filecopy.

(C) You should be prompted with

Filecopy what file?

TYPE RAM:bsimout.TEXT (or whatever name you used for output file name)

HIT [ENTER] key.

(D) You should now be prompted with

Filecopy to what?

TYPE #3:bsimout.TEXT (or whatever you would like the file to be named on the disk)

HIT [ENTER] key.

(E) When the copying is done. you will see the FILER command line at the top of the screen and the following message beneath it.

RAM:bsimout.TEXT => V3:bsimout.TEXT (or V4:bsimout.TEXT)

NOTE that V3 or V4 is the disk directory name selected by the system. You may change the directory name by doing the following.

HIT [C] key.

You should be prompted by

Change what file?

TYPE V3: (or V4: depends on which one was shown)

HIT [ENTER] key.

You should be prompted by

Change to what?

TYPE any directory name followed by a colon (:): directory name should not exceed 5 characters.

HIT [ENTER] key.

The FILER command line should again appear at the top of the screen. To get back to the main command level

HIT [Q] key.

2.1.10 LOADING PROBE FILES INTO RAM

(A) INSERT the disk containing the prober file in the right disk drive (unit#3).

(B) MAKE SURE that you are at the main command level.

HIT [F] key to invoke FILER.

HIT [F] key again to invoke Filecopy.

(C) You should be prompted by

Filecopy what file?

TYPE #3:probe.TEXT (or the name of the prober file stored on the disk)

(D) You should now be prompted by

Filecopy to what?

TYPE RAM:probe.TEXT (or any name you would like to name the file)

HIT [ENTER] key.

(E) When the loading is completed, the FILER command line should appear at the top of the screen. And now you have the prober file in RAM ready to be used.

2.1.11 LINKING HP 9836 TO VAX

(A) To transfer file to the VAX, the RS232 Data Communication Board must be in the back of the HP9836, and it must be connected to a port selector or a modem.

(B) **INSERT** the Pascal SYSVOL Disc in the right Disc Drive.

(C) Return to the main command line of the operating system.

(D) **PRESS** [P] {this selects the permanent load operating system option.}

Load what code file ? **#3:NEWKBD**

(E) **PRESS** [X]

Execute what file? **#3:VT2**

(F) When the program is loaded, a menu will appear, and the user must load in the configuration as follows:

Main> **4** {option to creat a configuration}

Selection? **1** {VAX/UNIX}

Rate? **9600** {baud rate}

Selection? **3** {modem}

Main> **1** {go to emulator mode}

PRESS the port selector switch, and log into a VAX account as using normal procedures. The terminal type is 2648. To perform file transfer, **PRESS** CTRL and EXECUTE at the same time. This returns the program to an execution menu.

Execute> **3** {file transfer to host}

Enter host file name: **Your VAX FILE NAME**

Enter local file name: **A name like RAM:bsimout.TEXT**

Multiple files can be transferred, or the HP9836 can be used as a VAX terminal. To xit the program, select the terminate emulator option.

2.1.12 INSTRUCTIONS FOR CREATING PROBER FILES

An automatic prober file should be constructed exactly as follows. All comments are surrounded by parenthesis. while all lines should begin at the left margin. and no blank lines should be present.

```
8200                    {The x-direction die size dimension in microns}
8200                    {The y-direction die size dimension in microns}
00000000000000000000  {the first two lines should be  omitted  when }
00000000000000000000  {using Mode 3 ( semi-automatic with manual }
00000000000000000000  {probe station}
00000000000000000000
00000010000010000000
000X0000000000000000
00000000000000000000  {This is the 20 by 20 array to designate the}
10000100000100000000  {die which are to be tested. All locations}
00000000000000000000  {marked with a "1" will be probed. and the}
00000000000000000000  {location marked with "X" is the origin die}
00000001000000010000
00000000000000000000  {The upper left corner of the array represents}
00000000000000000000  {location(1,1). and the x-values increase towards}
10000000001000000000  {the right. while the y-values increase towards}
00000000000000000000  {the bottom}
00001000000100000000
00000000000000000000  {In this example. the origin is (4,6)}
00000000000000000000
00000000000000000000
10000000010000000000
**                  {Device to device delimiters}
mx=800              {Distance in x-direction from die origin to device}
my=920              {Distance in y-direction from die origin to device}
w=50                {Width of device #1 in microns}
l=1                 {Length of device #1 in microns}
dt=1                {Device #1 is a NMOS device}
ed=1                {Device #1 is an enhancement device}
sd=1                {Drain of device #1 is connected to SMU 1}
sg=2                {Gate of device #1 is connected to SMU 2}
ss=3                {Source of device #1 is connected to SMU 3}
sb=4                {Substrate of device #1 is connected to SMU 4}
**
```

{Values for the next device are listed here. if multiple devices will be analyzed.}

{The last line of the prober file must be "**"}

{The devices can be in any order. and all device descriptions such as "mx=800"}

{can be in any arbitrary sequence. as long as they are all included}

Acceptable device description values are:

mx,my {in microns}

w,l {in microns}

dt {1=NMOS, -1=PMOS}

ed {1=enhancement, 0=zero-threshold, and -1=depletion}

sd,sg,ss,sb {any value from 1 to 4 which is not the value of}

" {any one of the remaining three}

2.2 Measurement Procedure

The measurement routine of the extraction program is composed of four parts: device type test, device functionality test, drain current measurement for drain current parameter extraction, and substrate current measurement for substrate current parameter extraction. Because it is important to find out the devices with abnormal behaviors before useful data can be stored, a device functionality test will always be performed before HP 4145 measures the drain current or the substrate current to help eliminating faulty devices, which may cause large errors in the final process files. Both the device-type test and device functionality test are based on the user specified maximum voltage (V_{DD}) to set up HP4145.

2.2.1 Device-Type Test

In the following paragraph, we will describe the basic principle used in the parameter extraction program to determine the device type (n-channel or p-channel) of the device under measurement. In an MOS transistor, the source/drain and substrate junctions are p-n junctions. The polarity of this p-n junction is a good indication of the device type. The initial HP4145 measurement setup measures the forward- and reverse-biased current of the source/drain-body junctions to find out the polarity of this p-n junction and determine the device type. Also, HP4145 measures the gate current at different biases to checks possible short-circuits between the gate and other terminals.

In the test, the source and drain are connected to the ground, and the gate is connected to V_{GG}, the maximum applied gate voltage specified by the user. Two voltages ($+V_{BB}$ and $-V_{BB}$) are applied to the substrate successively. V_{BB} is the maximum applied substrate voltage specified by the user. Junction currents (I_{bpos} and I_{bneg}) and gate currents (I_{gpos} and I_{gneg}) corresponding to these two substrate biases are measured. The Biasing conditions for an NMOS device in this test is shown in Fig. 8a. A sample of the test results displayed on HP4145 in shown in Fig. 8b. Notice that the gate potential is set at a different level with the source, drain and substrate. Any short-circuit between the gate and any of the other terminals can be detected. To find out the polarity of the

source/drain-body junctions, I_{bpos} and Ib_{neg} are compared to some pre-determined current level (e.g. 10 μA).

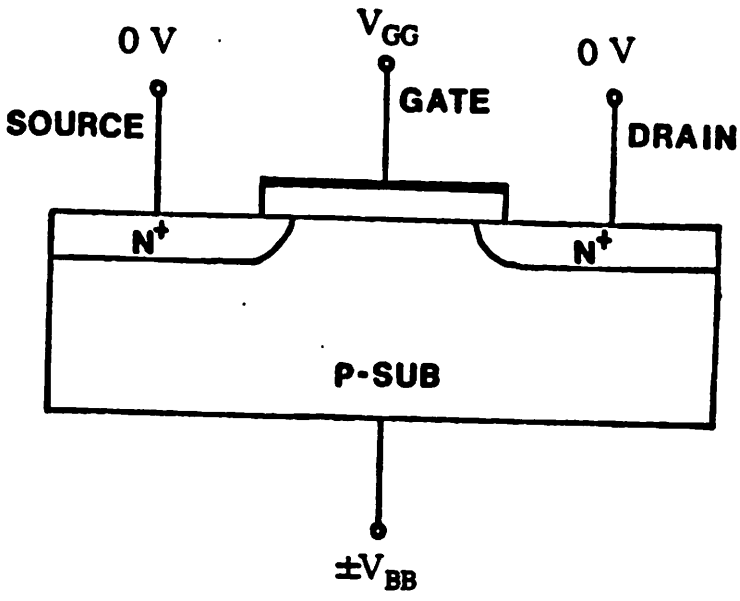


Fig.8a The baising condition on an N-channel device in the Device Type Test. For a P-channel device, the polarity of all biases are reversed.

******* LIST DISPLAY *******

VBODY = [-5.0000V to 5.0000V in 10.000V step] = [to] in step

LINE 0			
VBODY	IBODY	IGATE	
-5.0000V	-7.700pA	-5.300pA	
5.0000V	12.32mA	-5.300pA	

Fig.8b A sample of the display on HP4145 screen for the Device Type Test.

A list of all possible results are summarized below.

(1) If $I_{bpos} > 10\mu A$ and $I_{bneg} < 10\mu A$, then the device is an N-channel device.

(2) If $I_{bpos} < 10\mu A$ and $I_{bneg} > 10\mu A$, then the device is a P-channel device.

(3) If both I_{bpos} and I_{bneg} are less than $10\mu A$, then it is open-circuited between the source/drain and the substrate.

(4) If both I_{bpos} and I_{bneg} are larger than 10 μA, then it is short-circuited between the source/drain and the substrate. (Note: because of this test, devices with butting source and substrate contact will be rejected.)

(5) If either I_{gpos} or I_{gneg} is larger than $0.1\mu A$, then the quality of the gate oxide is not within the acceptable range.

Should any error occurs, the program will pause and wait for user's action. The user can readjust the probes or skip the present device according to the error message. The reference current levels used to judge device behaviors are set for the purpose of detecting obvious shorts and are not intended for screening junction or gate leakage currents. For devices with large leakage currents and/or high noise level, these reference levels should be changed accordingly.

2.2.2 Device-Functionality Test

If no error is detected in the Device-Type-Test procedure, the device type determined in the previous step is used to set up HP4145 in determining the device functionality. The test principle is based on the relationship between the drain saturation current and the gate voltage. The bias condition for an NMOS transistor in this test is illustrated in Fig. 9a. A sample of the test results is shown in Fig. 9b. For this test, the source is connected to ground potential, the drain voltage is set to V_{dd}, and the gate is set to 0 V and V_{GG}. For a P-channel device, the polarity of the voltages are reversed accordingly. Two drain current values, I_{DS0} and $I_{DSv_{gg}}$, corresponding to the two gate voltages are measured.

Syndromes of the device functionality test are listed below.

(1) If $I_{DS0} > 0.95I_{DSvgg}$, it indicates a short circuit between the source and drain.

(2) If $I_{DS0} < 10(W_{drawn}/L_{drawn})$ μA, it indicates an open circuit between the source and the drain.

(3) Otherwise, the device is functional.

where W_{drawn} and L_{drawn} denote the drawn channel width and length, respectively. If an error occurs, the program will pause and wait for the user's action as in last test.

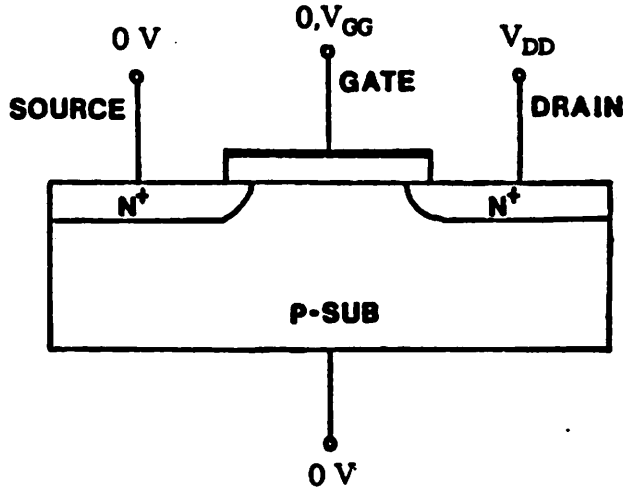


Fig.9a The baising condition on an N-channel device in the Device Functionality Test.

****** MATRIX DISPLAY ******

VGATE = .0000V to 5.0000V in 5.0000V step
= to in step

[IDRAIN MEASUREMENT]

LINE 0			
VGATE			
.0000V	105.9nA		
5.0000V	10.39mA		

Fig.9b A sample of the display on HP4145 screen for the Device Functionality Test.

2.2.3 Device Measurements for Parameter Extraction

If the device is functional, HP4145 is set to measure I_{DS}-V_{GS} data with V_{BS} as the Z-axis variable. V_{GS} is chosen to increase from 0 to V_{GG} with voltage steps ranging from 0.05 to 0.5 volt depending on the magnitude of V_{GG} (V_{Gstep} = 0.05 V for $V_{GG} \leq$ 3V, V_{Gstep} = 0.1 V for $V_{DGG} \leq$ 5V, V_{Gstep} = 0.2 V for $V_{GG} \leq$ 10 V, V_{Gstep} = 0.5 V for $V_{GG} \geq$ 10 V) V_{BS} is equally spaced between 0 and V_{BB}. V_{DS} is biased at four values: 0.05, $0.4V_{DD}$, $0.9V_{DD}$, and V_{DD}. Data with V_{DS} = 0.05 are used to extract linear-region parameters. The data with $V_{DS} = 0.9V_{DD}$ and V_{DD} are used to extract saturation-region parameters. After the 17 strong-inversion parameter values are extracted, the data with $V_{DS} = 0.4V_{DD}$ are used to refine these parameters. This refinement procedure improves the global accuracy of the calculated characteristics by fine tuning the parameter set. At each V_{DS} bias, thirty I_{DS} values corresponding to five equally spaced V_{GS} between the threshold voltage and V_{DD}, and six V_{BS} values are measured. They are stored in a three-dimensional array, and are passed to the extraction routine for parameter extraction.

The first V_{GS} value of the five equally-spaced biases is selected to be 5 V_{Gstep} above the threshold. This is to ensure that none of the stored I_{DS} data is in the subthreshold region. Depending on the fabrication process and bias conditions, the threshold voltage of MOS transistors may widely distributed below V_{DD}. A threshold current level is used to determine the approximate threshold voltage. The threshold current is set to be

$$I_{threshold} = 0.1 \frac{W_{drawn}}{L_{drawn}} \mu A \qquad \text{for } V_{DS} = 0.1 \text{ and } 0.2 \text{ Volt} \qquad (101)$$

$$I_{threshold} = 0.1 \left(\frac{V_{DS}}{0.1}\right)^{1/2} \frac{W_{drawn}}{L_{drawn}} \mu A \qquad \text{for } V_{DS} = 0.4V_{DD}, 0.9V_{DD} \text{ and } V_{DD} \qquad (102)$$

During the measurement, I_{DS}-V_{GS} curves are displayed on the screen of HP4145. If the user does not satisfy the quality of measured data, re-measurement of the device is possible at the end of each measurement. A typical display on the screen of HP4145 during the measurement is shown in Fig. 10.

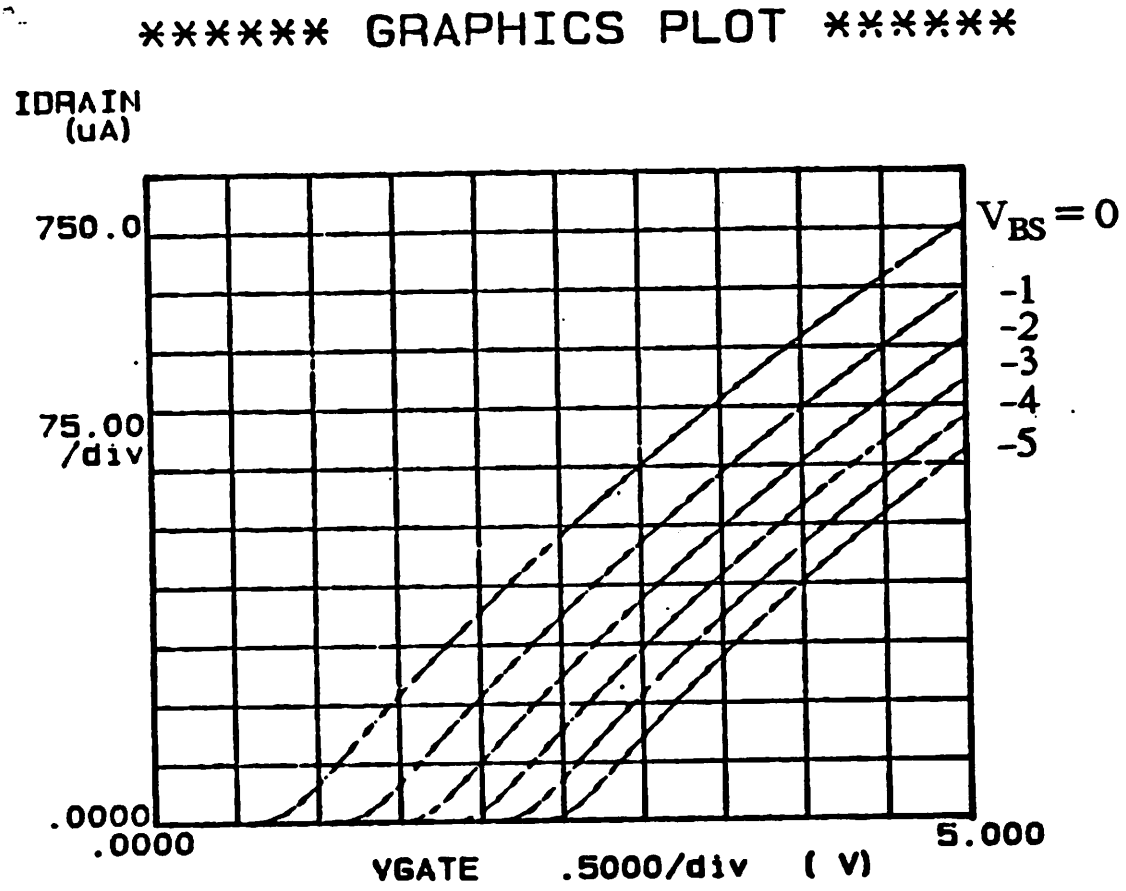


Fig.10A typical $I_{DS}-V_{GS}$ display on the screen of HP4145 during measurement. The six curves are corresponding to six substrate biases.

2.2.4 Measurement for Subthreshold Parameter Extraction

After the 17 strong-inversion parameters have been extracted, the user has the option to extract subthreshold parameters. The setup of HP4145 for subthreshold measurement is the same as that for strong inversion except that V_{GS} scans from V_{th} - 0.5 to V_{th} - $3V_{tm}$ and y-axis scale is set to be logarithmic rather than linear. V_{th} is calculated from the strong inversion parameters just extracted. The gate bias range is selected to be such that the device is operating in the subthreshold region, and the measured drain current is above noise level. Since threshold voltages are different for different drain and substrate biases, the biasing condition of HP4145 has to be set up again every time V_{DS} or V_{BS} changes. The subthreshold swing coefficient n for two drain biases (V_{DS} = 0 and V_{DD}) and two substrate biases (V_{BS} = 0 and $-V_{DD}$) are calculated from the measured subthreshold current. The subthreshold swing coefficient n is calculated according to the following equation.

$$n = \frac{1}{V_{tm}\ln(10)} S \tag{103}$$

where S is the subthreshold swing defined by

$$S = \ln(10) \frac{dV_{GS}}{d[\ln(I_{DS})]} \tag{104}$$

The slope of the transistor characteristics in the subthreshold region is the subthreshold swing S.

2.2.5 Measurements for the Substrate Current Parameter Extraction

Bias voltages to extract the substrate parameters have been carefully chosen so that leakage currents do not affect extraction. Extraction is performed for 5 V_{gs} values (starting at V_{th} + 0.3 to a maximum value of $0.8V_{dd}$ in equal increments, with V_{th} taken at maximum substrate bias) and four user-selectable V_{bs} values, in addition to measuring at a low gate bias ($V_{gs} - V_{th}$ = 0.3) for each V_{bs} value. A total of 48 measurements of both I_{bs} and I_{ds} are done in a time span of approximately 5 minutes.

Recently, an empirical method to determine V_{dsat} was proposed by T.Y. Chan et al

[13]. In that work, it was found that parallel contours could be plotted by plotting constant $\frac{I_{bs}}{I_{ds}}$ curves in I_{ds} - V_{ds} space (Fig. 11). V_{dsat} could then be found simply by noting the $\frac{I_{bs}}{I_{ds}}$ contour that intersected the origin at $I_{ds} = 0$ and $V_{ds} = 0$, and extracting the V_{ds} value at which this contour intersected the normal I_{ds} - V_{ds} graph.

The approach taken in the actual extraction process is as follows. A specific $\frac{I_{bs}}{I_{ds}}$ current contour is chosen internally by the program. Presently, this current ratio is set at 0.2 decades below the $\frac{I_{bs}}{I_{ds}}$ value at $V_{ds} = V_{dd}$ and maximum gate bias, or at 10^{-5}, whichever is smaller. This procedure ensures that a current ratio is chosen such that leakage current is not substantial, yet is less than the maximum current ratio measured at $V_{ds} = V_{dd}$ for all gate biases. Next, $\frac{I_{bs}}{I_{ds}}$ is measured at $V_{gs} - V_{th} = 0.3$ (Fig. 12). At this low gate bias, V_{dsat} is approximately $V_{gs} - V_{th}$. The program then notes the V_{ds} value at which the measured $\frac{I_{bs}}{I_{ds}}$ value is equal to the previously set current ratio. An offset voltage $V_{doffset}$ is then found by taking the difference between this drain voltage and $V_{dsat} \approx V_{gs} - V_{th}$ (Fig. 12). Now, because the $\frac{I_{bs}}{I_{ds}}$ current ratio contours are parallel for all gate bias, V_{dsat} for other gate voltages can be found simply by noting the V_{ds} value at which $\frac{I_{bs}}{I_{ds}}$ is equal to the preset current ratio, and then subtracting $V_{doffset}$ from it. This difference will equal V_{dsat} for that specific gate voltage. Generally, $V_{doffset}$ changes for different body biases so that the low gate bias measurement must be done for each body bias value.

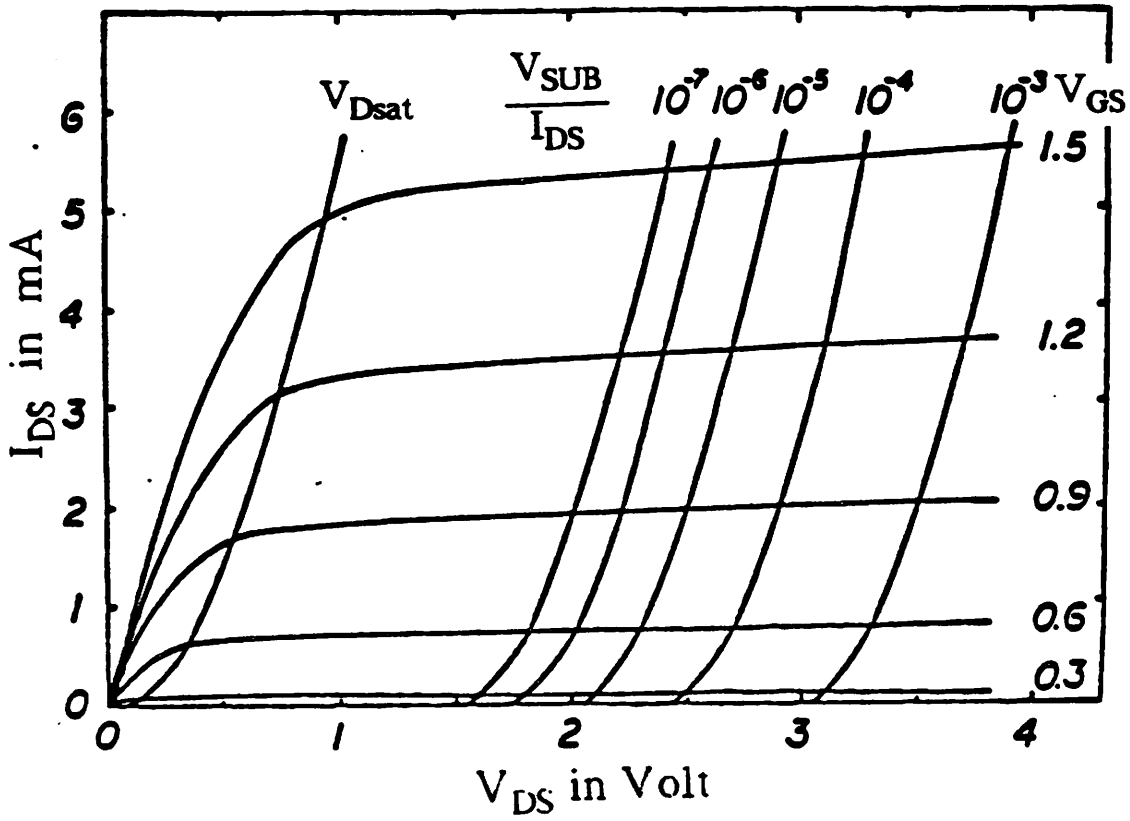


Fig.11Parallel contours of constant I_{BS}/I_{DS}.

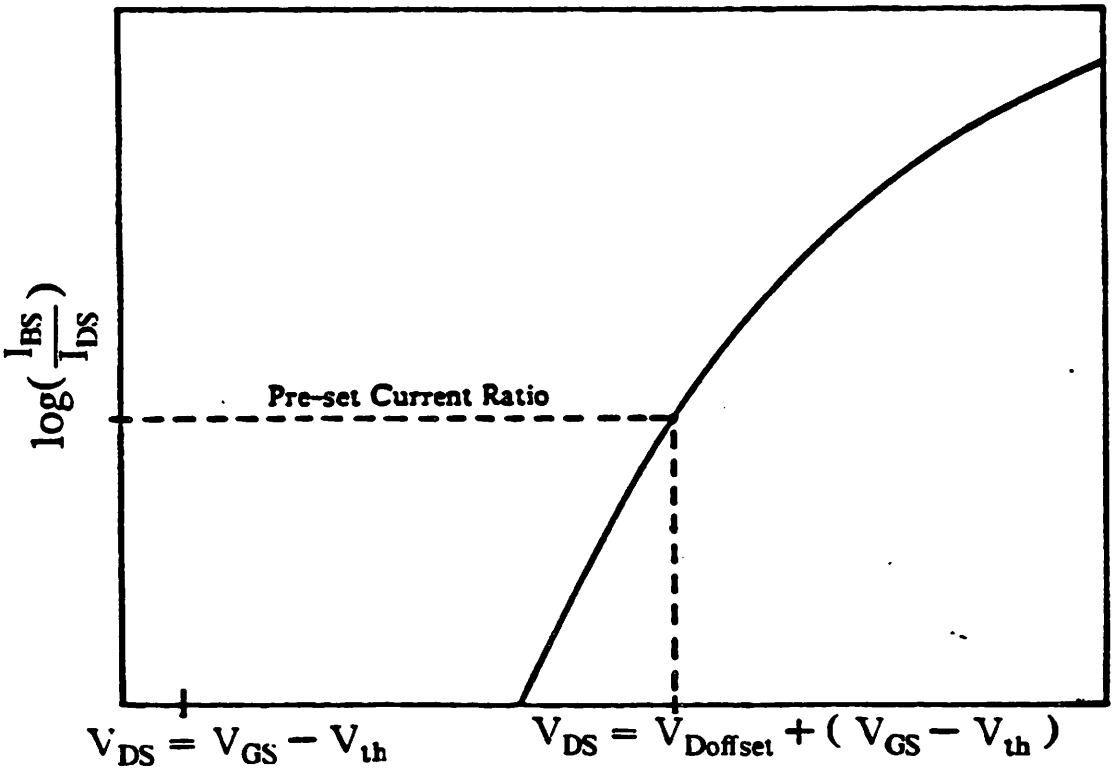


Fig.12Calculation of $V_{Doffset}$.

2.3 Parameter Extraction Theory and Algorithms

This routine is the most sophisticated portion of the BSIM parameter extraction program. It uses the I_{DS}–V_{GS} data stored in the measurement routine to extract 20 drain current parameters and I_{BS} data to extract 11 substrate current parameters. The extraction routine can be divided into five parts: linear-region parameter extraction, saturation-region parameter extraction, parameter refinement, subthreshold parameter extraction, and substrate current parameter extraction.

Nonlinear global optimization is the most common technique used in many parameter extraction system. However, because of the complicated physics involved in small geometry effects, many transistor parameters are inevitably correlated which makes the global optimization process difficult to converge and the optimum parameter set nonunique. What even worse is that the extracted parameter values are not physical, which makes the interpolation or extrapolation of parameter values from known parameter set extremely difficult. Also, as the number of parameters in a model increases, the global optimization technique becomes highly inefficient. Because the computation time increases superlinearly with the number of parameters. Several modified algorithms, like Levenberg-Marquardt method, modified Gauss method, the steepest descent method, or combinations of these methods are commonly adapted to help finding the global minimum error and to expedite the convergence.

To ensure that the physical parameters acquire meaningful values, physics based parameter extraction methods together with local parameter optimization routines are employed. This algorithm is simpler than most of the parameter optimizers published in the literature, avoids most problems encountered by global optimization, yet retains the required accuracy. Newton-Raphson's iteration and the linear least square fit are the only numerical procedures needed throughout the extraction routine. As a result of the simplicity of the extraction method, no special algorithm is required to enhance the convergence which makes the implementation of the extraction system easy and straight forward.

This local optimization technique has several advantages over the global optimization technique. First, physical parameters rather than model parameters are extracted that makes the extracted parameter values more physical and reliable. Because physical parameters are usually not as strongly coupled as model parameters, extracting only fundamental parameters avoids the problem of parameter redundancy which are crucial in the global optimization technique. Parameter redundancy makes the global optimization process difficult to converge and the extracted parameter values non-physical. Secondly, the algorithms involved in local optimization technique are much simpler than those in the global optimization technique. This makes the implementation easy and straight forward. Also, because less parameters are to be determined in each optimization procedure, the calculations are simple and fast. In our case, the total computation time is less than one minute on an HP9836 desk-top computer. Thirdly, the effect of a particular device measured on the final parameter set can be identified. Since each device is extracted individually, abnormal device parameter values can be easily observed. This feature provides the user higher confidence when the parameter set is used to predict device characteristics with other dimensions.

Instead of minimizing the global errors, that put all model parameters into the optimization process, this extraction method only extract some physical parameters at a time under different bias conditions by local optimization. The optimization algorithm is a combination of Newton-Raphson's iteration and the linear least square fit with one or two variables. The flowchart of the algorithm is shown in Fig. 13.

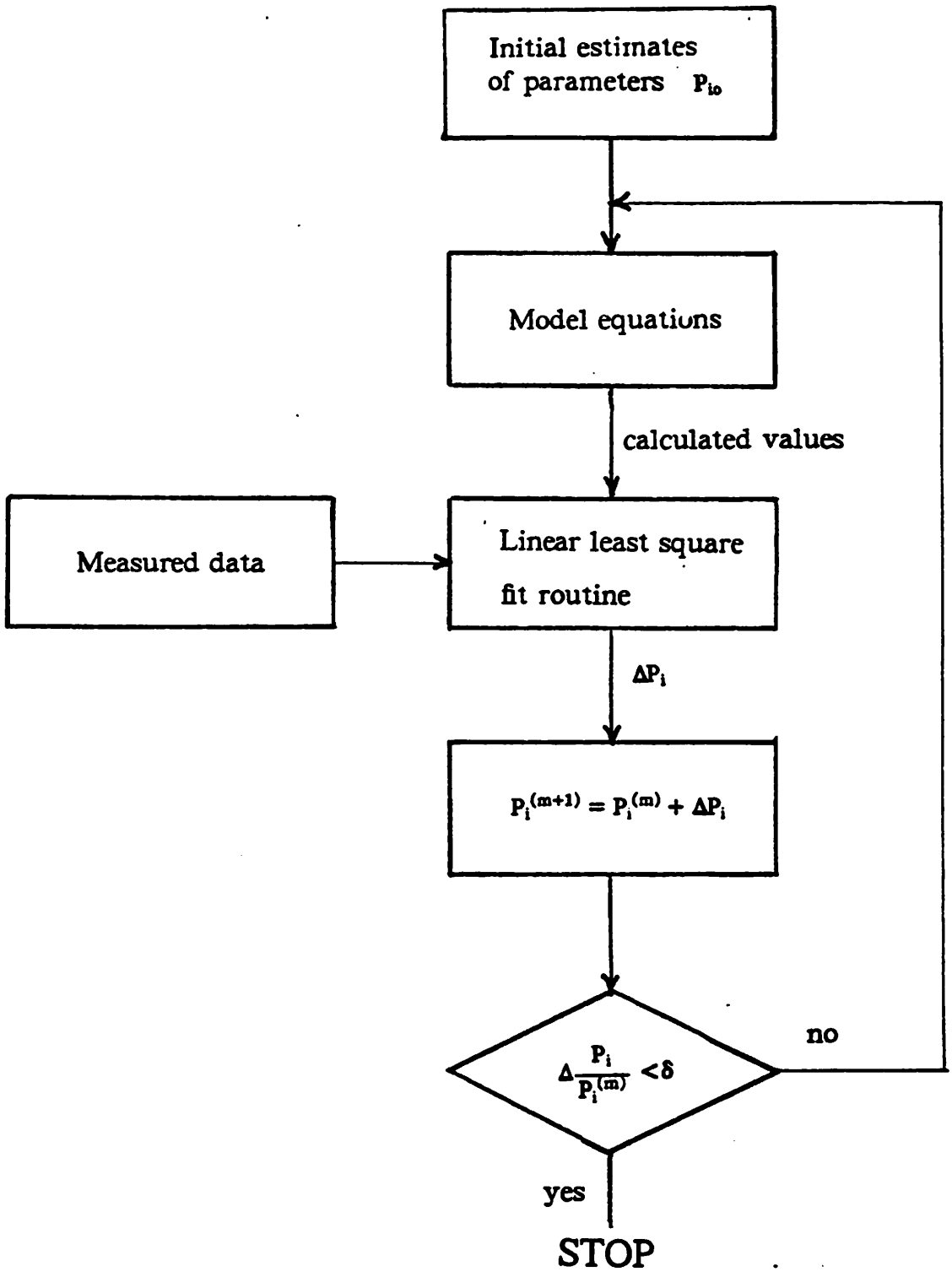


Fig.13 The flowchart of the combined Newton-Raphson's iteration and the least square fit routine.

Let f (P_1, P_2, P_3) be the function to be optimized, P_1, P_2 and P_3 be the fitting parameters, and P_{1o}, P_{2o} and P_{3o} be the true parameters. The Newton-Raphson's iteration algorithm gives

$$f(P_{1o},P_{2o},P_{3o}) - f(P_1^{(m)},P_2^{(m)},P_3^{(m)}) = \frac{\partial f}{\partial P_1}\Delta P_1^{(m)} + \frac{\partial f}{\partial P_2}\Delta P_2^{(m)} + \frac{\partial f}{\partial P_3}\Delta P_3^{(m)} \quad (105)$$

where the superscript (m) denotes the m^{th} iteration. To make (105) ready for the linear least square routine (a form of $y = a + bx_1 + cx_2$), we have to divide both sides of (105) by $\partial f/\partial P_1$. The measured data are then fitted to (105) through the least square fit to calculate ΔP_is. The parameters for the $(m+1)^{th}$ iteration are given by

$$P_i^{(m+1)} = P_i^{(m)} + \Delta P_i^{(m)} \qquad i = 1,2,3 \quad (106)$$

The iteration ends when ΔP_is are smaller than some pre-determined values. In real case, $f(P_{1o},P_{2o},P_{3o})$s may be the measured I_{DS}–V_{GS} data at fixed drain and substrate voltages over various gate biases and $f(P_1^{(m)},P_2^{(m)},P_3^{(m)})$s are the calculated values using parameters obtained from previous iteration. This procedure is repeated for different drain and substrate voltages. Finally, the extracted physical parameters at different drain and substrate voltages are fitted to some empirical equations as in (2), (14-18), (28), and (38-41) by the linear least square routine again to calculate the model parameters. The flowchart of the extraction procedure is shown in Fig. 14.

