Copyright © 1984, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

A FULLY AUTOMATED MOS DEVICE CHARACTERIZATION SYSTEM FOR PROCESS-ORIENTED INTEGRATED CIRCUIT DESIGN

by

Brian Scott Messenger

Memorandum No. UCB/ERL M84/18

30 January 1984

A FULLY AUTOMATED MOS DEVICE CHARACTERIZATION SYSTEM FOR PROCESS-ORIENTED INTEGRATED CIRCUIT DESIGN

by

Brian Scott Messenger

Memorandum No. UCB/ERL M84/18

30 January 1984

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

BRIAN SCOTT MESSENGER
Author

A FULLY AUTOMATED MOS DEVICE
CHARACTERIZATION SYSTEM FOR PROCESS-
ORIENTED INTEGRATED CIRCUIT DESIGN
Title

RESEARCH PROJECT

Submitted to the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, in partial satisfaction of the requirements for the degree of Master of Science, Plan II.

Approval for the Report and Comprehensive Examination:

Committee: David A. Hodges, Research Advisor

Jan 30, 1984 Date

D J Scharfetter

Jan 30, 1984 Date

A FULLY AUTOMATED MOS DEVICE CHARACTERIZATION SYSTEM FOR PROCESS-ORIENTED INTEGRATED CIRCUIT DESIGN

Brian Scott Messenger

Department of Electrical Engineering
and Computer Sciences
University of California, Berkeley

ABSTRACT

A fully automatic MOS device parameter extraction system has been developed to provide a systematic, reproducible and accurate means of determining device parameters for a new compact short channel MOSFET model. The system also provides the capability of evaluating parameter variations with device geometry for the purpose of generating process files which can model spectrums of device sizes. The parameter extraction software has been merged with device measurement, interactive graphics, and automatic prober capabilities to provide a powerful tool for device research and for statistical analysis of integrated circuit processes. Use of the model and the extraction system to analyze devices from various processes, has resulted in excellent simulations for devices with gate oxides ranging from 200 Angstroms to 700 Angstroms, and for channel lengths extending down to one-micron drawn dimensions. The parameter extraction system can be used for research, or in a production environment by a technician with no prior device physics background. This fully automatic MOS device characterization capability is expected to result in significant increases in integrated circuit design productivity.

January 31, 1984

ACKNOWLEDGEMENTS

I wish to thank my thesis advisors, Professor David A. Hodges and Dr. Don L. Scharfetter, for their encouragement, support and technical advice throughout the entire course of my research. Dr. Scharfetter provided invaluable technical background for the development of the Automatic Characterization System, and was responsible for the organization and management of the entire CSIM model team at Berkeley. I would also like to thank Bing Sheu, Ralph Liu and Jeff Burns who provided valuable technical discussions in the process of interfacing our collective efforts.

I am grateful to Dr. Lynn Roylance and Dr. Fred Perner of Hewlett-Packard Laboratories for the work opportunities, exposure, and insight to device modeling and characterization that they provided. I would like to thank Xerox Palo Alto Research Center, and especially Dr. Bob Tremain for technical discussions, and for sample wafers provided. Professor Ping Ko of U.C. Berkeley and previously of Bell Labs also provided experimental test wafers for analysis. Hewlett Packard Corporation and Xynetics/Electroglas Corporation made this project possible through generous equipment donations which are greatly appreciated. A portion of this research was also supported by the Semiconductor Research Corporation, Grant #82-11-008.

I deeply appreciate the years of help, support and encouragement that my parents, Miles and Frances, have provided me.

TABLE OF CONTENTS

1. INTRODUCTION 1

2. CSIM MODEL AND PROCESS FILE DESCRIPTION 6

2.1. CSIM Model Relationship to the Parameter Extraction Strategy 7

2.2. CSIM Threshold Voltage Model 8

2.3. Modeling the Transconductance with CSIM 9

2.4. The Parameter 'a' and its Function in the CSIM Model 9

2.5. Handling Depletion Mode MOS Devices 10

2.6. Parameter Extraction Sequence 10

2.7. Process File Development 11

3. AUTOMATIC MOS DEVICE PARAMETER EXTRACTION SYSTEM HARDWARE 13

3.1. The 4145 Parameter Analyzer 13

3.2. 2001X Automatic Wafer Prober 14

3.3. 2671G Thermal Printer 16

3.4. 9836 Computer and VAX Link 16

4. AUTOMATIC MOS DEVICE PARAMETER EXTRACTION SYSTEM SOFTWARE 17

4.1. Parameter Extraction Routines 18

4.1.1. Linear Region Extraction of Beta0x, U0x, and Vtx 18

4.1.2. Extraction of the Parameter 2Phif 20

4.1.3. General Linear Region Threshold Analysis and Data Reduction 22

4.1.4. Saturation Region Extraction of Beta0sat, Vthsat, and U1sat 23

4.1.5. Saturation Region Data Reduction 24

4.2. Device Measurement Software 25

4.2.1. Device Type Measurement 25

4.2.2. Device Functionality Verification 27

4.2.3. Device Measurement for Parameter Extraction 28

4.3. Process File Development Software 30

4.4. Interactive Graphics Software....................31

4.5. Automatic Prober Software....................33

5. SAMPLE PROGRAM EXECUTION....................36

5.1. Initial Program Mode Selection....................36

5.2. User Input Lists....................37

5.3. Continuous Extraction Display....................38

5.4. Graphic Subprogram Interaction....................39

5.5. Experimental Results for Various Devices....................40

6. POSSIBLE FUTURE PROGRAM ENHANCEMENTS....................42

APPENDIX 1. PARAMETER EXTRACTION SYSTEM SETUP INSTRUCTIONS............A1

APPENDIX 2. LAYOUT CONSIDERATIONS FOR AUTOMATIC PROBING..................A2

APPENDIX 3. INSTRUCTIONS FOR AUTOMATIC PROBER FILE DEVELOPMENT......A3

APPENDIX 4. CSIM SOFTWARE SOURCECODE (Available on Request).................A4

4.4. Interactive Graphics Software....................31
4.5. Automatic Prober Software....................33
5. SAMPLE PROGRAM EXECUTION....................36
5.1. Initial Program Mode Selection....................36
5.2. User Input Lists....................37
5.3. Continuous Extraction Display....................38
5.4. Graphic Subprogram Interaction....................39
5.5. Experimental Results for Various Devices....................40
6. POSSIBLE FUTURE PROGRAM ENHANCEMENTS....................42
APPENDIX 1. PARAMETER EXTRACTION SYSTEM SETUP INSTRUCTIONS....................A1
APPENDIX 2. LAYOUT CONSIDERATIONS FOR AUTOMATIC PROBING....................A2
APPENDIX 3. INSTRUCTIONS FOR AUTOMATIC PROBER FILE DEVELOPMENT....................A3
APPENDIX 4. CSIM SOFTWARE SOURCECODE (Available on Request)....................A4

1. INTRODUCTION

With the introduction of integrated circuit computer aided design programs such as SPICE in the late 1960's, there arose a critical need for accurate models to simulate the electrical characteristics of transistors. The means to extract consistent and reproducible sets of model parameters was and still remains an essential requirement for successful integrated circuit simulation. An analysis of the present state of affairs in MOS device modeling discloses a diverse collection of parameter sets in use by various universities and corporations. The means by which these organizations extract their model parameters is equally varied, and include such methods as: rapid curve tracer estimates, elaborate time-consuming device physics analysis studies, and global optimization routines providing for model generality. To fully understand the benefits of fully automatic parameter extraction, it is necessary to understand the limitations and shortcomings of the types of methods which are presently in use.

In many instances, a circuit designer is confronted with the need to extract a set of device parameters before any circuit simulations can be attempted. Unless this person has access to, and is familiar with, some form of computer controlled system to lead him through parameter extraction, the designer will likely resort to using a curve tracer to determine rough values for parameters such as threshold voltage, transconductance, etc.. The very nature of this method of parameter extraction induces a large degree of uncertainty, and non-reproducability. Error may be introduced in the actual visual extraction of the device parameters from the measured characteristics, or it may occur that the model and the choice of parameters are not sufficient for simulating all of the observed device properties.

At the opposite end of the modeling spectrum from the simple device model user, stands the device physicist who analyzes and develops models to explain recently observed device phenomena. This theoretician is generally not involved

with computer aided circuit design, and may develop models which can not be feasibly used in simulators such as SPICE. When extensive device models have been developed, the option is present to translate the elaborate set of model parameters into a smaller subset suitable for circuit simulation. This translation can create uncertainty, and this alone makes this method of parameter extraction undesirable. The fact that the methods used to extract large sets of parameters may require extensive time and knowledge places the greatest limitation on this approach. The probability that parameter extraction for a very complicated model can be done fully automatically and can handle devices over an entire range of device types and geometries is very slim.

In some instances programs have been written to systemize device parameter extraction using global optimization algorithms. This method allows for simple model modification, and is preferable to the previously mentioned approaches. Assuming a suitable model and parameter set are chosen, a set of parameters can be generated which, as a group, will generally simulate the measured device characteristics. Unfortunately, there are pitfalls that may arise when using a global optimization method. A major problem encountered is that limits on the variation of the parameters in the parameter set must be specified to enable the global optimizer to converge to an accurate solution. The designer may not have an intuitive feel for the acceptable ranges of some parameters, and these limits will be device type and device geometry dependent. Depending upon the initial selection of allowable parameter ranges, it is possible for various sets of parameters to be extracted. This results in the situation that though a set of parameters can be derived to model the electrical characteristics of an isolated device, it is very difficult to corelate the extracted model parameters with device geometry. The global optimization approach is not a method that can be easily automated for testing multiple devices of

with computer aided circuit design, and may develop models which can not be feasibly used in simulators such as SPICE. When extensive device models have been developed, the option is present to translate the elaborate set of model parameters into a smaller subset suitable for circuit simulation. This translation can create uncertainty, and this alone makes this method of parameter extraction undesirable. The fact that the methods used to extract large sets of parameters may require extensive time and knowledge places the greatest limitation on this approach. The probability that parameter extraction for a very complicated model can be done fully automatically and can handle devices over an entire range of device types and geometries is very slim.

In some instances programs have been written to systemize device parameter extraction using global optimization algorithms. This method allows for simple model modification, and is preferable to the previously mentioned approaches. Assuming a suitable model and parameter set are chosen, a set of parameters can be generated which, as a group, will generally simulate the measured device characteristics. Unfortunately, there are pitfalls that may arise when using a global optimization method. A major problem encountered is that limits on the variation of the parameters in the parameter set must be specified to enable the global optimizer to converge to an accurate solution. The designer may not have an intuitive feel for the acceptable ranges of some parameters, and these limits will be device type and device geometry dependent. Depending upon the initial selection of allowable parameter ranges, it is possible for various sets of parameters to be extracted. This results in the situation that though a set of parameters can be derived to model the electrical characteristics of an isolated device, it is very difficult to corelate the extracted model parameters with device geometry. The global optimization approach is not a method that can be easily automated for testing multiple devices of

various device types and sizes to gain meaningful statistical data.

It is easily perceived that a fully integrated solution which encompasses the entire range from device modeling to automatic parameter extraction and subsequent circuit simulation would greatly increase productivity and reliability in the field of circuit design. The university environment provides the perfect opportunity to develop, introduce, and enhance such a solution.

The approach to be outlined here revolves around a new compact short channel insulated gate field effect transistor model (CSIM) which was initially developed, at Bell Labs by H. Poon [1], and was enhanced at Berkeley and Xerox PARC by Bing Sheu et. al. [2]. The model is compact to allow for reasonable execution speed using SPICE, and has been implemented in SPICE version 2G.6 by Ralph Liu [3]. The means to automatically generate device parameters without requiring any initial parameter estimates or parameter ranges, was of prime importance in the development of the CSIM model. The use of a consistent reproducible parameter extraction approach allows for the modeling of parameter variations with device geometry. This device geometry dependent modeling is used to develop a **process file** which can model devices of a given device type over process ranges of channel lengths and widths.

The process file approach allows the circuit designer to consider a design in terms of the physical mask dimensions of a layout. This method of analysis replaces the cumbersome approach of requiring the circuit designer to determine parameters to model every device and section of interconnect encountered in an integrated circuit layout. In the case of a MOS integrated circuit, a process file would describe the following: all NMOS and PMOS devices categorized as being enhancement, depletion or zero-threshold, diffusion interconnects, polysilicon levels of interconnect, and metal levels of interconnect. A designer

will enter device and interconnect dimensions, and will provide the SPICE simulation program with a process file name. The electrical parameters determined by the automatic parameter extraction program described in this report are used directly in a CSIM Capacitance model under development by Bing Sheu. Programs to automatically determine the parasitic resistance and capacitance values to which model device interconnect will be the subject of future work at Berkeley.

The following report introduces the structure of the CSIM model, and the process file approach. It relates how the development of the model was linked to the need for a fully automatic parameter extraction capability, and how the program extracts all initial parameter estimates prior to final parameter optimization. The hardware implementation of the parameter extraction system at the University of California at Berkeley is described, and system setup and use are explained in the Appendices.

The integrated system parameter extraction software is written in Pascal, and has been partitioned into six major sections. The measurement of devices is fully program controlled, requiring the user to input only the supply voltage which will be used in circuit operation. The parameter extraction software is fully self contained, and requires the user to supply only the physical parameters of the gate oxide thickness and the temperature of the wafer test chuck before full parameter extraction can proceed. An extensive graphics package is described which allows the user to view firsthand the simulated extracted model parameter characteristics and compare them to the actual measured device electrical characteristics. Incorporated within the software is the capability to develop process files after multiple devices have been measured. These process files are transported to a mainframe computer to be accessed by SPICE.

will enter device and interconnect dimensions, and will provide the SPICE simulation program with a process file name. The electrical parameters determined by the automatic parameter extraction program described in this report are used directly in a CSIM Capacitance model under development by Bing Sheu. Programs to automatically determine the parasitic resistance and capacitance values to which model device interconnect will be the subject of future work at Berkeley.

The following report introduces the structure of the CSIM model, and the process file approach. It relates how the development of the model was linked to the need for a fully automatic parameter extraction capability, and how the program extracts all initial parameter estimates prior to final parameter optimization. The hardware implementation of the parameter extraction system at the University of California at Berkeley is described, and system setup and use are explained in the Appendices.

The integrated system parameter extraction software is written in Pascal, and has been partitioned into six major sections. The measurement of devices is fully program controlled, requiring the user to input only the supply voltage which will be used in circuit operation. The parameter extraction software is fully self contained, and requires the user to supply only the physical parameters of the gate oxide thickness and the temperature of the wafer test chuck before full parameter extraction can proceed. An extensive graphics package is described which allows the user to view firsthand the simulated extracted model parameter characteristics and compare them to the actual measured device electrical characteristics. Incorporated within the software is the capability to develop process files after multiple devices have been measured. These process files are transported to a mainframe computer to be accessed by SPICE.

Routines have been developed to allow for fully automatic probing of devices across entire wafers to develop a statistical base of process parameters. A set of simple menus provides a designer having no prior experience in device parameter analysis with the capability of selecting various system options, and of performing full reproducible parameter extractions. After the project implementation is discussed, a sample program execution session is presented. Experimental results for various integrated circuit processes are reported.

The parameter extraction system described here represents a key link in a fully integrated solution which has been developed to remove major bottlenecks in the process of achieving fast and accurate MOS circuit design. It is hoped that this entire integrated solution will evolve into a standard industry wide approach that can unify modeling, parameter extraction and simulation for MOS integrated circuits.

2. CSIM MODEL AND PROCESS FILE DESCRIPTION

The development of the CSIM model was intimately connected to the development of a fully automatic MOS device parameter extraction capability. It was essential to construct a model whose parameters could be determined using sets of local optimizations, to avoid the inherent difficulties in using global optimization for parameter extraction. The effects of mobility degradation due to increased vertical electric field in the channel and velocity saturation due to increased transverse electric fields in the channel, are modeled with CSIM. Velocity saturation is essentially nonexistent at very small drain-source bias potentials, while it may be a dominating factor with drain source bias on the order of the supply voltage, depending upon the supply voltage and the device geometry. This is a critically important fact, since it permits the decoupling of the velocity saturation and the low field mobility degradation effects in doing parameter extraction. Mobility degradation parameters can be extracted at very low drain-source bias, and can then be used in extracting the velocity saturation parameters at high drain-source biases. Experimentation with parameter extraction using the CSIM model has shown that by using drain-source biases near zero, and drain-source biases near the supply voltage, a set of parameters can be extracted which can accurately model the entire range of drain-source biases.

The CSIM model consists of 17 parameters which are used to simulate the electrical characteristics of a single device. Of these 17 parameters, eight primary parameters represent well understood phenomena such as transconductance or mobility reduction. These eight parameters make up a first order set of parameters which can be used as a complete set of device parameters. The remaining nine secondary parameters represent the variations of some primary parameters with drain-source and bulk-source potentials and provide a second-

2. CSIM MODEL AND PROCESS FILE DESCRIPTION

The development of the CSIM model was intimately connected to the development of a fully automatic MOS device parameter extraction capability. It was essential to construct a model whose parameters could be determined using sets of local optimizations, to avoid the inherent difficulties in using global optimization for parameter extraction. The effects of mobility degradation due to increased vertical electric field in the channel and velocity saturation due to increased transverse electric fields in the channel, are modeled with CSIM. Velocity saturation is essentially nonexistent at very small drain-source bias potentials, while it may be a dominating factor with drain source bias on the order of the supply voltage, depending upon the supply voltage and the device geometry. This is a critically important fact, since it permits the decoupling of the velocity saturation and the low field mobility degradation effects in doing parameter extraction. Mobility degradation parameters can be extracted at very low drain-source bias, and can then be used in extracting the velocity saturation parameters at high drain-source biases. Experimentation with parameter extraction using the CSIM model has shown that by using drain-source biases near zero, and drain-source biases near the supply voltage, a set of parameters can be extracted which can accurately model the entire range of drain-source biases.

The CSIM model consists of 17 parameters which are used to simulate the electrical characteristics of a single device. Of these 17 parameters, eight primary parameters represent well understood phenomena such as transconductance or mobility reduction. These eight parameters make up a first order set of parameters which can be used as a complete set of device parameters. The remaining nine secondary parameters represent the variations of some primary parameters with drain-source and bulk-source potentials and provide a second-

order, high-accuracy model. The CSIM approach analyzes the parameters that are extracted for devices of various channel lengths and widths, and models their variations with device geometry to create a **process file** which can describe an entire range of device geometries.

2.1. CSIM Model Relationship to the Parameter Extraction Strategy

The structure of the CSIM model was developed to allow three parameters to be extracted at small drain-source biases, and three parameters to be extracted at drain-source biases near the supply voltage. Given the structure of the model, it was possible to use a **Linear Least Square** algorithm in the process of determining these sets of three parameters. The optimization was performed using a combination of a **Newton-Raphson** algorithm and the Linear Least Square Algorithm to converge rapidly to a solution.

At small drain-source biases, values of gate-source bias are chosen so that a MOSFET under test is in the linear region of operation. The drain-source current in the linear region(Equation 1) contains three parameters which are extracted: the transconductance (Beta0lin), the threshold voltage (Vthlin), and the mobility reduction parameter (U0). The parameter **a** is a function of Vbs and its derivation will be described later.

$$Ids_{lin} = \frac{Beta0lin}{(1+U0*(Vgs-Vthlin))}(Vgs-Vthlin-\frac{a}{2}Vds)\,Vds \tag{1}$$

For parameter extraction purposes, the linear region equation for drain-source current is transformed to the equation for the conductance in the linear region. Figure 1 shows how curves of the conductance versus gate-source potential reflect the values of the linear region parameters.