* Note: in the extraction program, the parameter variables which account for the sensitivity to V_{BS} are prefixed by "x2", and the sensitivity to V_{DS} are prefixed by "x3". For example, η_B is denoted by x2eta; β_{SD} is denoted by x3beta0sat. The suffix "sat" means that this parameter is extracted in the saturation region.

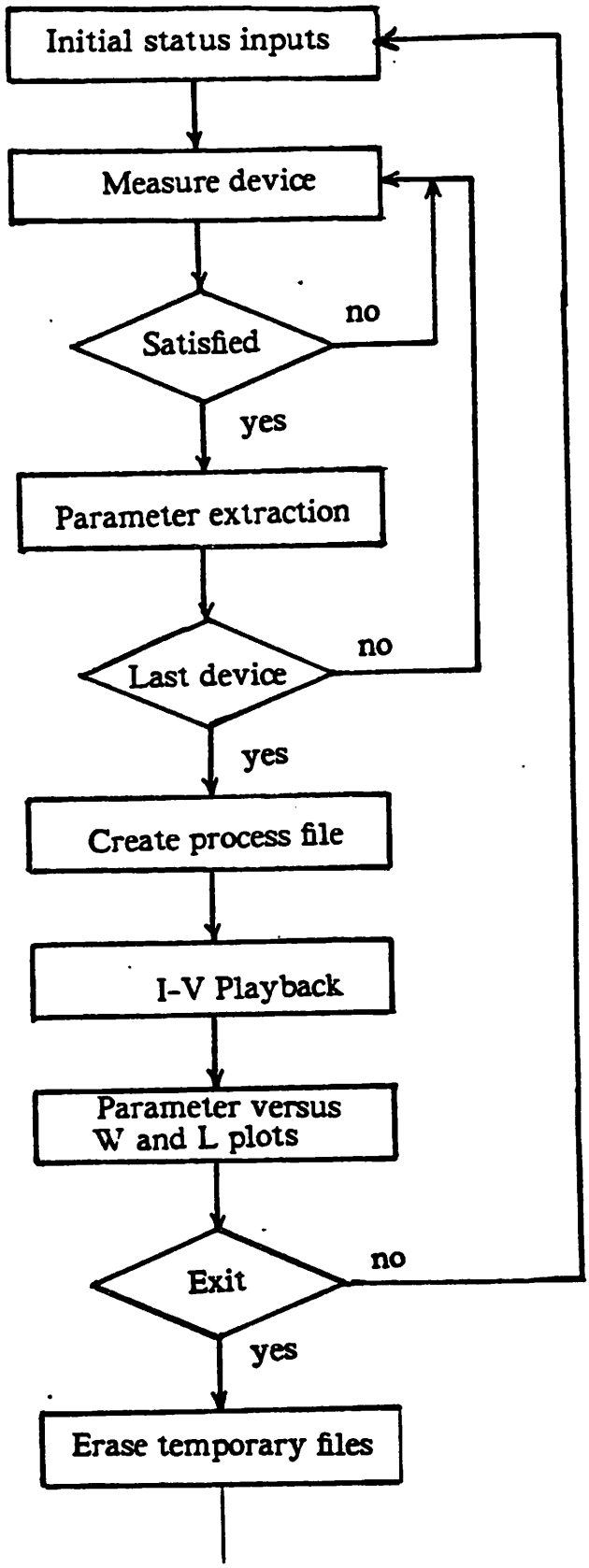


Fig.14a The flowchart of the parameter extraction program.

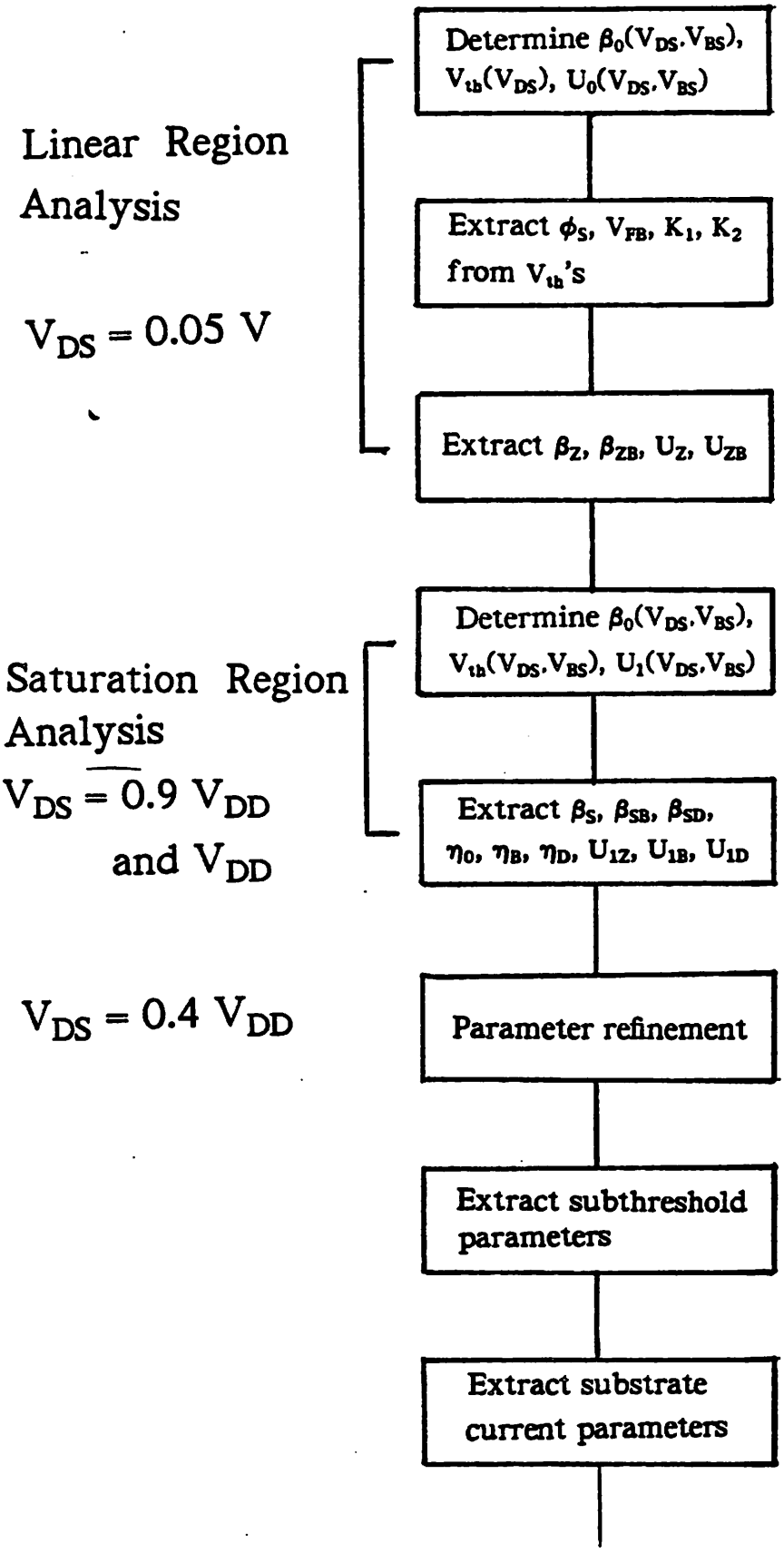


Fig.14b The flowchart of the parameter extraction routine.

2.3.1 Linear-Region Analysis

In this part, six parameters are extracted, namely, V_{FB}, ϕ_{2f}, K_1, K_2, U_0, and U_{0B}^*. The data used in this procedure are I_{DS} values stored in the measurement routine corresponding to V_{DS} = 0.05 V.

(a) Procedure "Linear-Region-Extraction"

This procedure calculates β_0, V_{th} and U_0 in the linear region. The drain current in the linear region can be rearranged as

$$f(\beta_0, V_{th}, U_0) = \beta' - \frac{G + G\,U_0\,(V_{GS} - V_{th})}{V_{GS} - V_{th} - \frac{a}{2} V_{DS}} = 0 \tag{107}$$

where $G = I_{DS}/V_{DS}$ is the measured channel conductance and $\beta' = \beta_0 / (1 + U_1 V_{DS})$. Substituting (107) into (105) gives

$$f(\beta_0^{(m)}, V_{th,lin}^{(m)}, U_0^{(m)}) = -\Delta\beta'^{(m)} + \frac{\partial f}{\partial V_{th}} \Delta V_{th,lin}^{(m)} + \frac{\partial f}{\partial U_0} \Delta U_0^{(m)} \tag{108}$$

where

$$\frac{\partial f(x)}{\partial V_{th}} = \frac{G\,(1 + U_0 \frac{a}{2} V_{DS})}{(V_{GS} - V_{th} - \frac{a}{2} V_{DS})^2} \tag{109}$$

$$\frac{\partial f(x)}{\partial U_0} = \frac{G\,(V_{GS} - V_{th})}{V_{GS} - V_{th} - \frac{a}{2} V_{DS}} \tag{110}$$

$V_{th,lin}$ represents the threshold voltage in the linear region. Eq. (108) is used in Newton-Raphson's iteration.

To solve (108), initial values of β, U_0 and V_{th} have to be provided in the Newton-Raphson's iteration. These initial values are calculated by the program automatically before the iteration using stored I_{DS}–V_{GS} data obtained in the measurement routine.

Five measured I_{DS}–V_{GS} data with V_{DS} = 50 mV and V_{BS} fixed are least-square-fitted to (108) to obtain the parameter increments for next iteration. At the end of each iteration, a test is made to detect whether convergence has been reached. The iteration is ended when the increments of all three parameters are less than 0.001% of the parameter values

in previous iteration or when a maximum allowable number of iteration is encountered. The above procedure is repeated for six V_{BS} values. The extracted values of U_0 and β_0 at different substrate biases are fitted to (14) and (17), respectively, to calculate U_{0Z}, U_{0B}, β_Z, and β_{ZB}.

If the maximum number of iteration is encountered, a warning message

"WARNING: NEWTON-RAPHSON'S ITERATION MAY NOT CONVERGE"

will be displayed on the screen. When this happens, it does not necessarily mean that the extracted parameters are not valid. The users should check the playback characteristics to determine the validity of the parameter set.

2.3.2 Extraction of Surface Potential and Body-Effect Coefficients

The name of this procedure in the extraction program is "large_device_20f_extraction" which is mutually exclusive with another similar procedure named "linear_region_threshold_analysis". Only one of these two procedures will be called during the extraction. The parameter ϕ_S is a direct measure of the effective channel doping concentration, and its accurate determination is essential for process analysis. Therefore two procedures were used in the extraction program to achieve high accuracy. The threshold voltage extracted in the previous procedure for different substrate biases are used to extract another three parameters, K_1, K_2, and V_{FB}. The procedure "large_device_20f_extraction" is called when the program is measuring the first device or the largest device in the probe file. The parameter ϕ_S is extracted from measured data. The procedure "linear_region_threshold_analysis" is called for all other devices in the probe file. This procedure uses the value of while in the latter procedure, it uses the ϕ_S value extracted from procedure "large_device_20f_extraction". It is recommended that the user always specify the largest device as the first measured device in the probe file.

For a large dimension device, K_1 is equal to

$$K_1 = \frac{\sqrt{2q\epsilon_{si}N_A}}{C_{ox}} \tag{111}$$

and ϕ_S is equal to

$$\phi_S = 2\,V_{tm}\ln\left(\frac{N_A}{n_i}\right) \tag{112}$$

where n_i is the intrinsic carrier concentration in silicon. Substituting (112) into (111), K_1 can be expressed as a function of ϕ_S, i.e.

$$K_1 = \frac{\sqrt{2\,q\,\epsilon_{si}.n_i\,e^{\frac{\phi_S}{2V_{tm}}}}}{C_{ox}} \tag{113}$$

Applying Newton-Raphson's algorithm to (1) gives

$$V_{th} - V_{th}^{(m)} = \Delta\,V_{FB} + \frac{\partial V_{th}}{\partial \phi_S}\Delta\,\phi_S + \frac{\partial V_{th}}{\partial K_2}\Delta\,K_2 \tag{114}$$

where

$$\frac{\partial V_{th}}{\partial \phi_S} = 1 - K_2 + \frac{K_1}{2\sqrt{\phi_S - V_{BS}}} + \frac{K_1}{4\,V_{tm}}\sqrt{\phi_S - V_{BS}} \tag{115}$$

and

$$\frac{\partial V_{th}}{\partial K_2} = -(\,\phi_S - V_{BS}\,) \tag{116}$$

Where $V_{th}^{(m)}$ is the calculated threshold voltage for the m^{th} iteration and V_{th} is the threshold voltage obtained from (108), and $\partial V_{th}/\partial$phis is evaluated with K_1 expressed by (113). The final values of VFB, ϕ_S, and K_2 can be calculated through a similar procedure as used in (108). For devices other than the largest one, VFB, K_1, and K_2 are calculated by fitting V_{th} obtained in (108) to (1) using $\sqrt{\text{phis} - V_{BS}}$ and phis $- V_{BS}$ as controlling variables as in (117).

$$V_{th} - \phi_s = (\,V_{FB} - \eta\,V_{DS}) + K_1\sqrt{\phi_S - V_{BS}} - K_2\,(\,\phi_S - V_{BS}\,) \tag{117}$$

For the single device operation mode, only procedure "linear_region_threshold_analysis" will be called, therefore the value of ϕ_S has to be specified by the user in the initial input menu page.

2.3.3 Linear-Region Parameter Validity Check

After parameters V_{FB}, ϕ_S, K_1, K_2, U_0, and U_{0B} have been calculated, they are checked against certain bounds. These bounds is set based on physical considerations. If any of these parameters is not within the bounds, all the parameters will not be saved and the program skip to the next device. Also, an error message will be displayed on the screen to notify the user.

The bounds of the linear region parameters are listed below:

$-5.0 \leqslant V_{FB} \leqslant 1.0$

$0.2 \leqslant \phi_s \leqslant 1.5$

$0 \leqslant K_1 \leqslant 5.0$

$-1.0 \leqslant K_2 \leqslant 1.0$

$-2.0 \leqslant U_0 \leqslant 2.0$

$-2.0 \leqslant U_{0B} \leqslant 2.0$

2.3.4 Saturation-Region Analysis

The I_{DS} data measured corresponding to $V_{DS} = 0.9V_{DD}$ and $V_{DS} = V_{DD}$ are used to determine U_1, β, and V_{th} in the saturation region. Calculations are based on (19).

$$I_{Dsat} = \frac{\beta (V_{GS} - V_{th})^2}{2 a K} \tag{19}$$

Expressing (19) in a form suitable for Newton-Raphson's iteration gives

$$I_{DSmeas} - I_{DSsim}^{(m)} = \frac{\partial I_{DSsim}}{\partial \beta_0} \Delta \beta_0^{(m)} + \frac{\partial I_{DSsim}}{\partial V_{th}} \Delta V_{th,sat}^{(m)} + \frac{\partial I_{DSsim}}{\partial U_1} \Delta U_1^{(m)} \tag{118}$$

where

$$\frac{\partial I_{DSsim}}{\partial \beta_0} = \frac{I_{DSsim}}{\beta_0} \tag{119}$$

$$\frac{\partial I_{DSsim}}{\partial V_{th}} = \frac{I_{DSsim} U_1 T}{a (1 + U_0 V_{GT})} + \frac{I_{DSsim} U_1 T}{a K} - \frac{2 I_{DSsim}}{V_{GT} (1 + U_0 V_{GT})} \tag{120}$$

$$\frac{\partial I_{DSsim}}{\partial U_1} = \frac{I_{DSsim} V_{GT} T}{a K} \tag{121}$$

$$T = 0.5 + \frac{0.5}{\sqrt{1 + 2 V_c}} \tag{122}$$

$V_{GT} = V_{GS} - V_{th}$. I_{DSmeas} is the measured drain saturation current. I_{DSsim} is the calculated

drain saturation current evaluated from (19). Similarly, a combined Newton-Raphson's iteration and linear least square routine is applied to (118) again at different substrate biases. The values of U_1, β_0, and V_{th} in the saturation region can be obtained. The initial values of β_0 and V_{th} in this procedure are calculated from the linear-region parameters. The initial value of U_1 is set to zero. A convergence test is performed at the end of each iteration to determine whether the required accuracy is observed. If the maximum number of iteration is reached, a warning will be displayed on the screen.

The threshold voltage extracted in the saturation region is compared with those extracted in the linear region to calculate the parameter η through (123).

$$\eta(V_{DS},V_{BS}) = \frac{V_{th,sat} - V_{th,lin}}{V_{DS}} \tag{123}$$

The final values of β_0, η, and U_1 at various V_{BS} and V_{DS} biases are fitted to (18), (2), and (15), respectively, to calculate η_0, η_B, η_D, β_S, β_{SB}, U_{1Z}, U_{1B}, and U_{1D}.

2.3.5 Saturation-Region Parameter Refinement

For very short-channel devices, hot-electron effects become appreciable at large drain voltages. The extracted parameters in the saturation region may not be correct throughout the drain voltage operating range due to the substrate current component in the measured data. Without this procedure, what usually happens for very short-channel devices is that the calculated drain currents match very well at small and large drain voltages, but introduce large errors at intermediate drain voltage values. This is because local optimization was used only at low and high V_{DS} values. This procedure uses the I_{DS}–V_{GS} data measured at $V_{DS} = 0.4V_{DD}$ to iterate ten times with the parameters obtained in previous iteration. This improves the global accuracy of fit.

To minimize the calculation time, only U_1 and β_S which are most sensitive to the hot-electron effects, are refined in this procedure. The procedure consists of two parts. The first part corrects the U_1 parameters. The second part corrects the β_S parameters. All other parameters are kept intact.

(a) U_1 refinement

Solving (12) and (19), we can express U_1 in terms of I_{DS} and other parameters.

$$U_1 = \frac{1}{V_{DS}} \left\{ \frac{\beta_0 [(V_{GS} - V_{th}) V_{DS} - \frac{a}{2} V_{DS}^2]}{I_{DS} [1 + U_0 (V_{GS} - V_{th})]} - 1 \right\} \quad \text{for linear region} \tag{124}$$

and

$$= \frac{2 a (\alpha - \sqrt{\alpha})}{V_{GS} - V_{th}} \quad \text{in saturation region} \tag{125}$$

where

$$\alpha = \frac{\beta_0 (V_{GS} - V_{th})^2}{2 a I_{DS} [1 + U_0 (V_{GS} - V_{th})]} \tag{126}$$

New U_1 values are obtained from Eq. (124) and (125) for various gate and substrate biases. These new U_1 values are fitted again to (15) to calculate U_{1Z}, U_{1B}, and U_{1D}.

(b) β_0 refinement

Through a similar procedure as used for refining U_1, new values of β_0 can also be calculated and fitted to (17) and (18). This process is repeated several times until the variation of U_1 and β_0 values are small.

Up to this point, the seventeen strong-inversion BSIM parameters have been extracted. The twelve parameters obtained in the saturation-region are compared with some pre-determined bounds to check their validities. The bounds for these parameters are:

$-1.0 \leq \eta_0 \leq 1.0$

$0 \leq \beta_{0Z} \leq 1.0$

$-1.0 \leq U_1 \leq 5.0$

$-1.0 \leq \beta_{0B} \leq 1.0$

$-1.0 \leq \eta_B \leq 1.0$

$-1.0 \leq \eta_D \leq 1.0$

$-1.0 \leq U_{1B} \leq 1.0$

$0 \leq \beta_S \leq 1.0$

$-1.0 \leq \beta_{SB} \leq 1.0$

$-1.0 \leq \beta_{SD} \leq 1.0$

$-1.0 \leq U_{1D} \leq 1.0$

2.3.6 Subthreshold Parameters Extraction

The BSIM subthreshold current expression is given in (25)-(27). The subthreshold swing coefficient (n) is the only parameter to be determined in the weak inversion region. By biasing devices below the threshold, the subthreshold swing coefficient can be calculated from (127).

$$n = \frac{1}{V_{tm}} \frac{dV_{GS}}{d \ln (I_{DS})} \tag{127}$$

Measured results of the subthreshold swing coefficient for different substrate and drain biases are fitted through (28) to calculate three subthreshold parameters, n_0, n_B, and n_D.

This measurement requires good electric and optical shielding to reduce leakage currents.

2.3.7 Substrate Current Parameter Extraction

(A) Extraction of E_{crit}

As mentioned in section 2.2.5, V_{DSAT} can be obtained from I_{BS}/I_{DS} contour plots. Once the V_{DSAT} values are extracted, they are fitted to the analytical model mentioned previously and repeated here for reference:

$$V_{DSAT} = \frac{E_{crit}L(V_{GS} - V_{th})}{E_{crit}L + (V_{GS} - V_{th})} \tag{37}$$

After measurements on various wafers, it was found that the V_{DSAT} values obtained could accurately be predicted by making the critical electric field parameter E_{crit} dependent on

both gate and body bias while using the first-order threshold voltage model.

$$V_{th} = V_{FB} + \phi_S + K_1\sqrt{\phi_S - V_{BS}} - K_2(\phi_S - V_{BS}) \quad (1)$$

V_{FB}, ϕ_S, K_1, and K_2 are parameters extracted in the drain current extraction process. The best fit was obtained by using a linear fit in terms of the bias voltages.

$$E_{crit}(V_{BS}, V_{GS}) = E_{crit0} + E_{critG}V_{GS} + E_{critB}V_{BS} \quad (38)$$

Fig. 15 clearly show the linear E_{crit} dependency to V_{GS} and V_{BS}. The increase in E_{crit} with increasing gate or substrate bias correlates with previously established results [7]. In that work, this increase is attributed to electron mobility degradation caused by vertical fields, which in turn causes an increase of the lateral field necessary to velocity-saturate the channel electrons.

An additional modification was found to be needed to accurately extract the E_{crit} parameters. Because equation (37) becomes inaccurate for low gate bias, it was found that E_{crit} actually decreased before increasing when plotted with increasing gate bias. Thus the program ignores all data that occur before this minima and uses data points only after it senses a positive slope in E_{crit}.

Fig. 16 compares the simulated and measured value of V_{DSAT} using this method. As can be seen, the predicted values correlate well with measured behavior.

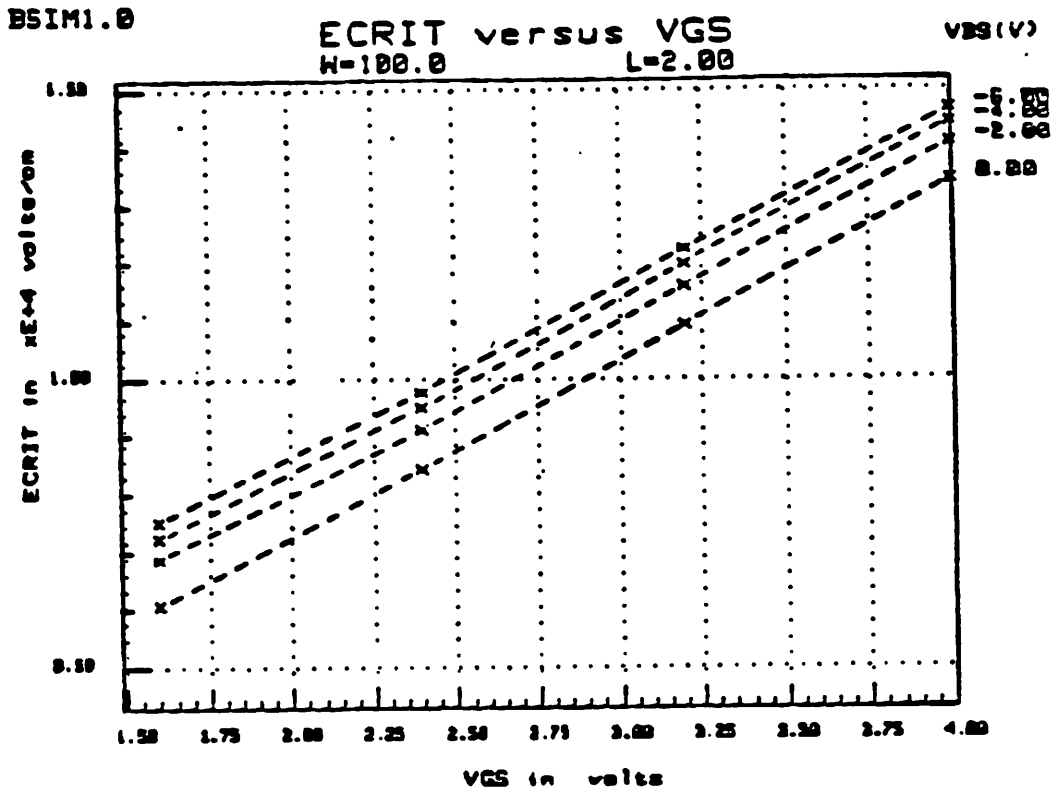


Fig.15a E_{crit} versus V_{GS}. W = 100 μm and L = 2 μm.

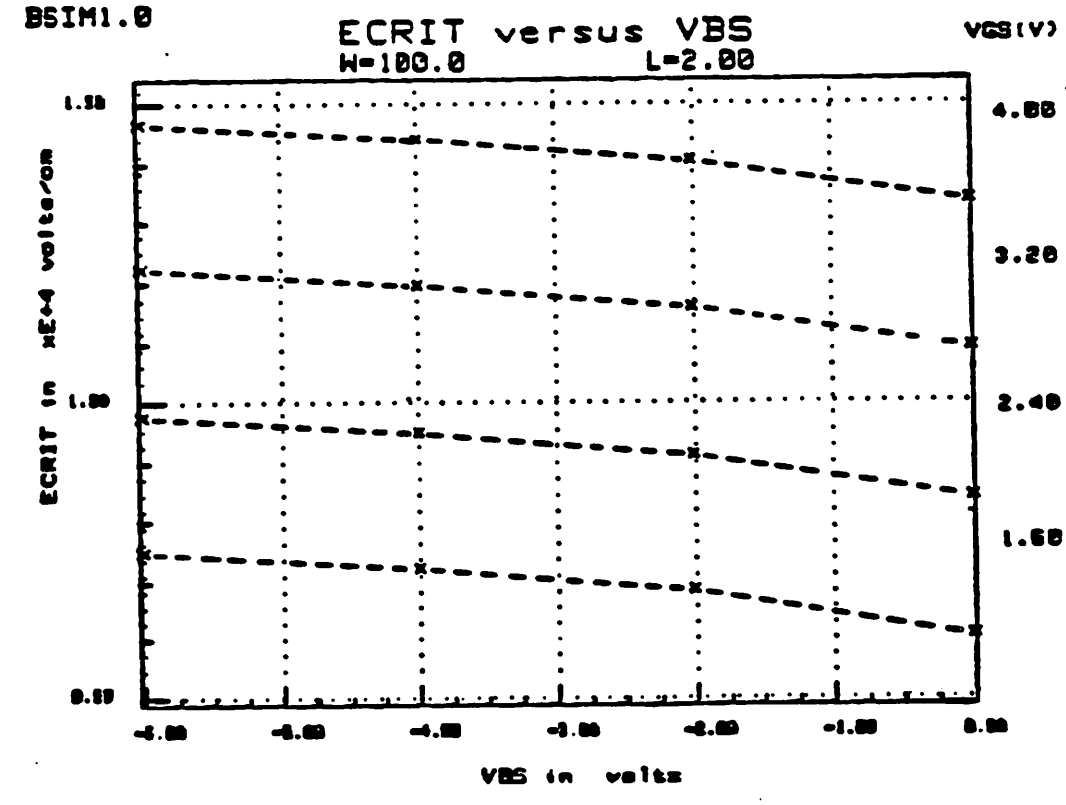


Fig.15b E_{crit} versus V_{BS}. W = 100 μm and L = 2 μm.

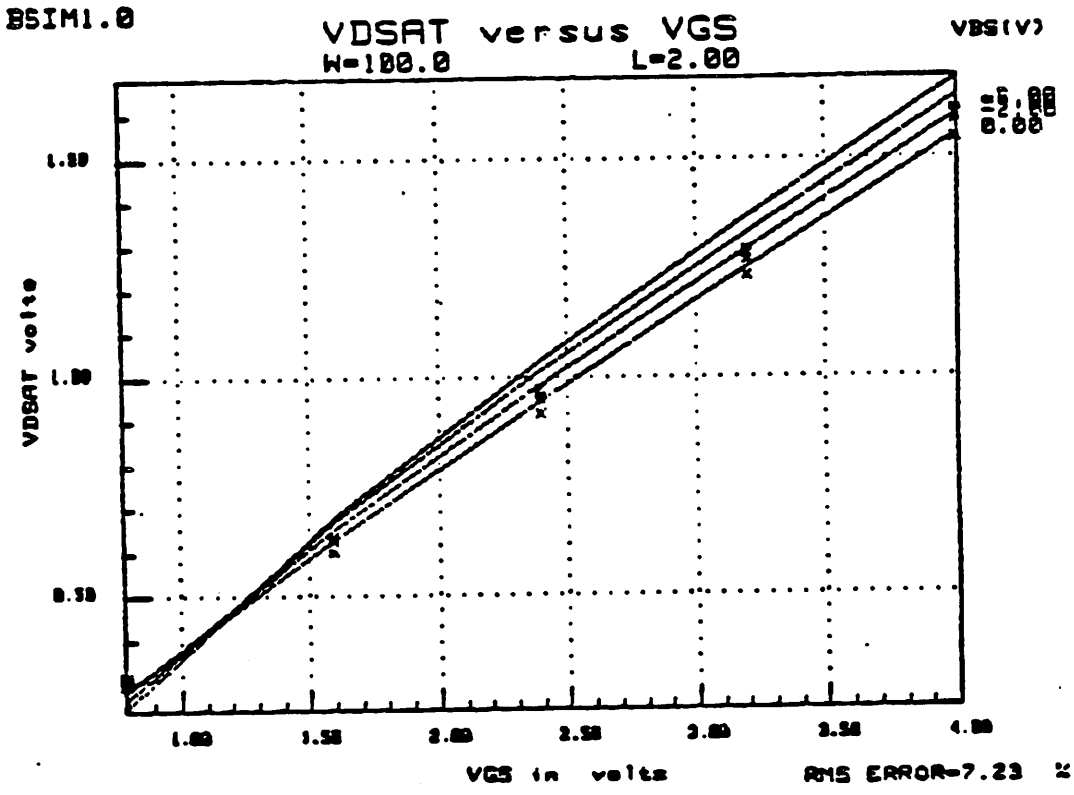


Fig.16 V_{DSAT} versus V_{GS} with V_{BS} as the third parameter. W = 100 μm and L = 2 μm.

(B) Extraction of l_c

Once all measured V_{DSAT} values are known, l_c can be extracted from measured $\frac{I_{BS}}{I_{DS}}$ values from equation (36),

$$\frac{I_{BS}}{I_{DS}} = \frac{A_i}{B_i}(V_{DS} - V_{DSAT}) e^{\frac{-B_i l_c}{(V_{DS} - V_{DSAT})}} \tag{36}$$

A_i and B_i, constants from the electron impact ionization coefficient α_n, are set at well-adopted values of 2×10^6 cm^{-1} and 1.7×10^6 V/cm as mentioned previously.

Extensive measurements were done on NMOS enhancement devices for various processes, with device sizes ranging from 4 μm to 100 μm in width, and 1.5 μm to 20 μm in length. All data could not be accurately predicted without making l_c bias dependent. After separating the various bias effects, the best form for l_c was found to be

$$l_c(V_{BS}, V_{GS}, V_{DS}) = \sqrt{t_{ox}} [l_1 + l_2 (\frac{1}{V_{GS} + 2})] \tag{39}$$

where

$$l_1 = l_{c0} + l_{c1}(\frac{1}{V_{BS} - 4}) + [l_{c2} + l_{c3}(\frac{1}{V_{BS} - 4})]V_{ds} \tag{40}$$

$$l_2 = l_{c4} + l_{c5}(\frac{1}{V_{BS} - 4}) + [l_{c6} + l_{c7}(\frac{1}{V_{BS} - 4})]V_{DS} \tag{41}$$

All data taken below a preset drain bias, $V_{Dthresh}$, is ignored so that leakage current effects will not alter the extraction. $V_{Dthresh}$ is set at 0.2 volts greater than the drain voltage where I_{BS}/I_{DS} falls to 10^{-8} on the maximum gate bias contour.

The motivation behind using the forms in equations (39) and (40) can be seen from Figs. 17. These plots show the effect of gate, drain, and substrate bias on extracted values of l_c. The importance of the cross-term parameters l_{c3} and l_{c7} can be realized by looking at Figs. 17b and 17c. Note that the slope of the measured curve is dependent on the third variable, V_{BS}. Thus, a simple linear relationship is not possible.

The inverse V_{GS} dependence of l_c is used to take into account an observable substantial decrease of I_{BS} (and therefore a non-linear increase of l_c) for low $V_{GS} - V_{th}$ (Fig. 17a). This effect as well as the inverse V_{BS} dependence may be qualitatively explained as being caused by the modulation of the inversion layer thickness of the channel electrons as a secondary effect in addition to the normal decrease in channel charge. As either gate or substrate bias decreases, more electrons flow in a lower field region further away from the silicon-silicon dioxide interface. This results in less electrons experiencing the critical field necessary for impact ionization, and therefore a greater decrease in substrate current than that predicted by theory is observed. This effect is more pronounced for shorter channel lengths and larger drain voltages. At present, no theoretical derivation for this behavior has been published.

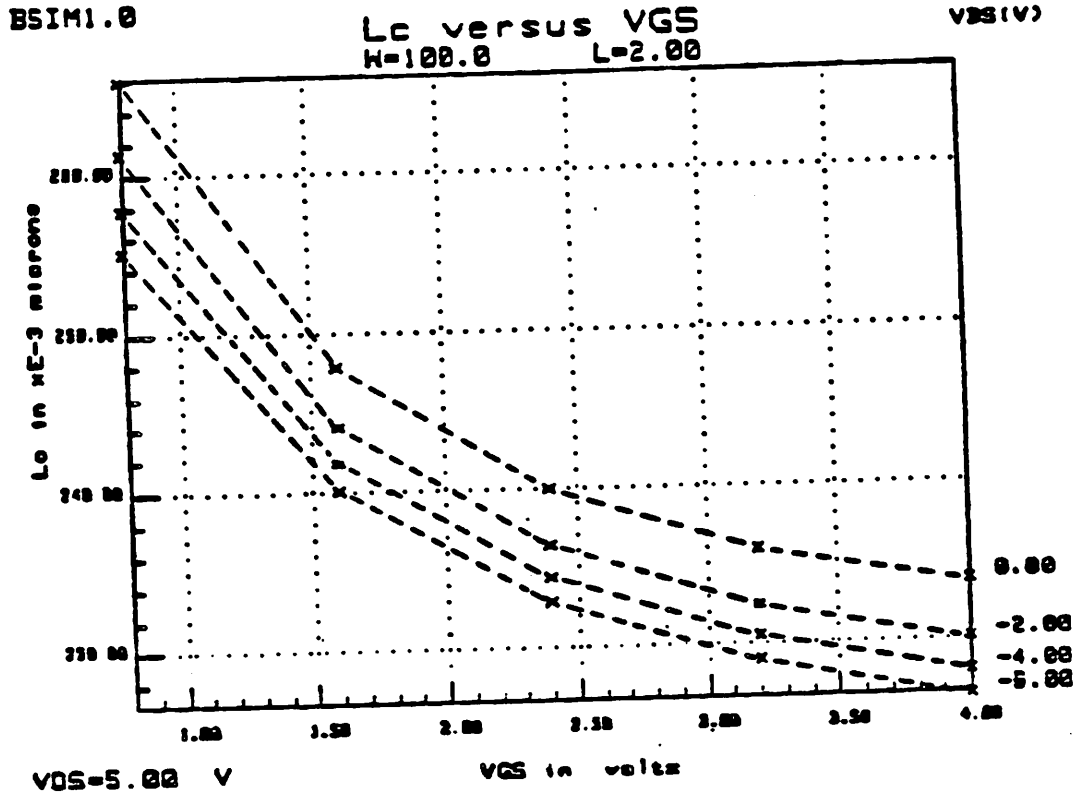


Fig.17a l_c versus V_{GS} with V_{BS} as the third parameter. W = 100 μm and L = 2 μm.