At large drain-source potentials, an enhancement MOSFET under test will operate in the saturation region. Special considerations for depletion MOSFETs

Figure 1

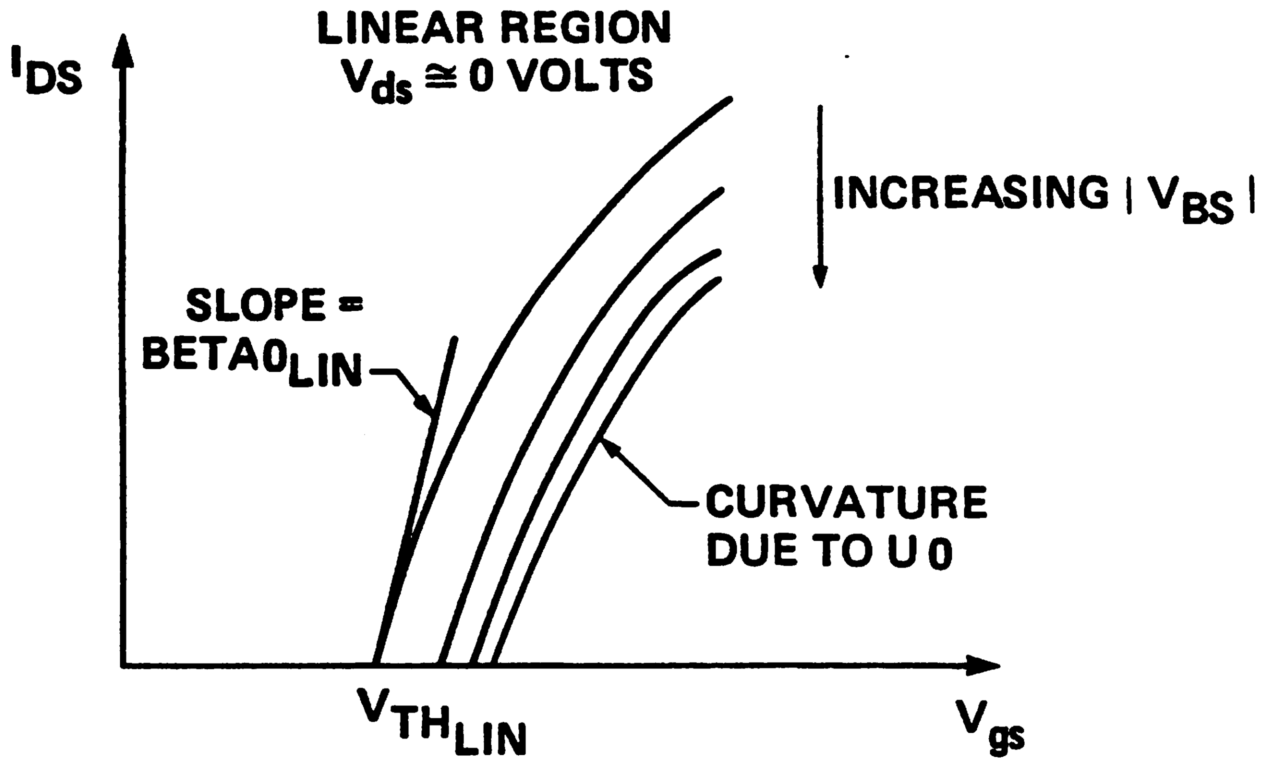


Figure 2

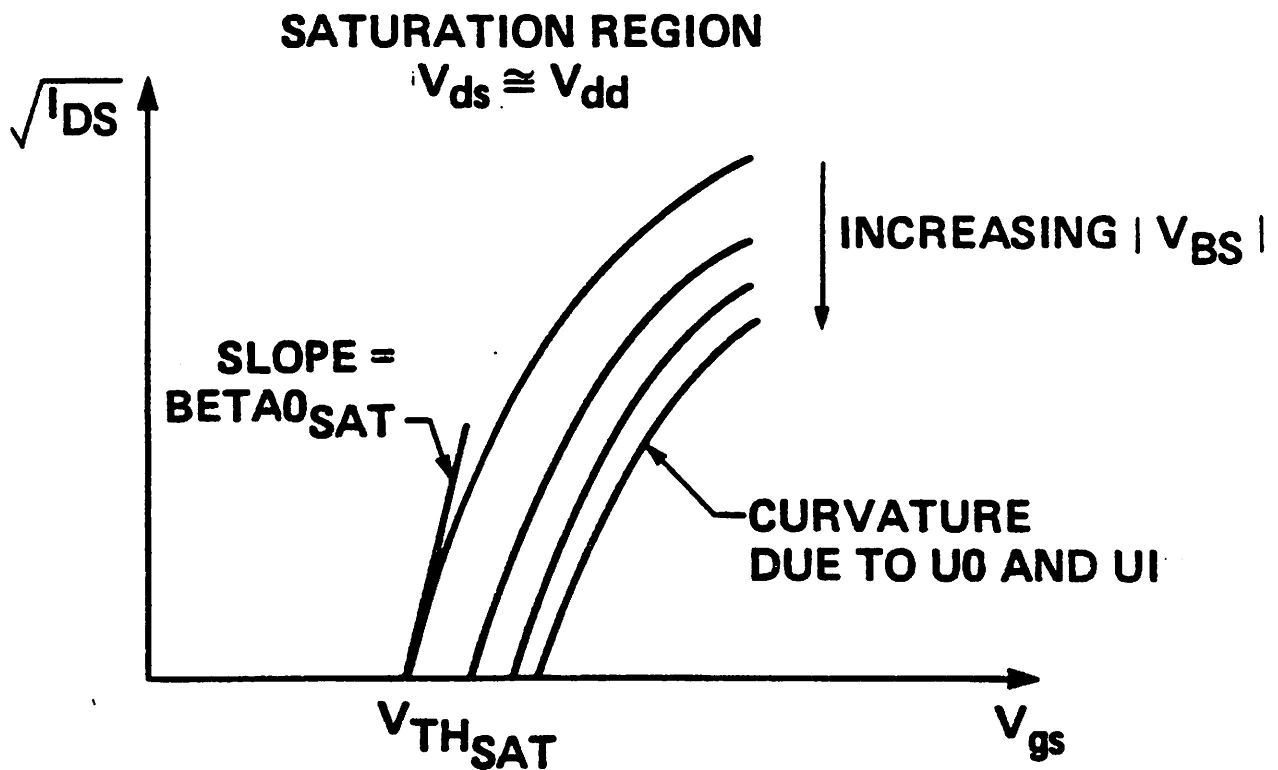


Figure 1

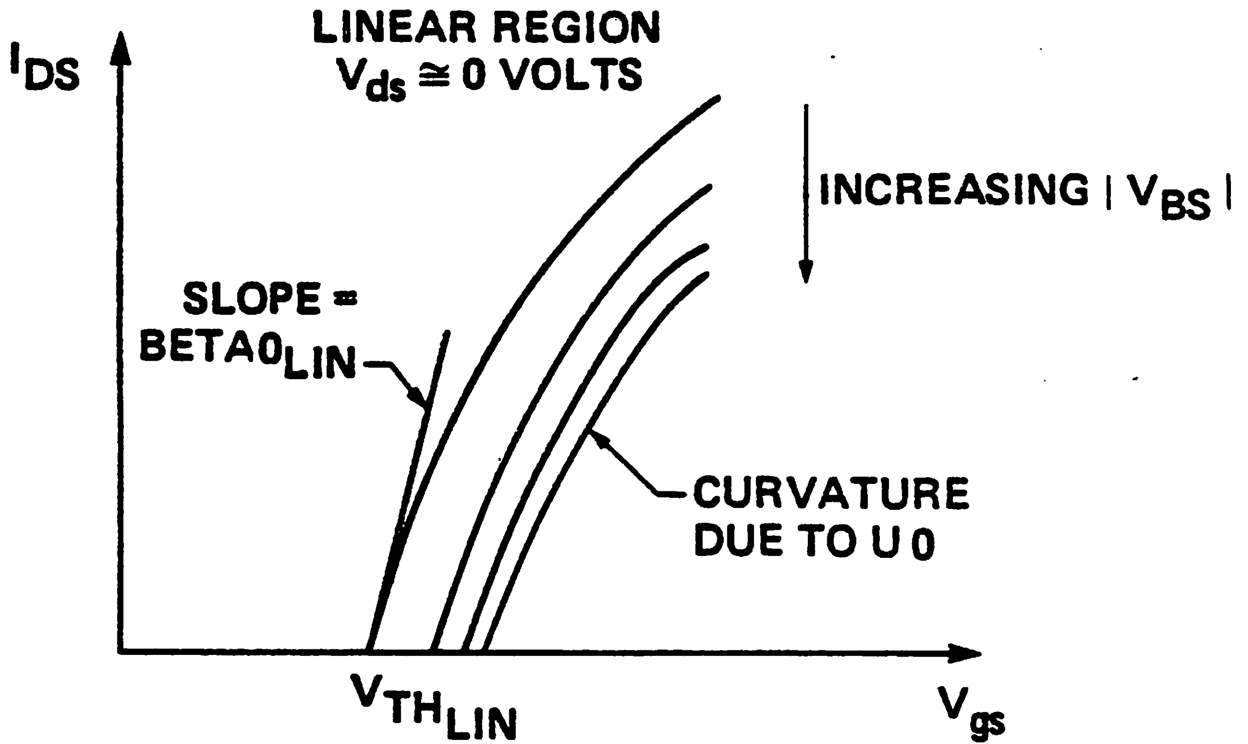
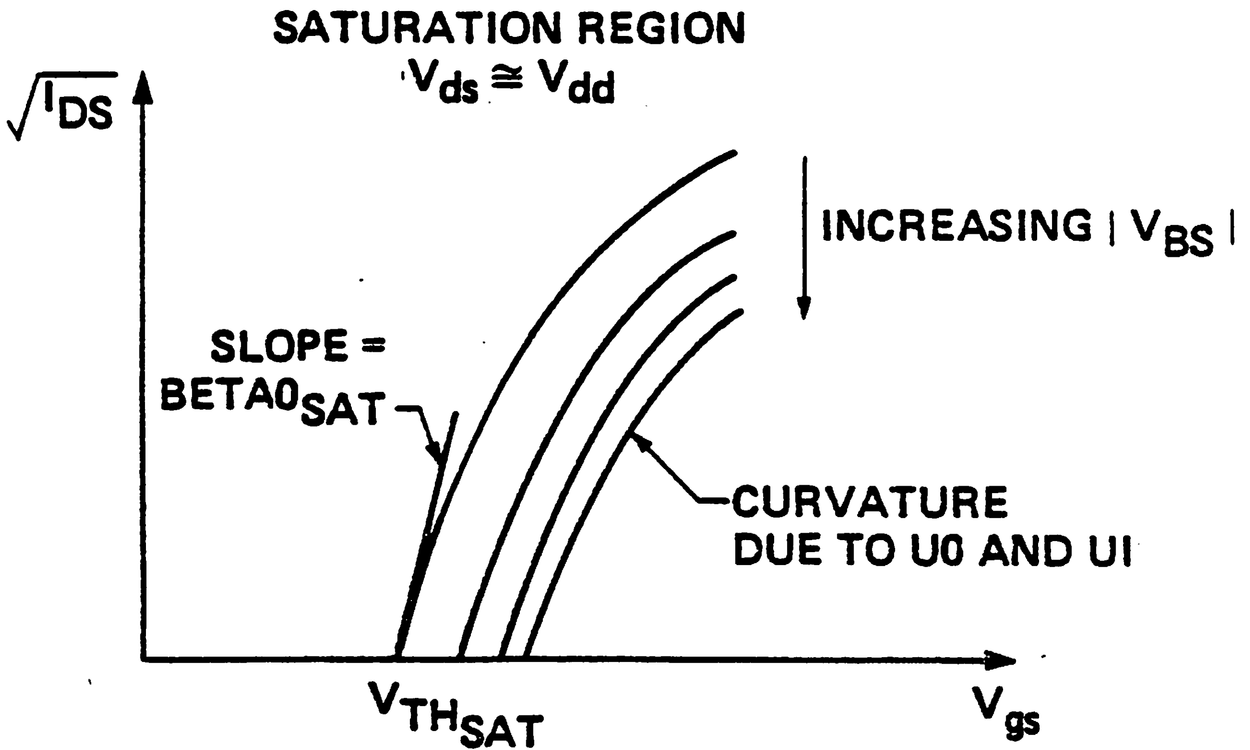


Figure 2



are discussed later. The drain-source current in the saturation region (Equation 2) contains three additional parameters which are extracted: the transconductance in saturation (Beta0sat), the threshold voltage in saturation (Vthsat), and the velocity saturation parameter U1. The velocity saturation model development for CSIM was done by Don Scharfetter, and is reported by Bing Sheu et. al.[2]. The voltage *Vdssat* at which saturation occurs is modeled in Equation (3).

$$Ids_{sat} = \frac{Beta0sat(Vgs - Vthsat)^2}{(1 + U0(Vgs - Vthsat))a\left(1.0 + \frac{U1}{a(Vgs - Vthsat)} + \sqrt{1.0 + \frac{2.0 * U1}{a(Vgs - Vthsat)}}\right)} \quad (2)$$

$$Vdssat = \frac{2(Vgs - Vth)}{a\sqrt{1 + \frac{U1 * (Vgs - Vth)}{a} + \left(1 + \frac{2U1(Vgs - Vth)}{a}\right)^{\frac{1}{2}}}} \quad (3)$$

Figure 2 shows how plots of the square-root of drain-source current reflect the values of the saturation parameters.

2.2. CSIM Threshold Voltage Model

There are five primary CSIM parameters which model the threshold voltage: VFB which is the flatband voltage, 2Phif which reflects the channel doping level, K1 which models the bulk doping effect, and the combination of K2 and ETA which model the drain induced barrier lowering effect. The basic CSIM threshold voltage equation (Equation 3) parameters are extracted from the values of Vthlin and Vthsat which have been determined at various *Vds* and *Vbs* bias conditions. The parameter ETA is modeled with three CSIM parameters (Equation 4) to reflect variations with *Vbs* and *Vds* bias.

$$Vth = VFB + 2Phif + K1 * (2Phif - Vbs)^{1/2} - K2 * (2Phif - Vbs) - ETA * (Vds) \quad (4)$$

$$ETA = ETA_0 + X2ETA * (2Phif - VBS) + X3ETA * (Vdd - Vds) \quad (5)$$

2.3. Modeling the Transconductance with CSIM

Modeling the transconductance of a MOS device is critical, and to reflect this importance, five CSIM parameters are used to model this device parameter. The values of Beta0lin which are extracted over ranges of Vbs and Vds biases are used to extract the parameters BETA0 and X2BETA0 to model the transconductance in the linear region (Equation 6).

$$Beta0_{linear} = BETA0 + X2BETA0 * Vbs \tag{6}$$

In a similar fashion the values of Beta0sat which have been extracted are used to extract the CSIM parameters BETA0SAT, X2BETA0SAT, and X3BETA0SAT which model Beta0 in the saturation region (Equation 7).

$$Beta0_{sat} = BETA0SAT + X2BETA0SAT * Vbs + X3BETA0SAT * (Vds - Vdd) \tag{7}$$

Using the parameters BETA0, BETA0SAT, and X3BETA0SAT, a parabolic fit for the transconductance versus Vds bias is performed. Allowing the transconductance to vary with bias, as is done in CSIM, provides greater accuracy, and has the added advantage that it can absorb some effects that may not be accounted for explicitly in the model. In this manner, phenomena associated with very short channel devices may be implicitly handled with the CSIM approach well before the device physics causing these effects can be well understood and modeled.

2.4. The Parameter 'a' and its Function in the CSIM Model

The basic first order charge theory for a MOS device, derived by Ihantola and Moll[4], contains 3/2 power terms (Equation 8) which are computationally very time consuming in circuit simulators such as SPICE.

$$IDS_{lin} = BETA0((Vgs - 2phif - Vfb)Vds - \frac{Vds^2}{2} - \frac{2}{3}\frac{\sqrt{2qe_{si}N_A}}{C_{ox}}(T))$$

$$T = (Vds + 2phif - Vbs)^{3/2} - (2phif - Vbs)^{3/2} \tag{8}$$

2.3. Modeling the Transconductance with CSIM

Modeling the transconductance of a MOS device is critical, and to reflect this importance, five CSIM parameters are used to model this device parameter. The values of Beta0lin which are extracted over ranges of *Vbs* and *Vds* biases are used to extract the parameters BETA0 and X2BETA0 to model the transconductance in the linear region (Equation 6).

$$Beta0_{linear} = BETA0 + X2BETA0 * Vbs \quad (6)$$

In a similar fashion the values of Beta0sat which have been extracted are used to extract the CSIM parameters BETA0SAT, X2BETA0SAT, and X3BETA0SAT which model Beta0 in the saturation region (Equation 7).

$$Beta0_{sat} = BETA0SAT + X2BETA0SAT * Vbs + X3BETA0SAT * (Vds - Vdd) \quad (7)$$

Using the parameters BETA0, BETA0SAT, and X3BETA0SAT, a parabolic fit for the transconductance versus *Vds* bias is performed. Allowing the transconductance to vary with bias, as is done in CSIM, provides greater accuracy, and has the added advantage that it can absorb some effects that may not be accounted for explicitly in the model. In this manner, phenomena associated with very short channel devices may be implicitly handled with the CSIM approach well before the device physics causing these effects can be well understood and modeled.

2.4. The Parameter 'a' and its Function in the CSIM Model

The basic first order charge theory for a MOS device, derived by Ihantola and Moll[4], contains 3/2 power terms (Equation 8) which are computationally very time consuming in circuit simulators such as SPICE.

$$IDS_{lin} = BETA0((Vgs - 2phif - Vfb)Vds - \frac{Vds^2}{2} - \frac{2}{3}\frac{\sqrt{2q\epsilon_{si}N_A}}{C_{ox}}(T))$$

$$T = (Vds + 2phif - Vbs)^{3/2} - (2phif - Vbs)^{3/2} \quad (8)$$

In the CSIM model, the three-halves power terms are replaced through the use of the term **a** shown in Equation (9). The parameter **a** was derived numerically by H. Poon [1], using a power series expansion about *Vbs* and *Vds*.

$$a = 1.0 - \frac{1.0}{1.744 + 0.8364(2phif - Vbs)} \frac{K1}{2} \sqrt{2phif - Vbs} \quad (9)$$

The use of the term **a** is needed in the linear region parameter extraction for the purpose of interpolating the threshold voltage and mobility reduction parameters to their values at zero drain-source bias. Equations 1,4 and 9 are solved simultaneously to derive the linear region parameters Beta0, U0, VFB, 2Phif, K1, and K2. This procedure is described in detail in the *Parameter Extraction Software* section.

2.5. Handling Depletion Mode MOS Devices

With *Vds* values near the supply voltage, a Depletion Mode transistor may or may not be in the saturation region of operation, depending upon the value of *Vgs*. In order to handle Depletion Mode devices with the same routines, and to extract the same parameters as in the case of Enhancement Mode devices, program modifications were necessary. At every value of *Vgs* which is analyzed, it is necessary to determine within the optimization routines whether the device is in the linear region or the saturation region. Once this is known, the coefficients for the Linear Least Square algorithm can be properly determined, and the optimization can continue to derive the necessary parameters.

2.6. Parameter Extraction Sequence

The parameter extraction strategy results in a set of 17 device parameters. The parameters VFB, 2Phif, K1, K2, BETA0, X2BETA0, U0 and X2U0 are derived from the linear region extraction, and in the case of BETA0 and X2BETA0 are

modified to account for the effects due to some saturation parameters. X2U0 is described in Equation (22). After the linear region threshold parameters have been derived, it is possible to perform parameter extraction for the saturation region which represents drain-source biases near *Vdd*. Analysis of the saturation region data results in the parameters U1, X2U1, X3U1, ETA, X2ETA, X3ETA, BETA0SAT, X2BETA0SAT, and X3BETA0SAT. Once the 17 CSIM parameters have been determined for a device, they are stored for later development of a process file.

The prefixes X2 and X3 used throughout this report and the CSIM model deserve further explanation. Parameters with the prefix X2 represent the bulk-source dependency of one of the eight main CSIM parameters, while those parameters with the prefix X3 represent the drain-source dependency of one of the eight main CSIM parameters. The reader should refer to Equations (29) through (33) for examples of these parameters.

2.7. Process File Development

The fact that variations between devices in close proximity on a wafer are usually quite small allows devices of varying sizes at the same die location to be analyzed to create a **process file**. Each type of device which is present on an integrated circuit test die is modeled using 54 parameters which are generated by the automatic parameter extraction program. As an example, a typical CMOS process file would have 54 parameters to model all NMOS enhancement devices present on a given test die, and 54 parameters to model all PMOS enhancement devices present on the same test die. This approach allows statistical analysis to be done on the process parameters that have been generated across a wafer to model variations in parameters such as substrate doping levels.

modified to account for the effects due to some saturation parameters. X2U0 is described in Equation (22). After the linear region threshold parameters have been derived, it is possible to perform parameter extraction for the saturation region which represents drain-source biases near *Vdd*. Analysis of the saturation region data results in the parameters U1, X2U1, X3U1, ETA, X2ETA, X3ETA, BETA0SAT, X2BETA0SAT, and X3BETA0SAT. Once the 17 CSIM parameters have been determined for a device, they are stored for later development of a process file.