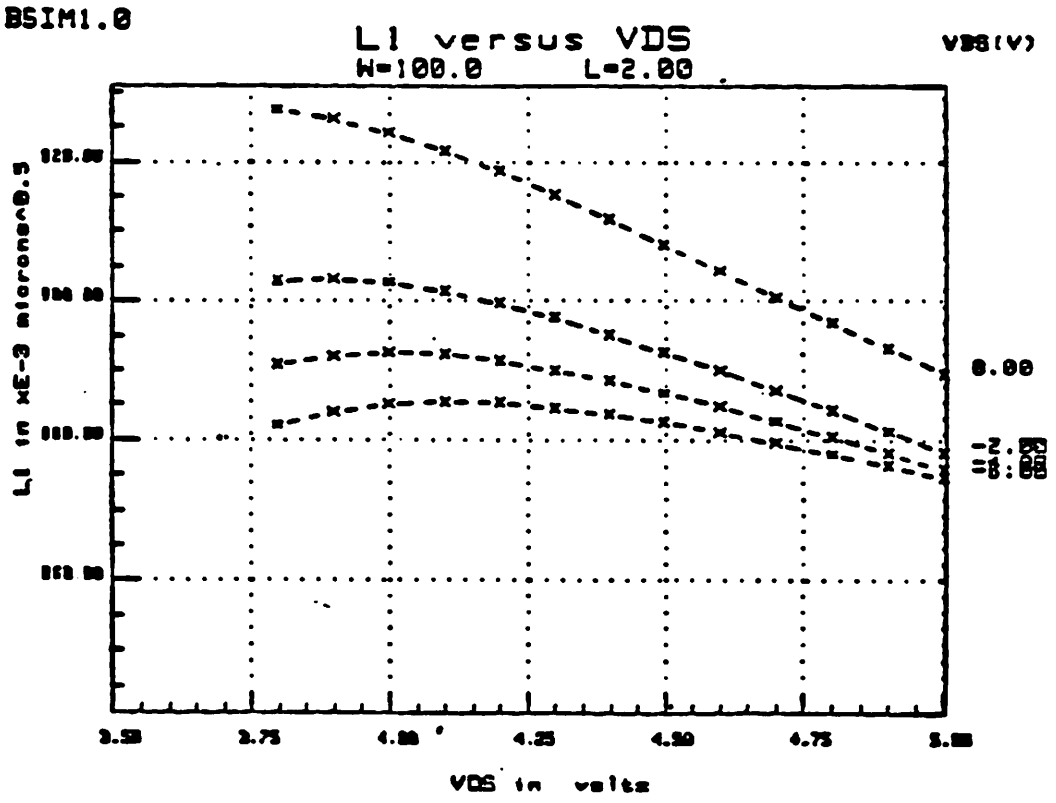


Fig.17b l_1 versus V_{DS} with V_{BS} as the third parameter. W = 100 μm and L = 2 μm.

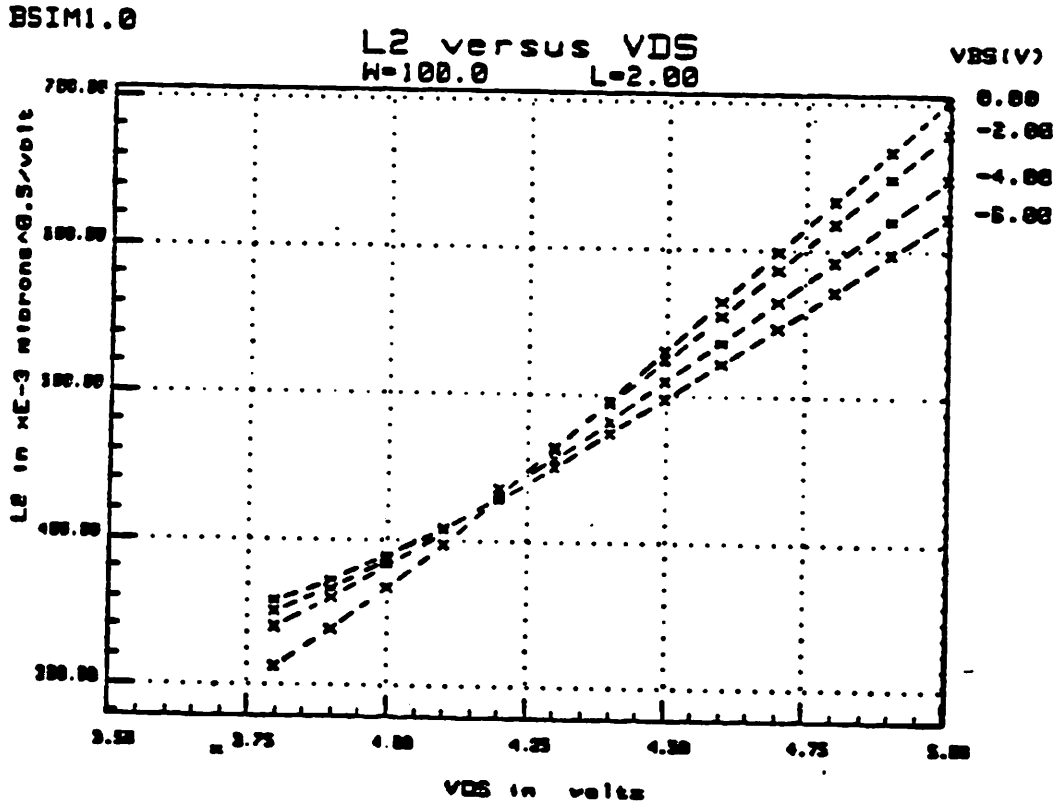


Fig.17c l_2 versus V_{DS} with V_{BS} as the third parameter. W = 100 μm and L = 2 μm.

2.3.8 PROCESS FILE DEVELOPMENT

After 20 (33 if you extract substrate current parameters) BSIM parameters are extracted from each device, they, together with the drawn channel length and channel width, are stored in a temporary file corresponding to the device type measured. Six temporary files are created for each device type at the begining of measurement for a new die. These temporary files that store all the parameters of each device are:

NEDFILE (for NMOS enhancement devices)

NZDFILE (for NMOS zero-threshold devices)

NDDFILE (for NMOS depletion mode devices)

PEDFILE (for PMOS enhancement devices)

PZDFILE (for PMOS zero-threshold devices)

PDDFILE (for PMOS depletion mode devices)

If a file with any of these names exists on the prefixed directory, it will be overwritten after the execution of the extraction program. These files will be purged when the program is normally exited by the user. If the program is abnormally terminated for some reasons, the stored BSIM parameters for those already measured devices are still retained in these files, which can be accessed by "FILER" of the Pascal system.

The parameters extracted from each device are size-dependent. It has been experimentally observed that most of the MOSFET parameters exhibit a $1/W_{eff}$ and $1/L_{eff}$ dependence. To generate a size-independent parameter set that can be used to predict device characteristics with arbitrary dimensions, these size-dependent parameters (except β_Z) are processed according to (128).

$$P_i = P_0 + \frac{P_L}{L_{drawn} - \Delta L} + \frac{P_W}{W_{drawn} - \Delta W} \qquad (128)$$

where W_{drawn} and L_{drawn} are the drawn channel-width and channel-length, respectively. ΔW and ΔL account for any process bias such as print bias, etch bias, and lateral diffusion of dopants. P_i's are the size-dependent parameters extracted from each device. P_0, P_L and P_W are the size-independent parameters to be generated and stored in the process file which

will be read by SPICE. ΔW and ΔL can be extracted from β_Z parameter. As pointed out in (8), in most text book β_Z is written as

$$\beta_Z = \mu_0 C_{ox} \frac{W_{drawn} - \Delta W}{L_{drawn} - \Delta L} \qquad (8)$$

Therefore, a plot of β_{0Z} against W_{drawn} ($1/\beta_{0Z}$ against L_{drawn}) at constant channel length (channel width), ΔW (ΔL) can be obtained. Some typical plots are shown in Fig. 18.

When the testing of all devices on a die is completed, process files for every device type are generated from parameter values stored in these six temporary files. These temporary files are also used to generate parameter versus W_{eff} & L_{eff} plots. The minimum number of devices for each device type required to generate a process file is 3 (two W's and two L's).

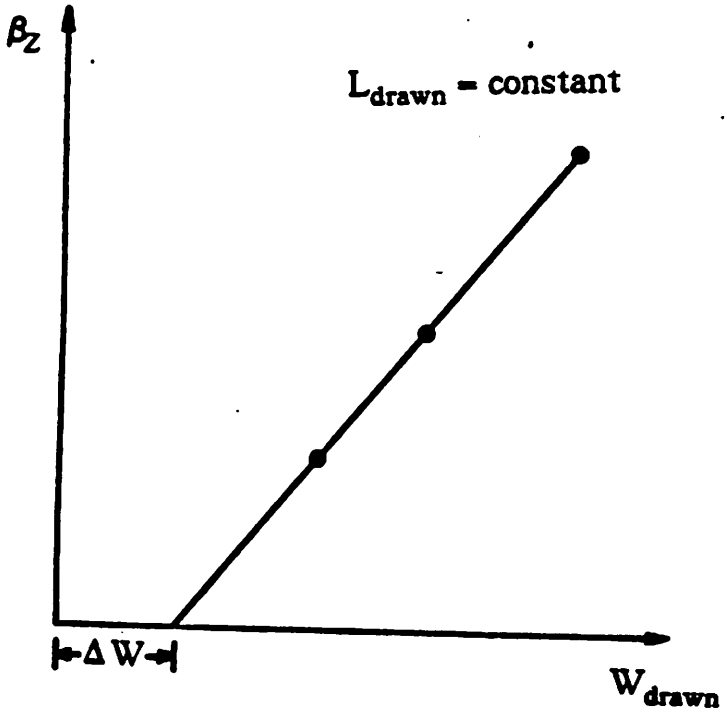


Fig.18a Extraction of ΔL from β_Z parameter.

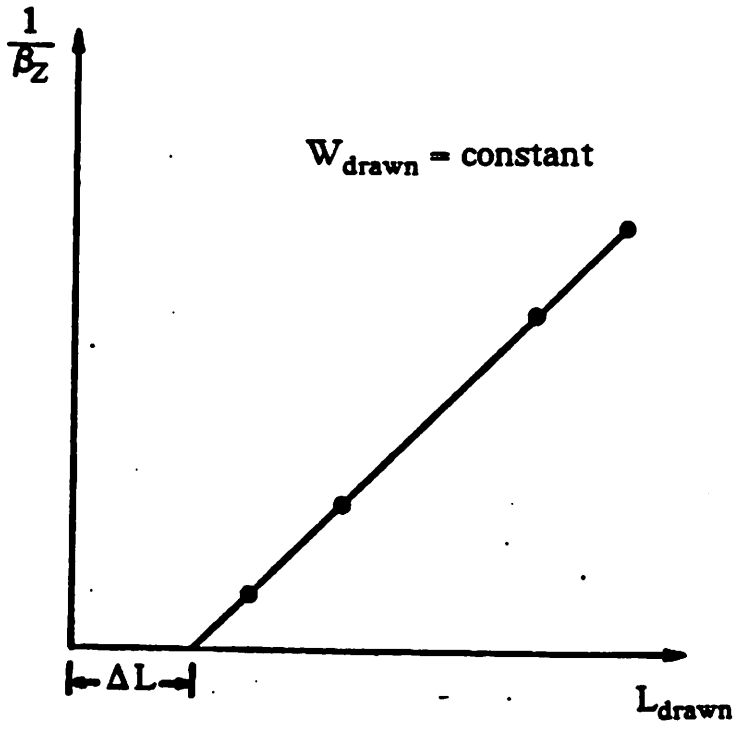


Fig.18b Extraction of ΔW from β_Z parameter.

The process contains 31 lines (43 lines if substrate current parameters are extracted) and 6 columns. A description of the parameters stored in the process file and some examples are shown in Fig. 19. The first line of the process file indicates the device type and process file number which will be referred by SPICE. For example: NM1 may represents NMOS enhancement devices process file No. 1. Lines 2 to 8 are just for user's reference and are ignored by SPICE. Lines 9 to 25 are the drain current parameters for the strong-inversion region. Line 26 stores the oxide thickness of the wafer in μm, temperature at which these parameters were extracted in degree C, and V_{DD} in Volt. Line 27 is for the overlap capacitances of gate to other terminals, CGDO, CGSO, and CGBO, respectively. In the extraction program, these overlap capacitances are all defaulted to 0. The user must put in the correct values for these overlap capacitances if he/she is goin to simulate transient behaviors. The first column of line 28 specifies the charge-partition option. It is defaulted to 1 (0/100 partitioning). It can be changed to 0 if 40/60 partitioning is to be used. Columns 2 and 3 of line 28 are not used. Lines 29 to 31 are for the subthreshold parameters. Lines 32 to 42 are for the substrate current parameters. The last line, line 43, stores the value of H, n, and m for modeling degradation. Except for lines 1-8, 14, 26-28, and 43, the first column of each line stores P_0 parameter as calculated in (120). The second column stores P_L and the third column stores P_W. The fourth to the sixth columns, which are not shown in the examples, are some statistical information which are ignored by SPICE. The format in line 14 is different from other lines. The first column of line 14 stores the low-field mobility calculated from β_0's using (8). The second and third columns of line 14 store Δ L and ΔW, respectively.

One the process file has been created, the extraction program uses it to generate calculated device characteristics. The user can compare the calculated curves with the measured curve to visually observe the goodness of fit. All the extracted results are shown in chapter 5.

	Name	L sens. factor	W sens. factor	Units of basic parameter
1	V_{FB} (VFB)	V_{FBl} (LVFB)	V_{FBw} (WVFB)	V
2	ϕ_S (PHI)	ϕ_{Sl} (LPHI)	ϕ_{Sw} (WPHI)	V
3	K_1 (K1)	K_{1l} (LK1)	K_{1w} (WK1)	$V^{1/2}$
4	K_2 (K2)	K_{2l} (LK2)	K_{2w} (WK2)	-
5	η_0 (ETA)	η_{0l} (LETA)	η_{0w} (WETA)	-
6	μ_Z (MUZ)	δ_l (DL)	δ_w (DW)	$cm^2/V\text{-}s$, μm, μm
7	U_{0Z} (U0)	U_{0Zl} (LU0)	U_{0Zw} (WU0)	V^{-1}
8	U_{1Z} (U1)	U_{1Zl} (LU1)	U_{1Zw} (WU1)	$\mu m\ V^{-1}$
9	μ_{ZB} (X2MZ)	μ_{ZBl} (LX2MZ)	μ_{ZBw} (WX2MZ)	$cm^2/V^2\text{-}s$
10	η_B (X2E)	η_{Bl} (LX2E)	η_{Bw} (WX2E)	V^{-1}
11	η_D (X3E)	η_{Dl} (LX3E)	η_{Dw} (WX3E)	V^{-1}
12	U_{0B} (X2U0)	U_{0Bl} (LX2U0)	U_{0Bw} (WX2U0)	V^{-2}
13	U_{1B} (X2U1)	U_{1Bl} (LX2U1)	U_{1Bw} (WX2U1)	$\mu m\ V^{-2}$
14	μ_S (MUS)	μ_{Sl} (LMS)	μ_{Sw} (WMS)	$cm^2/V^2\text{-}s$
15	μ_{SB} (X2MS)	μ_{SBl} (LX2MS)	μ_{SBw} (WX2MS)	$cm^2/V^2\text{-}s$
16	μ_{SD} (X3MS)	μ_{SDl} (LX3MS)	μ_{SDw} (WX3MS)	$cm^2/V^2\text{-}s$
17	U_{1D} (X3U1)	U_{1Dl} (LX3U1)	U_{1Dw} (WX3U1)	$\mu m\ V^{-2}$
18	T_{ox} (TOX)	T_{emp} (TEMP)	V_{dd} (VDD)	$\mu m, 1°C, V$
19	CGDO	CGSO	CGBO	F/m
20	XPART	DUM1	DUM2	-
21	N0	LN0	WN0	-
22	NB	LNB	WNB	-
23	ND	LND	WND	-

NOTE: In the process file, lines start with "*" are comment lines.

Fig.19a BSIM process file format (without substrate current parameters). This format is the one accepted by SPICE2.

```
NM1
*PROCESS=Bell.NMOS
*RUN=1
*WAFER=4
*XPOS=4
*YPOS=8
*OPERATOR=Min-Chie
*DATE=DEC-16-86
-9.1190E-001,4.50505E-002,8.07659E-002
7.15629E-001,0.00000E+000,0.00000E+000
9.11563E-001,3.09635E-002,9.10498E-002
-7.2610E-002,9.65815E-002,-1.2265E-002
-1.3534E-002,6.63392E-002,1.32316E-002
8.14334E+002,-6.6799E-001,6.57698E-001
8.38497E-002,2.15213E-001,-1.8507E-001
3.71972E-002,3.13453E-001,8.25056E-002
1.82478E+001,-3.0827E+001,3.23696E+001
-2.7497E-003,1.24943E-002,1.45979E-003
1.45349E-003,-6.0199E-003,-4.1096E-003
3.61136E-003,-1.2593E-002,9.10666E-003
-1.3209E-002,1.28809E-002,4.03320E-003
6.79946E+002,6.41401E+002,-1.0190E+002
-4.2992E+000,2.05127E+000,4.99043E+001
-8.0035E+000,1.16239E+002,-2.5747E+001
2.75281E-003,5.03620E-002,1.80500E-002
4.50000E-002,2.70000E+001,5.00000E+000
0.0,0
1.0,0
1.41981E+000,5.87905E-001,2.46860E-001
-5.1335E-002,1.78455E-001,2.93640E-002
3.83850E-002,1.56124E-001,9.59408E-003
```

Fig.19b Example of the process file without substrate current parameters.

	Name	L sens. factor	W sens. factor	Units of basic parameter
1	V_{FB} (VFB)	V_{FBl} (LVFB)	V_{FBw} (WVFB)	V
2	ϕ_S (PHI)	ϕ_{Sl} (LPHI)	ϕ_{Sw} (WPHI)	V
3	K_1 (K1)	K_{1l} (LK1)	K_{1w} (WK1)	$V^{1/2}$
4	K_2 (K2)	K_{2l} (LK2)	K_{2w} (WK2)	-
5	η_0 (ETA)	η_{0l} (LETA)	η_{0w} (WETA)	-
6	μ_Z (MUZ)	δ_l (DL)	δ_w (DW)	$cm^2/V\text{-}s.\ \mu m.\ \mu m$
7	U_{0Z} (U0)	U_{0Zl} (LU0)	U_{0Zw} (WU0)	V^{-1}
8	U_{1Z} (U1)	U_{1Zl} (LU1)	U_{1Zw} (WU1)	$\mu m\ V^{-1}$
9	μ_{ZB} (X2MZ)	μ_{ZBl} (LX2MZ)	μ_{ZBw} (WX2MZ)	$cm^2/V^2\text{-}s$
10	η_B (X2E)	η_{Bl} (LX2E)	η_{Bw} (WX2E)	V^{-1}
11	η_D (X3E)	η_{Dl} (LX3E)	η_{Dw} (WX3E)	V^{-1}
12	U_{0B} (X2U0)	U_{0Bl} (LX2U0)	U_{0Bw} (WX2U0)	V^{-2}
13	U_{1B} (X2U1)	U_{1Bl} (LX2U1)	U_{1Bw} (WX2U1)	$\mu m\ V^{-2}$
14	μ_S (MUS)	μ_{Sl} (LMS)	μ_{Sw} (WMS)	$cm^2/V^2\text{-}s$
15	μ_{SB} (X2MS)	μ_{SBl} (LX2MS)	μ_{SBw} (WX2MS)	$cm^2/V^2\text{-}s$
16	μ_{SD} (X3MS)	μ_{SDl} (LX3MS)	μ_{SDw} (WX3MS)	$cm^2/V^2\text{-}s$
17	U_{1D} (X3U1)	U_{1Dl} (LX3U1)	U_{1Dw} (WX3U1)	$\mu m\ V^{-2}$
18	T_{ox} (TOX)	T_{emp} (TEMP)	V_{dd} (VDD)	$\mu m, 1°C, V$
19	CGDO	CGSO	CGBO	F/m
20	XPART	DUM1	DUM2	-
21	N0	LN0	WN0	-
22	NB	LNB	WNB	-
23	ND	LND	WND	-
24	$E_{crit\,0}$ (ECRIT0)	$E_{crit\,0l}$ (LECRIT0)	$E_{crit\,0w}$ (WECRIT0)	V/cm
25	E_{critg} (ECRITG)	E_{critgl} (LECRITG)	E_{critgw} (WECRITG)	1/cm
26	E_{critb} (ECRITB)	E_{critbl} (LECRITB)	E_{critbw} (WECRITB)	1/cm
27	$l_{c\,0}$ (LC0)	$l_{c\,0l}$ (LLC0)	$l_{c\,0w}$ (WLC0)	$\mu m^{1/2}$
28	$l_{c\,1}$ (LC1)	$l_{c\,1l}$ (LLC1)	$l_{c\,1w}$ (WLC1)	$\mu m^{1/2}\text{-}V$
29	$l_{c\,2}$ (LC2)	$l_{c\,2l}$ (LLC2)	$l_{c\,2w}$ (WLC2)	$\mu m^{1/2}\text{-}V^{-1}$
30	$l_{c\,3}$ (LC3)	$l_{c\,3l}$ (LLC3)	$l_{c\,3w}$ (WLC3)	$\mu m^{1/2}$
31	$l_{c\,4}$ (LC4)	$l_{c\,4l}$ (LLC4)	$l_{c\,4w}$ (WLC4)	$\mu m^{1/2}\text{-}V$
32	$l_{c\,5}$ (LC5)	$l_{c\,5l}$ (LLC5)	$l_{c\,5w}$ (WLC5)	$\mu m^{1/2}\text{-}V^2$
33	$l_{c\,6}$ (LC6)	$l_{c\,6l}$ (LLC6)	$l_{c\,6w}$ (WLC6)	$\mu m^{1/2}$
34	$l_{c\,7}$ (LC7)	$l_{c\,7l}$ (LLC7)	$l_{c\,7w}$ (WLC7)	$\mu m^{1/2}\text{-}V$
35	H	n	m	-, -, -

Fig.19c BSIM process file format (with substrate current parameters). This format has to be processed by SCALP in order to be accepted by SPICE2.

```
NM1  DU1
*PROCESS=xerox
*RUN=1
*WAFER=
*XPOS=6
*YPOS=5
*OPERATOR=SCALP
*DATE=July-16-85
*  NMOS-1 PARAMETERS  (07-16-85)
-1.0087E+000,-2.1402E-001,3.44354E-001
7.96434E-001,0.00000E+000,0.00000E+000
1.31191E+000,3.23395E-001,-5.7698E-001
1.46640E-001,1.68585E-001,-1.8796E-001
-1.0027E-003,-9.4847E-003,1.47316E-002
5.34334E+002,7.9799E-001,4.7740E-001
4.38497E-002,6.38105E-002,-6.1053E-002
-5.7332E-002,1.01174E+000,1.62706E-002
8.25434E+000,-2.4197E+001,1.95696E+001
-7.6911E-004,9.62411E-003,-3.7951E-003
7.86777E-004,7.35448E-004,-1.7796E-003
1.06821E-003,-8.0958E-003,4.03379E-003
-1.9209E-002,-7.4573E-002,1.47520E-002
5.40612E+002,6.21401E+002,-1.9190E+002
-1.2992E+001,-6.4900E+001,4.29043E+001
-9.4035E+000,1.18239E+002,-2.9747E+001
0.0000E-002,0.00000E-001,0.0000E-002
3.00000E-002,2.70000E+001,5.00000E+000
0.00000E-000,0.00000E-000,0.00000E-000
1.0       0.0,      0.0
1.55,     0.0,      0.0
0.09,     0.0,      0.0
0.0,      0.0,      0.0
* The Substrate Parameters
1.01647E+004,3.94291E+003,-5.6175E+003
2.99713E+003,-5.0905E+002,3.70774E+002
1.65674E+002,5.63348E+002,-4.1546E+002
1.94944E+000,-1.5686E+000,1.40142E+000
4.22982E+000,-4.9338E+000,5.85560E+000
-1.5710E-001,2.45792E-001,-1.9944E-001
-6.0306E-001,6.84631E-001,-9.0453E-001
-2.8755E+000,4.84726E+000,-3.6936E+000
-1.1199E+001,1.53871E+001,-1.7046E+001
5.55000E-001,-8.4885E-001,4.46489E-001
1.57731E+000,-2.2878E+000,2.69803E+000
0.028,0.5,3.5
*  n+ diffusion layer
35.0,  2.75E-4,  1.90E-10,  1.0E-5,  0.7
0.8,  0.5,      0.33,     0,      0
```

Fig.19d Example of the process file with substrate current parameters.

Chapter 3: SPICE2 Implementation

3.1 Basic Considerations

It was desired that the addition of BSIM should not affect the execution of the existing SPICE2 MOS transistor models. Therefore, a new linked list was created to store BSIM model parameters. A SPICE2 example with BSIM is shown in Fig. 20.. To activate BSIM, an MOS transistor is described with "S" as the leading character in place of "M". The transistor card refers to a process file, instead of a model card, for the parameter values.

A model pointer has been added to the resistor and capacitor linked lists. Then, the resistor (or capacitor) value can also be calculated using the data corresponding to the interconnection layer from which it is made. This mimics actual integrated-circuit construction. Whenever the fabrication process is modified, the resistor (or capacitor) value will change according to the new data in the process file.

3.1.1 Process File Format

Figure 21 shows the process-file format. A process file consists of two parts: keywords on the first line, and data on the other lines. In a process file, there will be two lines of data associated with each interconnect keyword and 23 lines of data associated with each transistor keyword. The mapping is sequential. For example, data on the third and fourth lines are associated with the first keyword DU1 specified on the first line. A line beginning with "*" is treated as a comment line. At present, there are 14 keywords reserved for interconnect types (DU1 to DU6, PY1 to PY4, ML1 to ML4) and 10 keywords reserved for transistor types (NM1 to NM5, PM1 to PM5).

```
EXAMPLE1 :          CMOS INVERTER
VCC  1  0  5
SMN1  2  3  0  0  PC1_NM1    L=10U  W=50U
SMP1  2  3  1  1  PC1_PM1_DU2 L=10U  W=50U
VIN  4  0  PWL (0 0  5N 5  10N 5  20N 0)
RIN  4  3        PC1_PY1    L=20U  W=60U
C1   2  0  50F
.OPTIONS  NOMOD  RELTOL=1E-5 CHGTOL=1E-16
.TRAN 0.5N 20N
.PRINT TRAN  V(3)  V(2)
.PLOT  TRAN  V(3)  V(2)
.PROCESS PC1 FILENAME=PNMED
.WIDTH OUT=80
.END
```

Fig.20A SPICE example with BSIM

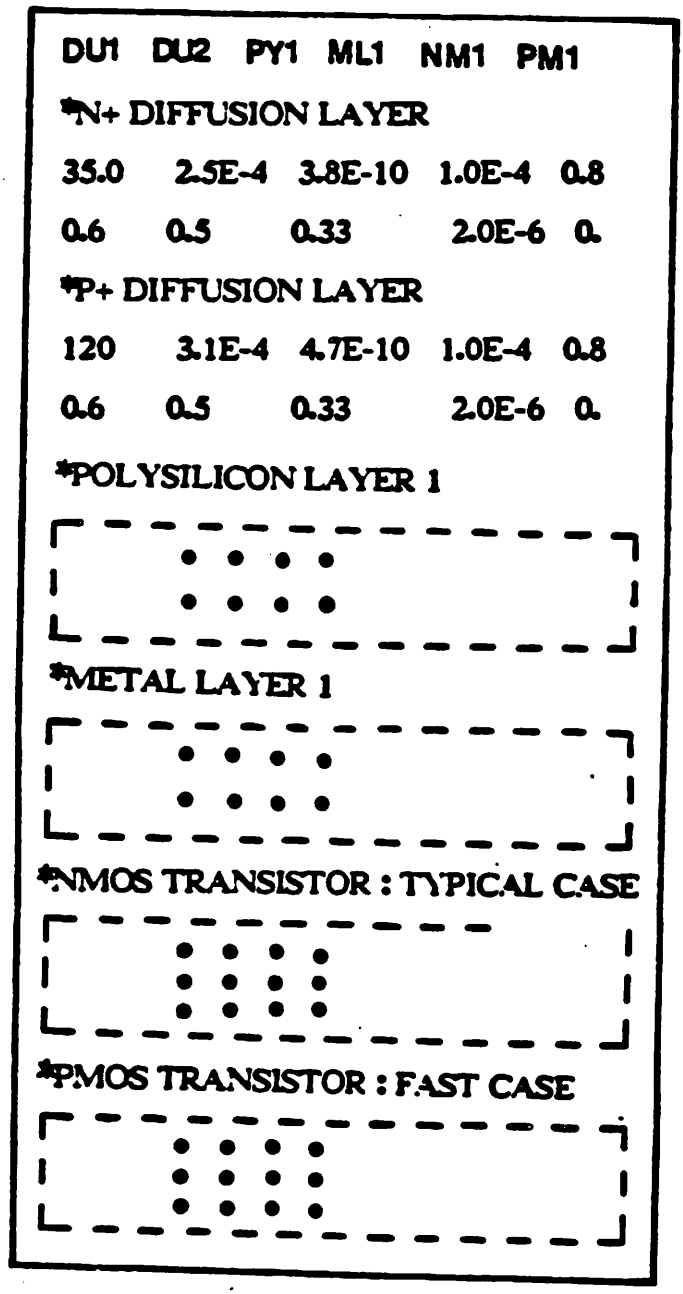


Fig.21Process file format

3.1.2 New and Modified Linked Lists

The implementation of BSIM and the process-oriented simulator structure involves a large number of changes in the SPICE2G.6 source code. New linked lists are created for the BSIM MOSFET with internal ID = 15 transistor process information with internal ID = 25, and interconnection-line process information with internal ID = 26.

Compared to the regular MOSFET linked list in SPICE, there are more entries reserved for the BSIM MOSFET to store the electrical parameters of each individual transistor. Two model pointers are used in the BSIM MOSFET linked list (ID = 15) while only one model pointer is reserved in the regular MOSFET linked list (ID = 14).

A new linked list with ID = 25 is created to store the 69 extracted BSIM process parameters and one of which serves as a flag to choose the channel-charge partitioning method. The model pointer 1 in the BSIM MOSFET linked list (ID = 15) points to the BSIM model linked list (ID = 25) for the intrinsic gate-region information. The model pointer 2 in the BSIM MOSFET linked list (ID = 15) points to the interconnection-line model linked list (ID = 26) for the source and drain junction information. The model pointer in the resistor linked list (ID = 1) or the capacitor linked list (ID = 2) points to the interconnection-line model linked list (ID=26) for resistance or capacitance information.

In the standard SPICE2 implementation, resistor and capacitor elements have no models, hence no pointers; now they have. For such an element two more entries are added to the original linked list, one for the interconnect type and the other for the model pointer. During the read-in stage, SPICE2 stores the identification code for the keyword (PY1, ML1, DU1, etc) appearing on the element card. Later the integer code of the element is used to search for the appropriate model to which the pointer entry of the element can point. The same strategy is used for BSIM MOS transistors. A com-

plete description of the new and modified linked lists is included in section 3.4.

Geometries of resistors, capacitors and MOS transistors are stored in their own linked lists as before. Given their sizes, the resistor and capacitor values are calculated from the corresponding interconnect-layer information in a pre-processing fashion. Electrical parameters for each BSIM MOS transistor are calculated from BSIM process parameters and stored in the linked list of each transistor. This is a one-time data reduction. Since the source and drain junction capacitances of an MOS transistor are bias dependent. They have to be evaluated during each iteration cycle.

3.2 Implementation in SPICE2

3.2.1 Functions of the Newly Added Subroutines

There are 5 new subroutines. Their names and functions are described in the following:

(1) subroutine PROCHK:

This subroutine converts size-independent parameters into electrical parameters for each transistor. It prints out a list of process-file parameters, and single-transistor electrical parameters according to the user's request. Its role is similar to that of MODCHK for SPICE2G.6 MOS LEVEL-1, 2 and 3 models.

(2) subroutine BSIMEQ:

BSIM dc and charge expressions and their derivatives are implemented in this subroutine. Given all the electrical parameters and the terminal voltages of an MOS transistor, this subroutine evaluates the drain current and conductances. It also calculates terminal charges and capacitances for transient and small-signal ac analysis. Its role is equivalent to a combination of subroutines MOSEQ2 and MOSQ2.

(3) subroutine BSIM:

This subroutine processes the MOS devices using the BSIM model for dc and transient analysis. Only the companion BSIM charge and capacitance model is used for the charge-storage effect associated with the thin-oxide region. It is similar to subroutine MOSFET in the regular SPICE2G.6 code.

(4) subroutine BSMCAP:

Given the active-region charge and capacitances from subroutine BSIMEQ and the overlap capacitances and junction capacitances, this subroutine computes the equivalent conductances and complete terminal charges for an BSIM transistor. It is like subroutine MOSCAP.

(5) subroutine FNDTYP:

This subroutine establishes the connection between an element linked list, as that of a resistor, capacitor, or BSIM MOS transistor, and the linked list of corresponding process parameters. It writes the address of the process-parameter linked list to the pointer location of the element linked list. Two keys are used to search for a match: the process name and the model type. It is similar to subroutine FNDNAM in the regular SPICE2G.6 code.

3.2.2. Modifications to the Original Subroutines

Eighteen subroutines in the original SPICE2G.6 code are partially modified. Their names and modifications are described in the following:

ACLOAD	DCTRAN	LOAD	SPICE
ADDELT	ERRCHK	MATLOC	TOPCHK
ALTER	FIND	MATPTR	TRUNC
CARD	GETLIN	NXTCHR	
DCOP	LNKREF	READIN	

(1) subroutine ACLOAD:

The task of initialization and loading of the complex coefficient matrix is extended to include BSIM MOS transistors (ID = 15).

(2) subroutines ADDELT. ALTER. and FIND:

The major changes in these subroutines are made to create linked lists with ID = 15. 25 and 26. and to expand the sizes of the linked lists with ID = 1 and 2.

(3) subroutines CARD. GETLIN:

Changes are made to handle the BSIM process file.

(4) subroutine DCTRAN:

Subroutine BSIM is called by this subroutine. The line "1 + JELCNT(14)". which is near the beginning of transient analysis source code. is changed to "1 + JELCNT(14) + JELCNT(15)".

(5) subroutine DCOP:

The printing of operating-point information is extended to include BSIM MOS transistors.

(6) subroutine ERRCHK:

Changes include translating node initial conditions to device initial conditions for BSIM MOS transistors when UIC is specified on the .TRAN card. and to assign default values for transistor geometries. The added subroutine PROCHK is called by this subroutine.

(7) subroutine LNKREF:

The task of resolving unsatisfied name references is extended to include BSIM MOS transistors. Subroutine FNDTYP is called by this subroutine twice to find addresses for both pointers. This subroutine calls FNDTYP. instead of FNDNAM. to handle resistors and capacitors.

(8) subroutine LOAD:

Subroutine BSIM is called by this subroutine.

(9) subroutines MATPTR. MATLOC and TRUNC:

Changes are made to include BSIM MOS transistors.

(10) subroutine NXTCHR:

The character "_" is also treated as a delimiter.