The prefixes X2 and X3 used throughout this report and the CSIM model deserve further explanation. Parameters with the prefix X2 represent the bulk-source dependency of one of the eight main CSIM parameters, while those parameters with the prefix X3 represent the drain-source dependency of one of the eight main CSIM parameters. The reader should refer to Equations (29) through (33) for examples of these parameters.

2.7. Process File Development

The fact that variations between devices in close proximity on a wafer are usually quite small allows devices of varying sizes at the same die location to be analyzed to create a **process file**. Each type of device which is present on an integrated circuit test die is modeled using 54 parameters which are generated by the automatic parameter extraction program. As an example, a typical CMOS process file would have 54 parameters to model all NMOS enhancement devices present on a given test die, and 54 parameters to model all PMOS enhancement devices present on the same test die. This approach allows statistical analysis to be done on the process parameters that have been generated across a wafer to model variations in parameters such as substrate doping levels.

Figure 3

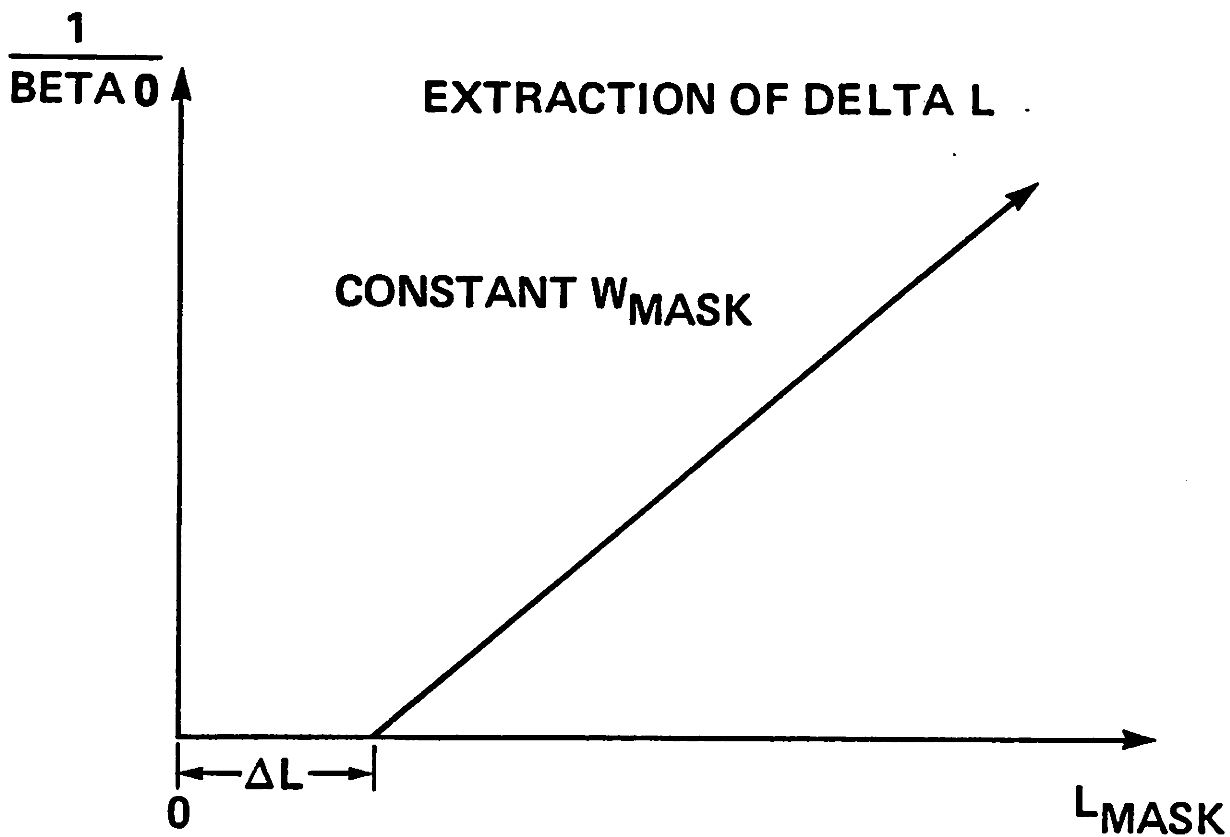
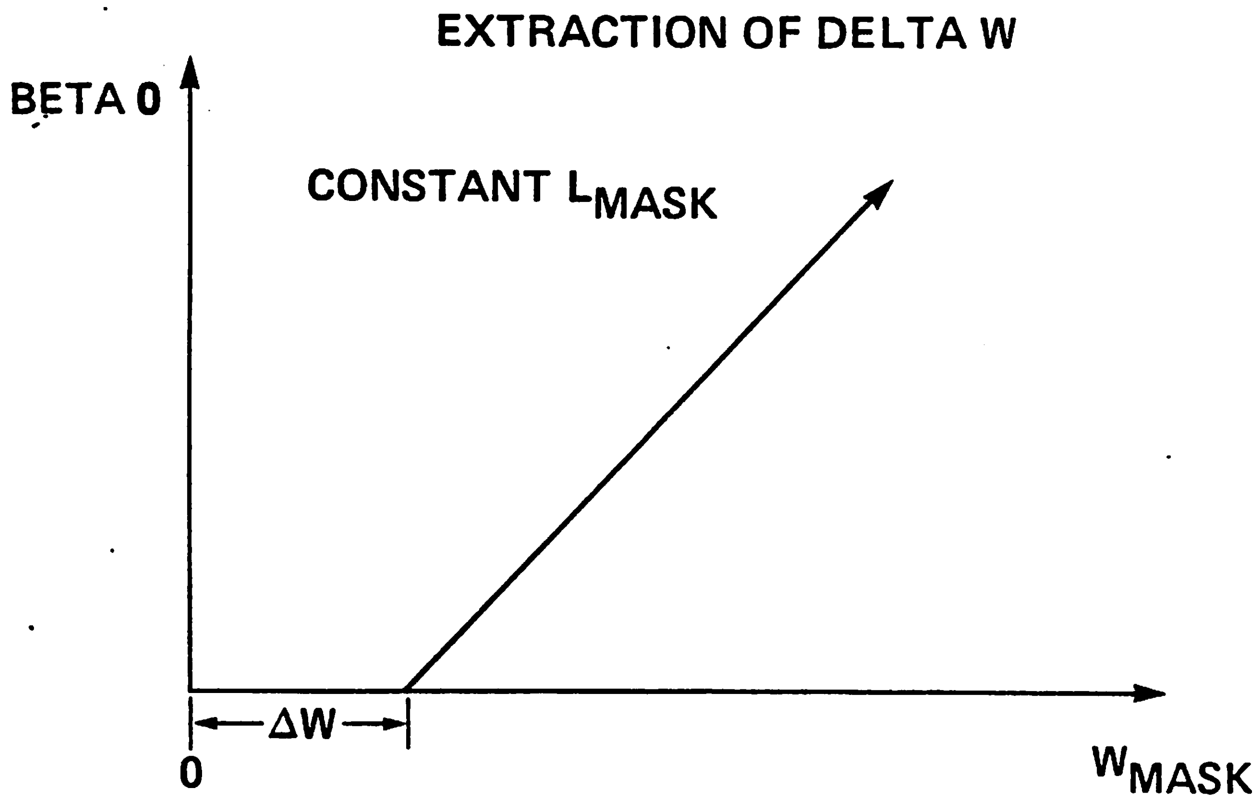


Figure 4



Process parameters are determined through analysis of the variation in the 17 CSIM model parameters with device channel length and width. The first calculation performed analyzes **BETA0** versus drawn length and drawn width (Equation 10) to determine **DELTAL** and **DELTAW**

$$BETA0=\mu_n\, Cox\frac{(W_m+\Delta W)}{(L_m+\Delta L)} \tag{10}$$

Figures 3 and 4 show the relationships between BETA0 and drawn width and drawn length that are used to determine DeltaL and DeltaW. The low field mobility, DELTAL and DELTAW are stored in the process file and are used by SPICE in determining the device gain.

Once DELTAL and DELTAW have been determined, they are used to convert all values of Mask length and width to the corresponding values of electrical channel length and width. At this point, the remaining 16 CSIM parameters are modeled versus electrical channel length and width using three parameters: **P0** **PL**, and **PW** (Equation 11).

$$Parameter(W,L)=P0(1+\frac{PL}{L_e}+\frac{PW}{W_e}) \tag{11}$$

The values of P0, PL and PW for the remaining 16 parameters are added to the process file. The user supplied operating voltage, temperature, and oxide thickness when added to the other process parameters result in a 54 parameter file for any given device type on a test die location.

Process parameters are determined through analysis of the variation in the 17 CSIM model parameters with device channel length and width. The first calculation performed analyzes **BETA0** versus drawn length and drawn width (Equation 10) to determine **DELTAL** and **DELTAW**

$$BETA0=\mu_n Cox\frac{(W_m+\Delta W)}{(L_m+\Delta L)} \tag{10}$$

Figures 3 and 4 show the relationships between BETA0 and drawn width and drawn length that are used to determine DeltaL and DeltaW. The low field mobility, DELTAL and DELTAW are stored in the process file and are used by SPICE in determining the device gain.

Once DELTAL and DELTAW have been determined, they are used to convert all values of Mask length and width to the corresponding values of electrical channel length and width. At this point, the remaining 16 CSIM parameters are modeled versus electrical channel length and width using three parameters: **P0** **PL**, and **PW** (Equation 11).

$$Parameter(W,L)=P0(1+\frac{PL}{L_e}+\frac{PW}{W_e}) \tag{11}$$

The values of P0, PL and PW for the remaining 16 parameters are added to the process file. The user supplied operating voltage, temperature, and oxide thickness when added to the other process parameters result in a 54 parameter file for any given device type on a test die location.

3. AUTOMATIC MOS DEVICE PARAMETER EXTRACTION SYSTEM HARDWARE

The automatic parameter extraction system at Berkeley consists of five integrated elements: a Hewlett-Packard 9836 desktop computer, a Xynetics/Electroglas 2001X fully automated wafer prober, a Hewlett Packard 4145 Semiconductor Parameter Analyzer, a Hewlett Packard 2671G Graphics Thermal Printer, and a link to a network of Digital Equipment VAX 11/780 and VAX 11/750 computers (Figure 5). The elements of this system, with the exception of the VAX mainframes, are linked together with an IEEE-488 Bus. The VAX mainframes communicate with the 9836 Desktop computer over a separate RS-232 connection. For a majority of the system functions, as in the case of device measurements, the 9836 computer acts as a controller while a peripheral such as the 4145 parameter analyzer is under remote supervision.

3.1. The 4145 Parameter Analyzer

All device measurements are taken with the 4145 Semiconductor Parameter Analyzer which, for the purposes of the automatic parameter extraction system, is operated in a programmable digital curve tracer fashion. Once the 9836 automatic extraction program has been given the supply voltage and device terminal connections to the 4145, all measurement setup for the 4145 and data aquisition from the 4145 is automatically controlled by the 9836. Preparing the 4145 for use with the extraction system is discussed in Appendix 1. This instrument has some limited MOS parameter extraction capabilities, and internal graphics that permit it to be used as a stand alone system for device analysis and research. Since the 4145 lacks major programming capabilities, it was not possible to exploit some of its features in developing an automatic extraction facility.

Figure 5

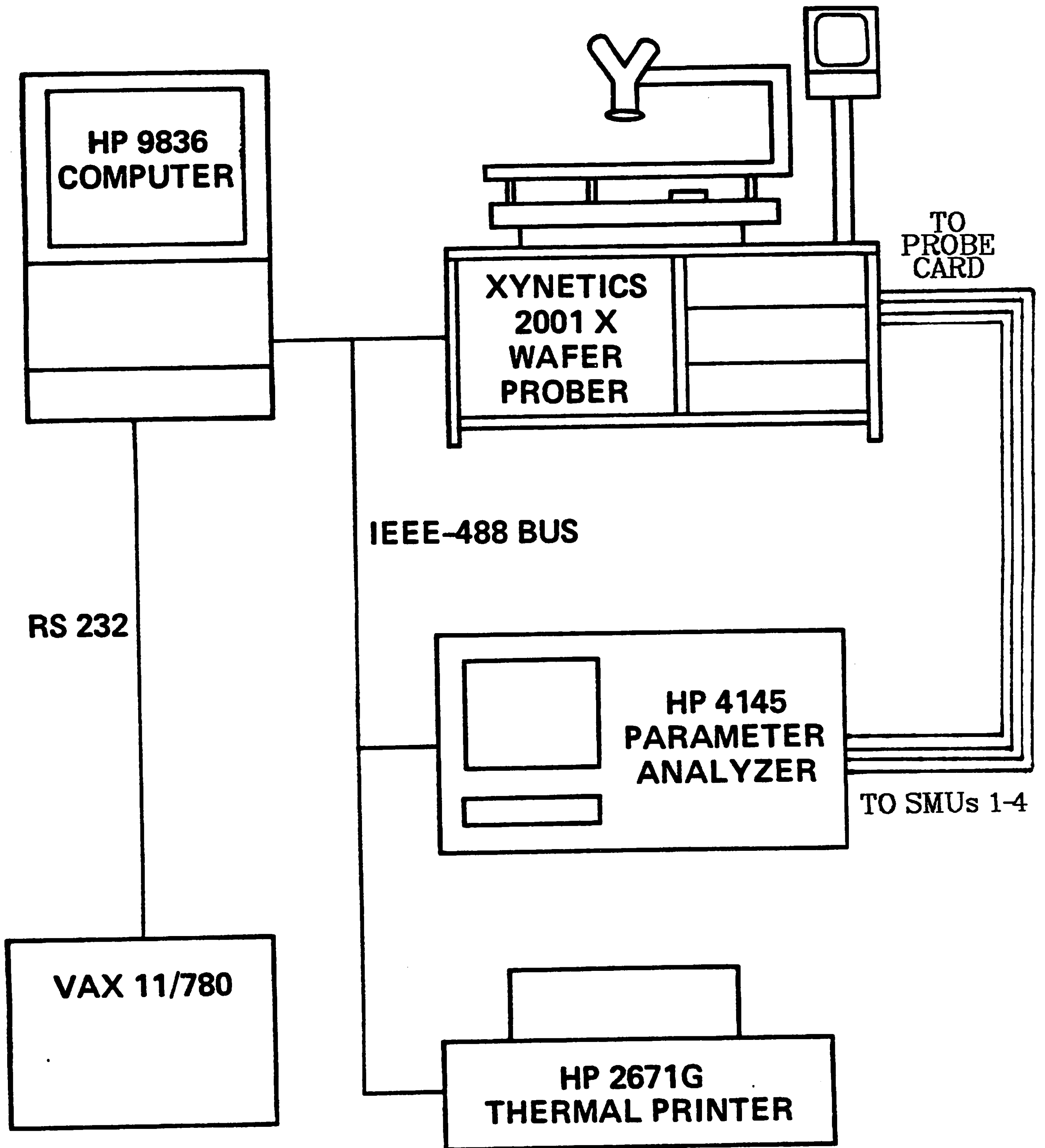
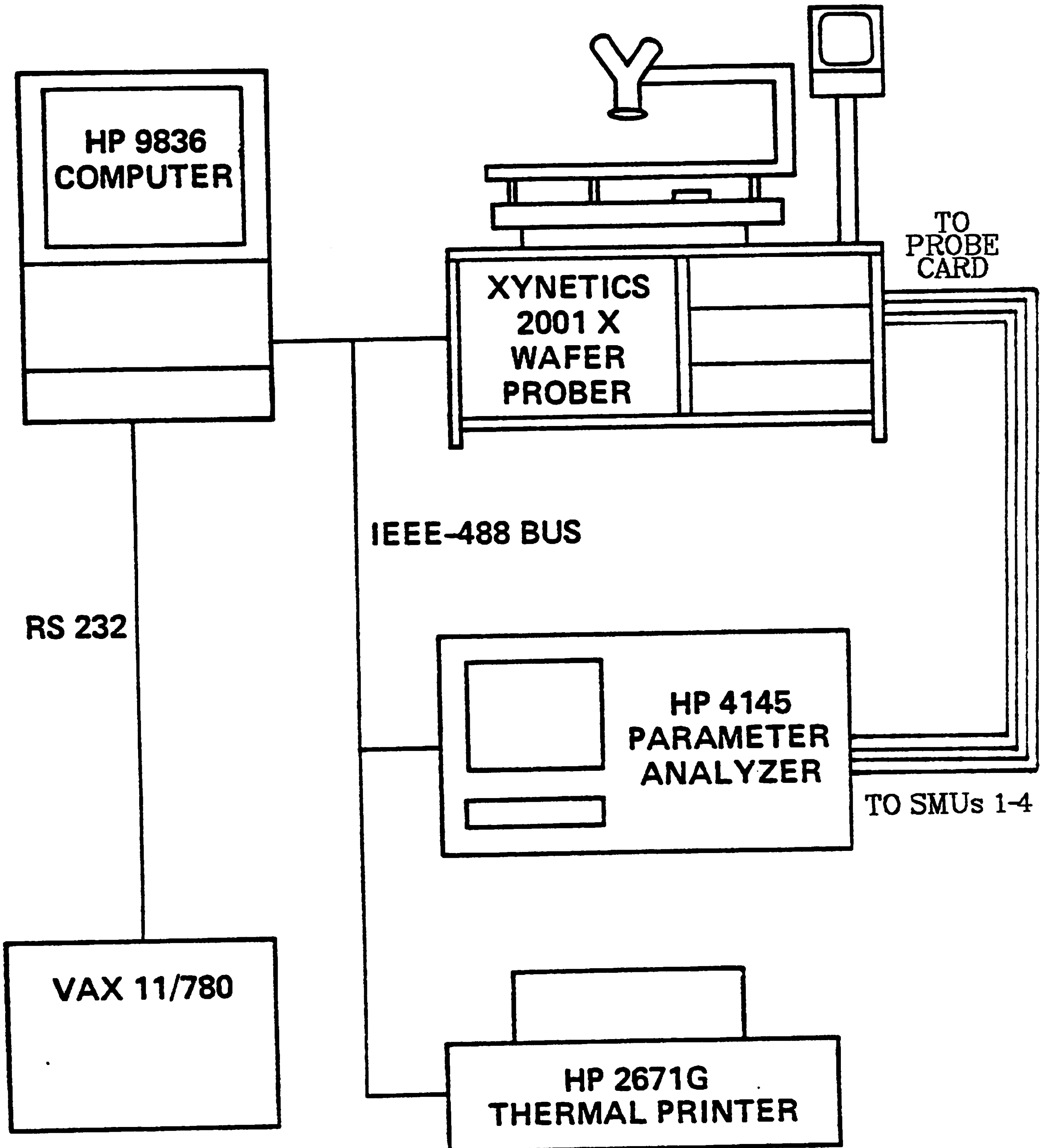


Figure 5



Fortunately, the 4145 is a flexible, high accuracy instrument which is ideal for measuring single transistors. It has four stimulus measurement units capable of being setup as constant or stepped voltage or current sources. These units can be programmed to set voltage levels up to 100V, or current levels from 1pA to 100mA. In addition to these four channels, two programmable voltage sources, and two voltage monitor channels are available to be used when needed for testing more complex structures. The major extraction system limitation resulting from the use of the 4145 as opposed to a more elaborate tester, lies in the fact that any test structure is limited to at most eight connections, and in some instances even less than eight are allowed.

To allow for layout flexibility, but within the constraints of the 4145 limitations, the extraction software allows any of the four required MOS device terminals to be connected to any of the four stimulus measurements. For best results this generally requires that test transistors be laid out in a regular pad arrangement. In general it is not recommended that any pads connected to device terminals be shared, as this may make fully automatic wafer probing impossible. Instructions concerning pad layout, and potential layout procedures to avoid are found in Appendix 2. A solution to these layout difficulties would be the use of a good quaility programmable switching matrix. A Hewlett Packard 4062 parameter analyzer system comes with such a switching matrix, and would allow more general and extensive test structure characterization.