(11) subroutine READIN:

Many modifications have been made in this subroutine. Variable strings BIDM(25), JPOLAR(24), NPROID(24), JPAR(3), JPBR(2), BMPAR(80), RESLIN(15), and ITYPES(25) are created and initialized to handle the read-in of the BSIM MOS transistor and interconnect parameters. One more entry is added to the string AIDC to recognize the keyword "PROCESS." New input syntax for resistors, capacitors, BSIM MOS devices, and interconnects is included. The keyword "PROCESS" is treated like the keyword "MODEL" except it is used with ID = 25 and 26. This subroutine opens process files and calls subroutine CARD to read the parameters.

(12) main program SPICE:

The line "WRITE (IOFILE.361) NUNODS, NCNODS, NUMNOD, NUMEL, (JELCNT(I), I=11,14)" is changed to

"WRITE (IOFILE.361) NUNODS, NCNODS, NUMNOD, NUMEL, (JELCNT(I),I=11,13), 1JELCNT(14) +JELCNT(15)" to correctly count the number of MOS transistors.

(13) subroutine TOPCHK:

The line "1 1HQ,1HJ,1HM,0.0D0,0.0D0,0.0D0,1HT,0.0D0,0.0D0,0.0D0,0.0D0 /" is changed to "1 1HQ,1HJ,1HM,1HS,0.0D0,1HT,0.0D0,0.0D0,0.0D0,0.0D0 /" to include BSIM MOS transistors.

3.3 Program Performance

Experimental results show that the BSIM model reduces the program execution time by a factor of 2 as compared with the popular SPICE2 LEVEL-2 MOSFET model

A comparison of selected SPICE2 simulation execution time is listed in Table 3.1

Table 3.1

A Comparison of Selected SPICE2 Simulation Execution Time		
Circuit Description	BSIM (sec.)	MOS LEV-2 Model (sec.)
Ratioless Dynamic Logic Ckt.	24.50	29.75
Five Stage Inverter Chain	18.30	44.25
MOS Amplifier (dc & ac)	40.02	52.70
MOS Amplifier (transient)	75.08	137.50
One Stage Op-Amp	15.83	70.77
Binary-to-Octal Decorder	262.37	586.28
Telecommunication Ckt.	1784.83	2717.32

3.4 BSIM User's Guide

****

This is a supplement to the original "SPICE Version 2G User's Guide," and is intended for SPICE users who have access to BSIM (Berkeley Short-channel Igfet Model). It contains descriptions of the various additional device features supported by BSIM and how to activate them. To be complete, it should be used together with the original User's Guide.

****

In addition to the regular resistor and capacitor formats, BSIM also supports resistors and capacitors generated with interconnects.

3.4.1 Resistors

General form:

RXXXXXXX N1 N2 PNAME_LT L=VAL <W=VAL> <TC=TC1<TC2>>

Examples:

R1 1 2 PC1_DU1 L=10U

RC1 12 17 PC1_DU3 L=20U W=4U TC=0.001, 0.015

N1 and N2 are the two element nodes. PNAME is the process name.

LT is the interconnect type. At present, there are fourteen interconnect types available (DU1 to DU6, PY1 to PY4, and ML1 to ML4).

L and W are the resistor length and width, in meters. W should be specified if the default value in the process file is not to be used.

TC1 and TC2 are the (optional) temperature coefficients: if not specified, zero is

assumed for both. The value of the resistor as a function of temperature is given by:

value(TEMP) = value(TNOM)*(1+TC1*(TEMP-TNOM)+TC2*(TEMP-TNOM)**2))

3.4.2 Capacitors

General form:

CXXXXXXX N1 N2 PNAME_LT L=VAL <W=VAL>

Examples:

CBYP 13 0 PC1_PY1 L=20U

COSC 17 23 PC1_ML2 L=30U W=30U IC=3V

N1 and N2 are the two element nodes. PNAME is the process name.

LT is the interconnect type. At present, there are fourteen interconnect types available (DU1 to DU6, PY1 to PY4, and ML1 to ML4).

L and W are the capacitor length and width, in meters. W should be specified if the default value in the process file is not to be used.

The (optional) initial condition is the initial (time-zero) value of capacitor voltage (in volts).

[Note that the initial conditions (if any) apply 'only' if the UIC option is specified on the .TRAN card.]

3.4.3 BSIM MOSFET's

General form:

SXXXXXXX ND NG NS NB PNAME_MT_DT<_STHD> <L=VAL> <W=VAL>

+ <AD=VAL> <AS=VAL> <PD=VAL> <PS=VAL> <NRD=VAL> <NRS=VAL>

+ <OFF> <IC=VDS,VGS,VBS>

Examples:

S1 24 2 0 20 PC1_NM1

S31 2 17 6 10 PC2_NM2_DU2_STHD L=5U W=2U

S31 2 16 6 10 PC2_PM1_DU3 5U 2U

S1 2 9 3 0 PC2_PM2_DU3 L=10U W=5U AD=100P AS=100P PD=40U PS=40U

S1 2 9 3 0 PC1_NM1_STHD 10U 5U 2P 2P

ND, NG, NS, and NB are the drain, gate, source, and bulk (substrate) nodes, respectively.

PNAME is the process name.

MT is the device type. At present, there are five device types for both N-channel and P-channel transistors (NM1 to NM5, and PM1 to PM5).

DT is the diffusion type to be used for the source/drain junctions. There are six diffusion types available (DU1 to DU6). The default is DU1 for N-channel transistors and DU2 for P-channel transistors.

STHD is used as a flag. If it is specified, the weak-inversion current characteristic will be included.

L and W are the channel length and width, in meters.

AD and AS are the areas of the drain and source diffusions, in sq-meters.

[Note that the suffix U specifies microns (1E-6 m) and P sq-microns (1E-12 sq-m). If any of L, W, AD, or AS are not specified, default values are used. The user may specify the values to be used for these default parameters on the .OPTIONS card. The use of defaults simplifies input deck preparation, as well as the editing required if device geometries are to be changed.]

PD and PS are the perimeters of the drain and source junctions, in meters. NRD and NRS designate the equivalent number of squares of the drain and source diffusions; these values multiply the sheet resistance RSH specified on the .process card for an accurate representation of the series drain and source resistance components of each transistor.

[PD, PS, NRD, and NRS all default to 0.0. OFF indicates an (optional) initial condition on the device for dc analysis. The (optional) initial condition specification using IC=VDS,VGS,VBS is intended for use with the UIC option on the .TRAN card, when a

transient analysis is desired starting from other than the quiescent operating point. See the .IC card for a better and more convenient way to specify transient initial conditions.]

3.4.4 .PROCESS Card

General form:

.PROCESS PNAME FILENAME=FNAME

Examples:

.PROCESS PC1 FILENAME=PNMED

.PROCESS PD2 FILENAME=FAST

The .PROCESS card specifies process parameter values that will be used by one or more devices.

PNAME is the process name. and FNAME is the name of the file containing the process parameter values. Special rules have to be observed in choosing FNAME. The leading character of FNAME should be alphabetical. Alphabetical characters will be recognized only in the capital form. Totally no more than 8 characters are allowed.
The process card is used together with resistors. capacitors. as well as MOS transisors.

3.4.5 Process File

This file is generated by the automated characterization program. and it contains the process information for the transistors as well as for the interconnects. For transistors. the L (channel-length) and W (channel-width) sensitivity factors of a basic electrical parameter are denoted by appending the italic characters 'l' and 'w' to the name of the parameter. For the example of the basic parameter V_{FB} (flat-band voltage). there are two corresponding sensitivity factors. V_{FBl}. V_{FBw}. If P_0 is the basic parameter and P_L and P_W are the corresponding L and W sensitivity factors. The formula

$$P = P_0 + \frac{P_L}{L_{eff}} + \frac{P_W}{W_{eff}}$$

is used to obtained the value for each transistor size with both L_{eff} (= L_{MK} - ΔL) and W_{eff} (= W_{MK} - ΔW) in μm.

(a) The format of the process parameters is listed below:

TRANSISTORS

	name	L sens. factor	W sens. factor	units of basic parameter
1	V_{FB} (VFB)	V_{FBl} (LVFB)	V_{FBw} (WVFB)	V
2	ϕ_S (PHI)	ϕ_{Sl} (LPHI)	ϕ_{Sw} (WPHI)	V
3	K_1 (K1)	K_{1l} (LK1)	K_{1w} (WK1)	$V^{1/2}$
4	K_2 (K2)	K_{2l} (LK2)	K_{2w} (WK2)	-
5	η_0 (ETA)	η_{0l} (LETA)	η_{0w} (WETA)	-
6	μ_Z (MUZ)	δ_l (DL)	δ_w (DW)	$cm^2/V-s(\mu m . \mu m)$
7	U_{0Z} (U0)	U_{0Zl} (LU0)	U_{0Zw} (WU0)	V^{-1}
8	U_{1Z} (U1)	U_{1Zl} (LU1)	U_{1Zw} (WU1)	$\mu m\ V^{-1}$
9	μ_{ZB} (X2MZ)	μ_{ZBl} (LX2MZ)	μ_{ZBw} (WX2MZ)	cm^2/V^2-s
10	η_B (X2E)	η_{Bl} (LX2E)	η_{Bw} (WX2E)	V^{-1}
11	η_D (X3E)	η_{Dl} (LX3E)	η_{Dw} (WX3E)	V^{-1}
12	U_{0B} (X2U0)	U_{0Bl} (LX2U0)	U_{0Bw} (WX2U0)	V^{-2}
13	U_{1B} (X2U1)	U_{1Bl} (LX2U1)	U_{1Bw} (WX2U1)	$\mu m\ V^{-2}$
14	μ_S (MUS)	μ_{Sl} (LMS)	μ_{Sw} (WMS)	cm^2/V^2-s
15	μ_{SB} (X2MS)	μ_{SBl} (LX2MS)	μ_{SBw} (WX2MS)	cm^2/V^2-s
16	μ_{SD} (X3MS)	μ_{SDl} (LX3MS)	μ_{SDw} (WX3MS)	cm^2/V^2-s
17	U_{1D} (X3U1)	U_{1Dl} (LX3U1)	U_{1Dw} (WX3U1)	$\mu m\ V^{-2}$
18	T_{ox} (TOX)	T_{emp} (TEMP)	V_{dd} (VDD)	$\mu m\ (^0C . V)$

19	CGDO	CGSO	CGBO	F/m
20	XPART	DUM1	DUM2	-
21	N0	LN0	WN0	-
22	NB	LNB	WNB	-
23	ND	LND	WND	-

INTERCONNECTS

1	Rsh (RSH)	Cj (CJ)	Cjw (CJW)	Ijs (IJS)	Pj (PJ)
2	P_{jw} (PJW)	Mj (MJ)	Mjw (MJW)	Wdf (WDF)	δ_l (DL)

(b) The names of the process parameters of transistors are listed below:

V_{FB}	flat-band voltage
ϕ_s	surface inversion potential
K_1	body effect coefficient
K_2	drain/source depletion charge sharing coefficient
η_0	zero-bias drain-induced barrier lowering coefficient
μ_Z	zero-bias mobility
U_{0Z}	zero-bias transverse-field mobility degradation coefficient
U_{1Z}	zero-bias velocity saturation coefficient
μ_{ZB}	sensitivity of mobility to the substrate bias at $V_{ds} = 0$
η_B	sensitivity of drain-induced barrier lowering effect to the substrate bias
η_D	sensitivity of drain-induced barrier lowering effect to the drain bias, at $V_{ds} = V_{dd}$

U_{0B}	sensitivity of transverse-field mobility degradation effect to the substrate bias
U_{1B}	sensitivity of velocity saturation effect to the substrate bias
μ_S	mobility at zero substrate bias and at $V_{ds}=V_{dd}$
μ_{SB}	sensitivity of mobility to the substrate bias at $V_{ds} = V_{dd}$
μ_{SD}	sensitivity of mobility to the drain bias at $V_{ds} = V_{dd}$
U_{1D}	sensitivity of velocity saturation effect to the drain bias, at $V_{ds}= V_{dd}$
T_{ox}	gate-oxide thickness
T_{emp}	temperature at which the process parameters are measured
V_{dd}	measurement bias range
N0	zero-bias subthreshold slope coefficient
NB	sensitivity of subthrehold slope to the substrate bias
ND	sensitivity of subthrehold slope to the drain bias
CGDO	gate-drain overlap capacitance per meter channel width
CGSO	gate-source overlap capacitance per meter channel width
CGBO	gate-bulk overlap capacitance per meter channel length
XPART	gate-oxide capacitance model flag

Note: XPART= 0, 0.5, and 1 selects the 40/60, 50/50, and 0/100 channel-charge partitioning methods, respectively.

The names of the process parameters of diffusion layers are listed below:

sheet resistance/square	Rsh	Ω/square
zero-bias bulk junction bottom capacitance/unit area	Cj	F/m^2

zero-bias bulk junction sidewall capacitance/unit length	Cjw	F/m
bulk junction saturation current/unit area	Ijs	A/m^2
bulk junction bottom potential	Pj	V
bulk junction sidewall potential	Pjw	V
bulk junction bottom grading coefficient	Mj	-
bulk junction sidewall grading coefficient	Mjw	-
default width of the layer	Wdf	m
average reduction of size due to side etching or mask compensation	δ_s	m

---

The names of the process parameters of poly and metal layers are listed as following:

sheet resistance/square	Rsh	Ω/square
capacitance/unit area	Cj	F/m^2
edge capacitance/unit length	Cjw	F/m
default width of the layer	Wdf	m
average variation of size due to side etching or mask compensation	δ_l	m

3.4.6 Examples

(a) The following is an example of a process file. The lines starting with "*" are used as comments.

```
NM1  PY1  ML1  ML2  DU1  DU2
*PROCESS=PC1
*RUN=12 medium-size devices
*OPERATOR=Bing
*DATE=07/16/85
*
*  NMOS-1 PARAMETERS
*
-1.0087E+000.   -2.1402E-001.   3.44354E-001
7.96434E-001.   0.00000E+000.   0.00000E+000
1.31191E+000.   3.23395E-001.   -5.7698E-001
1.46640E-001.   1.68585E-001.   -1.8796E-001
-1.0027E-003.   -9.4847E-003.   1.47316E-002
5.34334E+002.   7.97991E-001.   4.77402E-001
4.38497E-002.   6.38105E-002.   -6.1053E-002
-5.7332E-002.   1.01174E+000.   1.62706E-002
8.25434E+000.   -2.4197E+001.   1.95696E+001
-7.6911E-004.   9.62411E-003.   -3.7951E-003
7.86777E-004.   7.35448E-004.   -1.7796E-003
1.06821E-003.   -8.0958E-003.   4.03379E-003
-1.9209E-002.   -7.4573E-002.   1.47520E-002
5.40612E+002.   6.21401E+002.   -1.9190E+002
```

```
-1.2992E+001.   -6.4900E+001.   4.29043E+001
-9.4035E+000.   1.18239E+002.   -2.9747E+001
7.76925E-003.   -1.0940E-001.   -8.3353E-003
3.00000E-002.   2.70000E+001.   5.00000E+000
2.70000E-010.   2.70000E-010.   1.40000E-010
1.0.            0.0.            0.0
1.5.5           0.0.            0.0
0.09.           0.0.            0.0
0.0.            0.0.            0.0
*
*  poly layer-1
*
30.0.           7.0E-5.         0.0.        0.0.    0.0
0.0.            0.0.            0.0.        0.0.    0.0
*
*  metal layer-1
*
0.040.          2.60E-5.        0.0.        0.0.    0.0
0.0.            0.0.            0.0.        0.0.    0.0
*
*  metal layer-2 (top metal)
*
0.030.          1.3E-5.         0.0.        0.0.    0.0
0.0.            0.0.            0.0.        0.0.    0.0
*
*  n+ diffusion layer
*
```

```
35.0.          2.75E-4.      1.90E-10.     1.0E-8.   0.7
0.8.           0.5.          0.33.         0.0.      0.0
*
*  p+ diffusion layer
*
120.0.         3.1E-4.       3.0E-010.     1.0E-8.   0.7
0.8.           0.5.          0.33.         0.0.      0.0
```

(b) The following deck determines the transient characteristics of a resistive load inverter with a capacitor connected at the output node.

```
SAMPLE CMOS INVERTER
VCC  1  0  5
SMN1  2  3  0  0  PC1_NM1_DU1  W=50U L=10U
SMP1  2  3  1  1  PC1_PM1_DU2  W=50U L=10U
VIN  4  0  PWL (0 0  5N 5  10N 5  20N 0)
RIN  4  3  PC1_PY1  L=20U  W=60U
C1  2  0  50FF
.OPTIONS  RELTOL=1E-5  CHGTOL=1E-15
.TRAN  0.5N  20N
.PRINT TRAN  V(3)  V(2)
.PLOT TRAN  V(3)  V(2)
.PROCESS PC1 FILENAME=CM0716
.WIDTH OUT=80
.END
```

3.5 Linked List Specifications

3.5.1 Resistor (ID=1)

	-1: subckt info		
LOC	+0: next pointer	LOCV	+0: element name
	+1: LOCV		+1: g (TEMP)
	+2: n1		+2: r (TNOM)
	+3: n2		+3: temp.coef. 1
	+4: (n1,n2)		+4: temp.coef. 2
	+5: (n2,n1)		+5: length
	+6: (n1,n1)		+6: width
	+7: (n2,n2)		
	+8: model type		
	+9: model pointer		
	+10: cycle number		

3.5.2 Capacitor (ID=2)

	-1: subckt info		
LOC	+0: next pointer	LOCV	+0: element name
	+1: LOCV		+1: computed element value
	+2: n1		+2: initial condition
	+3: n2		+3: argument vector
	+4: function code		+4: length
	+5: (n1,n2)		+5: width
	+6: (n2,n1)		
	+7: tp (function coefficients)		
	+8: LXi offset	LXi	+0: q(capacitor)
	+9: exponent vector		+1: i(capacitor)
	+10: (n1,n1)		
	+11: (n2,n2)		
	+12: model type		
	+13: model pointer		
	+14: cycle number		

3.5.3 BSIM MOS Device (ID=15)

	-1: subckt info			
LOC	+0: next pointer	LOCV	+0: element name	
	+1: LOCV		+1: effective length	
	+2: nd		+2: effective width	
	+3: ng		+3: drain diffusion area	
	+4: ns		+4: source diffusion area	
	+5: nb		+5: IC: vds	
	+6: nd'		+6: IC: vgs	
	+7: ns'		+7: IC: vbs	
	+8: model pointer 1		+8: device mode	
	+9: off		+9: von	
	+10: (nd.nd')		+10: vdsat	
	+11: (ng.nb)		+11: drain perimeter	
	+12: (ng.nd')		+12: source perimeter	
	+13: (ng.ns')		+13: # square of drain diff.	
	+14: (ns.ns')		+14: # square of source diff.	
	+15: (nb.ng)		+15: vto	
	+16: (nb.nd')		+16: XPART	
	+17: (nb.ns')		+17: VFB	
	+18: (nd'.nd)		+18: PHI	
	+19: (nd'.ng)		+19: K1	
	+20: (nd'.nb)		+20: K2	
	+21: (nd'.ns')		+21: ETA	
	+22: (ns'.ng)		+22: BETAZ	
	+23: (ns'.ns)		+23: U0	
	+24: (ns'.nb)		+24: U1	
	+25: (ns'.nd')		+25: X2BZ	
	+26: LXi offset		+26: X2E	
	+27: (nd.nd)		+27: X3E	
	+28: (ng.ng)		+28: X2U0	
	+29: (ns.ns)		+29: X2U1	
	+30: (nb.nb)		+30: BETAS	
	+31: (nd'.nd')		+31: X2BS	
	+32: (ns'.ns')		+32: X3BS	
	+33: model type 1		+33: X3U1	
	+34: model pointer 2		+34: COVLGD	+37: N0
	+35: model type 2		+35: COVLGS	+38: NB
	+36: cycle number		+36: COVLGB	+39: ND

Lxi	+0: VBDO	+16: QD
	+1: VBSO	+17: iQD
	+2: VGSO	+18: CGGBO
	+3: VDSO	+19: CGDBO
	+4: id	+20: CGSBO
	+5: ibs	+21: CBGBO
	+6: ibd	+22: CBDBO
	+7: gm	+23: CBSBO
	+8: gds	+24: QBD
	+9: gmbs	+25: iQBD
	+10: gbd	+26: QBS
	+11: gbs	+27: iQBS
	+12: QB	+28: CDGBO
	+13: iQB	+29: CDDBO
	+14: QG	+30: CDSBO
	+15: iQG	

Comments :

(1) vto=VFB+PHI+K1*$\sqrt{PHI}$ -K2*PHI

(2) device mode: +1(-1) for normal(inverse).

(3) If RSH $\neq$ 0.0D0. and NRD $\neq$ 0.0D0

then at LOCV+13: GDPR=1.0D0/(RSH*NRD)

If RSH $\neq$ 0.0D0. and NRS $\neq$ 0.0D0

then at LOCV+14: GDPR=1.0D0/(RSH*NRS)

3.5.4 BSIM Model (ID=25)

LOC	-1: subckt info	+1: LOCV		
	+0: next pointer	+2: device type		
LOCV	+0: process name			
	+1: VFB	+22: U1	+43: X2MS	+64: NB
	+2: LVFB	+23: LU1	+44: LX2MS	+65: LNB
	+3: WVFB	+24: WU1	+45: WX2MS	+66: WNB
	+4: PHI	+25: X2MZ	+46: X3MS	+67: ND
	+5: LPHI	+26: LX2MZ	+47: LX3MS	+68: LND
	+6: WPHI	+27: WX2MZ	+48: WX3MS	+69: WND
	+7: K1	+28: X2E	+49: X3U1	
	+8: LK1	+29: LX2E	+50: LX3U1	
	+9: WK1	+30: WX2E	+51: WX3U1	
	+10: K2	+31: X3E	+52: TOX	
	+11: LK2	+32: LX3E	+53: TEMP	
	+12: WK2	+33: WX3E	+54: VDD	
	+13: ETA	+34: X2U0	+55: CGDO	
	+14: LETA	+35: LX2U0	+56: CGSO	
	+15: WETA	+36: WX2U0	+57: CGBO	
	+16: MUZ	+37: X2U1	+58: XPART	
	+17: DL	+38: LX2U1	+59: DUM1	
	+18: DW	+39: WX2U1	+60: DUM2	
	+19: U0	+40: MUS	+61: N0	
	+20: LU0	+41: LMS	+62: LN0	
	+21: WU0	+42: WMS	+63: WN0	

Comments :

(1) device type:

1 for NMOS

-1 for PMOS

(2) XPART:

1 for the 0/100 channel-charge partitioning method.

0 for the 40/60 channel-charge partitioning method.

0.5 for the 50/50 channel-charge partitioning method.

3.5.5 Interconnect Model (ID=26)

	-1: subckt info		
LOC	+0: next pointer	LOCV	+0: process name
	+1: LOCV		+1: RSH
	+2: interconnect type		+2: CJ
			+3: CJW
			+4: IJS
			+5: PJ
			+6: PJW
			+7: MJ
			+8: MJW
			+9: WDF
			+10: DL

Comments :

(1) interconnect type:

1, 2, 3, 4, 5, and 6 for DU1, DU2, DU3, DU4, DU5, and DU6.

7, 8, 9, and 10 for PY1, PY2, PY3, and PY4.

11, 12, 13, and 14 for ML1, ML2, ML3, and ML4.

Chapter 4: The Substrate Current And Lifetime Program (SCALP)

Both the substrate current model and the device degradation model introduced in chapter 1 have been implemented in SCALP (the Substrate Current And Lifetime Program). SCALP contains a pre- and a post- processor which when used in conjunction with SPICE allows for the simulation of substrate current and device lifetime in analog and digital circuits. No modification to the original SPICE code is necessary for this implementation; consequently the SCALP simulator can be used with minimal setup time and works independently of the specific models implemented in SPICE. Using SCALP and SPICE together as a simulator system, the circuit designer can isolate areas within his circuit susceptible to adverse hot-electron and degradation effects.

The following sections will summarize the basic operation of this new simulator system, describe its internal structures, and offer a guide to users wanting to implement these models in their circuit simulations. The degradation and lifetime models used in SCALP are described in chapter 1.

Users who are not interested in using the degradation model in their circuit simulation should go directly to the last paragraph of section 4.1.

4.1 Process file

The degradation model parameters needed for the process file are (m and B) or (m, n, and H) and should be experimentally obtained by the user before he runs SCALP. The following paragraphs will briefly describe how these parameters are determined.

A lifetime versus I_{BS} experiment should be performed for at least one ΔV_{thf}. A typical result is shown in Fig. 4.

(1) m: The ratio of the trap energy and impact ionization energy $\frac{\phi_{it}}{\phi_i}$ which is approximately the slope of the log τ versus log I_{bs} curve in Fig. 4. A typical value for m is about 3 [3].

(2) B or (H and n): B is approximately the y-intercept of the curve in Fig. 4, and the unit used for B in this program is A-sec/μm. For each ΔV_{thf}, there is a value of B corresponding to it. The values of n and H can be obtained by fitting B values at various ΔV_{thf}'s to (46). H is dependent on the device processing technology. Typical values for n range from 0.5 to 0.75.

Note : H and n are determined from stressing experiments for at least two different ΔV_{thf}'s. However, the determination of the value of n is difficult and inaccurate, because n is a function of processing technology and bias condition. At the present, no empirical model for n has been developed. On the other hand, B can be more easily and accurately determined from stressing data for only one ΔV_{thf}. Thus, using B and m instead of H, n, and m offers more accuracy in the simulation. SCALP provides the flexibility to use either (B and m) or (H, n, and m) as the degradation parameters in the process file. To activate the first option, the users have to set the value of parameter n to zero and replace H by B. For example, if the last line of the process file is "1E-2, 0.5, 3.5", SCALP assumes parameters H, n, and m are used. If the last line of the process file is "1e-6, 0, 3.5", SCALP assumes that parameters B and m are used, because the value of n is zero. The user should note that the lifetime calculated by using the first option is based on the same ΔV_{thf} used to extract B.

The degradation parameters should be appended to the process file directly after the substrate current parameters in one row and in the order of H, n, and m, (see Fig. 19). Users who do not need any device lifetime information should enter 0,0,0 for H, n, and m respectively and use the appropriate commands given in section 4.3. The location of all BSIM parameters in the process file for proper SPICE read-in and a sample of the process file can also be found in Fig. 19.

4.2 System Structure and Implementation

This system contains three parts: a pre-processor, the SPICE program, and a post-processor. The two newly developed models are installed in the post-processor and are external to SPICE.

The pre-processor takes an input deck which contains commands given in section four and commands prepared according to the SPICE syntax. SPICE is used in this system to simulate the circuit to determine the transient voltage waveforms at the drain, gate, source, and substrate of all user-selected devices. The post-processor calculates the transient substrate currents by using the node voltages provided by SPICE.

For the SPICE analysis, the user is free to choose any one of the models available in SPICE for the circuit simulation. To avoid confusion, from here on we shall refer to the SPICE2 BSIM model as "BSIM2", other SPICE2 models as "SPICE2", the SPICE3 BSIM model as "BSIM3", and other SPICE3 models as "SPICE3". Besides the basic tasks, the operations of the pre- and post- processors differ somewhat for the BSIM2, BSIM3, SPICE2, and SPICE3 users. This was necessary because BSIM2 reads in the process file created by the BSIM extraction program directly whereas BSIM3, SPICE2, and SPICE3 need .MODEL commands with all parameters appended to the input deck. The system configuration was designed to retain as much commonality as possible between the two schemes (see Fig. 22). The "rawsub" and "RAWMODEL" files are created for communication between the two processors and for storing model parameters. The "RAWPROC" files are the new process files created when BSIM2 is used. All these intermediate files are

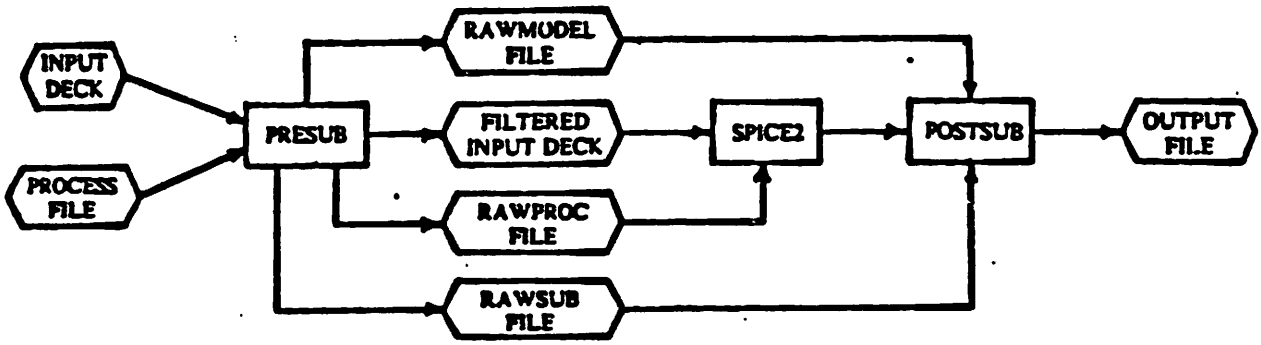


Fig.22a BSIM2 processor configuration.

removed at the end of the simulation. More detailed discussion will be given below.

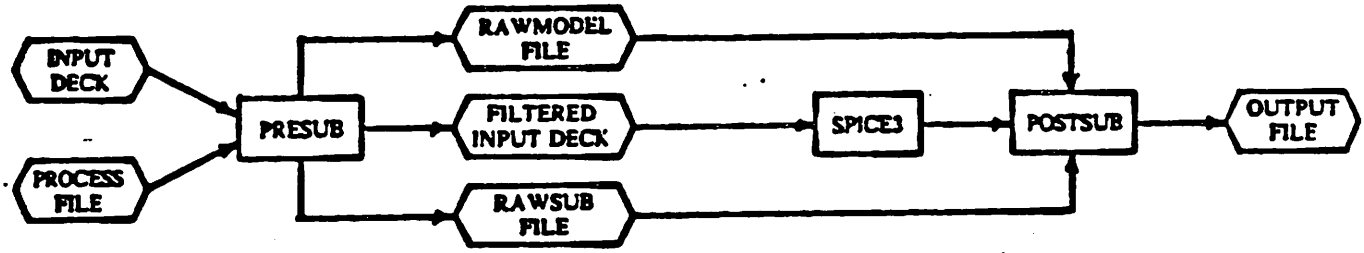


Fig.22b BSIM3/SPICE2/SPICE3 processor configuration.

4.2.1 Structure of the Pre-Processor:

The program name for the pre-processor is "prescalp.c". All global variables used by this program are declared in "BSIMpredefs.h". Other associated program names needed for compilation are "ACdefs.h", "CKTdefs.h", "FTEconstant.h", "FTEwritedata.h", "GENdefs.h", "INPdefs.h", "NIdefs.h", "OUTinterface.h", "Proc2ModSub.c", "SMPdefs.h", "SPerror.h", "TRCVdefs.h", "prefix.h", "suffix.h", "util.h", and "version.h". The program "prescalp.c" contains six main routines. The functions of these main routines are described below:

(1) PreFilter:

This routine scans the entire input file to determine the number of transistors in the circuit, the number of process files given, and the number of transistors given in the .PRINT or .PLOT ISUB commands. This is done in order to allocate the proper amount of memory spaces for data storage.

(2) FindProc:

The .PROCESS control line is searched for to obtain the process file name. For each process file given, a corresponding model file is created which contains all the process parameters with the substrate current and degradation parameters commented out. When BSIM2 is used, a new process file is created which is identical to the original process file except for the substrate and degradation parameters are commented out. The filenames for the model file and the new process file are "RAWMODEL0" and "RAWPROC0", respectively. When more than one process file is given, "RAWPROC1", "RAWPROC2", etc., will be created. Since SPICE only recognizes the first eight characters of a given process filename, the new process filename becomes "RAWPRO10" when more than nine process files are given. When more than 99 files are given, the new process filename becomes "RAWP100". Thus, the maximum number of .PROCESS lines that can be given in an input deck is set to be 9999.

(3) FindWidth, FindTran, FindOption:

FindWidth and FindTran search for the .ISUBWIDTH and .TRAN control lines and store these information in the rawsub file. The FindOption routine looks for the relevant information (i.e. DEFL, DEFW, NOMOD, etc.) that may appear in the .OPTIONS command.

(4) CreateInpFile:

The filtered input deck for SPICE is created in this routine. All non-SPICE-compatible commands are commented out, and the appropriate .PRINT node voltages commands are added to the input deck. Information pertaining to the user-selected transistors are stored in the "rawsub" file. For BSIM2 users, the filename in the .PROCESS control line is changed to "RAWPROC0"; for BSIM3 users, the "RAWMODEL0" file is appended to the input deck, and the .PROCESS command is deleted by the pre-processor; for SPICE2 and SPICE3 users, the .PROCESS command is deleted. All of these are done to maintain compatibility with the different versions of the SPICE codes.

4.2.2 Structure of the Post-Processor:

The code for the post-processor is stored in "scalp.c", and all global variables are defined in "BSIMpostdefs.h". This program is divided into two main sections. The first part reads in all transistor information from the "rawsub" and the "RAWMODEL" files and stores them. The second part reads in the voltage values from SPICE, calculates the substrate current and device lifetime, and prints the output. Several main functions from this part will be discussed here.

(1) AddSubParam:

If either SPICE2 or BSIM2 is being run and the NOMOD option is not present in the .OPTIONS control line as determined by the pre-processor earlier, then the substrate current parameters are appended to the model information section of the SPICE output.

(2) ReadVoltage:

This routine is used to search the right set of output voltages from SPICE for the user-selected transistors. This routine is dependent on the SPICE output format in its search and therefore may need modification if SPICE output format is changed.

(3) BSIMevaluate:

The BSIM substrate current model is implemented is this routine. Given all the process parameters and specified node voltages, this routine calculates the substrate current.

(4) BSIMDeltaVth, EvaluateLifetime:

Implementation of the degradation model is done in these two routines. Device lifetime is calculated based on the transient substrate current.

(5) SubOutput, PrintLifetime:

These two routines are responsible for the printing and plotting of the substrate current and the device lifetime.

4.3 User's Guide for the Processors

Since the input format for SPICE3, SPICE2, BSIM2 and BSIM3 are all different, the user needs to pay special attention to the command syntax that is applicable to his use. The following new commands have been added for substrate current and degradation analysis:

.PRINT ISUB(MXXX) < ISUB(MXXX) ... ISUB(MXXX) >

.PRINT ISUB(SXXX) < ISUB(SXXX) ... ISUB(SXXX) >

.PLOT ISUB(MXXX) < ISUB(MXXX) ... ISUB(MXXX) > < (MIN,MAX) >

.PLOT ISUB(SXXX) < ISUB(SXXX) ... ISUB(SXXX) > < (MIN,MAX) >

Examples:

.PLOT ISUB(S1) ISUB(S4) (0,7E-6)

.PRINT ISUB(M1) ISUB(M4) ISUB(M5)

These control lines are used to either print or plot out the substrate current. The device lifetime is always printed out following the output of the corresponding substrate current unless otherwise specified. The sums of the specified substrate current for the NMOS and PMOS devices are also automatically printed out. This is a good test to determine if the substrate-bias generator will be overloaded by the substrate current. SXXX is a syntax following the BSIM2 convention and should be used for both BSIM2 and SPICE2, while MXXX is used for SPICE3 and BSIM3. MIN and MAX are optional parameters specifying the minimum and maximum substrate current to be plotted.

.ISUBALL

This command will enable the SCALP simulator to provide the sums of the substrate current for all MOS devices in the circuit regardless of transistors specified in the .PRINT or .PLOT ISUB command.

.ISUBONLY

This command should be used if the user is not interested in the device lifetime calculation. When this command is given, only the substrate current information is provided by the SCALP simulator.

.LIFE

When the user wishes to know the lifetimes of all devices in the circuit, this command can be given. This command overrides the .ISUBONLY command and can be used together with the .PRINT or .PLOT ISUB command.

.ISUBWIDTH=COLWIDTH

Examples:

.ISUBWIDTH=80

.ISUBWIDTH=100

This command controls the width of the substrate current output printout. This is independent of the usual .WIDTH or .OPTIONS WIDTH command used in SPICE. Permissible values for COLWIDTH range from 80 to 200. The default value is 132. In SPICE3 or BSIM3, all non-substrate current analysis printout is outputted in 90 column format, regardless of the .OPTIONS WIDTH command. This was done to insure proper voltage read-in when none of the four nodes of the specified transistor is grounded.

.DELTAVT VALUE

Examples:

.DELTAVT 10MV

.DELTAVT 900UV

This control line is used for setting the amount of threshold voltage shift defined at device failure. The calculation of device lifetime is based on this given value.

.TRAN TSTEP TSTOP <TSTART>

Examples:

.TRAN 1NS 100NS

.TRAN 5NS 1000NS 2NS

Since this simulator system is designed to calculate the transient substrate currents, the SPICE .TRAN command should always be included whenever this system is used. In order for the calculated device lifetime to be meaningful, the difference between TSTOP and TSTART should equal to a multiple of the period of the input signal.

.PROCESS PNAME FILENAME=FNAME

Examples:

.PROCESS PC1 FILENAME=RUN

.PROCESS MK1 FILENAME=MMK

This control line specifies the process name and the corresponding process file name which contains all the process parameters. The format is identical to that already implemented in BSIM2, but it is new for SPICE2, SPICE3 and BSIM3. It is important to realize that ".MODEL" commands are no longer necessary for BSIM3, but that a ".PROCESS" command is now mandatory.

General form for MOSFETs:

SXXXXXXX ND NG NS NB <MNAME> PNAME_MT_DT <L=VAL> <W=VAL> ...

etc.

MXXXXXXX ND NG NS NB <MNAME> PNAME_MT_DT <L=VAL> <W=VAL> ...

etc.

Examples:

S1 1 2 3 4 MODN PC1_NM1_DU1 L=1U W=20U

```
S1 1 2 3 4 PC1_NM1_DU1 L=1U W=20U

M1 1 2 3 4 MODP PC3_PM1_DU2 L=10U W=5U AD=100P AS=100P

+ PD=40U PS=40U
```

To describe a MOSFET, the user should use SXXXXXXX for BSIM2 or SPICE2, MXXXXXXX for BSIM3 or SPICE3. MNAME is the model name which should be given only if SPICE2 or SPICE3 is used. PNAME_MT_DT should be given regardless of the type of model used in the SPICE analysis. PNAME is the process name which must be specified in a .PROCESS command. MT is the mos type. The possible choices are NM1 to NM5, and PM1 to PM5. DT is the source/drain junction diffusion type. DU1 to DU3 ar the three available diffusion types with DU1 used as the default value. For users who are not familiar with SPICE commands, please consult the manual for SPICE. For users who wish to learn more about the BSIM model implemented in SPICE or about the BSIM extraction program, please refer to chapters 1 and 2.

In an UNIX environment, the commands for compilation are:

```
cc -o prescalp prescalp.c -lm

cc -o scalp scalp.c -lm
```

where "prescalp" and "scalp" are the names of the executable files and "prescalp.c" and "scalp.c" are the names of the programs. To execute the programs, the command is:

```
prescalp -x -y deck | spice | scalp > outfile
```

where x is "b" if the BSIM model is used in the SPICE analysis, or "o" if other model is used, y is either "2" or "3" depending on whether SPICE2/BSIM2 or SPICE3/BSIM3 is used, "deck" is the input deck file, "spice" is the name of the executable version of the SPICE program the user is using, and "outfile" is the output file desired. If no "-x" option is specified, the BSIM model is assumed to be used; if no "-y" option is given, the simulator defaults to SPICE2/BSIM2.

Chapter 5: Examples

In this chapter, we will show examples and results of the parameter extraction program, SPICE, and SCALP. Some of figures have been shown in previous chapters, but for users' convenience, we will show them in this chapter too. There are three sections in this chapter. Each section is associated with the parameter extraction program, SPICE, and SCALP, respectively.

5.1 Parameter Extraction Program

5.1.1 Probe File

To extract parameters for multiple devices (Mode 1,2,3), a probe file which contains information of the device dimensions, device positions on the die, and SMU connections for HP 4145 has to be created by the user. Because the program will read the probe file to determine the measuring sequence. Detailed description of the format of a probe file can be found in Section 2.6.

Examples of the probe file for operation mode 1, 2 (using automatic probe station) and mode 3 (using manual probe station) are shown in Figs. 23a and 23b. In both examples, the number of devices specified in the probe file is 4.

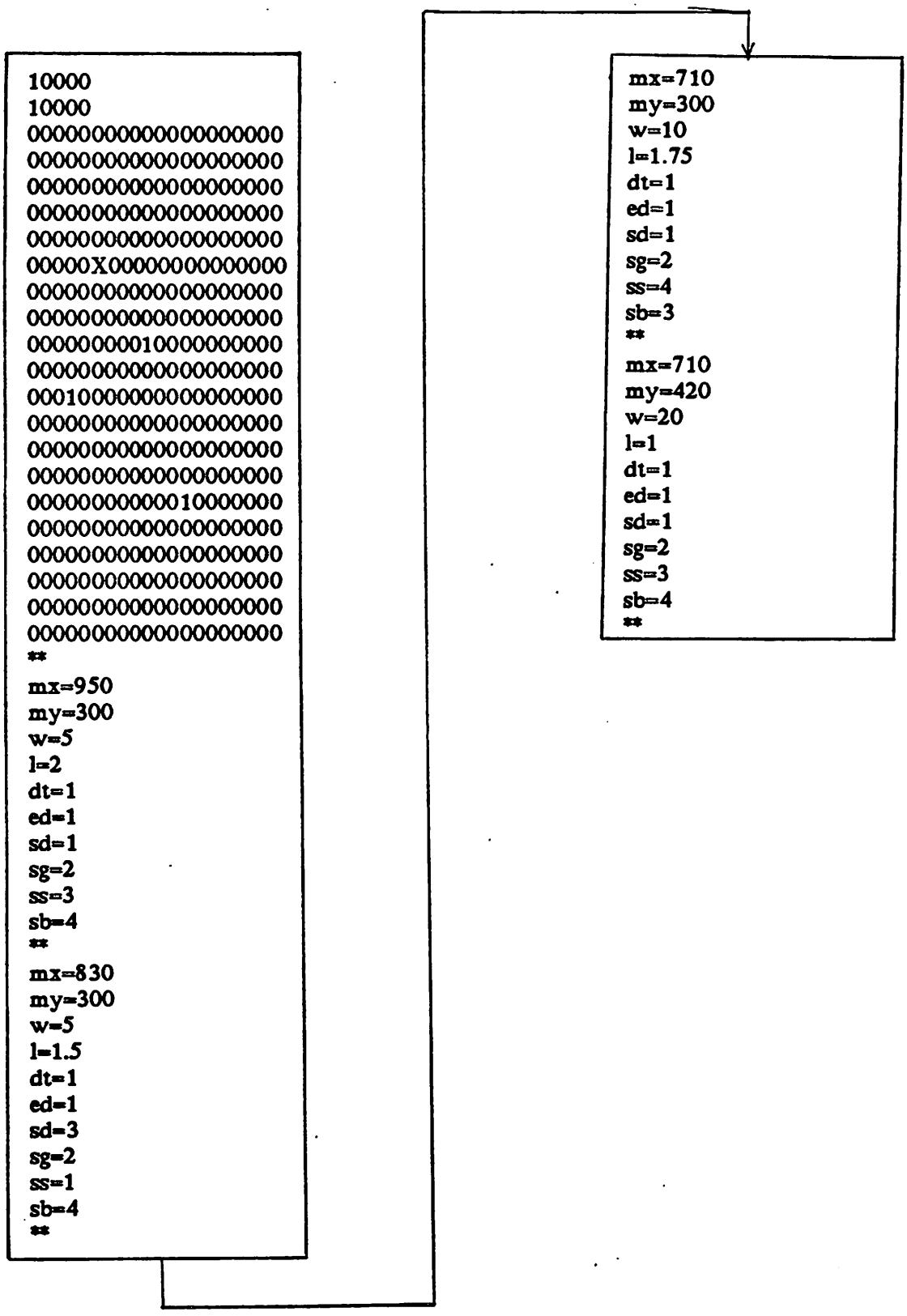


Fig.23a An Example of the prober file for automatic probe stations.

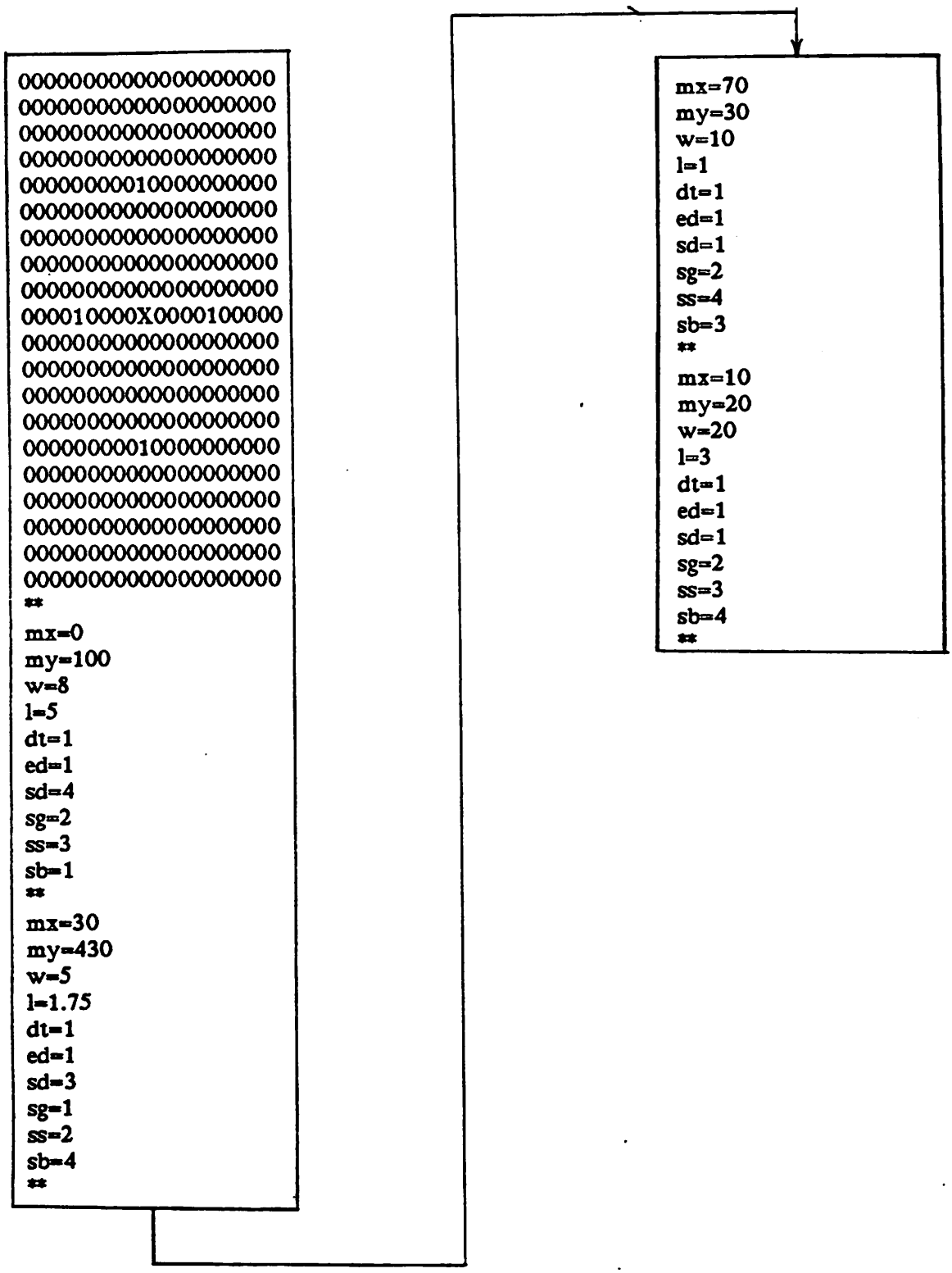


Fig.23b An example of the prober file for manual probe stations.

Note: (1). for operation model 4 (single device). the probe file is not needed. (2) for operation mode 3. the device locations on the die (given by mx=? and my=?) can be any number. because it will not be used. (3) a blank line is not allowed in the probe file. This usually happens on the last line of the probe file. When you edit a probe file using the "EDITOR" of the Pascal system. be sure that the cursor is on the same line as the last "**" and stops at the position next to the right of the last "**" in the probe file. Otherwise. an error message "error -8: value range error" appears at the bottom of the screen. (4) the probe file has to be copied into RAM before the program read it.

5.1.2 Process File

The process file generated by the parameter extraction program has to different formats depending on whether the substrate current parameters are extracted as were explained in chapter 2. Both of these formats are shown in Fig.24.

```
NM1
*PROCESS=
*RUN=
*WAFER=
*XPOS=6
*YPOS=5
*OPERATOR=Min-Chie
*DATE=
-8.9759E-001,3.53530E-002,-2.1152E-001,0.00000E+000,1.00000E+001,1.00000E+000
8.86997E-001,0.00000E+000,0.00000E+000,0.00000E+000,1.00000E+001,1.00000E+000
8.86404E-001,4.01989E-002,1.28249E-001,0.00000E+000,1.00000E+001,1.00000E+000
3.05574E-002,3.27008E-002,5.39528E-002,0.00000E+000,1.00000E+001,1.00000E+000
-2.7162E-003,2.07369E-002,-6.6789E-002,1.21497E-012,2.00000E+000,1.00000E+000
5.26958E+002,7.06713E-002,-1.5136E+000,0.00000E+000,0.00000E+000,0.00000E+000
1.47767E-001,1.18850E-001,-1.5492E-001,1.33999E-014,5.00000E+000,1.50000E+000
-2.0636E-001,3.90394E-001,1.33614E+000,3.36664E-014,1.00000E+001,1.00000E+000
4.43613E+001,-2.9336E+001,-5.9551E+001,1.06881E-013,2.00000E+000,1.00000E+000
6.27101E-003,-2.7073E-003,-2.4331E-002,5.22654E-014,1.00000E+001,1.00000E+000
4.22817E-003,-6.8250E-003,1.37042E-002,1.39256E-014,5.00000E+000,1.50000E+000
4.20822E-002,-3.4792E-002,-8.0149E-002,6.38279E-014,5.00000E+000,1.50000E+000
-2.0904E-002,3.33535E-003,7.39045E-002,3.84853E-013,2.00000E+000,1.00000E+000
3.48429E+002,2.67006E+002,4.14273E+002,0.00000E+000,1.00000E+001,1.00000E+000
2.67995E+001,-3.5421E+001,-1.5121E+001,3.47548E-013,5.00000E+000,1.50000E+000
-7.1520E+001,1.28544E+002,1.16723E+002,3.91123E-014,5.00000E+000,1.50000E+000
-6.7912E-002,1.51994E-001,1.76767E-001,2.11660E-014,5.00000E+000,1.50000E+000
8.80000E-003,2.70000E+001,3.00000E+000
0.00000E+000,0.00000E+000,0.00000E+000
1.00000E+000,0.00000E+000,0.00000E+000
1.50095E+000,2.03825E-002,-5.0062E-002,1.47183E-014,2.00000E+000,1.00000E+000
9.81471E-002,3.00207E-003,8.89480E-003,0.00000E+000,1.00000E+001,1.00000E+000
2.12815E-003,-1.1620E-003,-9.4879E-003,7.81198E-013,1.00000E+001,1.00000E+000
```

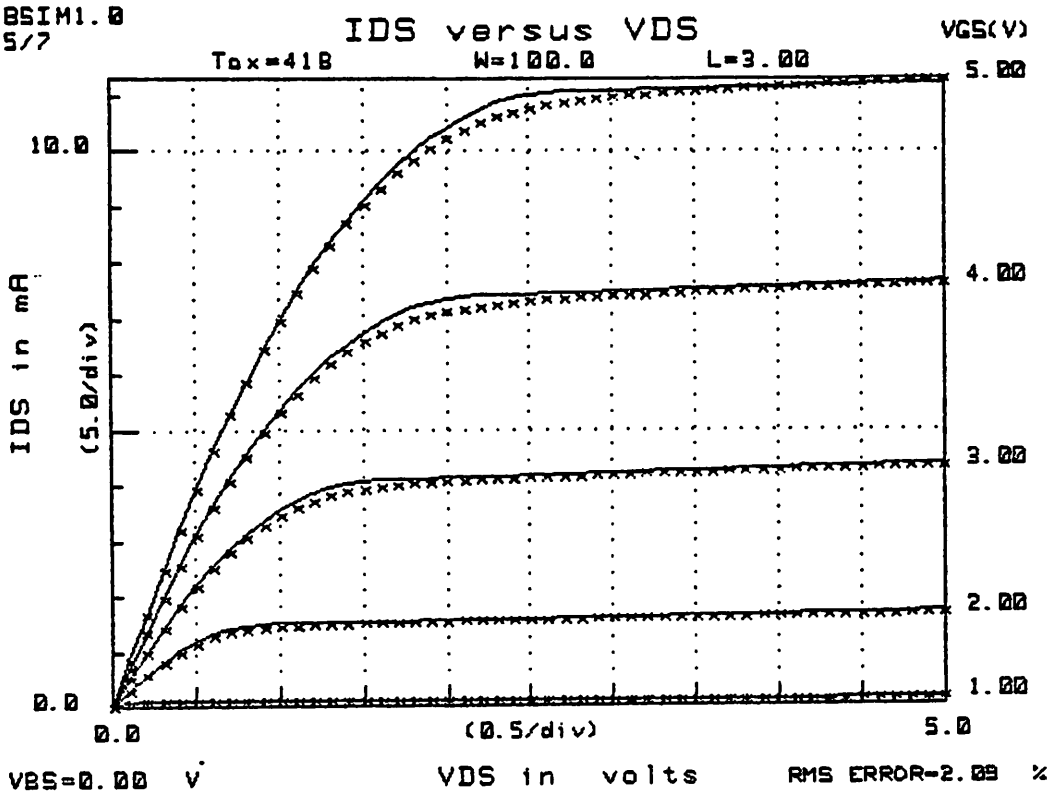
Fig.24a An example of the process file generated by the parameter extraction program (without substarte parameter).

```
NM1
*PROCESS=
*RUN=
*WAFER=
*XPOS=6
*YPOS=5
*OPERATOR=Min-Chie
*DATE=
-8.3024E-001,-7.8436E-002,1.10017E-001,4.47423E+000,1.00000E+001,1.00000E+000
8.85520E-001,0.00000E+000,0.00000E+000,0.00000E+000,5.00000E+001,2.00000E+000
8.56061E-001,9.84218E-002,-1.5991E-001,6.57344E+000,1.00000E+001,1.00000E+000
2.50033E-002,4.79264E-002,-6.7575E-002,4.00007E+001,1.00000E+001,1.00000E+000
1.28359E-002,-1.5952E-003,-3.0589E-002,3.03806E+002,5.00000E+000,1.50000E+000
4.15104E+002,4.10997E-001,-9.3679E-001,0.00000E+000,0.00000E+000,0.00000E+000
1.30577E-001,1.09196E-001,-1.8703E-001,1.39848E+001,1.00000E+001,1.00000E+000
-1.4536E-002,1.10045E-001,5.86745E-001,1.09906E+001,1.00000E+001,1.00000E+000
3.13590E+001,-1.7583E+001,-1.4715E+001,9.68274E+001,1.00000E+001,1.00000E+000
5.91513E-003,-4.9733E-003,-1.8650E-003,1.51074E+003,2.00000E+000,1.20000E+000
-1.4312E-003,1.54331E-003,-2.5037E-004,5.82962E+002,2.00000E+000,1.20000E+000
3.46888E-002,-2.6133E-002,-2.3228E-002,4.09408E+002,1.00000E+001,1.00000E+000
-1.0250E-003,3.95145E-003,-2.9756E-002,2.88871E+002,2.00000E+000,1.00000E+000
3.12029E+002,1.43396E+002,1.07716E+002,2.68085E+001,1.00000E+001,1.00000E+000
1.63953E+001,-1.0309E+001,-3.8036E+001,3.04027E+002,5.00000E+000,1.50000E+000
-3.8569E+001,5.85863E+001,5.55863E+001,1.29377E+002,5.00000E+001,2.00000E+000
-1.1784E-002,3.69518E-002,1.25711E-001,2.74758E+001,5.00000E+001,2.00000E+000
8.80000E-003,2.70000E+001,3.00000E+000
0.00000E+000,0.00000E+000,0.00000E+000
1.00000E+000,0.00000E+000,0.00000E+000
1.52212E+000,8.91195E-004,-4.3505E-002,5.36756E-001,5.00000E+000,1.50000E+000
1.05451E-001,-5.6017E-003,1.75837E-002,4.95397E+000,5.00000E+000,9.00000E-001
1.80311E-003,-1.9350E-003,-2.0996E-003,3.21277E+003,1.00000E+001,1.00000E+000
1.87491E+003,6.06880E+003,1.59818E+004,2.40178E+002,5.00000E+001,2.00000E+000
1.10318E+004,-2.9716E+003,-1.5572E+004,7.26017E+001,5.00000E+000,1.50000E+000
-4.6886E+003,2.43117E+003,4.80083E+003,2.27912E+002,5.00000E+000,1.50000E+000
7.73605E+000,-2.2467E+000,-4.0804E+000,1.91337E+001,5.00000E+000,9.00000E-001
2.85779E+001,-7.9501E+000,-1.2002E+001,2.33365E+001,5.00000E+000,9.00000E-001
-1.6989E+000,6.39380E-001,1.28226E+000,1.67552E+002,5.00000E+000,9.00000E-001
-7.7179E+000,2.20032E+000,3.53552E+000,2.75710E+001,5.00000E+000,9.00000E-001
-2.0741E+001,7.90330E+000,1.69143E+001,1.62472E+002,5.00000E+000,9.00000E-001
-9.3658E+001,2.90995E+001,5.18305E+001,3.11843E+001,5.00000E+000,9.00000E-001
5.40688E+000,-2.3481E+000,-5.3418E+000,5.22114E+001,5.00000E+000,9.00000E-001
2.64737E+001,-8.4534E+000,-1.5533E+001,3.98943E+001,5.00000E+000,9.00000E-001
0.00000E+000,0.00000E+000,0.00000E+000
```

Fig.24b An example of the process file generated by the parameter extraction program (with substarte parameter).

5.1.3 Playback

The parameter extraction program has the capability to plot the following calculated and measured device characteristics: (1) I_{DS} - V_{DS}. (2) I_{DS} - V_{GS}. (3) $\log(I_{DS})$ - V_{GS}. (4) G_m - V_{GS}. (5) G_d (or R_d) - V_{DS}. (6) $\log(I_{BS}/I_{DS})$ - V_{DS}. (7) I_{BS} - V_{GS}. In Fig. 25 we will show the extracted and measured results for all the device characteristics mentioned above for various process technologies.



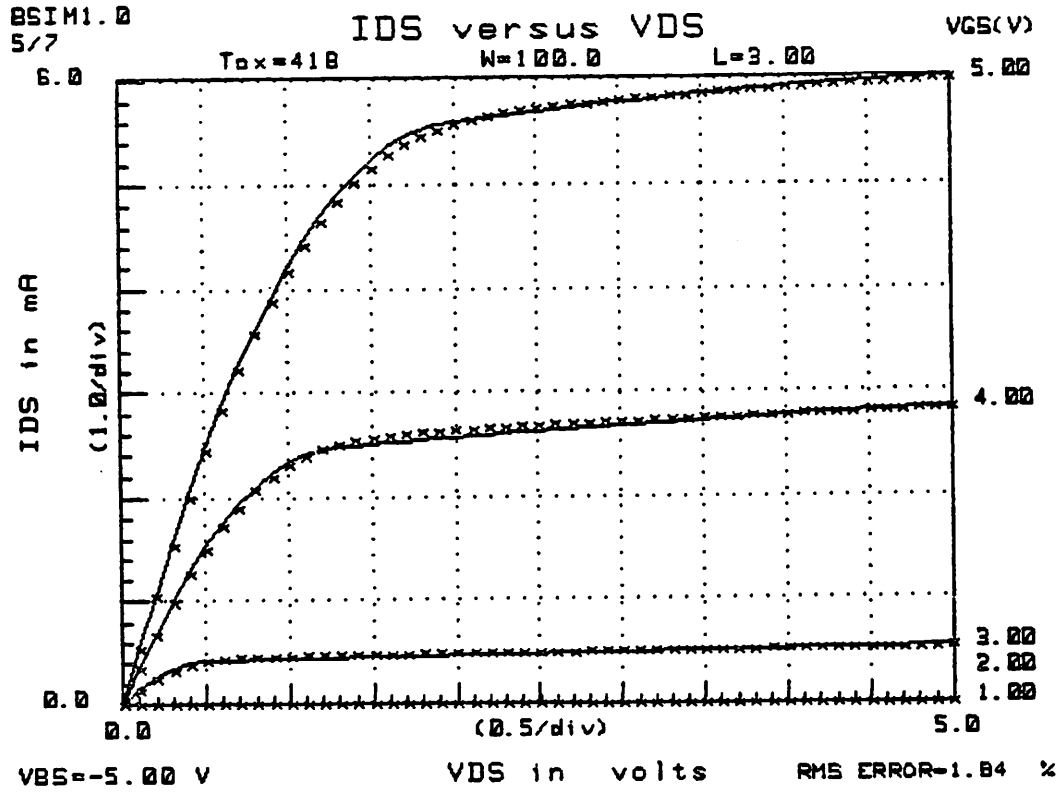
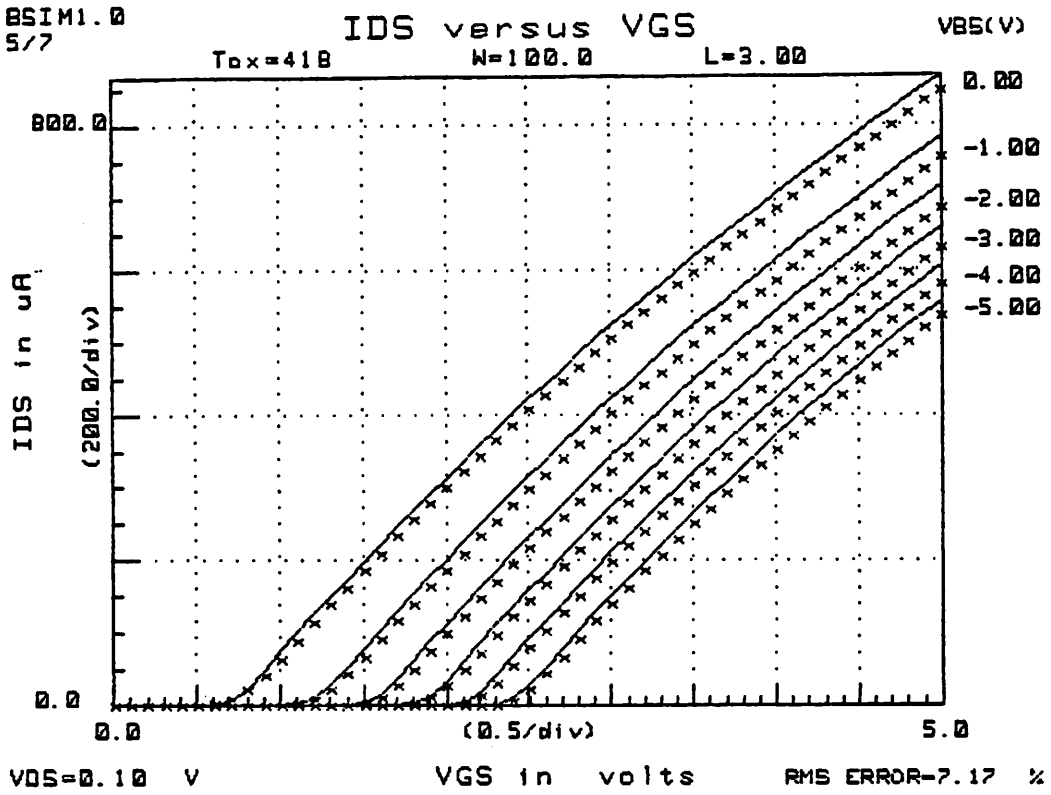


Fig.25a Bell Lab. device. W_{eff} = 100 μm. L_{eff} = 2.2 μm.



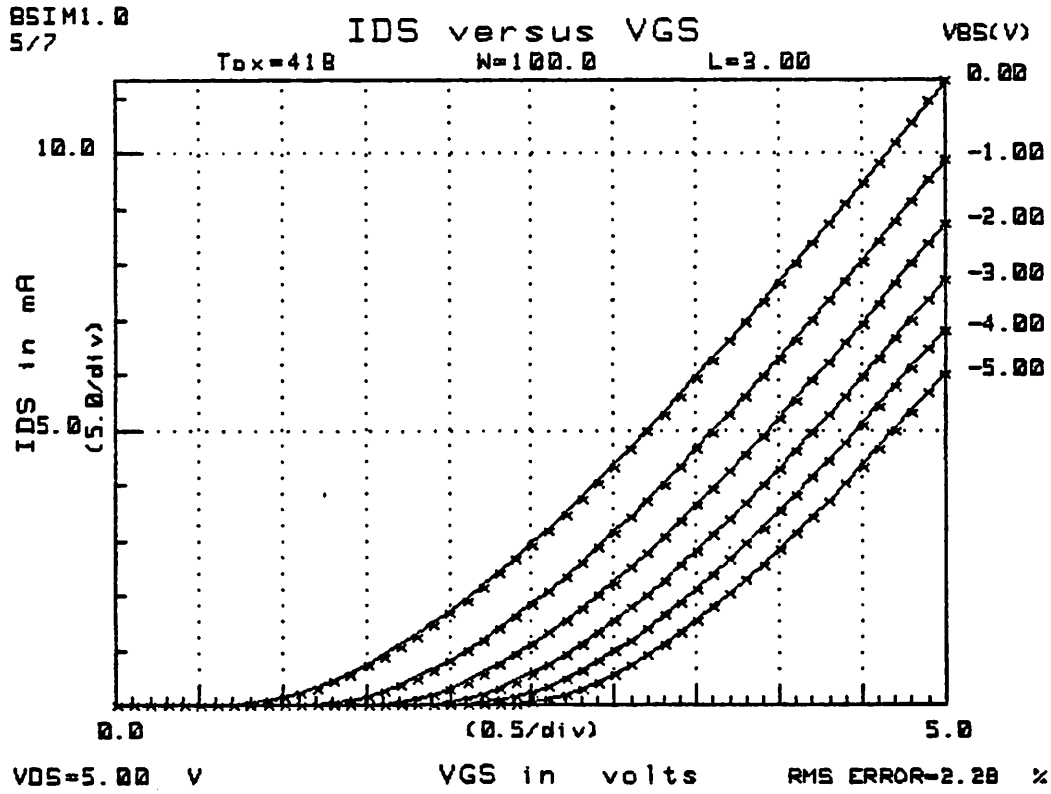
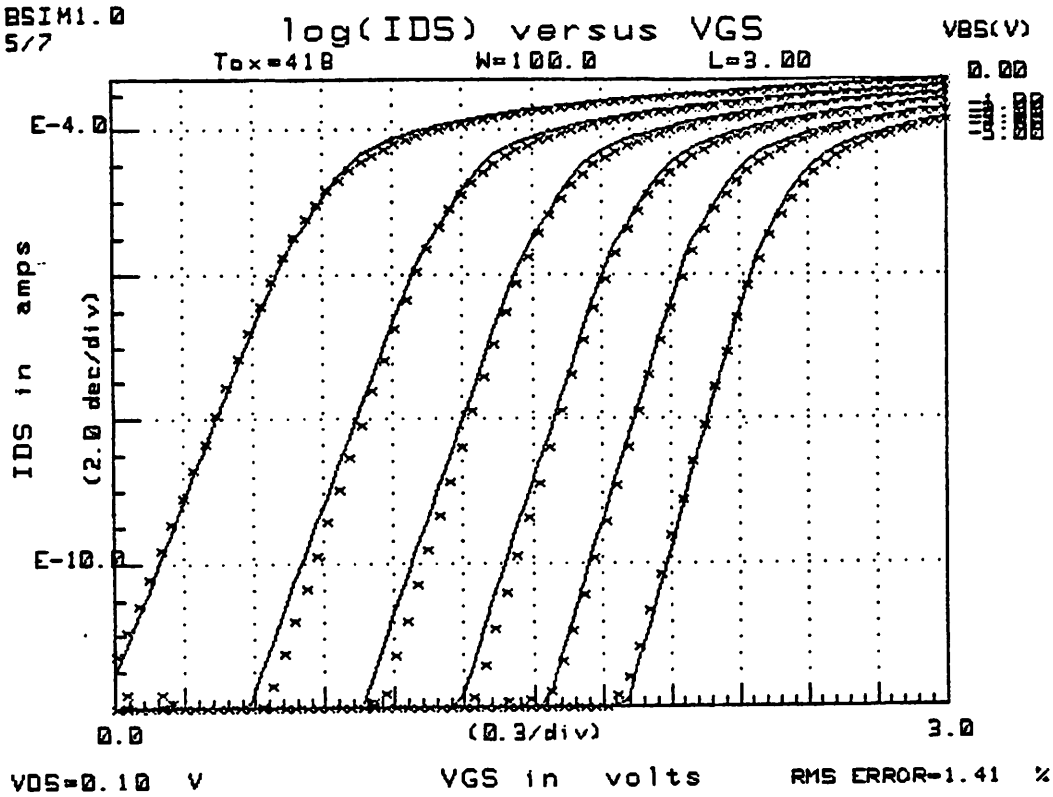


Fig.25b Bell Lab. device. W_{eff} = 100 μm, L_{eff} = 2.2 μm.