3.2. 2001X Automatic Wafer Prober

The 2001X wafer prober in use at Berkeley is a precision programmable wafer prober with full capability for external control from the 9836 computer. As in the case of the 4145, the 2001X prober has extensive internally programmed wafer stepping and probing capabilities, many of which were not

necessary for the automatic parameter extraction project, but could find use in a number of other applications at Berkeley. The automatic prober setup, which is required prior to external control by the 9836, involves turning on the prober, initializing the prober I/O system, and aligning a wafer. This is described in Appendix 1, and takes approximately two minutes for an experienced user.

To provide for flexibility in use of the prober the parameter extraction software provides for three different modes of prober operation: **fully automatic, semi-automatic, and single device**. Fully automatic and semi-automatic prober operation are used to test multiple devices for the purpose of generating process files, while single device operation is organized towards model experimentation and analysis. Fully automatic or semi-automatic prober operation requires that a prober file containing such information as die size, device locations, and device sizes be stored on the 9836 computer system. Presently, any die pattern contained within a 20 by 20 square array of die, can be tested, and provisions are provided to test up to 20 total devices per die location. Both of these limits are easily expandable, but they are reasonable maximums considering the memory available on the 9836 for process file and device parameter storage. Details of the prober file organization and use are explained in Appendix 3.

The semi-automatic prober mode of operation was never intended to be used in a fully operating system , but is included as an interim solution to be used when pad layout is not compatible with system requirements. This mode of operation uses a prober file, but stops the program after steps between devices to allow the user to manually switch probe card edge connector cables which are routed to the 4145. In single device prober operation, as opposed to automatic or semi-automatic prober operation, the prober is used for the sole purpose of contacting to a device, and it does not receive any external commands.

necessary for the automatic parameter extraction project, but could find use in a number of other applications at Berkeley. The automatic prober setup, which is required prior to external control by the 9836, involves turning on the prober, initializing the prober I/O system, and aligning a wafer. This is described in Appendix 1, and takes approximately two minutes for an experienced user.

To provide for flexibility in use of the prober the parameter extraction software provides for three different modes of prober operation: **fully automatic, semi-automatic, and single device.** Fully automatic and semi-automatic prober operation are used to test multiple devices for the purpose of generating process files, while single device operation is organized towards model experimentation and analysis. Fully automatic or semi-automatic prober operation requires that a prober file containing such information as die size, device locations, and device sizes be stored on the 9836 computer system. Presently, any die pattern contained within a 20 by 20 square array of die, can be tested, and provisions are provided to test up to 20 total devices per die location. Both of these limits are easily expandable, but they are reasonable maximums considering the memory available on the 9836 for process file and device parameter storage. Details of the prober file organization and use are explained in Appendix 3.

The semi-automatic prober mode of operation was never intended to be used in a fully operating system , but is included as an interim solution to be used when pad layout is not compatible with system requirements. This mode of operation uses a prober file, but stops the program after steps between devices to allow the user to manually switch probe card edge connector cables which are routed to the 4145. In single device prober operation, as opposed to automatic or semi-automatic prober operation, the prober is used for the sole purpose of contacting to a device, and it does not receive any external commands.

3.3. 2671G Thermal Printer

A 2671G thermal graphics printer provides the extraction system user with the capability of obtaining fine quality plots off the screen of the 9836. This is a preferred means of documenting measured or simulated device characteristics. The extraction system permits the user to simultaneously analyze measured and simulated characteristics, and examples of 2671G hard-copy plots for these graphs are shown in the sample execution section. The Berkeley extraction system, also has a Hewlett Packard 82905B dot matrix printer which can be used to obtain hard-copy of text such as parameter lists which appear on the 9836 screen.

3.4. 9836 Computer and VAX Link

The 9836 desktop computer is a Motorola 68000 based machine, and is used as the parameter extraction system controller. The Berkeley 9836 contains approximately 1.4MByte of RAM, which is useful in program development and parameter extraction system execution. Process file storage and manipulation is orders of magnitude faster if done in RAM instead of with floppy disc accesses. The amount of RAM present in a system will limit the number of devices and die which can be tested between file transfers to a VAX mainframe.

The 9836 connection to the network of VAX computers is done by using a Video Terminal Emulator Program called 'VT2' which was written by Hewlett Packard to connect the 9836 to a variety of mainframe computers. This program handles file transfers to and from the 9836, and is used to send process files to the VAX where SPICE simulations are performed.

4. AUTOMATIC MOS DEVICE PARAMETER EXTRACTION SYSTEM SOFTWARE

The parameter extraction software for the 9836 desktop computer is written in Hewlett-Packard Pascal Version 2.0. Pascal was chosen due to the fact that it is faster executing and easier to modify and maintain than is BASIC. These benefits are derived due to the fact that Pascal is compiled as opposed to interpreted, and is modular in form. New versions of BASIC are slowly adapting some of the better features of Pascal, and in the future, BASIC may be a better choice for developing programs such as the one described here. Hewlett-Packard Pascal employs some procedures which are not available in Standard Pascal, but software modifications to use the entire software package on a non-Hewlett-Packard Pascal system would not be extremely difficult.

The software has been divided into six distinct sections:

parameter extraction,

device measurement control,

interactive graphics,

process file development,

automatic prober routines,

and menu and display handling.

The parameter extraction software, with the exception of continuous display updating procedure calls, is written entirely in standard Pascal, and could be transported with very little effort to any Pascal system. The other sections incorporate I/O and graphics modules that have been written by Hewlett-Packard to control its equipment. These modules would have to be simulated in software if another machine and interface setup were to be used.

4. AUTOMATIC MOS DEVICE PARAMETER EXTRACTION SYSTEM SOFTWARE

The parameter extraction software for the 9836 desktop computer is written in Hewlett-Packard Pascal Version 2.0. Pascal was chosen due to the fact that it is faster executing and easier to modify and maintain than is BASIC. These benefits are derived due to the fact that Pascal is compiled as opposed to interpreted, and is modular in form. New versions of BASIC are slowly adapting some of the better features of Pascal, and in the future, BASIC may be a better choice for developing programs such as the one described here. Hewlett-Packard Pascal employs some procedures which are not available in Standard Pascal, but software modifications to use the entire software package on a non-Hewlett-Packard Pascal system would not be extremely difficult.

The software has been divided into six distinct sections:

parameter extraction,

device measurement control,

interactive graphics,

process file development,

automatic prober routines,

and menu and display handling.

The parameter extraction software, with the exception of continuous display updating procedure calls, is written entirely in standard Pascal, and could be transported with very little effort to any Pascal system. The other sections incorporate I/O and graphics modules that have been written by Hewlett-Packard to control its equipment. These modules would have to be simulated in software if another machine and interface setup were to be used.

4.1. Parameter Extraction Routines

For the purpose of parameter extraction, the voltages and currents of a PMOS device are negated to conform to the sign convention of a NMOS transistor. From this point on, all voltage designations will refer to NMOS devices unless otherwise noted. Prior to parameter extraction, a three dimensional array of *Ids* currents is measured, corresponding to six *Vbs*, five *Vgs*, and four *Vds* voltage biases. The *Vbs* values are equally spaced between -*Vdd* and 0V. The Vgs bias values are optimally determined in a threshold voltage estimation routine, and are spaced between a point exceeding the threshold and *Vdd*. Details of the Vgs bias determination and device measurement are described in the Device Measurement Control section. *Vds* biases of 0.1V and 0.2V are chosen for use in the linear region extraction, while biases of *Vdd* -0.5V, and *Vdd* are chosen for the saturation region extraction routines. These bias levels can be changed through very simple program modifications. In the case of Depletion Mode Devices these same biases are used in the saturation extraction routines where provisions have been made to handle devices which may not be operating in the saturation region.

4.1.1. Linear Region Extraction of BetaOx, UOx, and Vtx

The Linear Region Extraction revolves around the general equation for the trans-conductance in the linear region (Equation 12).

$$G = \frac{Ids}{Vds} = \frac{Beta0x(Vgs - Vtx)}{1 + U0x(Vgs - Vtx)} \tag{12}$$

Linear region extraction is performed on all combinations of *Vds* and *Vbs* biases that have been measured, to determine the parameters U0X, Vtx and Beta0x. These parameters are stored in arrays corresponding to the *Vds* and *Vbs* values used. The parameters Beta0x, Vtx and U0x include dependencies upon *Vds* bias

which can be removed once a value for **a** is determined. In keeping with the philosophy of not requiring the user to supply any initial parameter estimates, two routines are used to determine initial estimates for Beta0x, Vtx and U0x. These estimates are then used as inputs to the local optimization routine for the linear region.

Initial estimates for Beta0x and Vtx are determined by approximating **G** as a quadratic in *Vgs* for the smaller values of Vgs that have been measured, and by setting U0x equal to zero. A plot of *G* versus Vgs (Figure 6) shows that the intercept is an excellent approximation to Vtx, and the slope at the intercept is similarly an excellent approximation to Beta0x. This quadratic fit is done using a linear least square algorithm for two variables.

An initial estimate for U0x is calculated using a similar approach on a slightly modified equation. (Equation 12) In this instance U0x and Beta0x are variables, while the value of Vtx is a constant taken from the previous estimation.

$$\frac{Vgs - Vtx}{G} = \frac{1}{Beta0x} + \frac{U0x}{Beta0x(Vgs - Vtx)} \tag{13}$$

The linear least square method is employed to solve for both U0x, and Beta0x, but only the value of U0x is saved for use in the final linear region optimization.

The final values of Beta0x, U0x, and Vtx are determined through the use of Equation 14.

$$f(x) = \frac{G + G^* U0x^* (Vgs - Vth)}{Vgs - Vth} - Beta0x = 0$$

$$f(X) = \Delta Beta0x - \frac{\partial f(x)}{\partial Vtx} \Delta Vtx - \frac{\partial f(x)}{\partial U0x} \Delta U0x \tag{14}$$

This equation is solved at a given *Vds* and *Vbs* bias over the range of *Vgs* values measured. A combination linear least square algorithm with three variables, and Newton-Raphson algorithm is used. The linear least square algorithm is

which can be removed once a value for **a** is determined. In keeping with the philosophy of not requiring the user to supply any initial parameter estimates, two routines are used to determine initial estimates for BetaOx, Vtx and U0x. These estimates are then used as inputs to the local optimization routine for the linear region.

Initial estimates for BetaOx and Vtx are determined by approximating **G** as a quadratic in *Vgs* for the smaller values of Vgs that have been measured, and by setting U0x equal to zero. A plot of *G* versus Vgs (Figure 6) shows that the intercept is an excellent approximation to Vtx, and the slope at the intercept is similarly an excellent approximation to BetaOx. This quadratic fit is done using a linear least square algorithm for two variables.

An initial estimate for U0x is calculated using a similar approach on a slightly modified equation. (Equation 12) In this instance U0x and BetaOx are variables, while the value of Vtx is a constant taken from the previous estimation.

$$\frac{Vgs - Vtx}{G} = \frac{1}{Beta0x} + \frac{U0x}{Beta0x(Vgs - Vtx)} \tag{13}$$

The linear least square method is employed to solve for both U0x, and BetaOx, but only the value of U0x is saved for use in the final linear region optimization.

The final values of BetaOx, U0x, and Vtx are determined through the use of Equation 14.

$$f(x) = \frac{G + G^{*} U0x^{*} (Vgs - Vth)}{Vgs - Vth} - Beta0x = 0$$

$$f(X) = \Delta Beta0x - \frac{\partial f(x)}{\partial Vtx} \Delta Vtx - \frac{\partial f(x)}{\partial U0x} \Delta U0x \tag{14}$$

This equation is solved at a given *Vds* and *Vbs* bias over the range of *Vgs* values measured. A combination linear least square algorithm with three variables, and Newton-Raphson algorithm is used. The linear least square algorithm is

used to determine the next set of partial derivatives for the following iteration of the Newton-Raphson algorithm. At the completion of each iteration by the Newton-Raphson algorithm, a test is made to detect whether a certain level of convergence has been reached. Full convergence has been defined based on accuracy requirements, and when it is reached or if a certain limit on the number of iterations is reached, the iteration is stopped and the estimates are stored in arrays corresponding to the terminal biases. During the iteration process, U0x is not allowed to be forced negative, and if this is occurring, the value of U0x is alternatively divided by ten. In the limiting case, U0x will become 0 which matches the physical situation of no mobility reduction.

4.1.2. Extraction of the Parameter 2Phif

The parameter 2Phif is a direct measure of the effective channel doping level, and its accurate derivation is essential for process analysis. The different modes of program operation utilize various methods to derive this parameter, with the highest degree of accuracy resulting from fully automatic wafer level probing. In the case of single device analysis, it is necessary for the user to specify either 2Phif or the channel doping level **Na**. This value should have been previously determined for a given die using the Fully Automatic or the Manual Modes of Prober operation. Using these modes, the device with the largest area and the largest minimum length or width is chosen to determine 2Phif. If multiple die are tested, further accuracy in measuring 2Phif is derived due to the fact that a better analysis of the parameter K1 has been performed.

The parameter K1 is dependent on 2Phif, temperature, oxide thickness, and device geometry. Equations 15, 16, and 17 represent the expressions for K1, the substrate doping, and the K1 geometry dependence term **s** respectively.

$$K1=s*\sqrt{\frac{2*q*e*Na}{Cox}} \tag{15}$$

$$Na = ni * e^{\frac{2Phif}{2*k*Temp*q}} \quad (16)$$

$$s = 1 + \frac{LK1}{L_{mask} + \Delta L} + \frac{WK1}{W_{mask} + \Delta W} \quad (17)$$

The parameters LK1 and WK1 are the parameters derived when a process file is constructed. It is evident from the expression for **s** that a large Lmask and a large Wmask will result in the expression for **s** being approximately unity. For the first die measured on a test chip, **s** is set to unity, and the largest device available is used to extract 2Phif. The value of 2Phif extracted for the largest device of a given device type on a die, is used as the value of 2Phif for all other devices of the same device type on the same die. For all subsequent die, the geometry dependent terms in the expression for **s** are known, and can be used to derive a more accurate value of 2Phif for the largest device on the die.

The actual extraction of 2Phif involves the solution of the basic CSIM threshold voltage equation (Equation 17) over various *Vbs* biases at a fixed *Vds* value using the Newton-Raphson, Linear Least Square approach.

$$Vtx = \frac{a}{2} * Vds + (VFB - ETA * Vds) + 2Phif + K1 * \sqrt{2Phif - Vbs} - (K2) * (2Phif - Vbs) \quad (18)$$

An analysis of the threshold equation shows that the terms **a** and K1 can be evaluated using the latest iteration value of 2Phif. Initially, 2Phif is set at 0.6V as a reasonable estimate. The terms VFB and ETA*Vds will be constant over all values of Vbs, and therefore they cannot be distinguished from each other. K2 represents further, the effect on the threshold voltage due to drain induced barrier lowering. The solution of the threshold equation results in the parameters (VFB-ETA*Vds), 2Phif, K1, and K2 which are evaluated for $Vds = 0.1V$ and $Vds = 0.2V$.

$$Na = ni * e^{\frac{2Phif}{2*k*Temp*q}} \quad (16)$$

$$s = 1 + \frac{LK1}{L_{mask} + \Delta L} + \frac{WK1}{W_{mask} + \Delta W} \quad (17)$$

The parameters LK1 and WK1 are the parameters derived when a process file is constructed. It is evident from the expression for s that a large Lmask and a large Wmask will result in the expression for s being approximately unity. For the first die measured on a test chip, s is set to unity, and the largest device available is used to extract 2Phif. The value of 2Phif extracted for the largest device of a given device type on a die, is used as the value of 2Phif for all other devices of the same device type on the same die. For all subsequent die, the geometry dependent terms in the expression for s are known, and can be used to derive a more accurate value of 2Phif for the largest device on the die.

The actual extraction of 2Phif involves the solution of the basic CSIM threshold voltage equation (Equation 17) over various *Vbs* biases at a fixed *Vds* value using the Newton-Raphson, Linear Least Square approach.

$$Vtx = \frac{a}{2} * Vds + (VFB - ETA * Vds) + 2Phif + K1 * \sqrt{2Phif - Vbs} - (K2) * (2Phif - Vbs) \quad (18)$$

An analysis of the threshold equation shows that the terms a and K1 can be evaluated using the latest iteration value of 2Phif. Initially, 2Phif is set at 0.6V as a reasonable estimate. The terms VFB and ETA*Vds will be constant over all values of Vbs, and therefore they cannot be distinguished from each other. K2 represents further, the effect on the threshold voltage due to drain induced barrier lowering. The solution of the threshold equation results in the parameters (VFB-ETA*Vds), 2Phif, K1, and K2 which are evaluated for $Vds = 0.1V$ and $Vds = 0.2V$.

4.1.3. General Linear Region Threshold Analysis and Data Reduction

With the exception of the largest device of a given device type on a die, 2Phif is not extracted. In general, VFB-ETA*Vds, K1 and K2 are calculated using a Linear Least Square Fit to the threshold equation (Equation 19) for the range of *Vbs* values measured at a given *Vds* value.

$$Vtx - 2Phif - \frac{a}{2} * Vds = (VFB - ETA * Vds) + K1 * \sqrt{2Phif - Vbs} - K2 * (2Phif - Vbs) \quad (19)$$

Once the parameter K1 is known, it is possible to calculate the parameter **a** for a given *Vbs* bias. Knowing a, the values of U0x and Vtx can be adjusted to remove the threshold dependency on a/2*Vds using Equations (20) and (21).

$$Vth(Vds, Vbs) = Vtx(Vds, Vbs) - \frac{a(Vds, Vbs)}{2} * Vds \quad (20)$$

$$U0(Vds, Vbs) = \frac{U0x(Vds, Vbs)}{1 - U0x(Vds, Vbs) * \frac{a(Vds, Vbs)}{2} * Vds} \quad (21)$$

U0 and **X2U0** (Equation 21) are derived using a linear least square algorithm to model the *Vbs* and *Vds* dependencies of U0. The parameter X3U0 is not saved as a CSIM parameter, but is used to determine the values of U0(Vds,Vbs) in the limit of *Vds=0*.

$$U0(Vds, Vbs) = U0 + X2U0 * Vbs + X3U0 * Vds \quad (22)$$

VFB, 2Phif, K1 and K2 are determined in Equations (23) to (26) using the values for *Vds=0.1V* and *Vds=0.2V* as the inputs to a Linear Least Square algorithm with two variables.

$$VFB(Vds) = VFB + (discarded\ term) * Vds \quad (23)$$

$$2Phif(Vds) = 2Phif + (discarded\ term) * Vds \quad (24)$$

$$K1(Vds) = K1 + (discarded\ term) * Vds \quad (25)$$

$$K2(Vds) = K2 + (discarded\ term) * Vds \quad (26)$$

In summary, the threshold parameters are determined by requiring that the term **a** be explicitly included in the linear region parameters, allowing those parameters to be extrapolated to $Vds=0$. This allows the parameters VFB, K1, K2, 2Phif U0 and X2U0 to be extracted without the use of any approximation relating to measurements employing finite values of Vds.

4.1.4. Saturation Region Extraction of Beta0sat, Vthsat and U1sat

The Ids currents measured corresponding to $Vds=Vdd$ and $Vds=Vdd\text{-}0.5V$ are used to determine Beta0sat, Vthsat and U1sat as functions of Vds and Vbs. A combination Newton-Raphson algorithm, Linear Least Square algorithm is used to solve Equation (27) at each Vds and Vbs bias.

$$Ids_{meas} - Ids_{sim} = \frac{\partial Ids_{sim}}{\partial Beta0sat} * \Delta Beta0sat + \frac{\partial Ids_{sim}}{\partial Vthsat} * \Delta Vthsat + \frac{\partial Ids_{sim}}{\partial U1sat} * \Delta U1sat \quad (27)$$

The initial estimate for Beta0sat is set to be the value of Beta0 at the same Vbs bias but with $Vds=0.2V$ as was determined in the Linear Region Extraction. Vthsat can be determined initially from the values of VFB, Phif2, K1, and K2 that are known from the Linear Region Extraction, while U1 is initially approximated as zero.