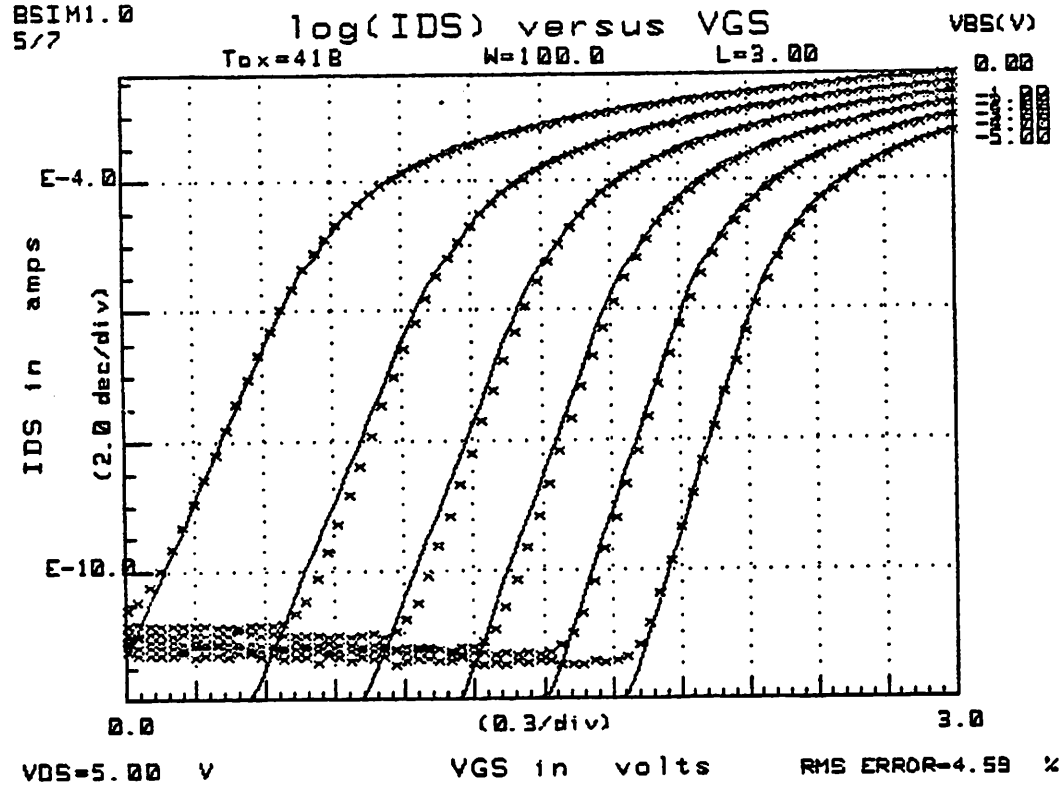
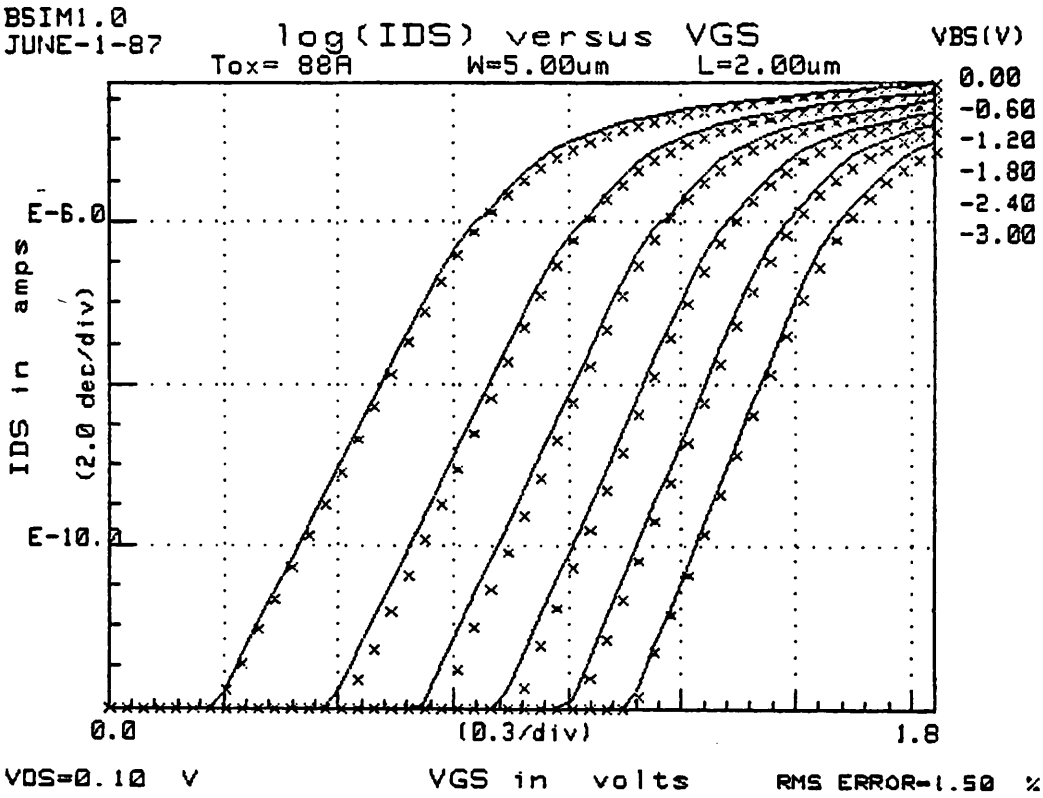


Fig.25c Bell Lab. device. W_{eff} = 100 μm, L_{eff} = 2.2 μm.



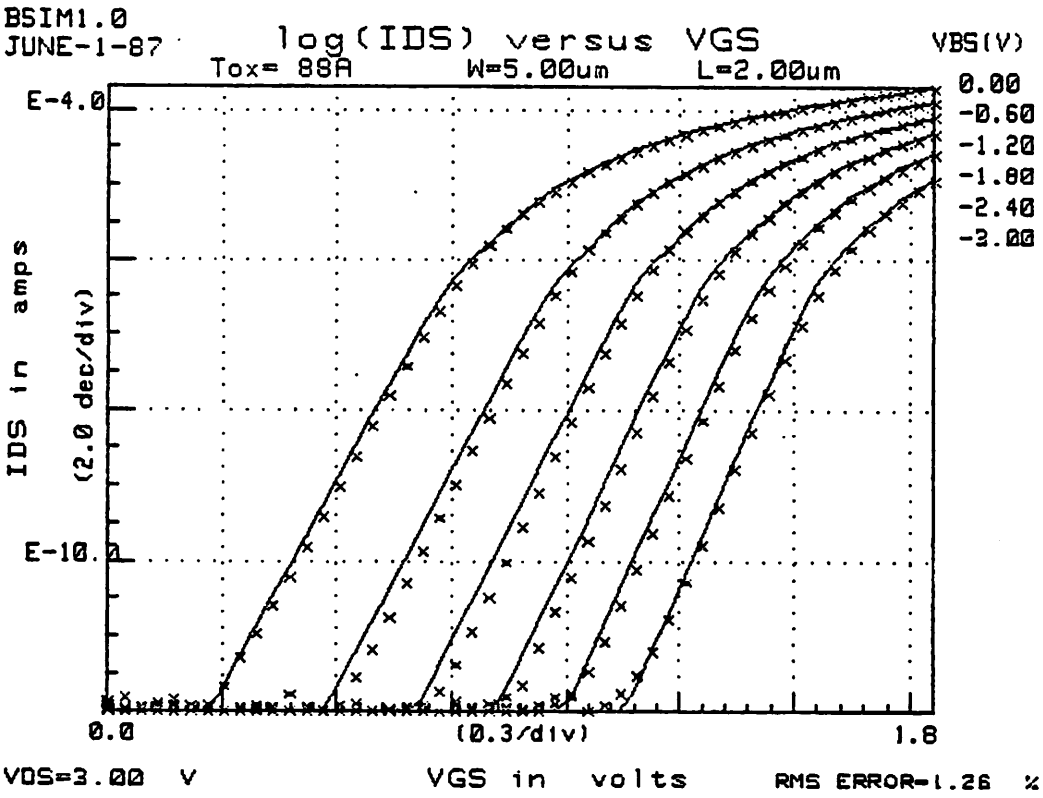
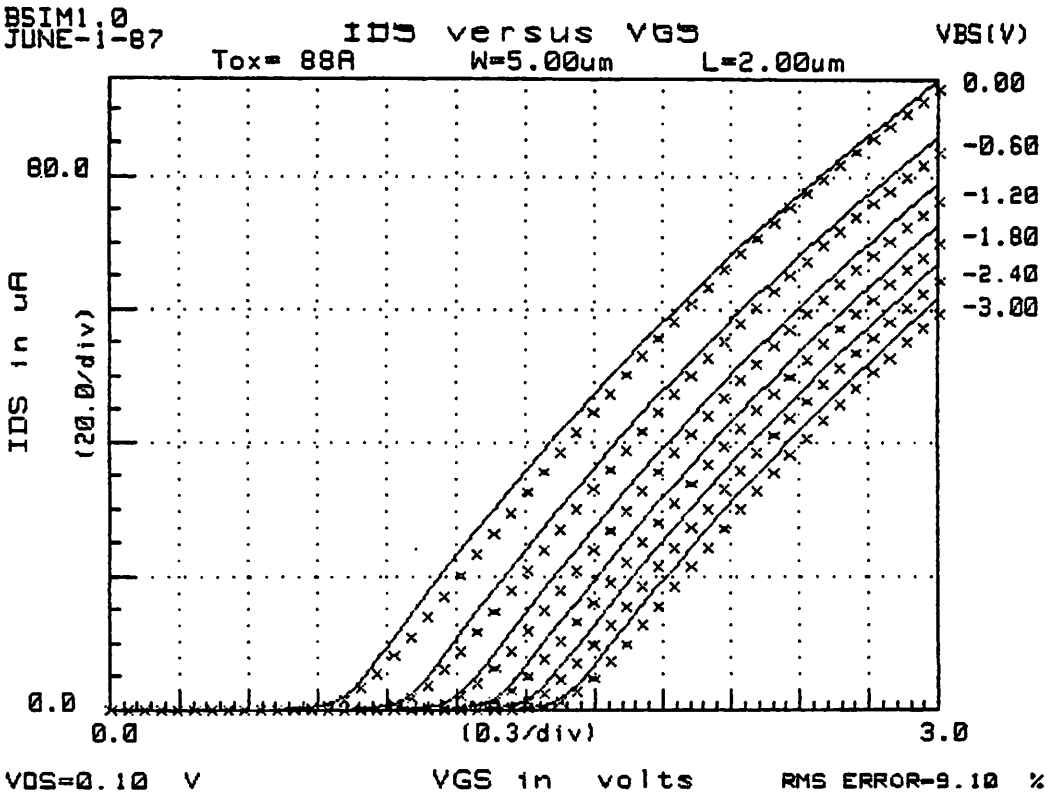


Fig.25d UCB Micro. Lab. device. W_{eff} = 5μm. L_{eff} = 1.5 μm.



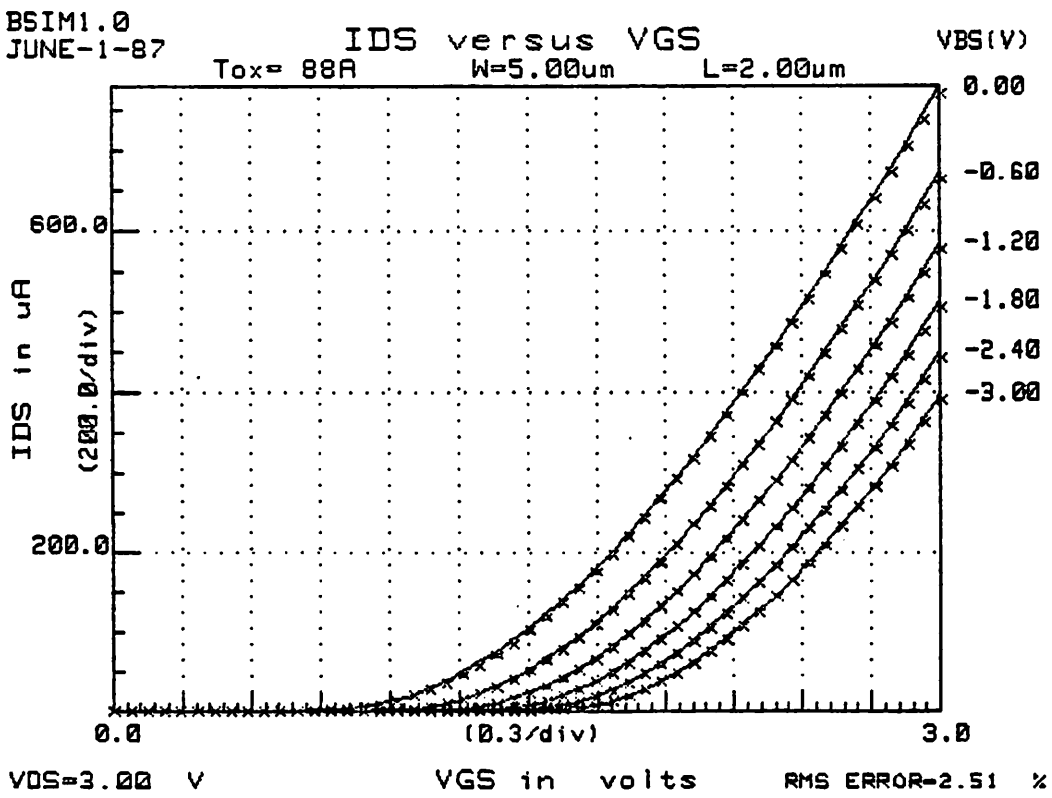
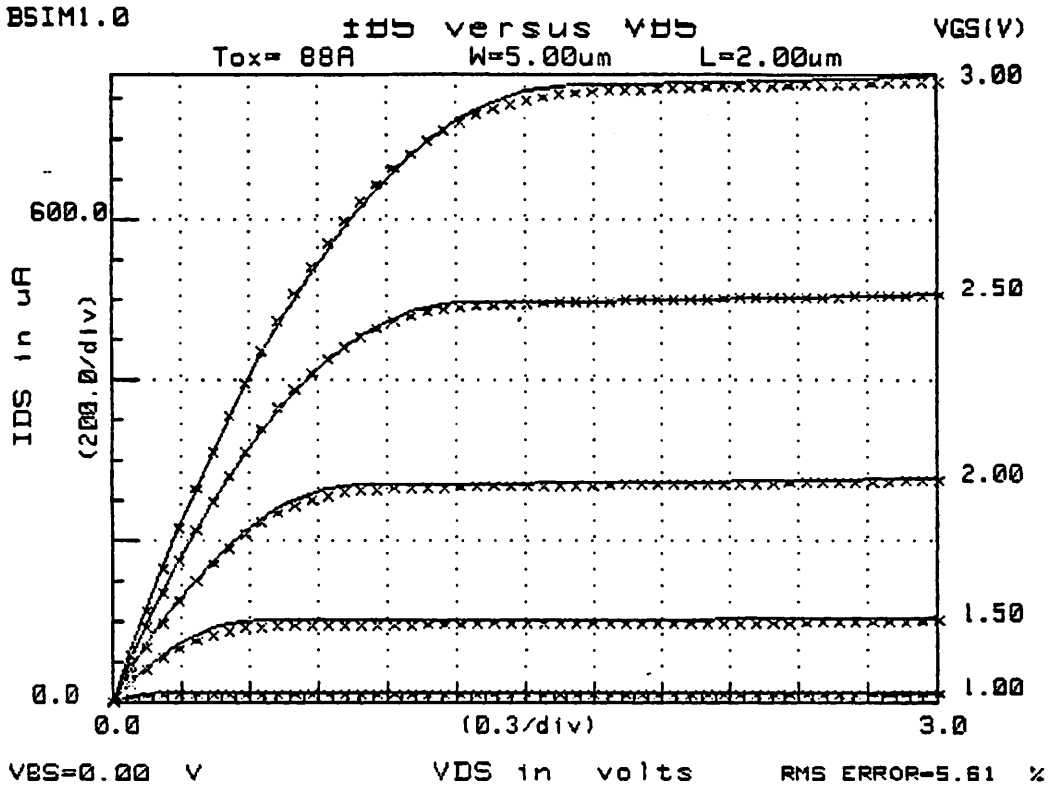


Fig.25e UCB Micro. Lab. device. W_{eff} = 5μm. L_{eff} = 1.5 μm.



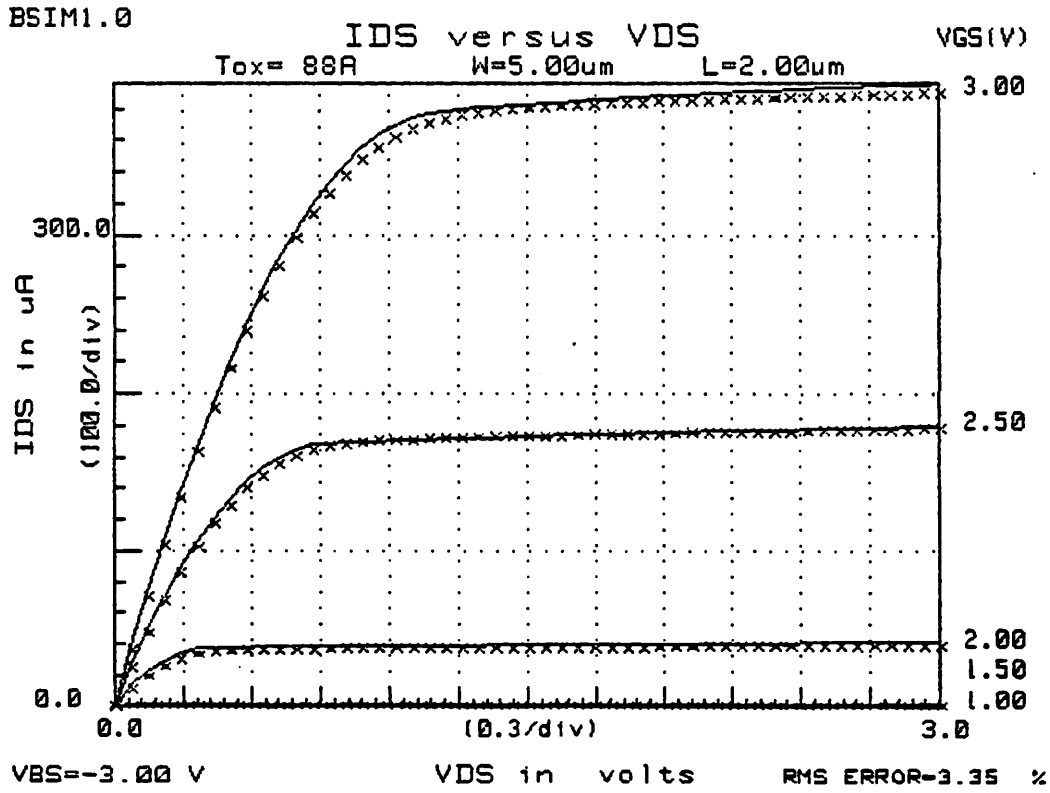
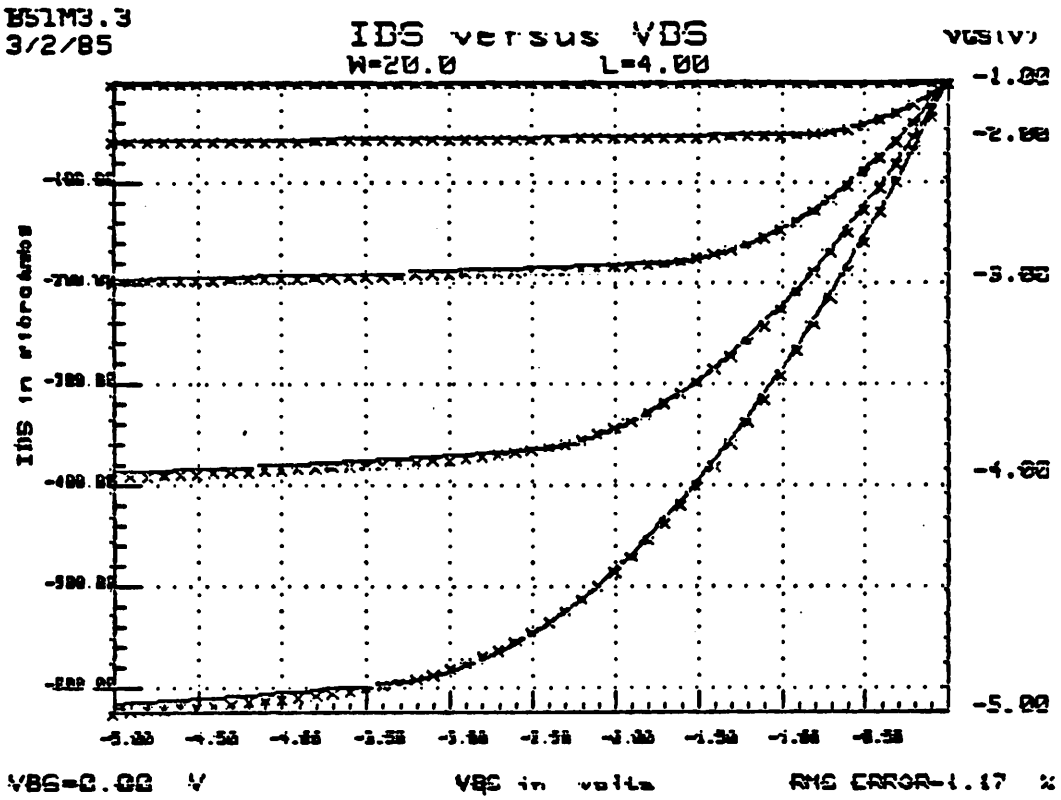


Fig.25f UCB Micro. Lab. device. $W_{eff} = 5\mu m$. $L_{eff} = 1.5\ \mu m$.



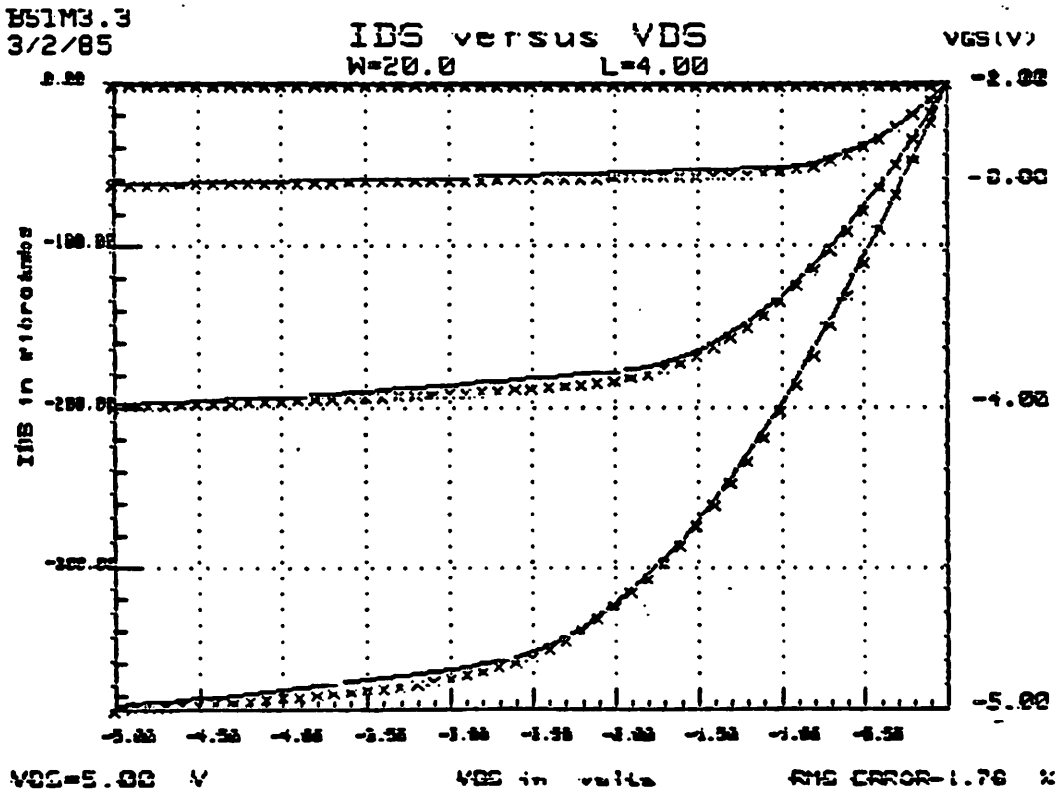
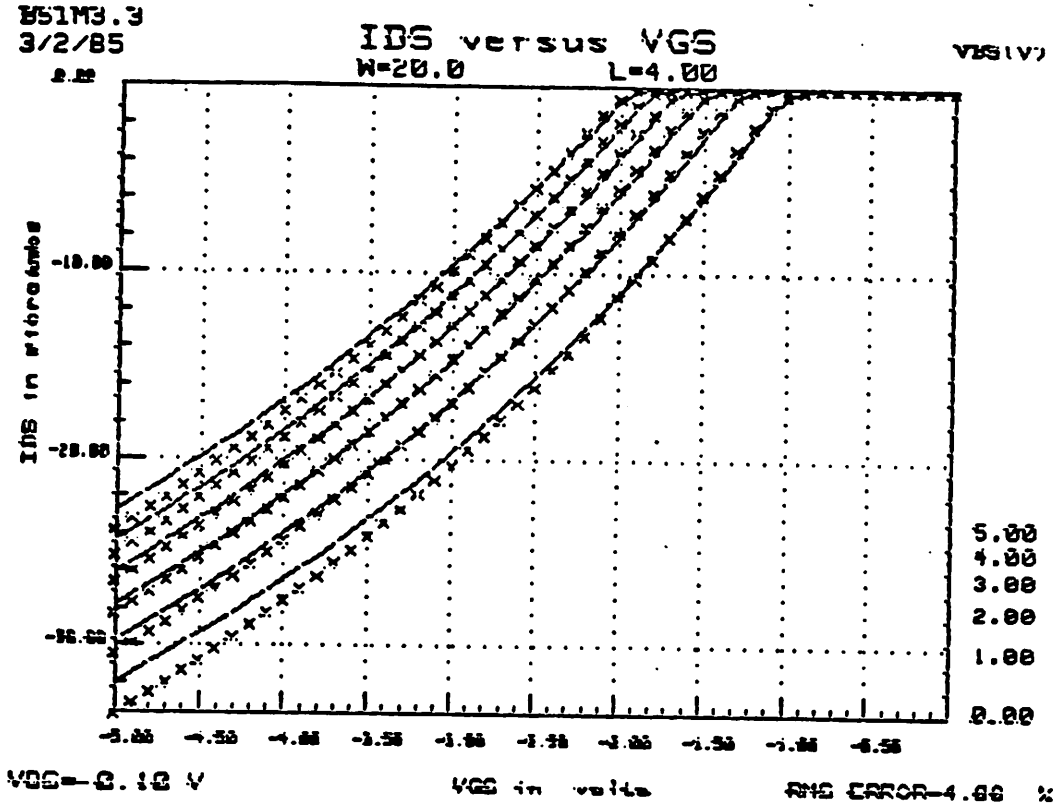


Fig.25g XEROX device. W_{eff} = 20μm, L_{eff} = 3.3 μm.



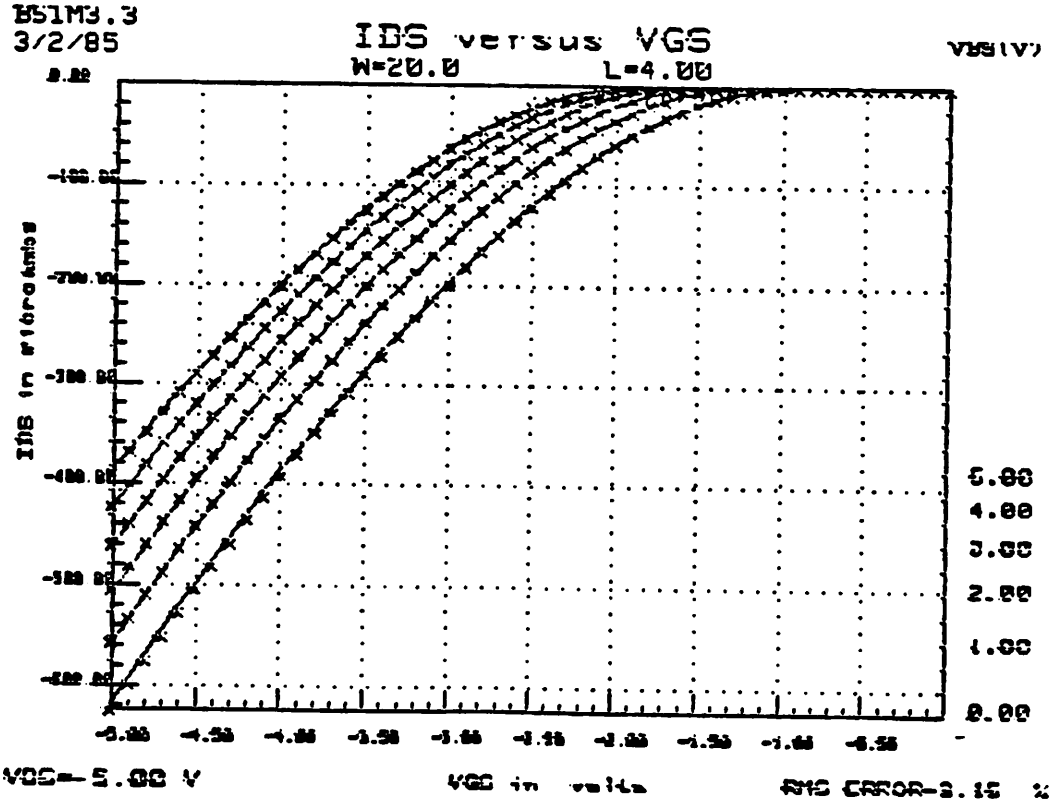
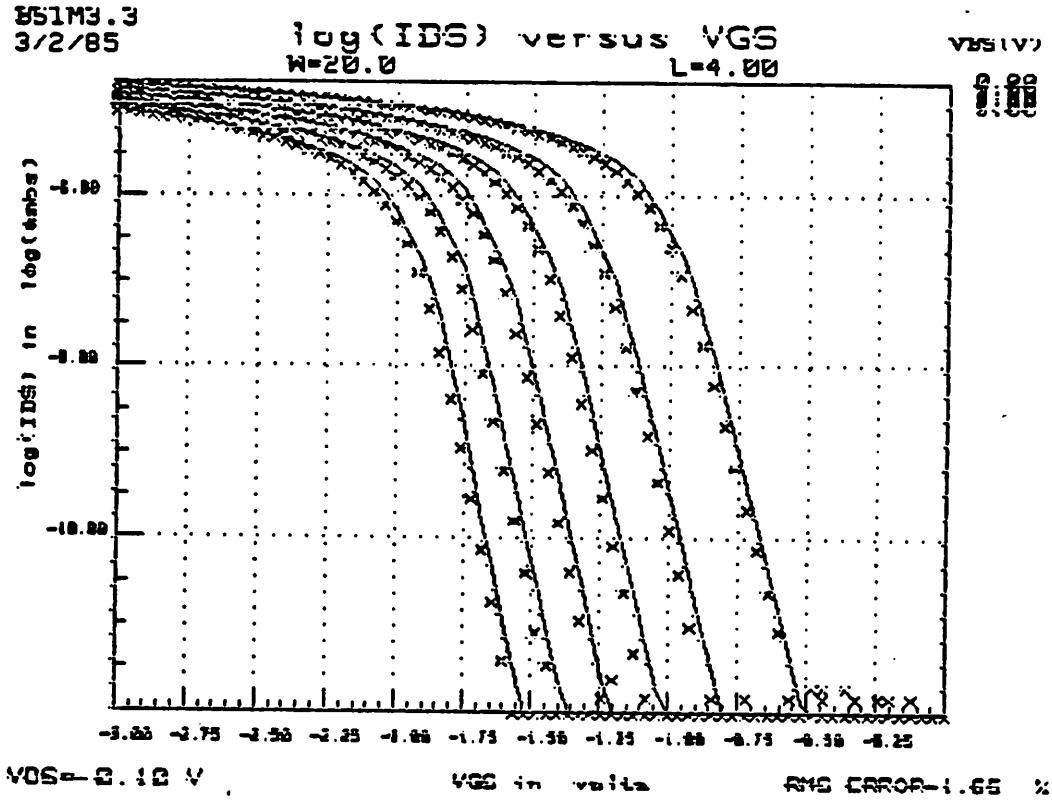


Fig.25h XEROX device. W_{eff} = 20μm. L_{eff} = 3.3 μm.



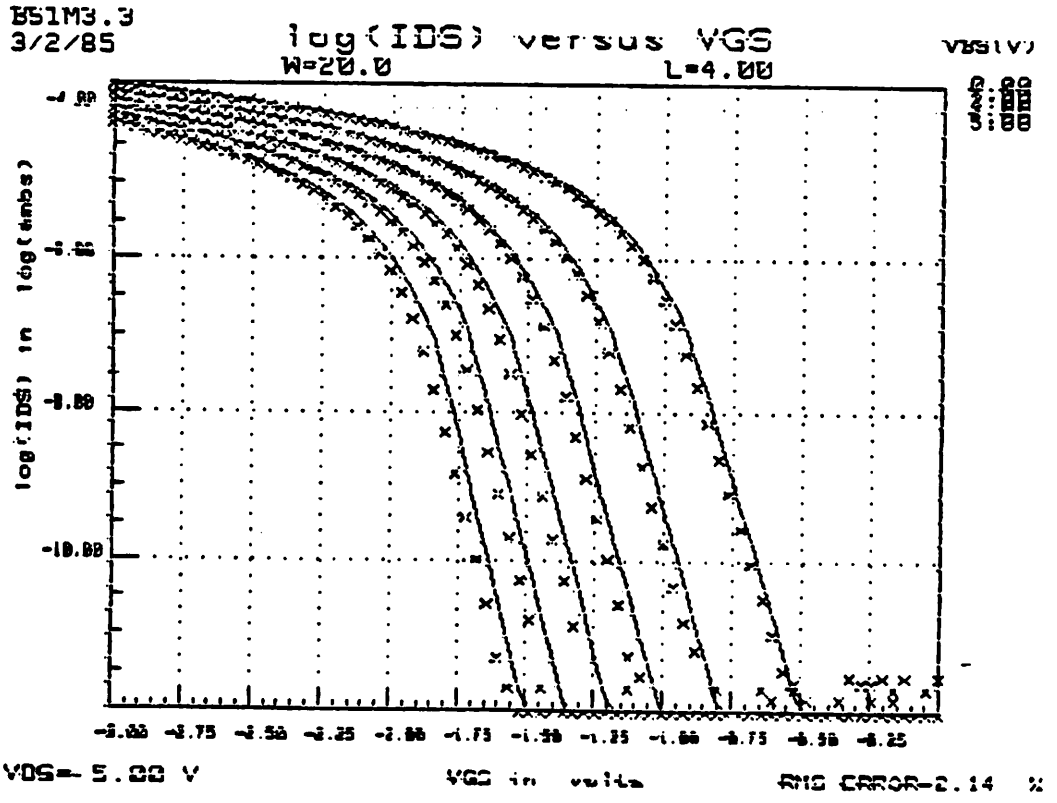
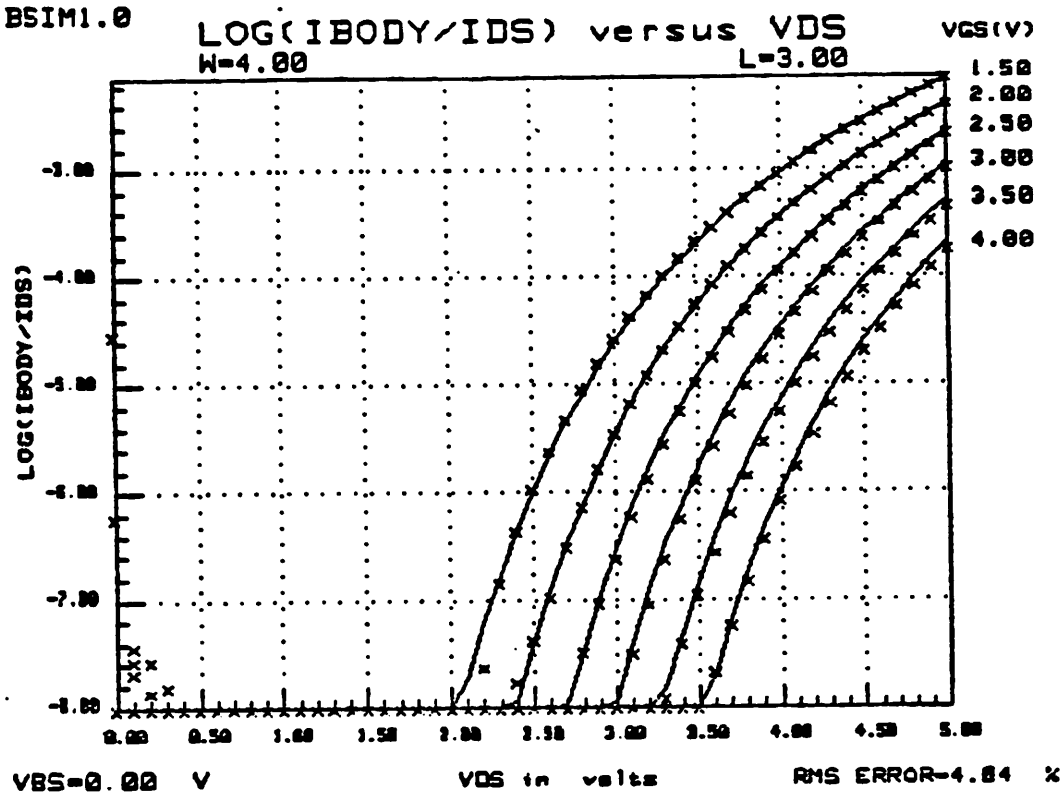


Fig.25i XEROX device. W_{eff} = 20μm, L_{eff} = 3.3 μm.