As previously mentioned, depletion mode MOS devices may not be operating in the saturation region, and this required program modification. It has to be determined whether a device is in saturation or linear operation and this is done by evaluating Equation (3) and comparing $Vdssat$ to the Vds bias. Analytical expressions for the derivatives in Equation (27), and the equation for Ids are selected based on saturation or linear region operation.

If the value of U1 is driven negative while the optimization is being performed, the value is divided by 10 and the optimization continues. In the limit, this reflects the case where there is no velocity saturation. If U1 is equal to zero

In summary, the threshold parameters are determined by requiring that the term **a** be explicitly included in the linear region parameters, allowing those parameters to be extrapolated to $Vds=0$. This allows the parameters VFB, K1, K2, 2Phif U0 and X2U0 to be extracted without the use of any approximation relating to measurements employing finite values of Vds.

4.1.4. Saturation Region Extraction of Beta0sat, Vthsat and U1sat

The Ids currents measured corresponding to $Vds=Vdd$ and $Vds=Vdd\text{-}0.5V$ are used to determine Beta0sat, Vthsat and U1sat as functions of Vds and Vbs. A combination Newton-Raphson algorithm, Linear Least Square algorithm is used to solve Equation (27) at each Vds and Vbs bias.

$$Ids_{meas} - Ids_{sim} = \frac{\partial Ids_{sim}}{\partial Beta0sat} * \Delta Beta0sat + \frac{\partial Ids_{sim}}{\partial Vthsat} * \Delta Vthsat + \frac{\partial Ids_{sim}}{\partial U1sat} * \Delta U1sat \quad (27)$$

The initial estimate for Beta0sat is set to be the value of Beta0 at the same Vbs bias but with $Vds=0.2V$ as was determined in the Linear Region Extraction. Vthsat can be determined initially from the values of VFB, Phif2, K1, and K2 that are known from the Linear Region Extraction, while U1 is initially approximated as zero.

As previously mentioned, depletion mode MOS devices may not be operating in the saturation region, and this required program modification. It has to be determined whether a device is in saturation or linear operation and this is done by evaluating Equation (3) and comparing $Vdssat$ to the Vds bias. Analytical expressions for the derivatives in Equation (27), and the equation for Ids are selected based on saturation or linear region operation.

If the value of U1 is driven negative while the optimization is being performed, the value is divided by 10 and the optimization continues. In the limit, this reflects the case where there is no velocity saturation. If U1 is equal to zero

when the saturation region extraction converges, a special routine to handle this condition is executed. This is necessary, because the values of Beta0sat and Vthsat may not be optimal, and is done by simplifying Equation (27) through eliminating the U1 terms and U1 derivatives. Equation (27) is then resolved with a Newton-Raphson/Linear Least Square algorithm to determine more optimal values for Beta0sat and Vthsat. Beta0sat, Vthsat, and U1sat are all determined over the range of *Vbs* biases, and the saturation region *Vds* biases. The parameter Vthsat is used to determine the entries for the array ETA[Vds,Vbs] in the saturation region, through the use of Equation (28).

$$ETA[Vds,Vbs]=\frac{(VFB+2Phif+K1*\sqrt{2Phif-Vbs}-K2*(2Phif-Vbs))-Vthsat}{Vds} \quad (28)$$

4.1.5. Saturation Region Data Reduction

For parameters such as Beta0sat, U1 and ETA which are of much greater significance for *Vds* values in the proximity of *Vdd*, it is preferred to extract the parameters *Vds* dependence to *Vdd* as *Vds=0V*. The U1 data array is modeled by this means (Equation 29) and it has been experimentally determined that the Vds dependency can be sufficient enough to warrant the inclusion of the parameter X3U1 in the CSIM model.

$$U1[Vds,Vbs]=U1+X2U1*Vbs+X3U1*(Vdd-Vds) \quad (29)$$

The threshold drain induced barrier lowering term, ETA, is modeled with both *Vds* and *(Phif2-Vbs)* dependency. Since this term is insignificant at low *Vds* biases, only the saturation region extracted values for ETA[Vds,Vbs] are used to to determine ETA, X2ETA and X3ETA as is shown in Equation (30).

$$ETA[Vds,Vbs]=ETA+X2ETA*(Phif2-Vbs)+X3ETA*(Vds-Vdd) \quad (30)$$

Figure 6

CSIM PARAMETER EXTRACTION FLOWCHART

LINEAR REGION ANALYSIS

DETERMINE USING OPTIMIZATION
BETAφX(V_{ds}, V_{bs})
VTX (V_{ds}, V_{bs}) AND
UφX (V_{ds}, V_{bs})
OVER LINEAR REGION V_{ds} VALUES
AND ALL V_{bs} VALUES

↓

EXTRACT: PHIF2 (V_{ds}), KI (V_{ds})
VFB (V_{ds}), AND K2 (V_{ds})
FROM VTX (V_{BS}) AT
CONSTANT V_{ds} VALUES

↓

REDUCE ALL ARRAYS TO V_{ds} = 0
VOLT VALUES

EXTRACT PHIF2, KI, VFB, K2, Uφ,
AND X2Uφ

↓

SATURATION REGION ANALYSIS

DETERMINE USING OPTIMIZATION
BETAφSAT (V_{ds}, V_{bs})
VTHSAT (V_{ds}, V_{bs})
UISAT (V_{ds}, V_{bs})
OVER SATURATION REGION V_{ds}
VALUES AND ALL VBS VALUES

↓

DEVELOP ETA ARRAY FOR
SATURATION VALUES MODEL
BETAφSAT, ETA AND UI AT V_{ds} = V_{dd}

↓

INCLUDE EFFECT OF UI ON BETAφ

MODEL BETA AS A PARABOLA
VERSUS V_{DS}

↓

END OF EXTRACTION

Figure 6

CSIM PARAMETER EXTRACTION FLOWCHART

LINEAR REGION ANALYSIS

DETERMINE USING OPTIMIZATION
BETAφX(V_{ds}, V_{bs})
VTX (V_{ds}, V_{bs}) AND
UφX (V_{ds}, V_{bs})
OVER LINEAR REGION V_{ds} VALUES
AND ALL V_{bs} VALUES

EXTRACT: PHIF2 (V_{ds}), KI (V_{ds})
VFB (V_{ds}), AND K2 (V_{ds})
FROM VTX (V_{BS}) AT
CONSTANT V_{ds} VALUES

REDUCE ALL ARRAYS TO $V_{ds} = 0$
VOLT VALUES

EXTRACT PHIF2, KI, VFB, K2, Uφ,
AND X2Uφ

SATURATION REGION ANALYSIS

DETERMINE USING OPTIMIZATION
BETAφSAT (V_{ds}, V_{bs})
VTHSAT (V_{ds}, V_{bs})
UISAT (V_{ds}, V_{bs})
OVER SATURATION REGION V_{ds}
VALUES AND ALL VBS VALUES

DEVELOP ETA ARRAY FOR
SATURATION VALUES MODEL
BETAφSAT, ETA AND UI AT $V_{ds} = V_{dd}$

INCLUDE EFFECT OF UI ON BETAφ

MODEL BETA AS A PARABOLA
VERSUS V_{DS}

END OF EXTRACTION

A final data manipulation for both the array of Beta0[Vds,Vbs] and Beta0sat[Vds,Vbs] is necessary to remove the effects of U1 on these parameters as they were extracted.

$$Beta[Vds,Vbs]=Beta[Vds,Vbs]*(1+(U1+X2U1*Vbs)*Vds) \qquad (31)$$

Once this is complete, the Beta0 array is modeled as in Equation (32) and the Beta0sat array is modeled as in Equation (33).

$$Beta0[Vds,Vbs]=BETA0SAT+X2BETA0*Vbs+(discarded\ term)*Vds \qquad (32)$$

$$Beta0sat[Vds,Vbs]=BETA0SAT+X2BETA0SAT*Vbs+X3BETA0SAT*(Vds-Vdd) \qquad (33)$$

4.2. Device Measurement Software

For the purpose of automatic parameter extraction, it is important to have the capability to detect those devices which are affected by any one of a variety of error conditions. Such a procedure can save valuable testing time, and helps to eliminate the possibility that parameter extraction will be attempted on devices exhibiting anomalous electrical behavior. The program described here employs the user specified supply voltage for the purpose of setting up the 4145 to obtain an accurate analysis of a device over its expected operating range. Prior to parameter extraction, independent 4145 measurement setups are used to: check for error conditions, determine device functionality, and establish the measurement data base necessary for the extraction software.

4.2.1. Device Type Measurement

The initial 4145 measurement setup analyzes the source-body and drain-body junctions, and checks for gate to terminal shorts (TABLE 1).

TABLE 1 Device Type Measurement Setup	
Terminal	*Voltage*
Drain	0.0V
Source	0.0V
Gate	Vdd
Body	-Vdd and Vdd

In the process of verifying that the source-body and drain-body junctions are present and are not shorted, the program uses the polarity of the junctions to determine whether a NMOS or a PMOS device is present. By setting the gate potential at a distant level from the potentials of the source, drain or bulk, and by measuring the gate current, a short to one of the other terminals can be detected. The program definitions for tolerable junction and gate leakage currents (TABLE 2) are not dependent on device geometry. They have been set for the purpose of detecting obvious shorts and are presently not intended for screening junction or gate leakage currents. In the case of either gate or junction leakage currents, the noise affecting the measurement setup may be significantly greater than the actual leakage current, depending upon the type of cables and connections used.

TABLE 1 Device Type Measurement Setup

Terminal	*Voltage*
Drain	0.0V
Source	0.0V
Gate	Vdd
Body	-Vdd and Vdd

In the process of verifying that the source-body and drain-body junctions are present and are not shorted, the program uses the polarity of the junctions to determine whether a NMOS or a PMOS device is present. By setting the gate potential at a distant level from the potentials of the source, drain or bulk, and by measuring the gate current, a short to one of the other terminals can be detected. The program definitions for tolerable junction and gate leakage currents (TABLE 2) are not dependent on device geometry. They have been set for the purpose of detecting obvious shorts and are presently not intended for screening junction or gate leakage currents. In the case of either gate or junction leakage currents, the noise affecting the measurement setup may be significantly greater than the actual leakage current, depending upon the type of cables and connections used.

TABLE 2 Device Type Measurement Results

Condition	Message
Igate>0.1uA	ERROR **Gate Short**
abs(Ibody)>10uA for both Vbs=Vdd and Vbs=-Vdd	ERROR **Shorted Junction**
abs(Ibody)<10uA for both Vbs=Vdd and Vbs=-Vdd	ERROR **No Junction**
None of the above errors, and Ibody>10uA for Vbs=Vdd	DEVICE=NCHANNEL
None of the above errors, and Ibody<-10uA for Vbs=-Vdd	DEVICE=PCHANNEL

4.2.2. Device Functionality Verification

If no error conditions are detected in the *Device Type Measurement* section, the **device type** is used in setting up the 4145 to determine whether the device is functional. The functionality test biases the device at the point in its operating region where maximum drain-source current is expected, and determines whether the device is actually operating. *Vgs* is then set to 0V to determine whether approximately the same drain source current flows, which would indicate that the drain and source are shorted together. Table 3 shows the device functionality measurement setup, while Table 4 lists the error conditions that may arise.

Table 3 Device Functionality Measurement Setup

Bias	*NMOS*	*PMOS*
Vds	5V	-5V
Vgs	0V and 5V	0V and -5V
Vbs	0V	0V

Table 4 Device Functionality Measurement Results

Condition	*Message*
abs(Ids@Vgs=0V)>0.95*abs(Ids@abs(Vgs)=5V)	**Drain Source Short**
abs(Ids@abs(Vgs)=5V)<Wmask/Lmask*10uA	**Drain Source Open**
None of the above errors	(No Message)

4.2.3. Device Measurement for Parameter Extraction

The parameter extraction approach requires *Ids* measurements at: six equally spaced *Vbs* biases between 0V and *-Vdd*, five equally spaced *Vgs* biases between the threshold and *Vdd*, and *Vds* biases at 0.1V, 0.2V, *Vdd-0.5V*, and *Vdd*. The setup of the 4145 to perform a measurement requires a few seconds, and for speed performance reasons, it was essential to minimize the number of times that the 4145 needs to be set up with new bias conditions. The *Vds* voltage steps are non linear, and this led to the use of a 4145 setup for each of the four Vds

Table 3 Device Functionality Measurement Setup		
Bias	*NMOS*	*PMOS*
Vds	5V	-5V
Vgs	0V and 5V	0V and -5V
Vbs	0V	0V

Table 4 Device Functionality Measurement Results	
Condition	*Message*
abs(Ids@Vgs=0V)>0.95*abs(Ids@abs(Vgs)=5V)	**Drain Source Short**
abs(Ids@abs(Vgs)=5V)<Wmask/Lmask*10uA	**Drain Source Open**
None of the above errors	(No Message)

4.2.3. Device Measurement for Parameter Extraction

The parameter extraction approach requires *Ids* measurements at: six equally spaced *Vbs* biases between 0V and *-Vdd*, five equally spaced *Vgs* biases between the threshold and *Vdd*, and *Vds* biases at 0.1V, 0.2V, *Vdd-0.5V*, and *Vdd*. The setup of the 4145 to perform a measurement requires a few seconds, and for speed performance reasons, it was essential to minimize the number of times that the 4145 needs to be set up with new bias conditions. The *Vds* voltage steps are non linear, and this led to the use of a 4145 setup for each of the four Vds

biases to be used.

For a given *Vds* bias, the program sets up the 4145 to measure a set of *Ids* versus *Vgs* characteristic curves, with Z-axis steps corresponding to the *Vbs* biases required by the program. Depending upon the IC process, and the *Vbs* and *Vds* bias, the threshold voltage for any functioning MOS device could occur anywhere within or below the operating range of 0V to *Vdd*. The parameter extraction strategy requires that the *Vgs* biases used correspond to device operation in the linear or saturation region, but not in the subthreshold region. In the case of depletion mode devices, device measurements will correspond to the region of device operation, and the actual threshold region will not be directly measured. The extraction program will model the threshold voltage to best fit the measured data over the actual region of device operation.

To select values of *Vgs* to guarantee that the device under test is not operating in the subthreshold region, a large set of *Ids* currents are measured over *Vgs* bias range of 0V to VddV. An approximate threshold voltage can be determined, for each *Vds* and *Vbs* bias, and the data points needed for parameter extraction can then be selected from the larger set. The approximate threshold current level used in the selection process is a function of device geometry (Equation 34).

$$Ids_{threshold} = \frac{W_{mask}}{L_{mask}} * 0.1uA \tag{34}$$

The first *Vgs* bias selected is chosen at 0.5V above the threshold voltage to guarantee that all of the points measured at $Vds=0.1V$ and $Vds=0.2V$ correspond to device operation in the linear region.

4.3. Process File Development Software

Immediately after the 17 CSIM parameters have been calculated for a MOS device, they are stored in a file which will later be used to create a Process File for the given die location. When the testing of all devices on a die is complete, Process Files are created for every device type analyzed on the die. In addition to the 54 CSIM Process File Parameters described earlier, each Process File contains statistical information which is sent to the VAX for future analysis.

The 17 CSIM parameters, and the mask Length and Width for a specific device, are stored in a temporary file which contains the CSIM parameters for all devices of a given device type on a die. These files are purged when the CSIM extraction program is exited by the user, and are erased when testing on a new die commences. If the program is abnormally terminated the files for device parameters will remain on the directory that has been prefixed by the user. The Temporary files are **NEDFILE**, **NZDFILE**, **NDDFILE**, **PEDFILE**, **PZDFILE**, and **PDDFILE** and they correspond to NMOS or PMOS, enhancement, zero-threshold and depletion mode devices. A file named **TEMP** is also used in the Process File Development. **If a file by any of these names exists on the prefixed directory, it will be written over during the parameter extraction system software execution.**

When testing of a die is complete, any Temporary parameter storage file containing data is used to develop a Process File. If various width and length devices have been measured, the parameter **BETA0** is analyzed versus Mask Width and Length to determine DELTAL, DELTAW and the low field mobility. If only one device of a given type has been used, the process file contains the 17 CSIM parameters, the supply voltage, the temperature, and the oxide thickness. A Process File representing a single device can be read and used by the SPICE 2G.6 version at Berkeley to simulate a device with the exact dimensions as the

4.3. Process File Development Software

Immediately after the 17 CSIM parameters have been calculated for a MOS device, they are stored in a file which will later be used to create a Process File for the given die location. When the testing of all devices on a die is complete, Process Files are created for every device type analyzed on the die. In addition to the 54 CSIM Process File Parameters described earlier, each Process File contains statistical information which is sent to the VAX for future analysis.

The 17 CSIM parameters, and the mask Length and Width for a specific device, are stored in a temporary file which contains the CSIM parameters for all devices of a given device type on a die. These files are purged when the CSIM extraction program is exited by the user, and are erased when testing on a new die commences. If the program is abnormally terminated the files for device parameters will remain on the directory that has been prefixed by the user. The Temporary files are **NEDFILE**, **NZDFILE**, **NDDFILE**, **PEDFILE**, **PZDFILE**, and **PDDFILE** and they correspond to NMOS or PMOS, enhancement, zero-threshold and depletion mode devices. A file named **TEMP** is also used in the Process File Development. **If a file by any of these names exists on the prefixed directory, it will be written over during the parameter extraction system software execution.**

When testing of a die is complete, any Temporary parameter storage file containing data is used to develop a Process File. If various width and length devices have been measured, the parameter **BETA0** is analyzed versus Mask Width and Length to determine DELTAL, DELTAW and the low field mobility. If only one device of a given type has been used, the process file contains the 17 CSIM parameters, the supply voltage, the temperature, and the oxide thickness. A Process File representing a single device can be read and used by the SPICE 2G.6 version at Berkeley to simulate a device with the exact dimensions as the

device which was measured. This method of operation could be used if extreme model accuracy is needed, or if only a single size of device will be used.

When using devices of various sizes, to determine parameter length and width variations, the extraction software determines whether multiple mask widths and/or multiple mask lengths are present in the group of devices. As an example, if devices of varying lengths but the same width have been used, DELTAW, and all of the process parameters reflecting parameter variation with Width will be set to zero.

When parameter variation with respect to electrical channel length and channel width is modeled as in Equation 11, the given parameter is resimulated with the three process parameters for each device size that was measured. The resimulated parameter values for each device size are compared to the original extracted parameter values, and the maximum parameter re-simulation error is reported. For each set of three process parameters, representing a single model parameter, a fourth, fifth and sixth parameter are added to the process file for statistical analysis. The fourth parameter is the maximum percent error encountered in resimulating the CSIM parameter from the three process parameters over the range of device sizes analyzed. The fifth and sixth parameters are the width and length of the device which experienced the error reported by the fourth parameter. A sample Process File is described in the *Program Execution* section.

4.4. Interactive Graphics Software

When the user operates the extraction software in the single device mode, an interactive graphics subprogram is entered after the CSIM device parameters have been extracted and stored in a process file. The graphics program is fully self-contained, and includes 4145 device measurement capability which is

independent of the measurement capability used prior to parameter extraction. This program was designed to allow the user to verify that the extracted CSIM parameters simulate the actual device electrical characteristics. Special graphics features were added to provide the user with zoom and redraw capabilities to enhance potential graphics documentation. The graphics program is controlled by the user through interaction with a graphics menu which has three sections: measured and/or simulated plot selection, graph type selection, and the zoom/redraw selection.

The graphics program allows the user to plot either measured device data or simulated device data, or to superimpose both measured and simulated data on a graph. The graphs which are presently included are for *Ids* versus *Vds* stepping *Vgs*, *Ids* versus *Vgs* stepping *Vbs*, and *ln(Ids)* versus *Vds* stepping *Vgs*. The *ln(Ids)* versus *Vds* curves are experimental, and will be transformed in later work into curves to display subthreshold behavior. The three possible sets of curves allow the user to define arbitrarily the value of the undefined terminal to source potential. Given this user flexibility, it is necessary to measure and simulate the data to be graphed after the user completes the definition of the graph.