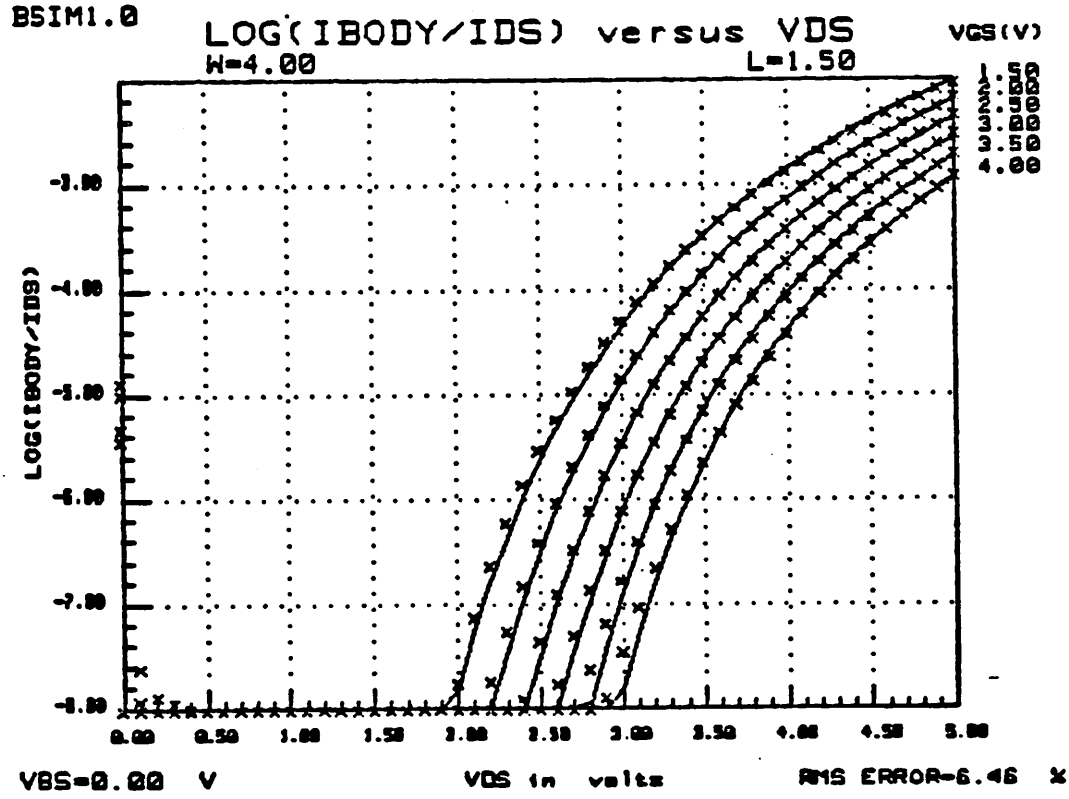
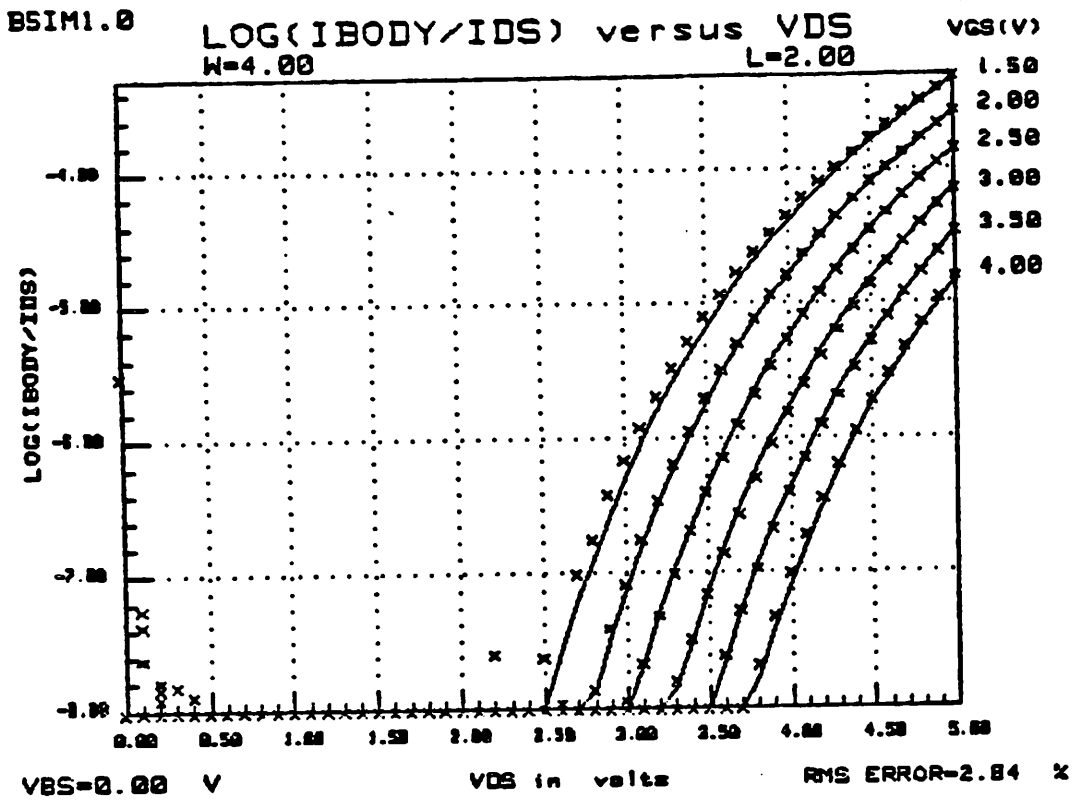


Fig.25j AMD conventional device.



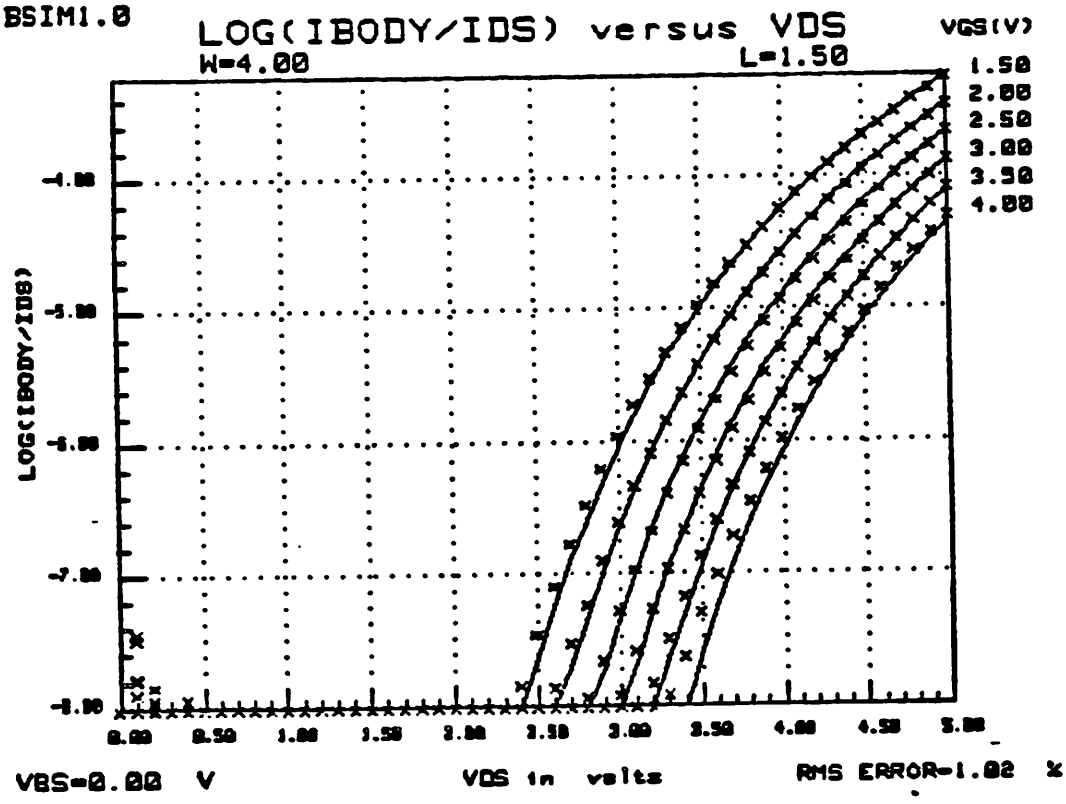
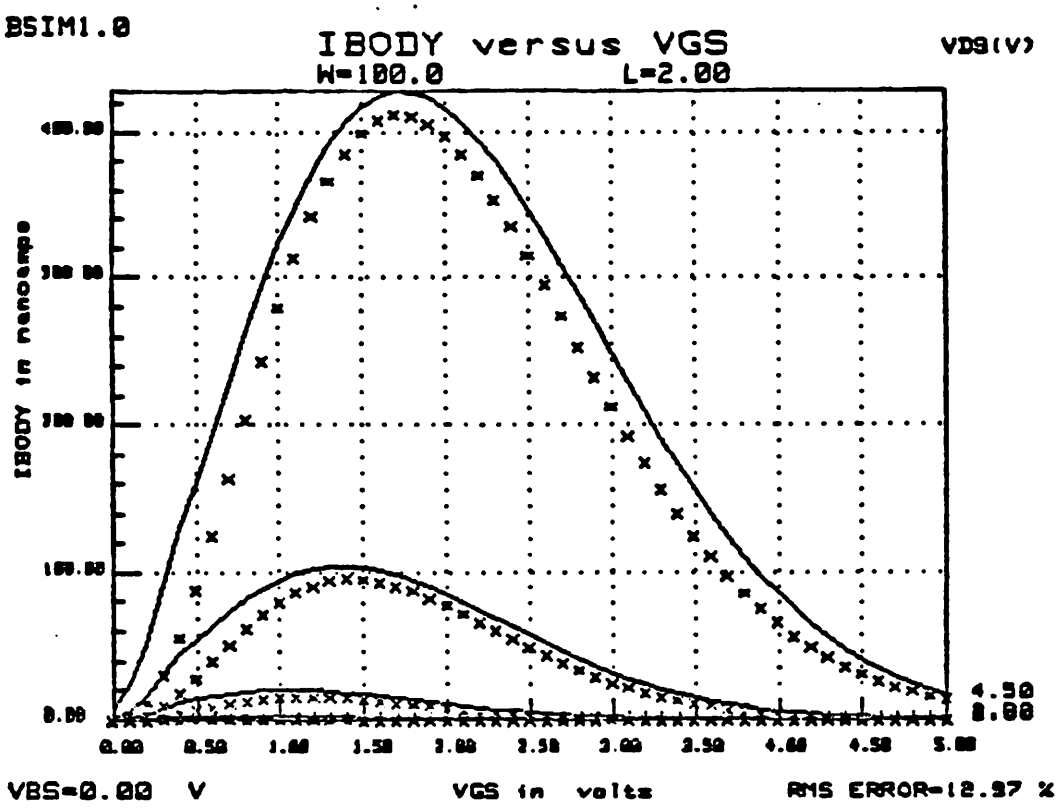


Fig.25k AMD LDD device.



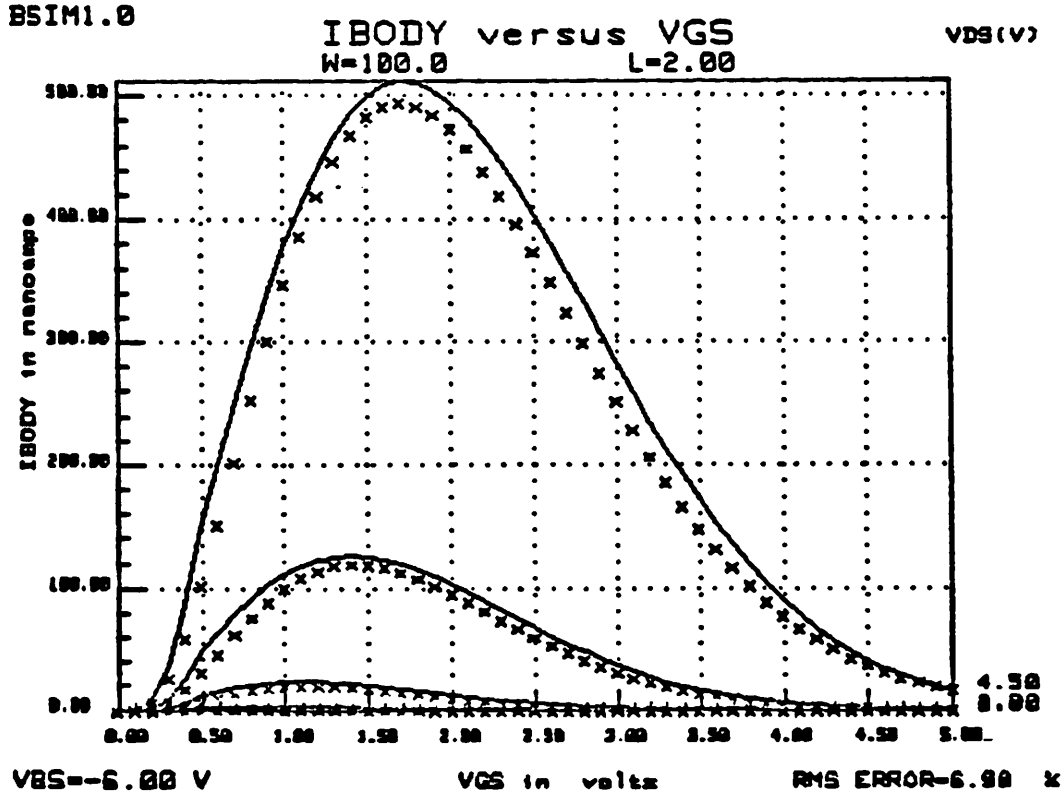
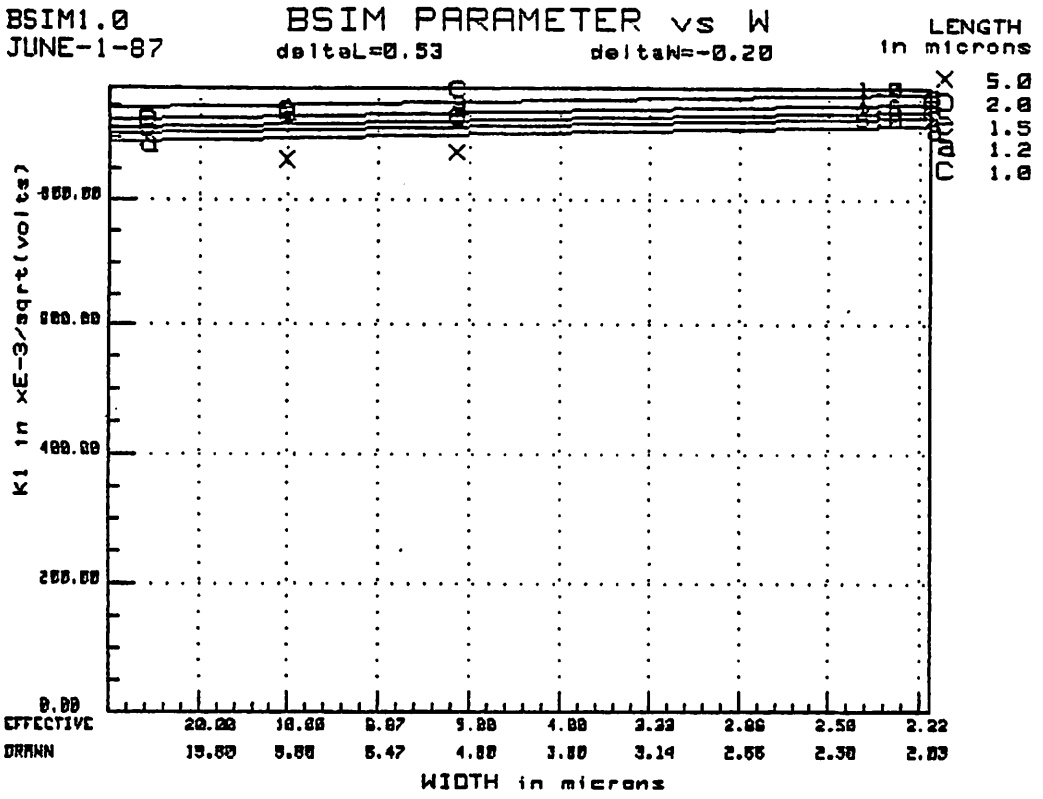


Fig.251 Reticon conventional device.

5.1.4 Parameters versus W and L Plots

After the extraction, all parameters can be plotted as a function of the channel length or channel width. Figure 26 are plots of some of the major parameters to the channel width or channel length. In general, the parameters are sensitive to the channel length and not to the channel width.



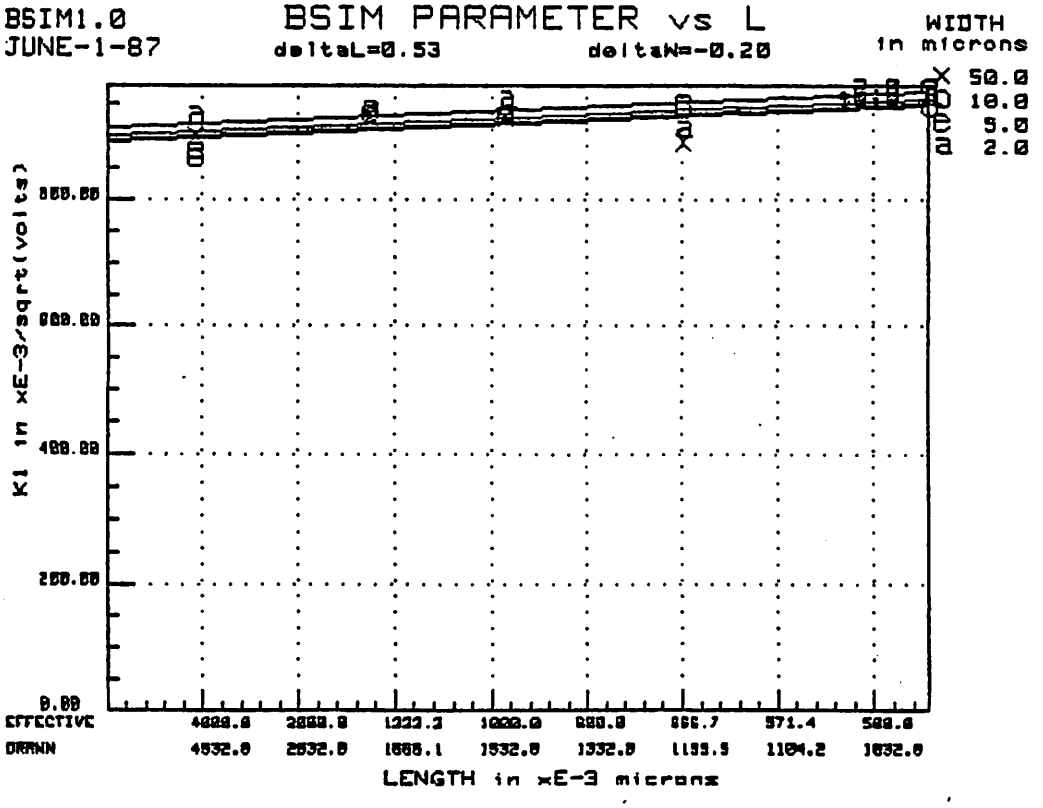
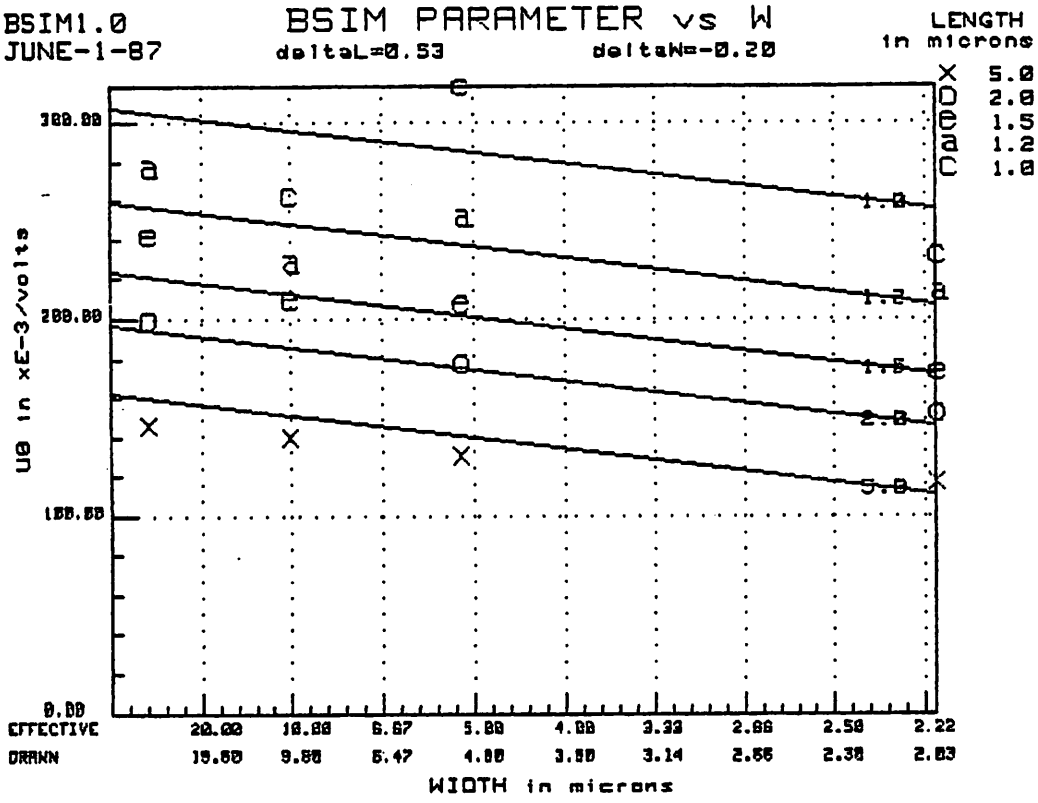


Fig.26a BSIM parameters versus channel length and channel width.



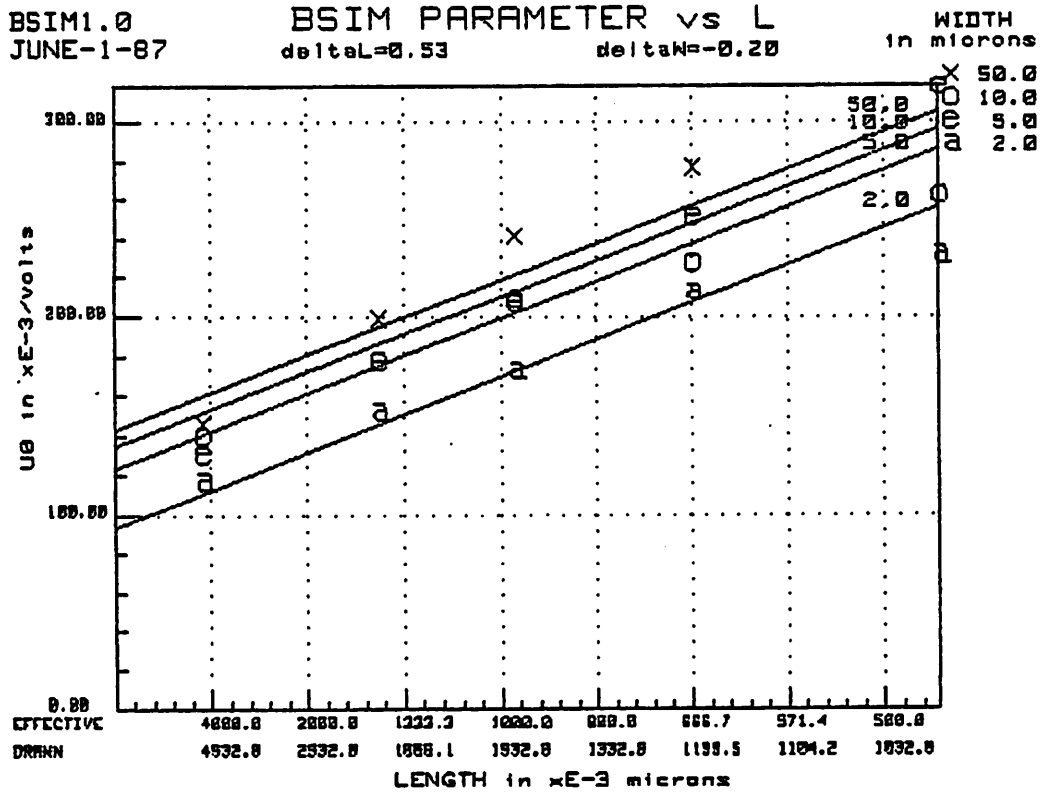
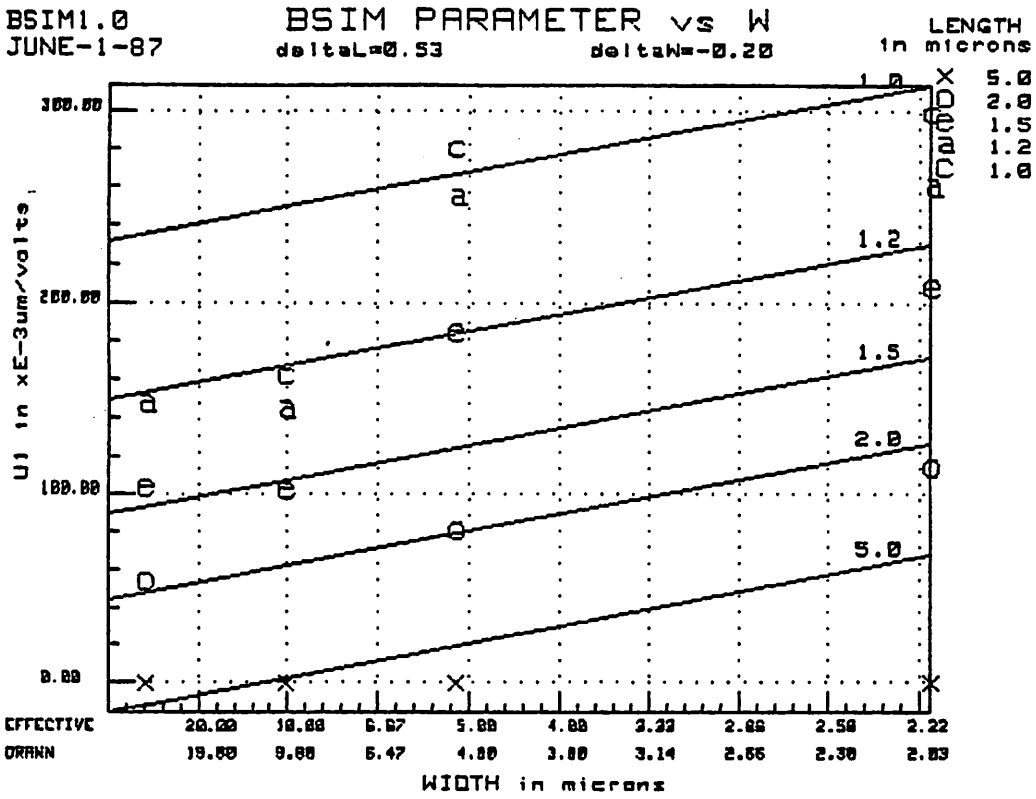


Fig.26b BSIM parameters versus channel length and channel width.



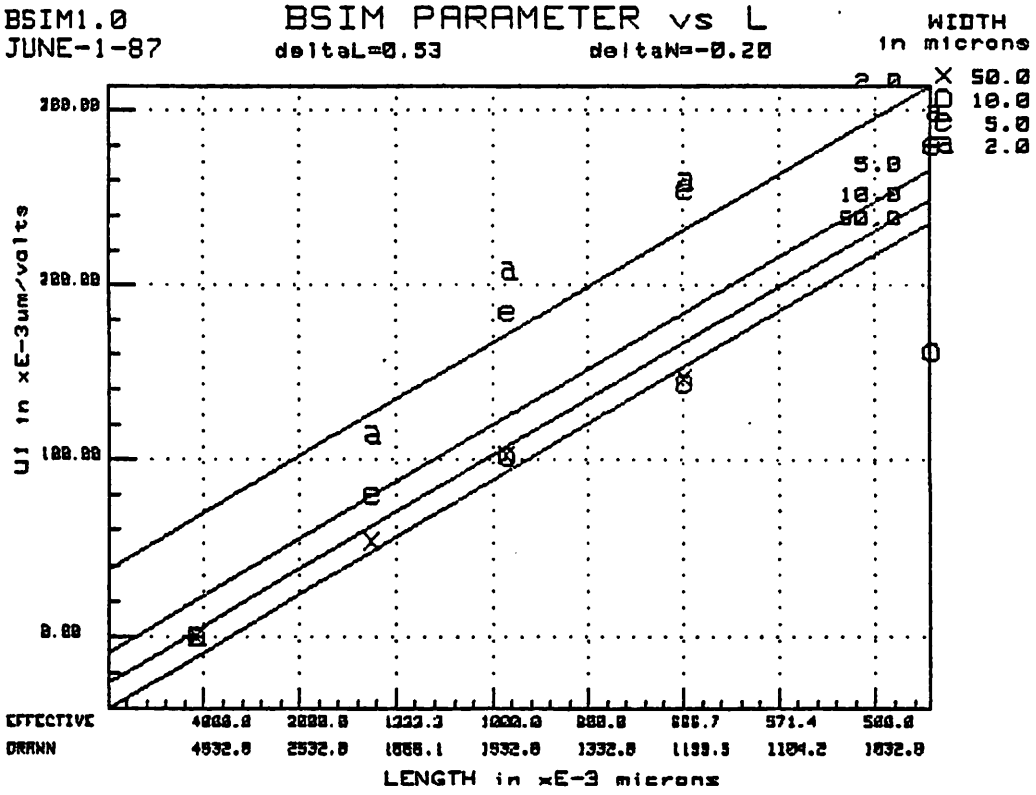
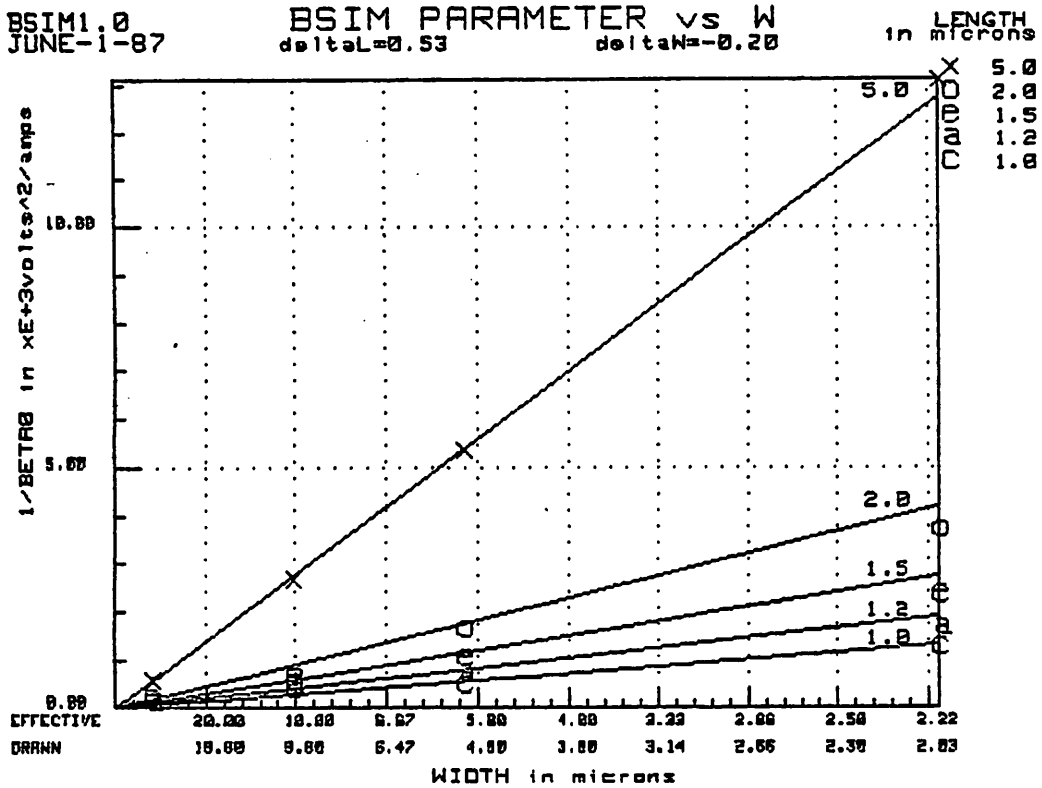


Fig.26c BSIM parameters versus channel length and channel width.



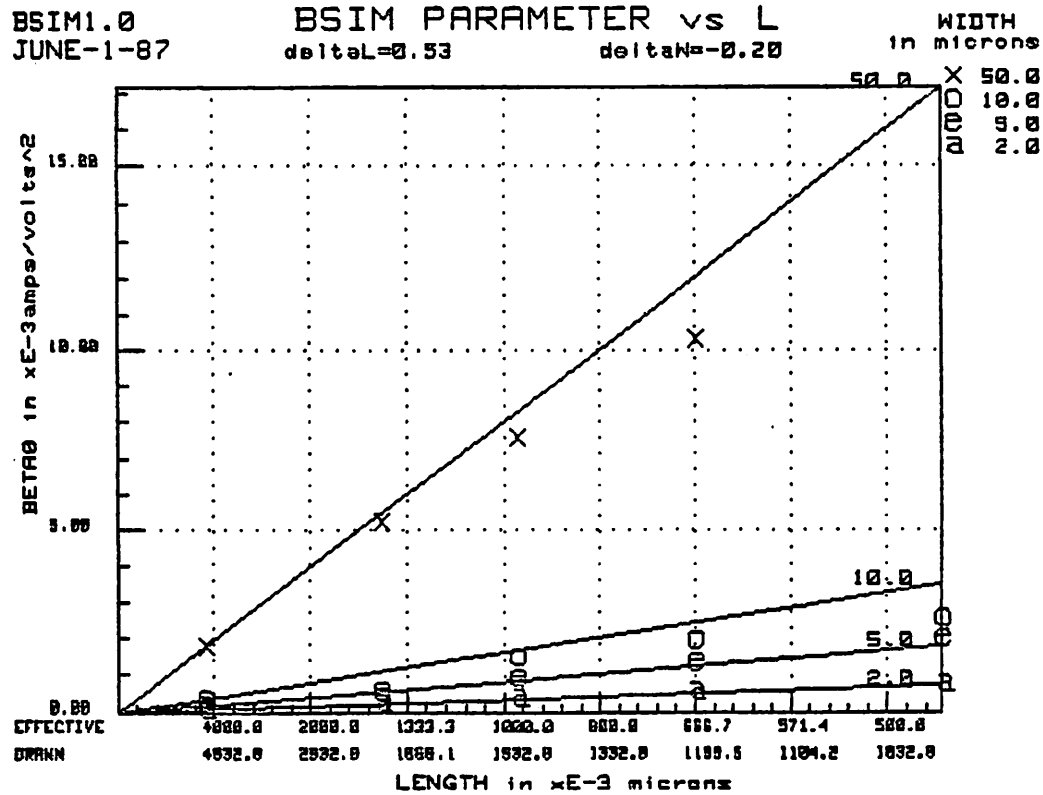
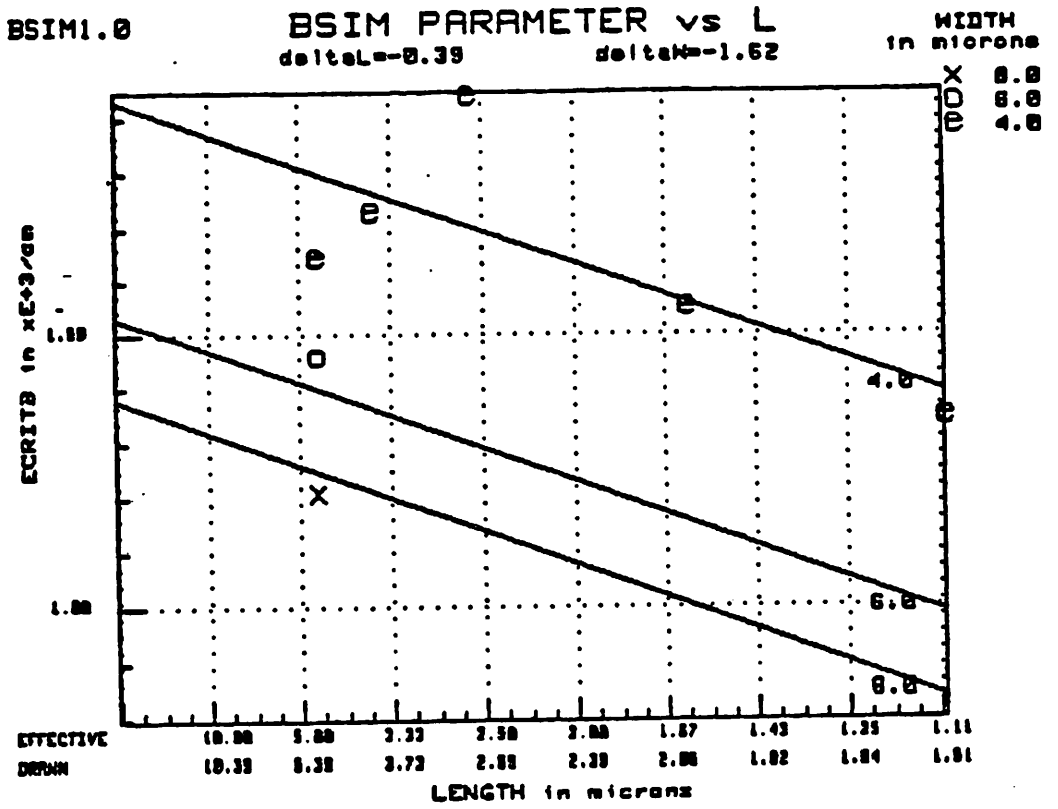


Fig.26d BSIM parameters versus channel length and channel width.



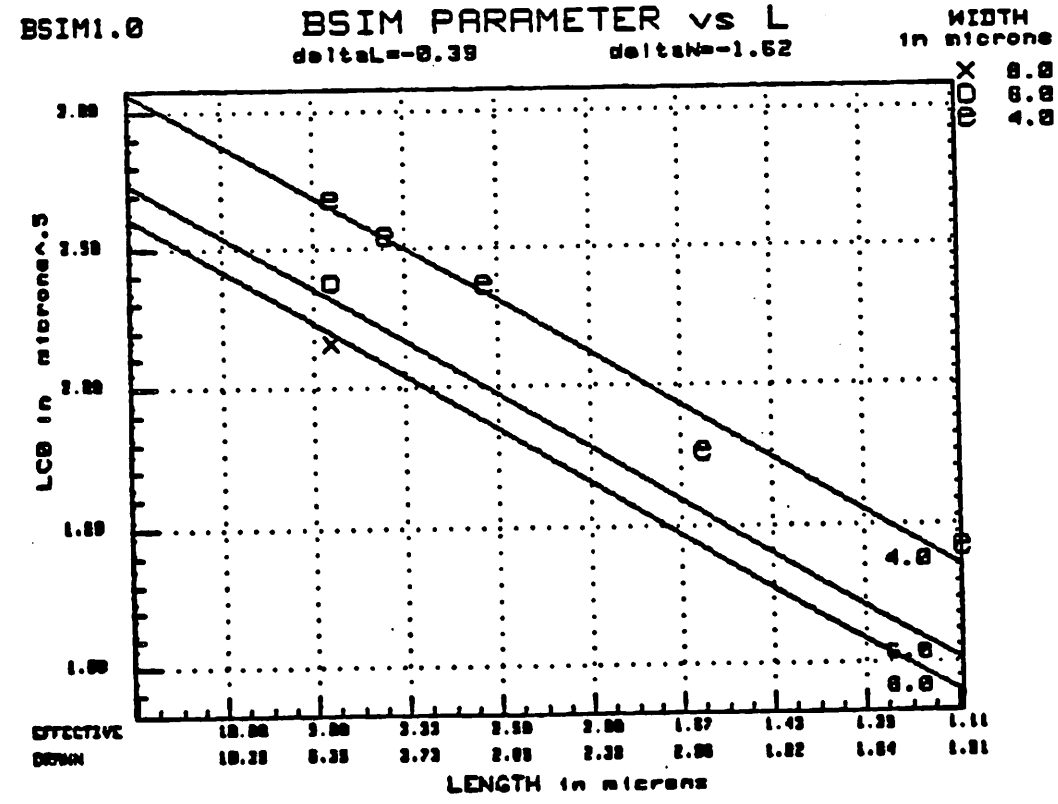
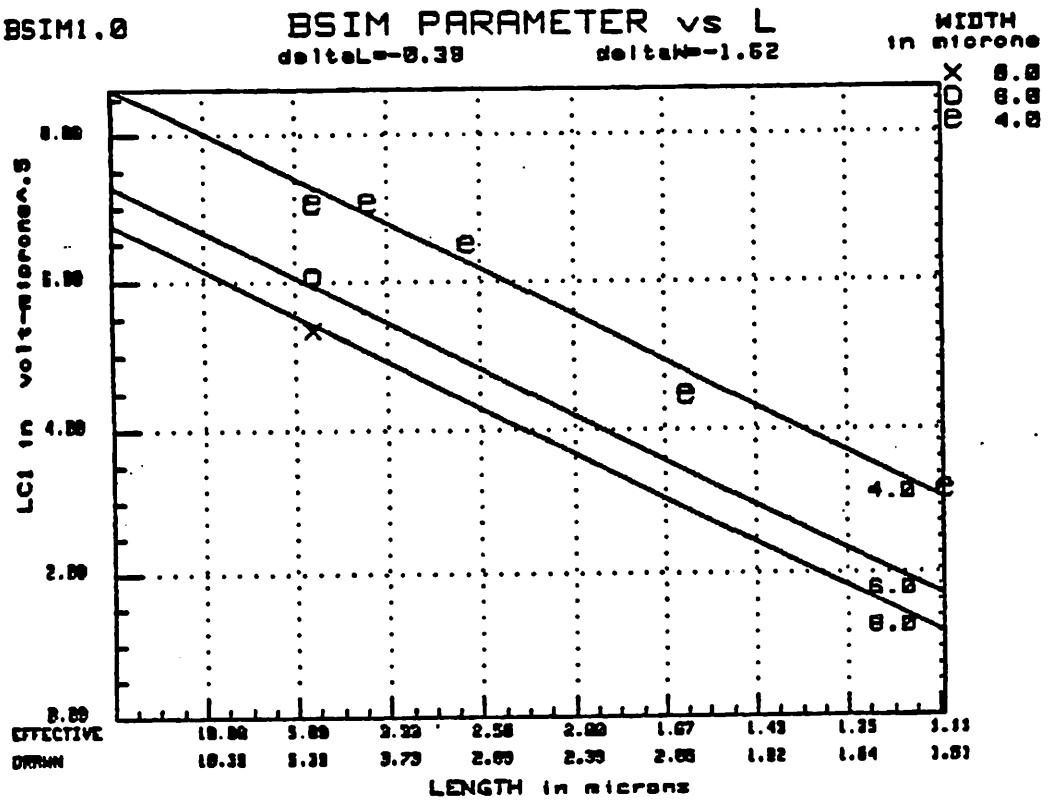


Fig.26e BSIM parameters versus channel length and channel width.



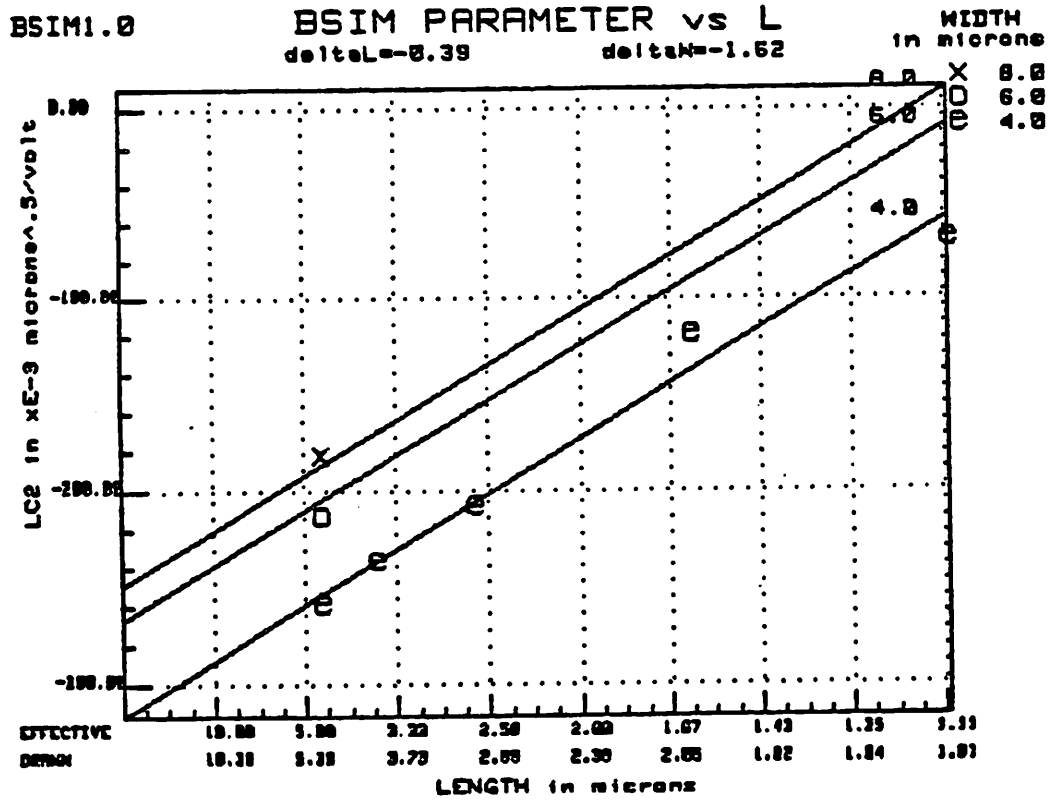


Fig.26f BSIM parameters versus channel length and channel width.

5.2 SPICE Examples

The input source program for SPICE2 with BSIM implemented is similar to those for standard SPICE2. An example is shown in Fig. 27 The process file ("PNMED" in this example) has to be in the same directory as the input source program.

The transient analysis of a simple resistive-load inverter is used as an example for the SPICE output. The schematic diagram of this circuit is shown in Fig. 28. The input and output waveforms are shown in Fig. 29. Two channel-charge partitioning methods (40/60 and 0/100) are used in this example.

```
EXAMPLE1 :          CMOS INVERTER
VCC  1  0  5
SMN1 2  3  0  0  PC1_NM1     L=10U  W=50U
SMP1 2  3  1  1  PC1_PM1_DU2 L=10U  W=50U
VIN  4  0  PWL (0 0  5N 5  10N 5  20N 0)
RIN  4  3       PC1_PY1     L=20U  W=60U
C1   2  0  50F
.OPTIONS NOMOD RELTOL=1E-5 CHGTOL=1E-16
.TRAN 0.5N 20N
.PRINT TRAN  V(3)  V(2)
.PLOT  TRAN  V(3)  V(2)
.PROCESS PC1 FILENAME=PNMED
.WIDTH OUT=80
.END
```

Fig.27An example of the input file for SPICE2.

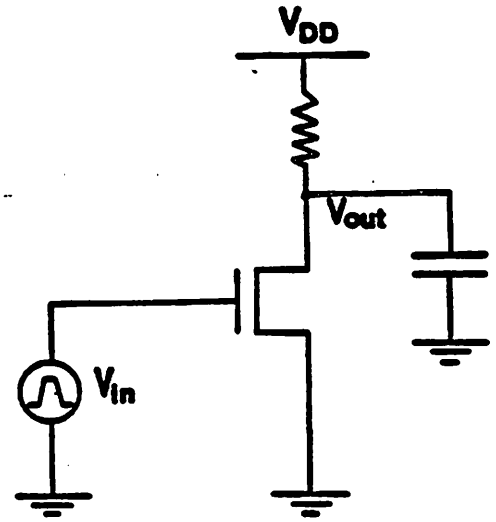


Fig.28Schematic diagram of a simple resistive-load inverter.

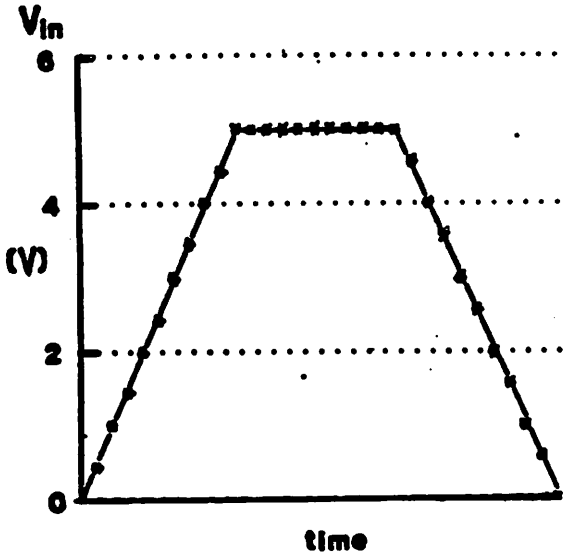


Fig.29a The input waveform

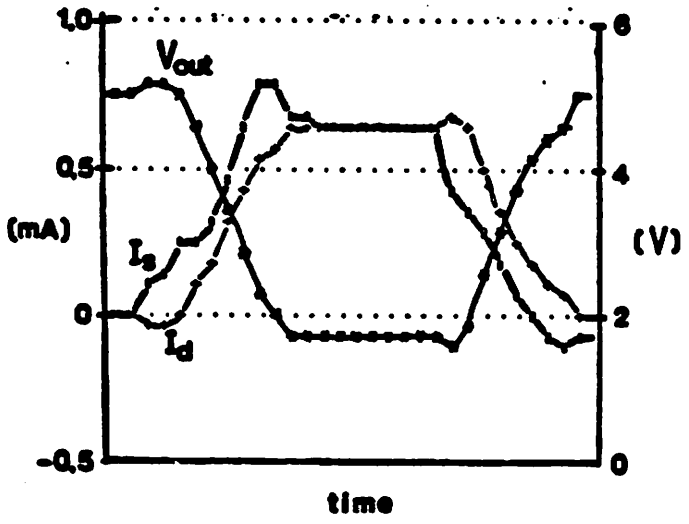


Fig.29b The output voltage and currents for 40/60 channel-charge partitioning method.

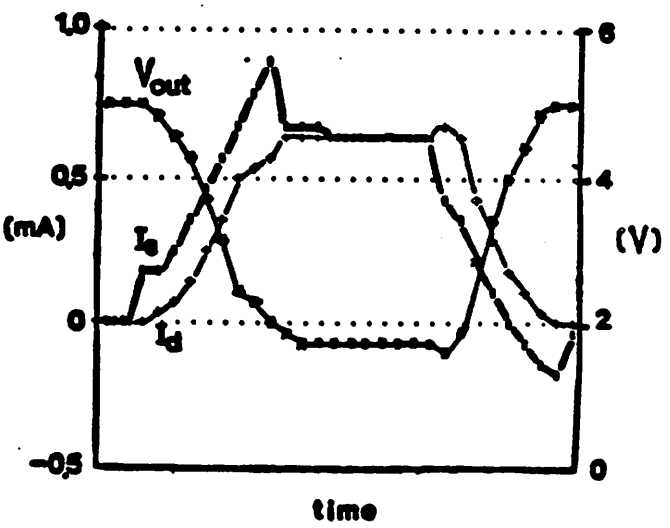


Fig.29c The output voltage and currents for 0/100 channel-charge partitioning method.

5.3 SCALP Examples

Fig. 30 illustrate sample BSIM2 and BSIM3 input decks respectively for a CMOS EEPROM sense amplifier shown in Fig. 31 M1 and M7 are EEPROM reference and memory cells, and M3 and M5 are the sense amplifier's NMOS devices of which substrate current simulations are requested. The essential difference between the two input decks is the BSIM transistor designations (SXXX for BSIM2, MXXX for BSIM3). Fig. 32 shows the resulting output for BSIM2 with a 5 volt input pulse of 4ns rise and fall time, and 25ns pulse width applied at V_{in} = V(6) at t = 0ns. Output for BSIM3 is virtually identical to the one shown here.

```
SIMPLE CMOS SENSE AMPLIFIER
s1 2 5 0 0 PC1_nm1_du1 w=3u l=2u
s2 2 2 5 5 PC1_pm1_du2 w=140u l=2u
s3 3 2 1 0 PC1_nm1_du1 w=20u l=2u
s4 3 2 5 5 PC1_pm1_du2 w=60u l=2u
s5 4 1 0 0 PC1_nm1_du1 w=5u l=2u
s6 4 3 5 5 PC1_pm1_du2 w=10u l=2u
s7 1 6 0 0 PC1_nm1_du1 w=3u l=2u
vdd 5 0 5 dc
vin 6 0 pulse(0.5.0ns.4ns.4ns.25ns.58ns)
.process PC1 filename=PF.SUB
.plot isub(s3) isub(s5)
.deltavt 10mV
c1 1 0 2pf
c2 4 0 0.2pf
.tran 1ns 58ns
.isubwidth=80
.width in=80 out=80
.end
```

Fig.30a Sample BSIM input deck for SPICE2.

```
SIMPLE CMOS SENSE AMPLIFIER
m1 2 5 0 0 PC1_nm1_du1 w=3u l=2u
m2 2 2 5 5 PC1_pm1_du2 w=140u l=2u
m3 3 2 1 0 PC1_nm1_du1 w=20u l=2u
m4 3 2 5 5 PC1_pm1_du2 w=60u l=2u
m5 4 1 0 0 PC1_nm1_du1 w=5u l=2u
m6 4 3 5 5 PC1_pm1_du2 w=10u l=2u
m7 1 6 0 0 PC1_nm1_du1 w=3u l=2u
vdd 5 0 5 dc
vin 6 0 pulse(0.5.0ns.4ns.4ns.25ns.58ns)
.process PC1 filename=PF.SUB
.plot isub(m3) isub(m5)
.deltavt 10mV
c1 1 0 2pf
c2 4 0 0.2pf
.tran 1ns 58ns
.isubwidth=80
.end
```

Fig.30b Sample BSIM input deck for SPICE3.

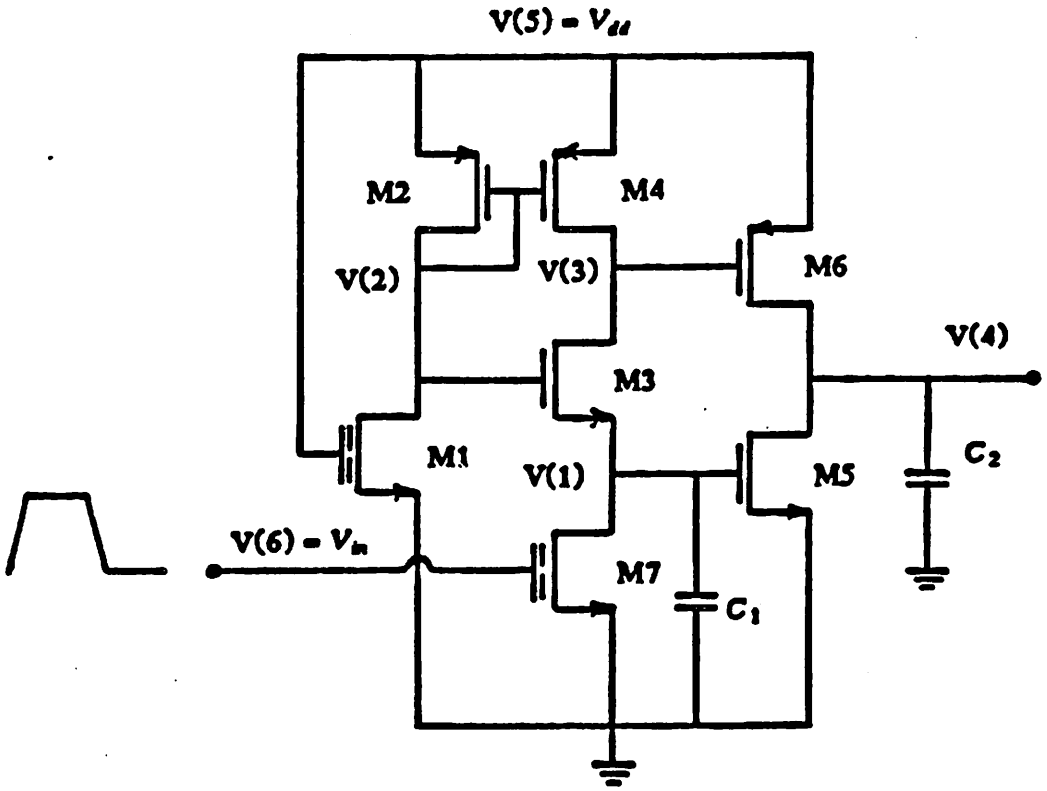
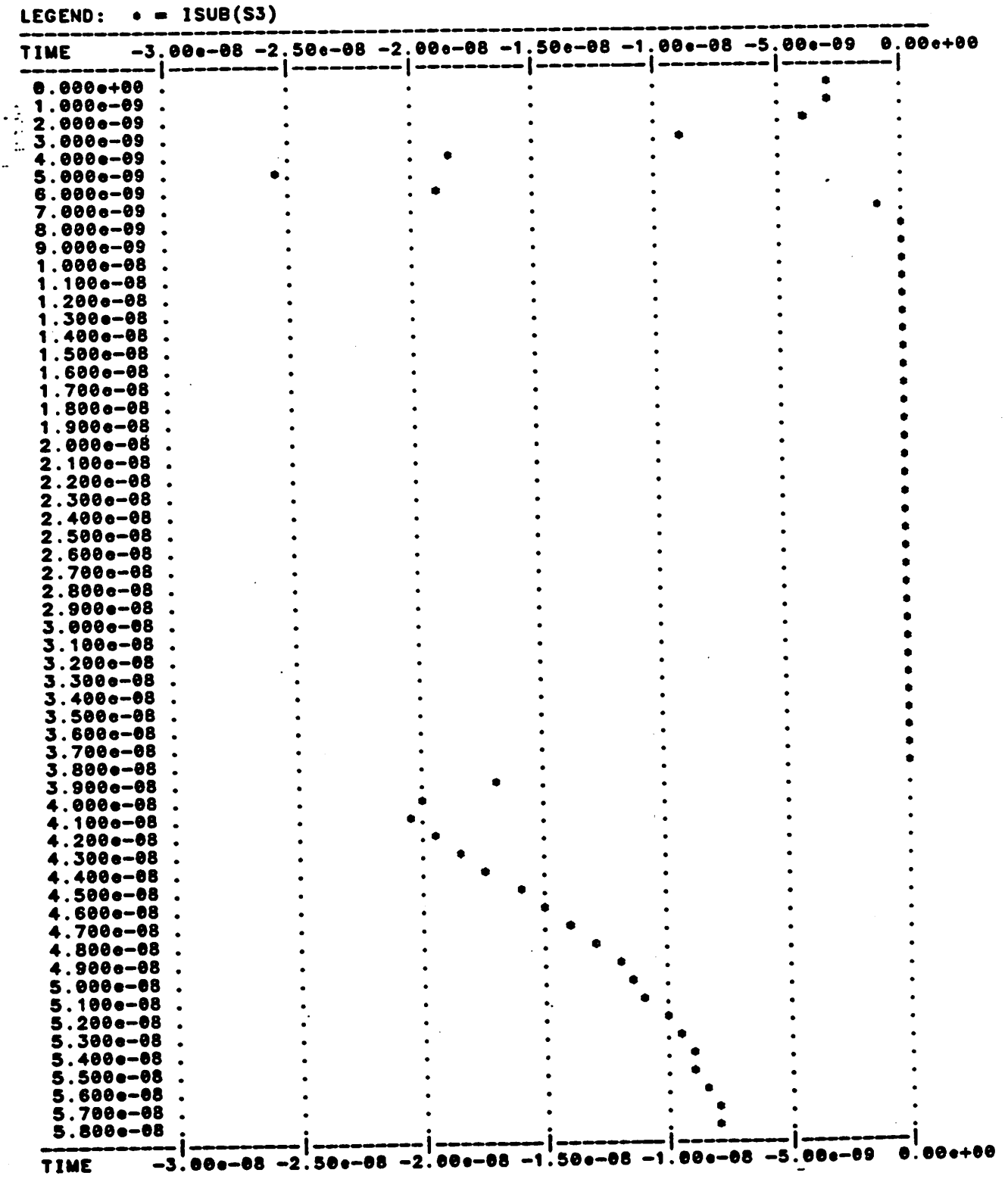


Fig.31 CMOS sense amplifier circuit used for SPICE simulation.

SUBSTRATE CURRENT TRANSIENT ANALYSIS
TRANSISTOR S3

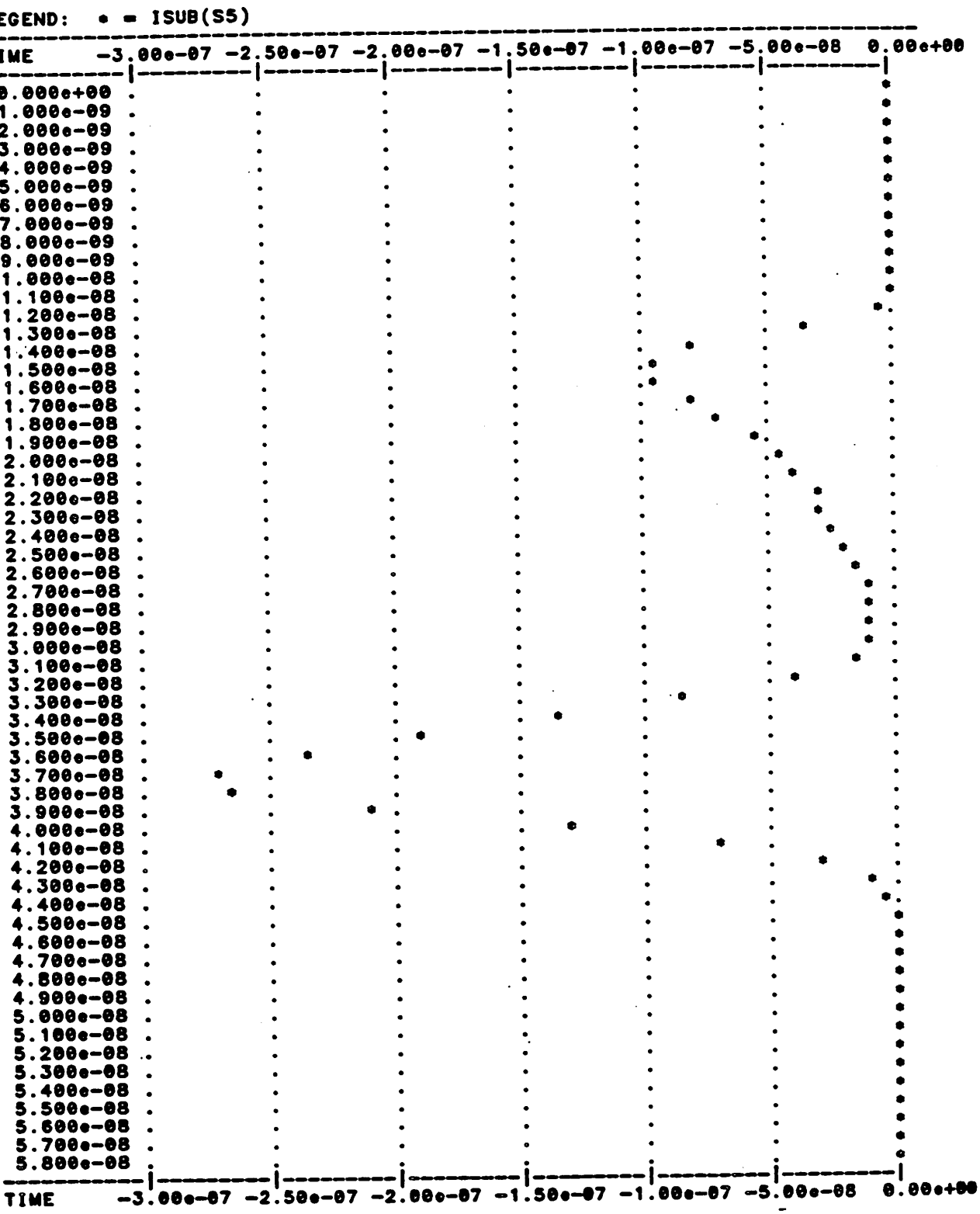


DEVICE LIFETIME AT DELTA VTH = 0.01 VOLTS

>>>>>> TAU(S3) = 3.42072e+10 HRS. (1.23146e+14 SEC.) <<<<<<

Fig.32SPICE2 output.

SUBSTRATE CURRENT TRANSIENT ANALYSIS
TRANSISTOR S5

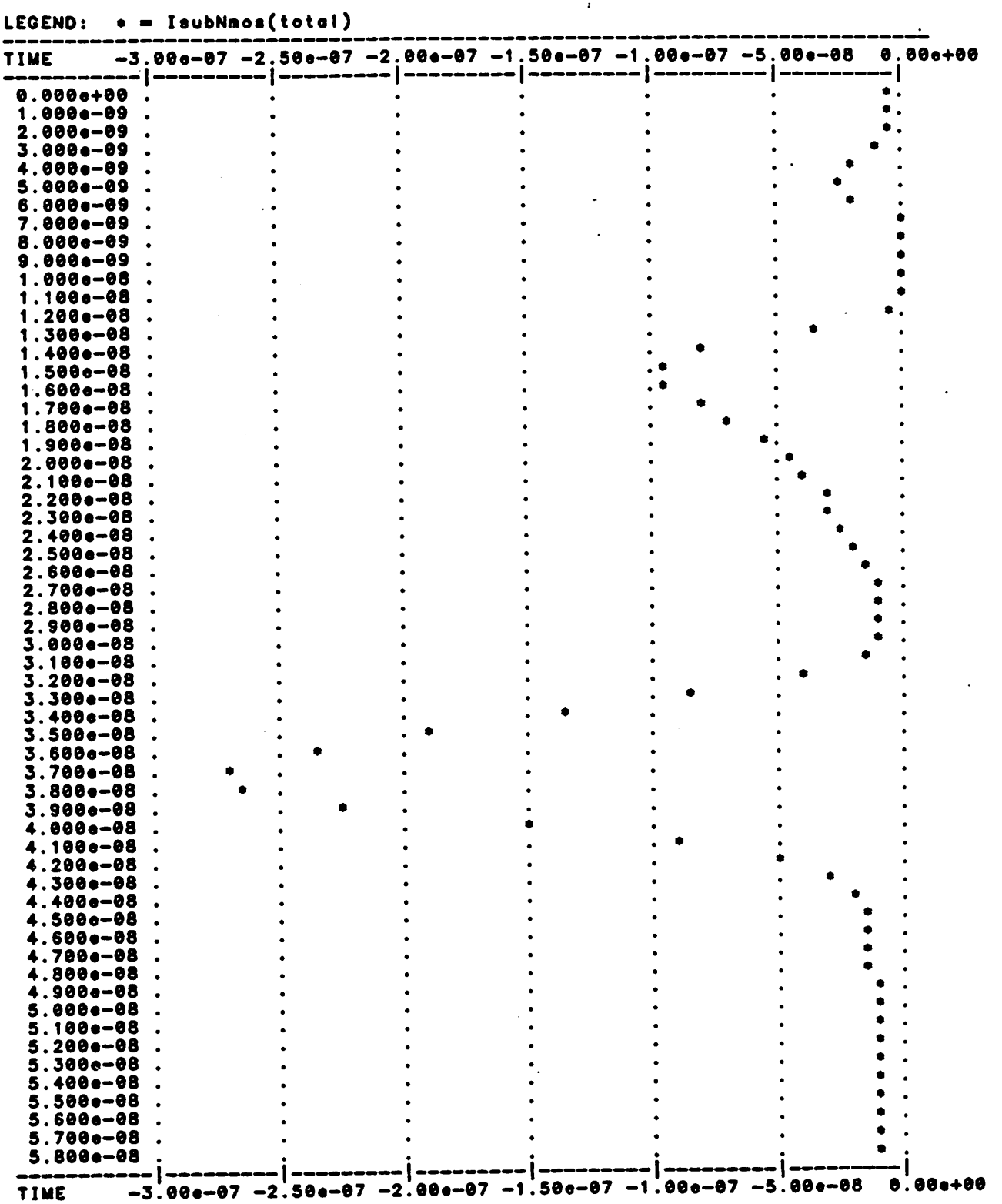


DEVICE LIFETIME AT DELTA VTH = 0.01 VOLTS

>>>>>> TAU(S5) = 84523 HRS. 56 MIN. 14 SEC. (3.04286e+08 SEC.) <<<<<<

Fig.32SPICE2 output (continued).

SUBSTRATE CURRENT TRANSIENT ANALYSIS



SPICE2 output (continued).

[1] B.J. Sheu, D.L Scharfetter, and H.C. Poon "Compact Short-Channel IGFET Model", Univ. of California, Berkeley, ERL memo. UCB/ERL M84/20.

[2] A. H.-C. Fung, "A Subthreshold Conduction Model for BSIM", Univ. of California, Berkeley, ERL memo. UCB/ERL M85/22.

[3] C. Hu, S. Ram, F.-C. Hsu, P.K. Ko, T.Y. Chan, K.W. Terill, "Hot-Electron_induced MOSFET Degradation - Model, Monitor, Improvement," IEEE Trans. Electron Devices, Vol. ED-32, pp. 375-385, February 1985.

[4] Y.A. El-Mansy and D.M. Caughey, "Modeling Weak Avalanche Multiplication Current in IGFET's and SOS Transistors for CAD," IEDM, pp.31-34, 1975.

[5] Y.A. El-Mansy and A.R. Boothroyed, "A Simple Two-Dimensional Model for IGFET Operation in the Saturation Region," IEEE Trans. Electron Devices, Vol. ED-24, pp.254-262, March 1977.

[6] P.K. Ko, "Hot-Electron Effects in MOSFET's," Univ. of California, Berkeley, Ph.D. Thesis, 1981.

[7] C. Sodini and P.K. Ko, "The Effect of High-Fields on MOS Device and Circuit Performance," IEEE Trans. Electron Devices, Vol. ED-31, pp.1386-1393, Oct. 1984.

[8] P. Yang, B.D. Epler, and P.K. Chatterjee, "An Investigation of the Charge Conservation Problem for MOSFET Circuit Simulation," IEEE Jour. of Solid-State Circuit, Vol. SC-15, no. 4, pp.636-643, Aug. 1980.

[9] S.-Y. Oh, D.E. Ward, and R.W. Dutton, "Transient Analysis of MOS Transistors," IEEE Jour. of Soild-State Circuit, Vol. SC-15, pp.636-643, Aug. 1980.

[10] D.E Ward, "Charge-Based Modeling of Capacitance in MOS Transistor," Ph.D. Dissertation, Tech. Rep G 201-11, Integrated Circuit Lab., Stanford University, June 1981.

[11] Y. Tsividis, "problems with Precision Modeling of Analog MOS LSI," IEEE IEDM, pp.274-277, 1982.

[12] N. Yuen, "A Fully Automated BSIM Parameter Extraction System Using the HP 4062 Test System," Univ. of California, Berkeley, ERL memo. UCB/ERL M86/41.

[13] T.Y. Chan, P.K. Ko, and C. Hu, "A simple Method to Characterize Substrate Current in MOSFETs," IEEE Trans. Electron Device Letters, Vol. EDL-5, p505, Dec. 1984.