The graphics software automatically scales the graph axis so that the data to be plotted entirely fills the screen. The entire set of data is used to generate a RMS error between all measured data points and those that are simulated with the CSIM model. The error is plotted on the full scale graph, and is not recalculated when a subsection of the graph is analyzed. The bias that is coarsely stepped is referred to as the z-axis value, and is plotted on the right axis of the graph at a point coinciding with where the curve exits the page. X,Y and Z axes labels are generated automatically, and account for the exponents of the values being plotted. The biases at which measured and simulated data points are

independent of the measurement capability used prior to parameter extraction. This program was designed to allow the user to verify that the extracted CSIM parameters simulate the actual device electrical characteristics. Special graphics features were added to provide the user with zoom and redraw capabilities to enhance potential graphics documentation. The graphics program is controlled by the user through interaction with a graphics menu which has three sections: measured and/or simulated plot selection, graph type selection, and the zoom/redraw selection.

The graphics program allows the user to plot either measured device data or simulated device data, or to superimpose both measured and simulated data on a graph. The graphs which are presently included are for *Ids* versus *Vds* stepping *Vgs*, *Ids* versus *Vgs* stepping *Vbs*, and *ln(Ids)* versus *Vds* stepping *Vgs*. The *ln(Ids)* versus *Vds* curves are experimental, and will be transformed in later work into curves to display subthreshold behavior. The three possible sets of curves allow the user to define arbitrarily the value of the undefined terminal to source potential. Given this user flexibility, it is necessary to measure and simulate the data to be graphed after the user completes the definition of the graph.

The graphics software automatically scales the graph axis so that the data to be plotted entirely fills the screen. The entire set of data is used to generate a RMS error between all measured data points and those that are simulated with the CSIM model. The error is plotted on the full scale graph, and is not recalculated when a subsection of the graph is analyzed. The bias that is coarsely stepped is referred to as the z-axis value, and is plotted on the right axis of the graph at a point coinciding with where the curve exits the page. X,Y and Z axes labels are generated automatically, and account for the exponents of the values being plotted. The biases at which measured and simulated data points are

taken depend upon the supply voltage. For a 5 Volt process, X axis data points are taken every 0.1V from 0V to 5V, while Z axis data points are taken every 1V.

A zoom capability is provided to allow closer inspection of any section of the graph. Zooming is done by drawing a box around the selected region with a cursor, and it can be repeated as often as liked. It is especially useful for studying the transition from subthreshold to threshold device operation. After zooming in on a section of the graph, the user has the opportunity to redraw the original full scale graph which covers the entire operating range of the device. Other options include selecting a new graph, or displaying the same graph with a different bias condition. Examples of actual graphs and the graphics menu are provided in the *Program Execution* section.

4.5. Automatic Prober Software

The Xynetics 2001X prober can be fully controlled by the 9836 computer once the prober is set up and initialized (see Appendix 1 for setup information). A simple set of prober procedures has been written to move from device to device, set the die size, step from die to die, initialize the prober units to microns, and unload the wafer. The set of prober control procedures that has been written, are modular, and should be easily modifiable if another prober is to be controlled. To provide for generality, all prober information is stored in a prober file which must be developed by the user prior to use of the Automatic or Manual Probing Modes of operation. The Manual Probing Mode requires all of the information in the prober file, and provides for prober movement, but stops to allow the user to switch the connections from the 4145 to the device under test.

A prober file contains all of the prober stepping, device connection, device type, and device geometry information needed by the system. Appendix 4 describes the construction of a prober file. The top two elements are the device

size in microns, and they are followed by a 20 by 20 array which describes the die to be tested. Initially, the 20 by 20 array is filled with 0 values, and then 1's are inserted at every die location to be tested. The die to which the prober is initially aligned, is marked with an X. Die are tested from top to bottom, left to right.

Within a specific die, the prober must know the positions of the devices to be tested, relative to the origin location on the die. The origin location is where the probes are initially set when the program starts. All device positions are in microns relative to the origin location with the positive x-axis being toward the right, and the positive y-axis being toward the bottom of the wafer. The connection of the device terminals to the individual Stimulus Measurement Units must be known for every device. This allows for generality in device pad orientation, and for more efficient use of pads. See Appendix 2 for details on pad layout. The mask length and width for each device to be tested must also be included in the prober file.

To allow the user to determine in which process file a device will be included, the user must specify whether the device is enhancement, depletion or zero threshold, and whether it is NMOS or PMOS. This avoids problems such as having the program attempt to distinguish between a zero-threshold device and an enhancement mode device. The device type is necessary so that the program can determine the best devices to be used for extracting Phif2.

It is preferable to use the largest device of any given device type to measure Phif2. After the prober file has been read by the extraction program, it is possible to switch the order of the devices to be probed. Within the automatic prober routines, is a procedure which will scan the device geometries for every device type present, and will adjust the prober file so that the best device for measuring Phif2 is probed before any other device of the same device type. This

size in microns, and they are followed by a 20 by 20 array which describes the die to be tested. Initially, the 20 by 20 array is filled with 0 values, and then 1's are inserted at every die location to be tested. The die to which the prober is initially aligned, is marked with an X. Die are tested from top to bottom, left to right.

Within a specific die, the prober must know the positions of the devices to be tested, relative to the origin location on the die. The origin location is where the probes are initially set when the program starts. All device positions are in microns relative to the origin location with the positive x-axis being toward the right, and the positive y-axis being toward the bottom of the wafer. The connection of the device terminals to the individual Stimulus Measurement Units must be known for every device. This allows for generality in device pad orientation, and for more efficient use of pads. See Appendix 2 for details on pad layout. The mask length and width for each device to be tested must also be included in the prober file.

To allow the user to determine in which process file a device will be included, the user must specify whether the device is enhancement, depletion or zero threshold, and whether it is NMOS or PMOS. This avoids problems such as having the program attempt to distinguish between a zero-threshold device and an enhancement mode device. The device type is necessary so that the program can determine the best devices to be used for extracting Phif2.

It is preferable to use the largest device of any given device type to measure Phif2. After the prober file has been read by the extraction program, it is possible to switch the order of the devices to be probed. Within the automatic prober routines, is a procedure which will scan the device geometries for every device type present, and will adjust the prober file so that the best device for measuring Phif2 is probed before any other device of the same device type. This

is done by interchanging all of the information for the best device with the information for the first device, for each specific device type that occurs in the prober file.

5. SAMPLE PROGRAM EXECUTION

The automatic extraction program is initialized by selecting the type of program operation desired, and by providing a list of necessary input values. Once the program setup is completed, the device measurement and parameter extraction are initiated, and a continuous display informs the user as to the latest status of the entire program. This extraction display allows the user to analyze the latest set of device parameters as they are extracted, and provides information as to the length of time remaining for program completion. In the *Single Device Mode* of operation, the program proceeds to the interactive graphics package after the parameter extraction and process file development is completed.

5.1. Initial Program Mode Selection

The *Initial Program Display* (Figure 7) introduces the program and allows the user to select a mode of prober operation or to exit the program. Prior to selecting the Automatic or Semi Automatic modes of operation, the user is required to have constructed a prober file. (See Appendix 3 for Prober File construction guidelines) Using these modes of operation, the program will return to the initial program display after all devices have been tested, and all process files have been created. The single device mode of operation can be accessed without any prior user action, and returns to the initial program display after the graphics subprogram is exited. The selection to exit the program results in the purging of all of the temporary files constructed for the purpose of developing process files.

5. SAMPLE PROGRAM EXECUTION

The automatic extraction program is initialized by selecting the type of program operation desired, and by providing a list of necessary input values. Once the program setup is completed, the device measurement and parameter extraction are initiated, and a continuous display informs the user as to the latest status of the entire program. This extraction display allows the user to analyze the latest set of device parameters as they are extracted, and provides information as to the length of time remaining for program completion. In the *Single Device Mode* of operation, the program proceeds to the interactive graphics package after the parameter extraction and process file development is completed.

5.1. Initial Program Mode Selection

The *Initial Program Display* (Figure 7) introduces the program and allows the user to select a mode of prober operation or to exit the program. Prior to selecting the Automatic or Semi Automatic modes of operation, the user is required to have constructed a prober file. (See Appendix 3 for Prober File construction guidelines) Using these modes of operation, the program will return to the initial program display after all devices have been tested, and all process files have been created. The single device mode of operation can be accessed without any prior user action, and returns to the initial program display after the graphics subprogram is exited. The selection to exit the program results in the purging of all of the temporary files constructed for the purpose of developing process files.

FIGURE 7 INITIAL PROGRAM DISPLAY

CSIM AUTOMATIC MOS DEVICE CHARACTERIZATION PROGRAM

UC BERKELEY FALL 1983 VERSION

This Program can be used in any of the following modes:
Fully Automatic, Semi Automatic and Single Device Prober Operation.

FULLY AUTOMATIC PROBER OPERATION requires the user to input the name of a prober file, and tests all devices in the file without interuption.

SEMI AUTOMATIC PROBER OPERATION requires the user to input the name of a prober file, and automatically moves to each device in the file. This mode of operation stops at each device to allow the user to switch connections.

SINGLE DEVICE OPERATION allows the user to analyze an individual device, extract CSIM parameters, and compare simulated versus measured data.

Select a Mode of Operation>
1:FULLY AUTOMATIC
2:SEMI AUTOMATIC
3:SINGLE DEVICE
4:EXIT CSIM

5.2. User Input Lists

Depending upon the mode of operation selected, one of two possible *Data Entry Input Lists* will be displayed. Figure 8 shows the list for Automatic and Semi Automatic operation, while Figure 9 shows the Single Device input list. The Process Name, Lot Number, Wafer Number, Date, and Operator are common to both lists, and are used purely for statistical purposes. These values can be defaulted to blanks by using the 'Enter' key. The Single Device Mode of operation also prompts for the X and Y position of the die on which the device under test is located. These two values will default to zero using the 'Enter' key, and have no influence on the parameter extraction program.

The Output File, Supply Voltage, Temperature and Oxide Thickness are all necessary for proper program execution, and are included in both input lists. The program automatically appends *.TEXT* to the Output File name, **and it is highly recommended that the system be prefixed to RAM so that all file manipulation, including the construction of the Output File can be done efficiently without unnecessary disc access delay.** If multiple die are tested, the Output File will contain all of the process files that have been created for each die location. This is set up so that the single Output File can be sent to the VAX Mainframe, where it can be divided into individual process files. Each Process File within the Output File contains its own individual statistical information, so that breaking the Output File into process files does not require any additional processing.

The Automatic and Semi Automatic modes of operation require that the name of the Prober File be entered. The program automatically appends *.TEXT* to the name of the prober file. It must also be noted that file names should not exceed eight characters. After the probes are aligned to the origin location on the wafer, according to the process file description, the user presses a key to

5.2. User Input Lists

Depending upon the mode of operation selected, one of two possible *Data Entry Input Lists* will be displayed. Figure 8 shows the list for Automatic and Semi Automatic operation, while Figure 9 shows the Single Device input list. The Process Name, Lot Number, Wafer Number, Date, and Operator are common to both lists, and are used purely for statistical purposes. These values can be defaulted to blanks by using the 'Enter' key. The Single Device Mode of operation also prompts for the X and Y position of the die on which the device under test is located. These two values will default to zero using the 'Enter' key, and have no influence on the parameter extraction program.

The Output File, Supply Voltage, Temperature and Oxide Thickness are all necessary for proper program execution, and are included in both input lists. The program automatically appends *.TEXT* to the Output File name, **and it is highly recommended that the system be prefixed to RAM so that all file manipulation, including the construction of the Output File can be done efficiently without unnecessary disc access delay.** If multiple die are tested, the Output File will contain all of the process files that have been created for each die location. This is set up so that the single Output File can be sent to the VAX Mainframe, where it can be divided into individual process files. Each Process File within the Output File contains its own individual statistical information, so that breaking the Output File into process files does not require any additional processing.

The Automatic and Semi Automatic modes of operation require that the name of the Prober File be entered. The program automatically appends *.TEXT* to the name of the prober file. It must also be noted that file names should not exceed eight characters. After the probes are aligned to the origin location on the wafer, according to the process file description, the user presses a key to

FIGURE 8 AUTOMATIC OR SEMI AUTOMATIC INPUT DISPLAY

AUTOMATIC OR SEMI-AUTOMATIC OPERATION

Process Name=? >
Lot=? >
Wafer=? >
Date=? >
Operator=? >
Output File=? >
VDD(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)
Press a "C" for changes, or press any other key to start. >

FIGURE 9 SINGLE DEVICE INPUT DISPLAY

SINGLE DEVICE OPERATION

Process Name=? >
Lot=? >
Wafer=? > XPOSITION=? > YPOSITION=? >
Date=? >
Operator=? >
Output File=? >
VDD(volts)=? >
TEMPERATURE(deg. C)=? >
TOX(angstroms)=? >
PHIF2 or NSUB=? >
drawn width (microns)=? >
drawn length (microns)=? >

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

Press a "C" for changes or press any other key to start. >

start program execution. At this point, the prober file will be read, and there will be a temporary delay before the extraction status display is viewed.

The Single Device mode of operation requires the user to input those pieces of information which are generally contained in the prober file. The dimensions of the device under test, in microns, are needed in some of the measurement routines for scaling, but are not used by the program for any parameter extraction when using the single device mode. The length and width cannot be defaulted to zero. The connection between the 4145 SMUs and the device terminals must be described so that the measurement can be properly set up. Finally, the single device mode of operation requires that a value for Phif2 or Nsub be input by the user. This value cannot be defaulted to zero, and the program automatically determines whether the input value is Phif2 or Nsub.

5.3. Continuous Extraction Display

The *Continuous Extraction Display* (Figure 10) is displayed throughout the measurement and extraction sections of the program and is periodically updated to reflect changes in status or to return new values for extracted parameters. The top section of this display contains all of the statistical information which was input in the Data Entry section, and contains the position of the die under test. The display will be updated to show NCHANNEL or PCHANNEL once the device type has been determined by the program.

The center section of this display contains two timers and an extraction location meter. The timers are clocked at critical locations in the program, but are not perfectly linear. If a device is determined to be non-functional, the time remaining will instantly decrease to reflect the fact that a device will not be fully analyzed. The time remaining on the timers assumes that each device to be tested will be functional, and will require approximately two and one-half

start program execution. At this point, the prober file will be read, and there will be a temporary delay before the extraction status display is viewed.

The Single Device mode of operation requires the user to input those pieces of information which are generally contained in the prober file. The dimensions of the device under test, in microns, are needed in some of the measurement routines for scaling, but are not used by the program for any parameter extraction when using the single device mode. The length and width cannot be defaulted to zero. The connection between the 4145 SMUs and the device terminals must be described so that the measurement can be properly set up. Finally, the single device mode of operation requires that a value for Phif2 or Nsub be input by the user. This value cannot be defaulted to zero, and the program automatically determines whether the input value is Phif2 or Nsub.

5.3. Continuous Extraction Display

The *Continuous Extraction Display* (Figure 10) is displayed throughout the measurement and extraction sections of the program and is periodically updated to reflect changes in status or to return new values for extracted parameters. The top section of this display contains all of the statistical information which was input in the Data Entry section, and contains the position of the die under test. The display will be updated to show NCHANNEL or PCHANNEL once the device type has been determined by the program.

The center section of this display contains two timers and an extraction location meter. The timers are clocked at critical locations in the program, but are not perfectly linear. If a device is determined to be non-functional, the time remaining will instantly decrease to reflect the fact that a device will not be fully analyzed. The time remaining on the timers assumes that each device to be tested will be functional, and will require approximately two and one-half

FIGURE 10 CONTINUOUS EXTRACTION DISPLAY

```
                    ***CSIM EXTRACTION STATUS***

PROCESS=XEROX PARC  NMOS                    VDD=5.00 VOLTS
LOT=23                                      TEMP=25.0 DEG C
WAFER=14                                    TOX=295 ANGSTROMS
DATE=JANUARY 14,1984                        XPOS=2    YPOS=2
OPERATOR=BRIAN MESSENGER                    DEVICE=NCHANNEL
OUTPUT FILE=csimout.TEXT                    WIDTH=50.00 MICRONS
PROBER FILE=MANUAL OPERATION                LENGTH=10.00 MICRONS

MINUTES TO DIE COMPLETION=2.5           MINUTES TO WAFER COMPLETION=2.5
DEVICE EXTRACTION LOCATION X                          FINISHED

PRESENT DEVICE CSIM PARAMETERS              X2U0=
VFB=                                        X2U1=
PHIF2=0.624                                 X3U1=
K1=                                         X2BETA0=
K2=                                         X2ETA=
ETA=                                        X3ETA=
BETA0=                                      BETA0SAT=
U0=                                         X2BETA0SAT=
U1=                                         X3BETA0SAT=

message from program=
```

minutes to be fully tested. Many factors, including the supply voltage, the number of devices to be tested, and the location of the output file will affect the program execution time. If a problem is encountered within the measurement or extraction sections of the program, the extraction meter is a very useful means for determining exactly where the difficulty occurred.

The lower section of the continuous display is devoted to the latest set of CSIM parameters that have been extracted. The linear region parameters are displayed seconds before the saturation region parameters, and both sets are visible only during the final stages of the program execution for a particular device. If the single device mode of operation is being used, the message **Press Enter to Enter Graphics Routine** will be displayed after development of the process file is complete. Prior to entering the graphics mode, it is possible to use the *Dump Alpha* key to print a hard copy of the extraction status and the list of CSIM parameters to be used as documentation. In the case of Semi Automatic Prober execution, the message **Press Enter to Continue Probing** will be displayed after the program has completed all operations on a specific device, and this allows the user to view the CSIM parameters for the latest device. The Automatic Probing Mode rapidly changes to a new display for the next device under test, and it is difficult to view all of the extracted parameters unless the program is *paused*.

5.4. Graphic Subprogram Interaction

The *Interactive Graphics Program* must be accessed from the Continuous Extraction Display, and the graphics menu is shown in Figure 11. Within this program, it is necessary to switch between the *Alpha* and the *Graphics* displays on the 9836 terminal. It is possible to superimpose both displays so that the graphics menu and the latest graph are simultaneously visible. When the

minutes to be fully tested. Many factors, including the supply voltage, the number of devices to be tested, and the location of the output file will affect the program execution time. If a problem is encountered within the measurement or extraction sections of the program, the extraction meter is a very useful means for determining exactly where the difficulty occurred.

The lower section of the continuous display is devoted to the latest set of CSIM parameters that have been extracted. The linear region parameters are displayed seconds before the saturation region parameters, and both sets are visible only during the final stages of the program execution for a particular device. If the single device mode of operation is being used, the message **Press Enter to Enter Graphics Routine** will be displayed after development of the process file is complete. Prior to entering the graphics mode, it is possible to use the *Dump Alpha* key to print a hard copy of the extraction status and the list of CSIM parameters to be used as documentation. In the case of Semi Automatic Prober execution, the message **Press Enter to Continue Probing** will be displayed after the program has completed all operations on a specific device, and this allows the user to view the CSIM parameters for the latest device. The Automatic Probing Mode rapidly changes to a new display for the next device under test, and it is difficult to view all of the extracted parameters unless the program is *paused*.

5.4. Graphic Subprogram Interaction

The *Interactive Graphics Program* must be accessed from the Continuous Extraction Display, and the graphics menu is shown in Figure 11. Within this program, it is necessary to switch between the *Alpha* and the *Graphics* displays on the 9836 terminal. It is possible to superimpose both displays so that the graphics menu and the latest graph are simultaneously visible. When the

FIGURE 11 INTERACTIVE GRAPHICS MENU

CSIM GRAPHICS MENU

The CSIM graphics routines use the CSIM parameters from the last device that has been measured to produce simulated data. When plots of measured data are requested, the 4145 is called to measure appropriate data points to produce a given set of curves.

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=? >
2)IDS versus VGS VDS=? >
3)ln(IDS) versus VDS VBS=? >

SELECT A NUMBER FOR A GIVEN ACTION CAPABILITY >

1)Zoom Using Knob and Keys
2)Redraw Full Graph
3)Select New Graph
4)Exit Graphics Menu

program is in the process of measuring or simulating a device under test, the *Alpha* display is cleared, but the past graph can be viewed on the *Graphics* display. The use of the zoom and redraw capability can be done while viewing the graphics display only, but the first time user may need to flip between the *Alpha* and *Graphics* displays to use these capabilities. Examples of the graphics capabilities are shown in the next section.

5.5. Experimental Results for Various Devices

This section includes sample 9836 graphics displays from the CSIM automatic parameter extraction system. Measured values of data are displayed with X's while simulated data is plotted with solid lines. Data points corresponding to subthreshold transistor operation are not screened from the RMS error calculations, and can bias the RMS error if they are present on the display.

Figures 12 and 13 show experimental *IDS versus VDS* results for a Width=50 micron, Length=4 micron device. This transistor has a 700 Angstrom gate oxide, and was manufactured at Xerox PARC. *IDS versus VGS* curves for the same device are displayed in figures 14 and 15, showing the characteristics at various drain-source biases. Figures 16 and 17 show how the zoom capability can be used to analyze the threshold region of operation for the same Xerox device. A box is drawn by the user around any region of the graph, as in Figure 16. The selected region is then displayed, as in Figure 17, and the user then has the option of zooming for even greater magnification, or returning to the original graph.

A W=20 micron, L=20 micron NMOS device fabricated at Xerox PARC is shown in Figures 18 and 19. This device is from a CMOS process using a 300 Angstrom gate oxide. The graphics program displays four quadrant graphics, and an example can be seen in the characteristic curves in Figures 20 and 21.

program is in the process of measuring or simulating a device under test, the *Alpha* display is cleared, but the past graph can be viewed on the *Graphics* display. The use of the zoom and redraw capability can be done while viewing the graphics display only, but the first time user may need to flip between the *Alpha* and *Graphics* displays to use these capabilities. Examples of the graphics capabilities are shown in the next section.

5.5. Experimental Results for Various Devices

This section includes sample 9836 graphics displays from the CSIM automatic parameter extraction system. Measured values of data are displayed with X's while simulated data is plotted with solid lines. Data points corresponding to subthreshold transistor operation are not screened from the RMS error calculations, and can bias the RMS error if they are present on the display.

Figures 12 and 13 show experimental *IDS versus VDS* results for a Width=50 micron, Length=4 micron device. This transistor has a 700 Angstrom gate oxide, and was manufactured at Xerox PARC. *IDS versus VGS* curves for the same device are displayed in figures 14 and 15, showing the characteristics at various drain-source biases. Figures 16 and 17 show how the zoom capability can be used to analyze the threshold region of operation for the same Xerox device. A box is drawn by the user around any region of the graph, as in Figure 16. The selected region is then displayed, as in Figure 17, and the user then has the option of zooming for even greater magnification, or returning to the original graph.

A W=20 micron, L=20 micron NMOS device fabricated at Xerox PARC is shown in Figures 18 and 19. This device is from a CMOS process using a 300 Angstrom gate oxide. The graphics program displays four quadrant graphics, and an example can be seen in the characteristic curves in Figures 20 and 21.

These curves are from a W=20 micron, L=4 micron PMOS device from a Xerox PARC wafer.

Experimental results for devices ranging down to 1 micron drawn channel lengths have been obtained using the parameter extraction system. Figures 22 through 25 are from a W=50 micron, L=1 micron device fabricated at Bell Labs Holmdale. These devices are from the SIGMOS process, and have a 200 Angstrom gate oxide thickness, and a DELTAL of 0.25 microns.

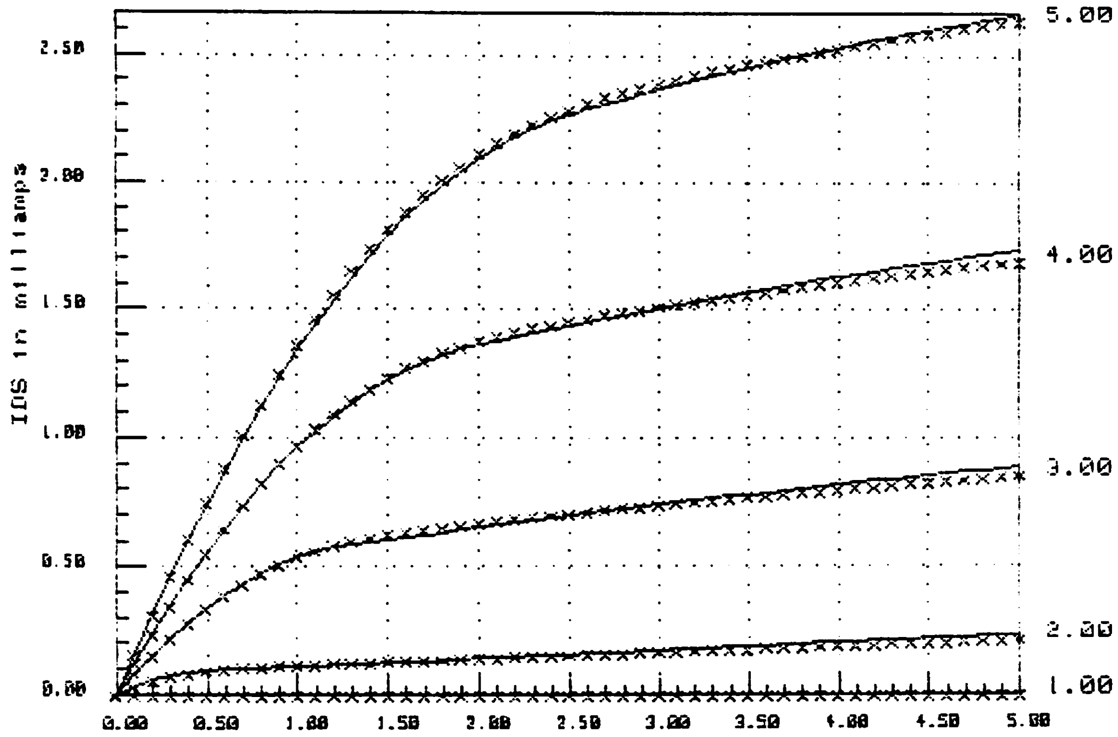


Figure 12
XEROX PARC W=50U L=4U

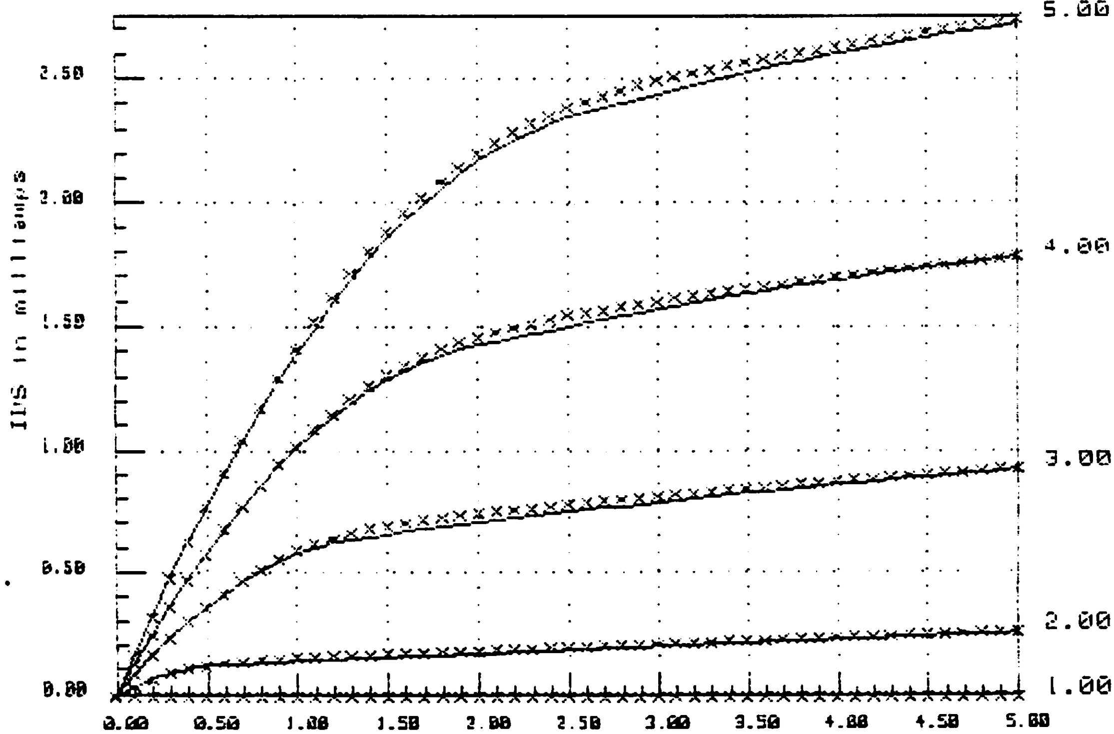


Figure 13
XEROX PARC W=50U L=4U

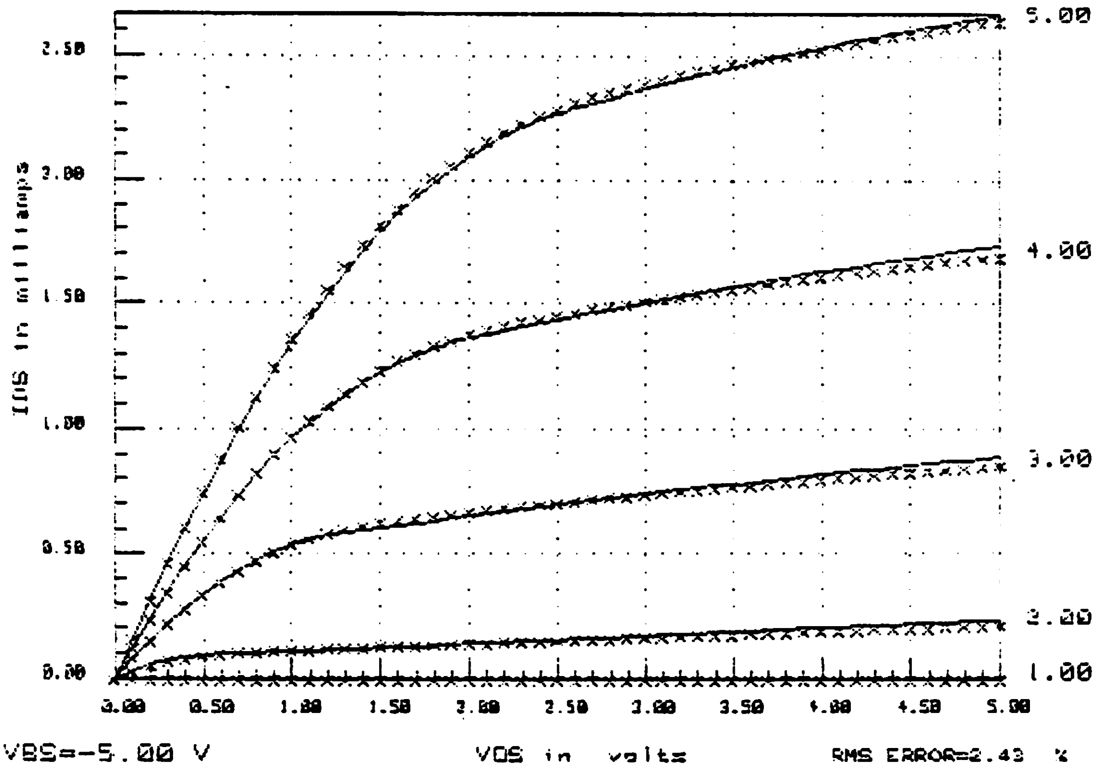


Figure 12
XEROX PARC W=50U L=4U

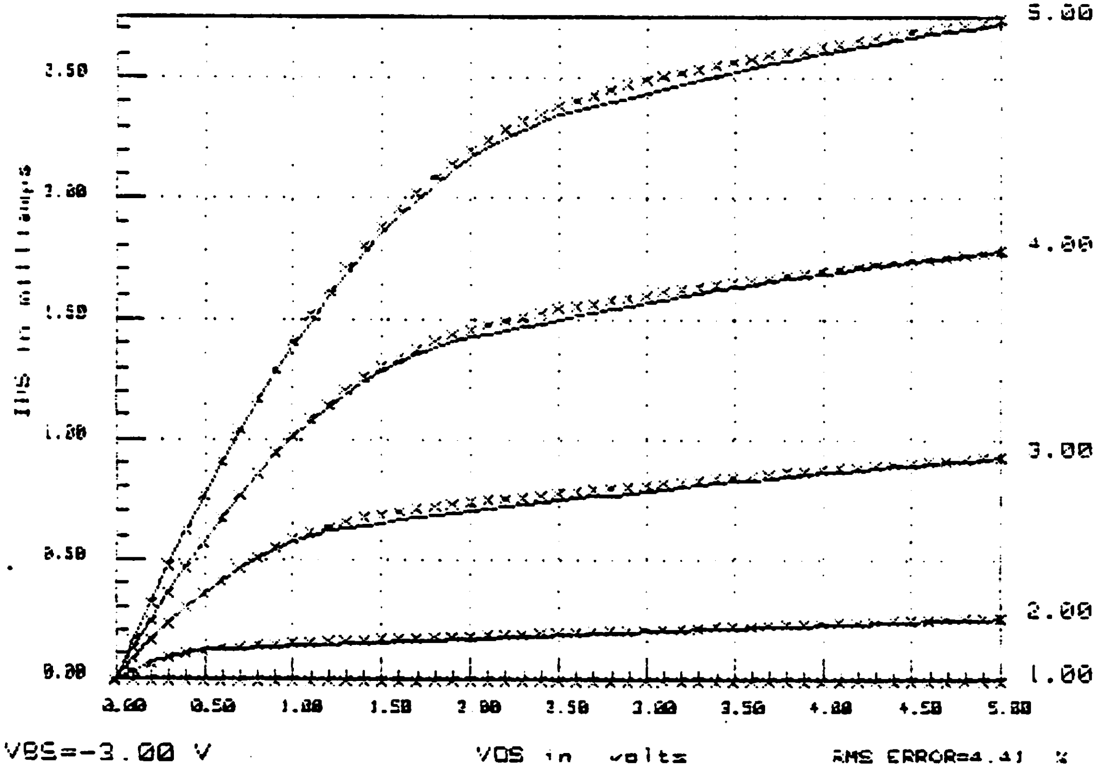


Figure 13
XEROX PARC W=50U L=4U

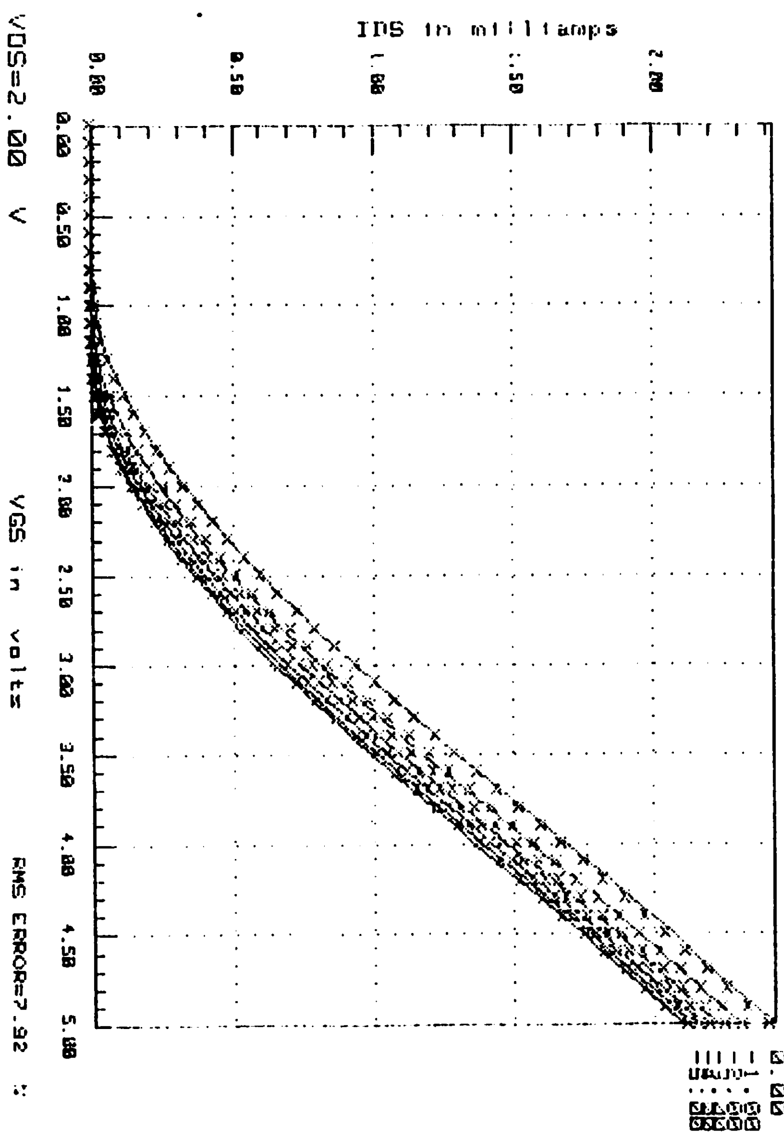


Figure 15
XEROX PARC W=50U L=4U

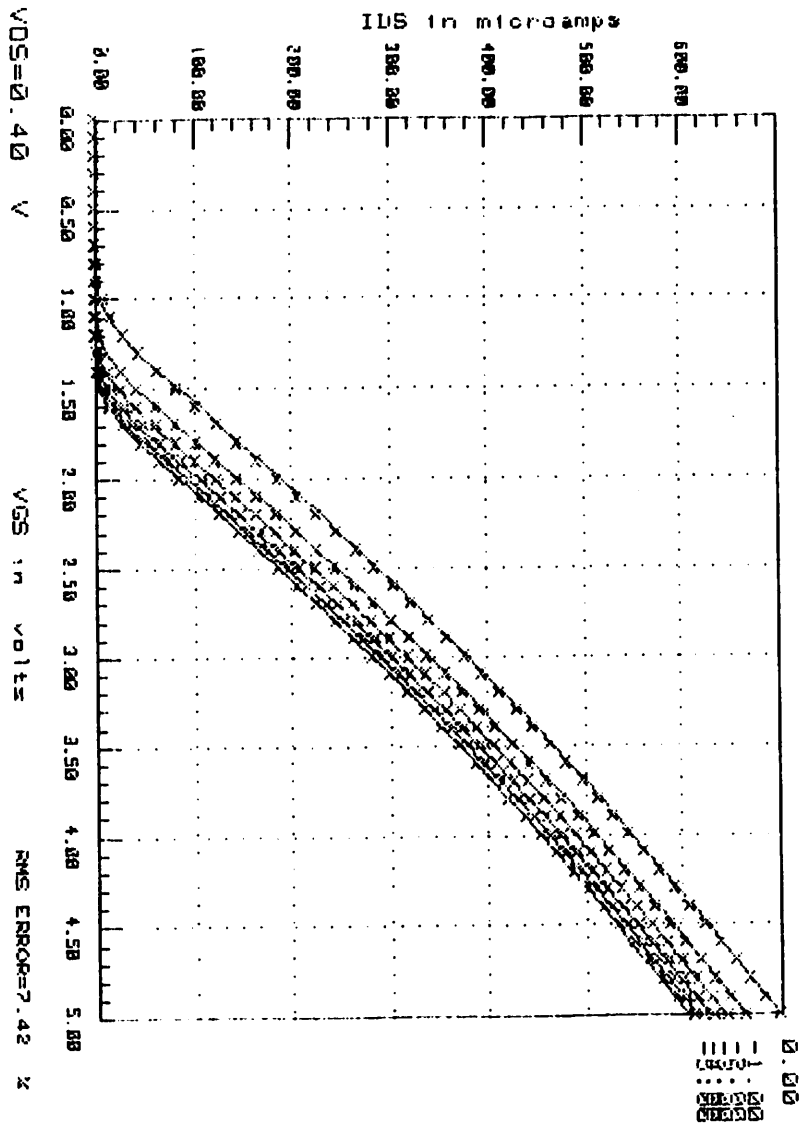


Figure 14
XEROX PARC W=50U L=4U

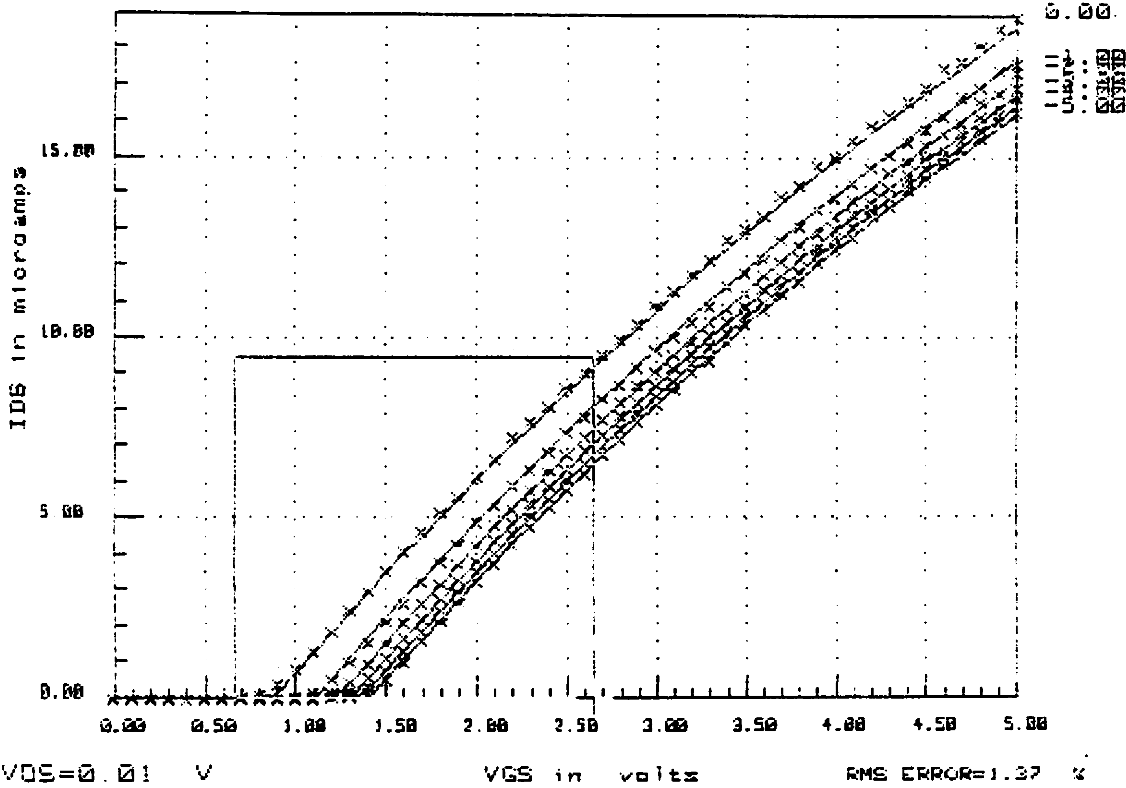


Figure 16
XEROX PARC W=50U L=4U

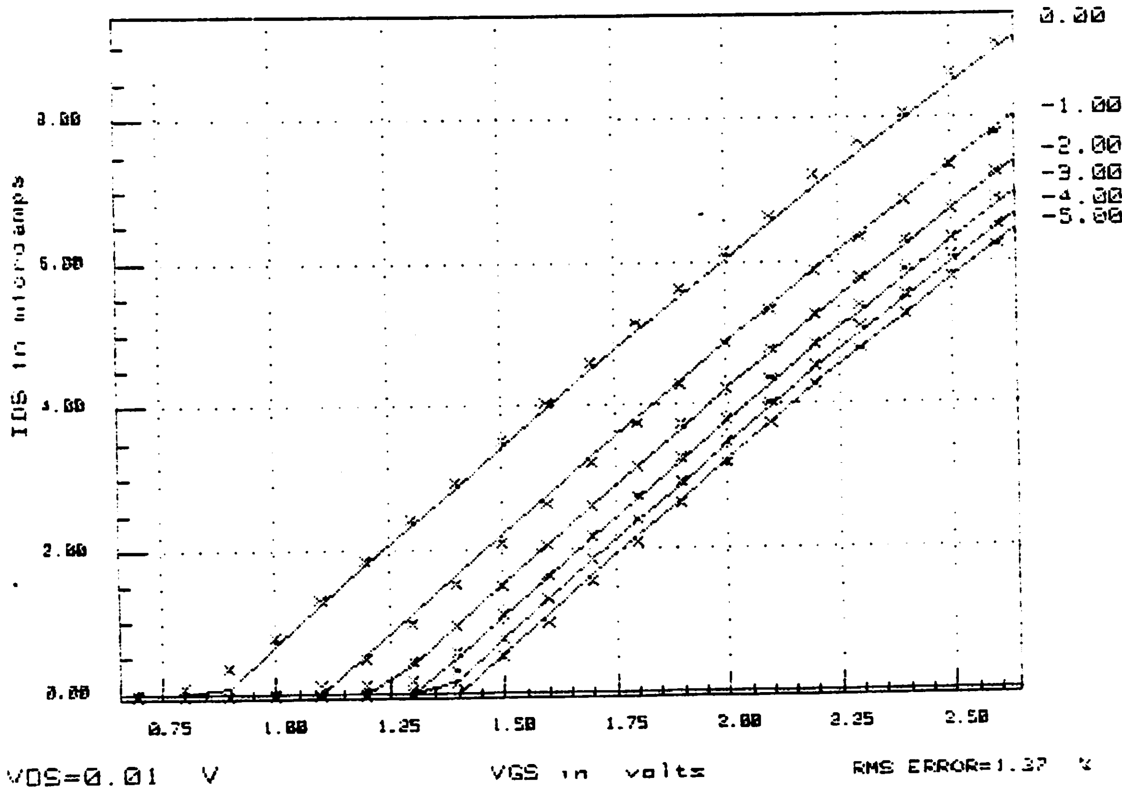


Figure 17
XEROX PARC W=50U L=4U

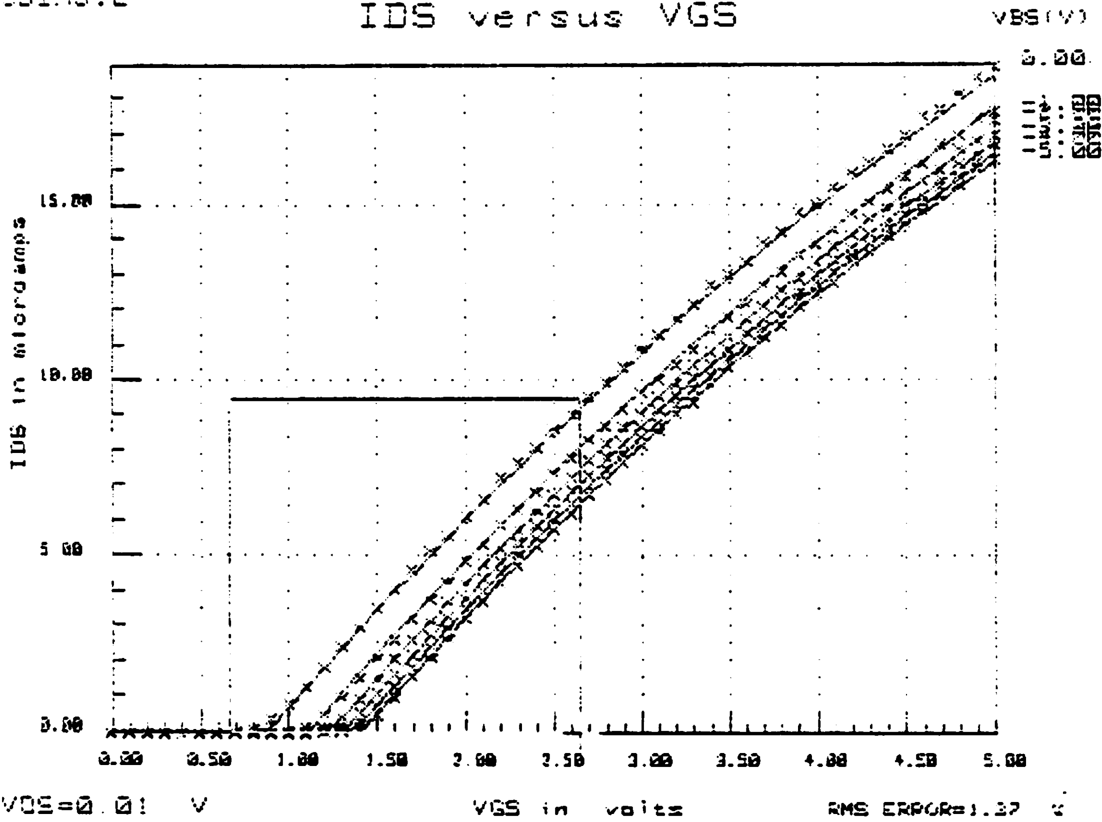


Figure 16
XEROX PARC W=50U L=4U

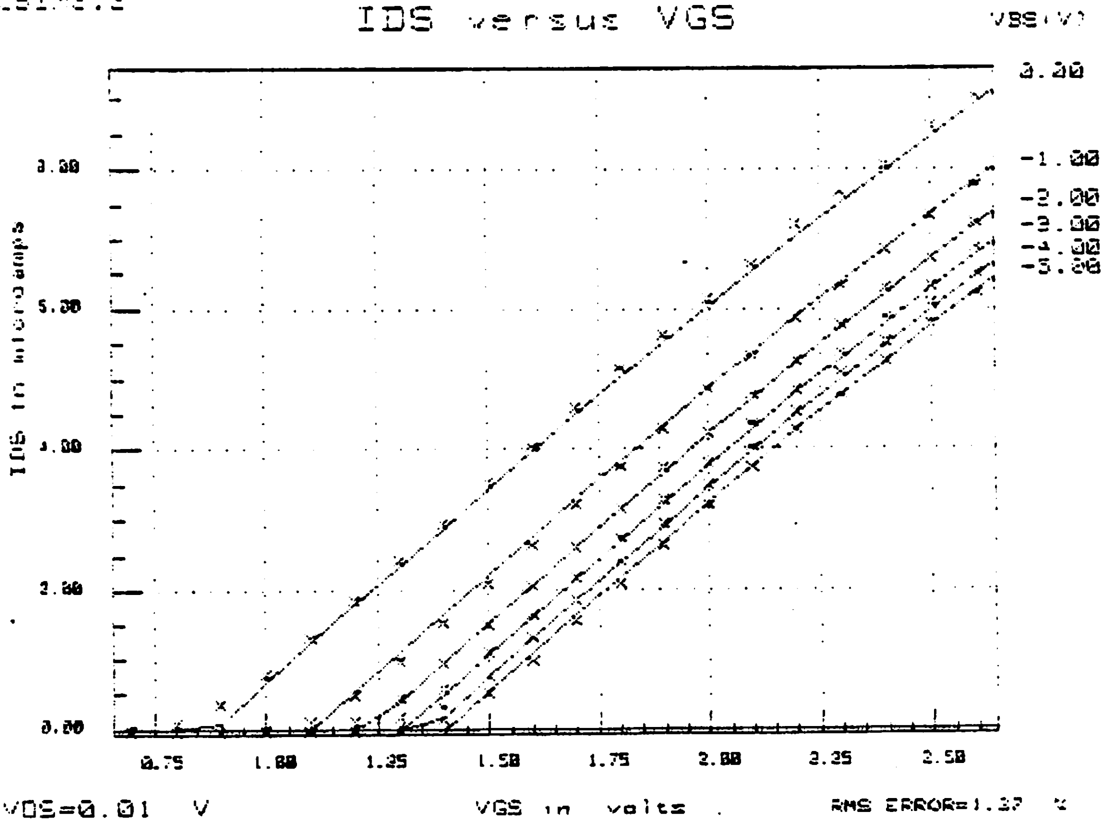


Figure 17
XEROX PARC W=50U L=4U

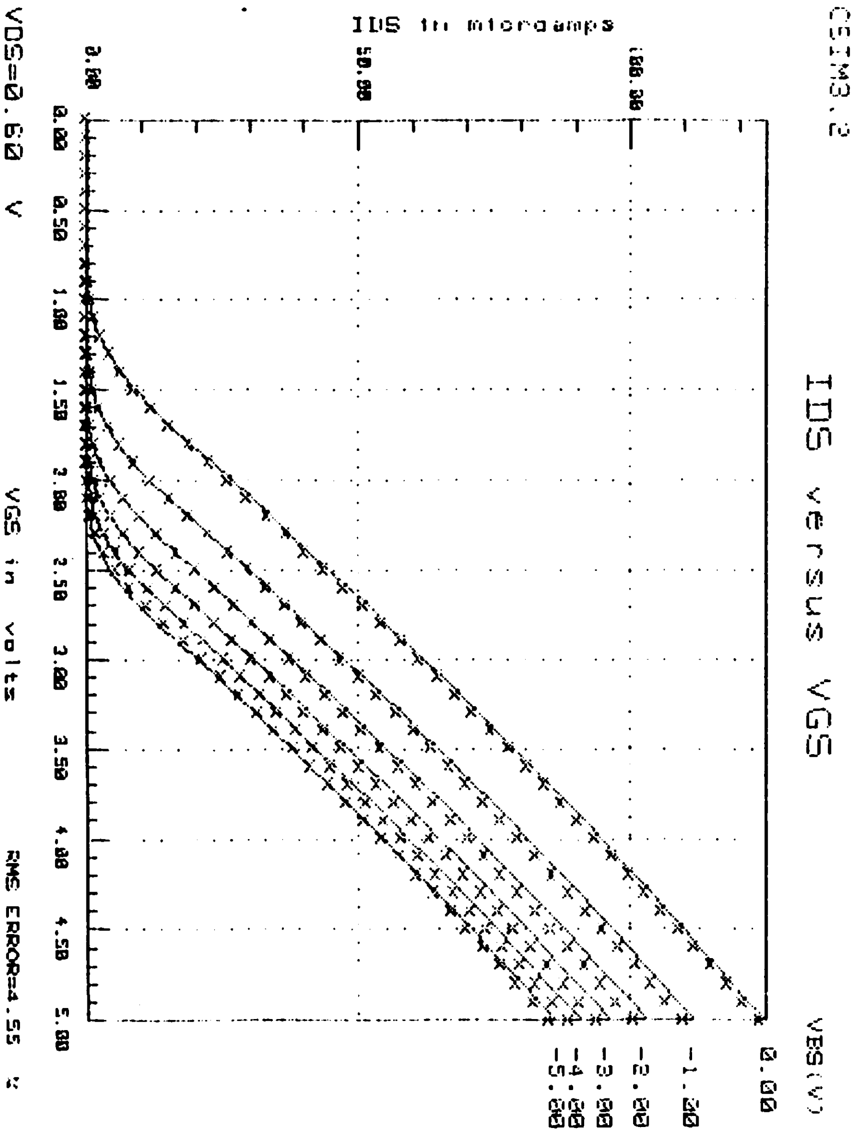


Figure 19
XEROX PARC W=20U L=20U

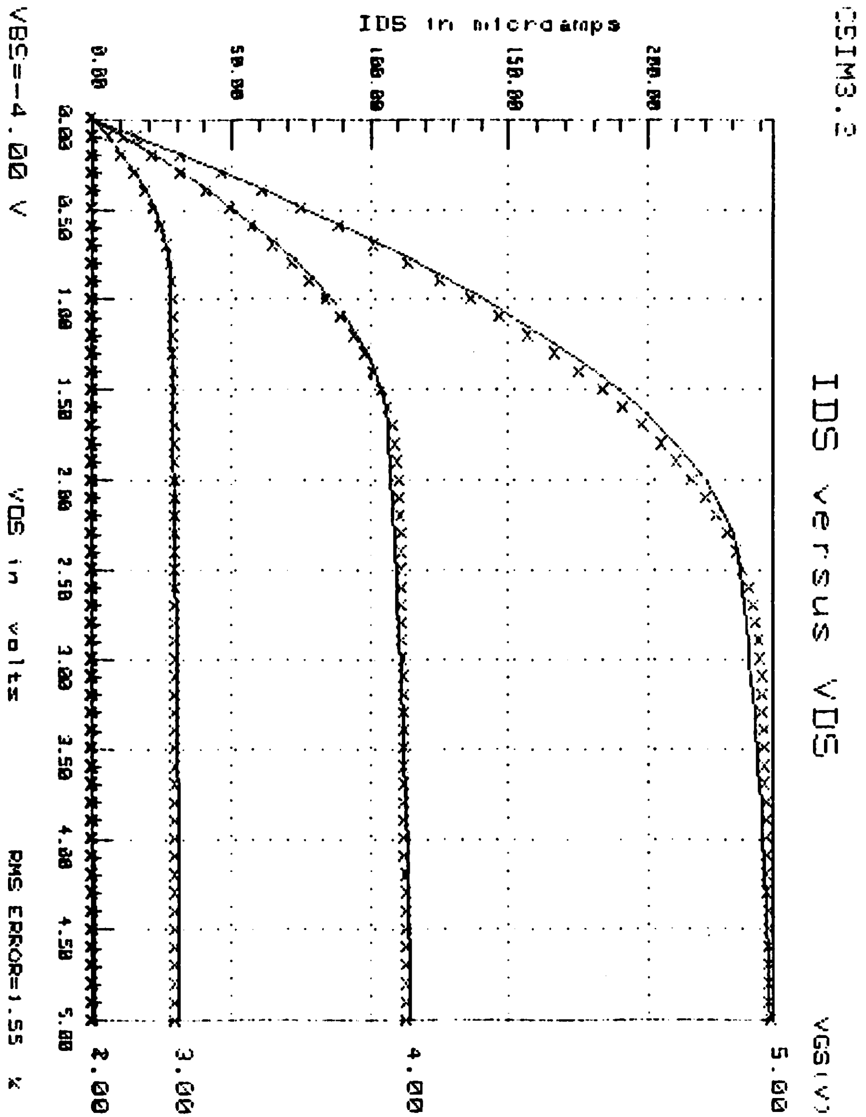


Figure 18
XEROX PARC W=20U L=20U

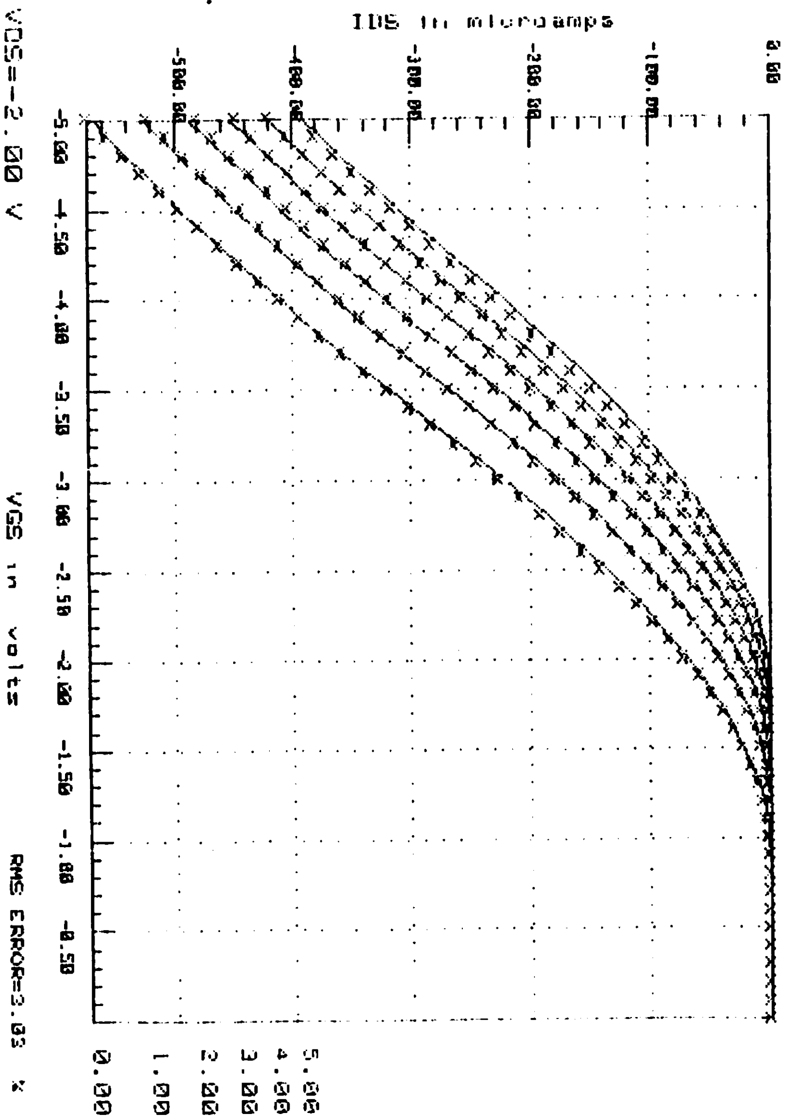


Figure 21
XEROX PARC W=20U L=4U

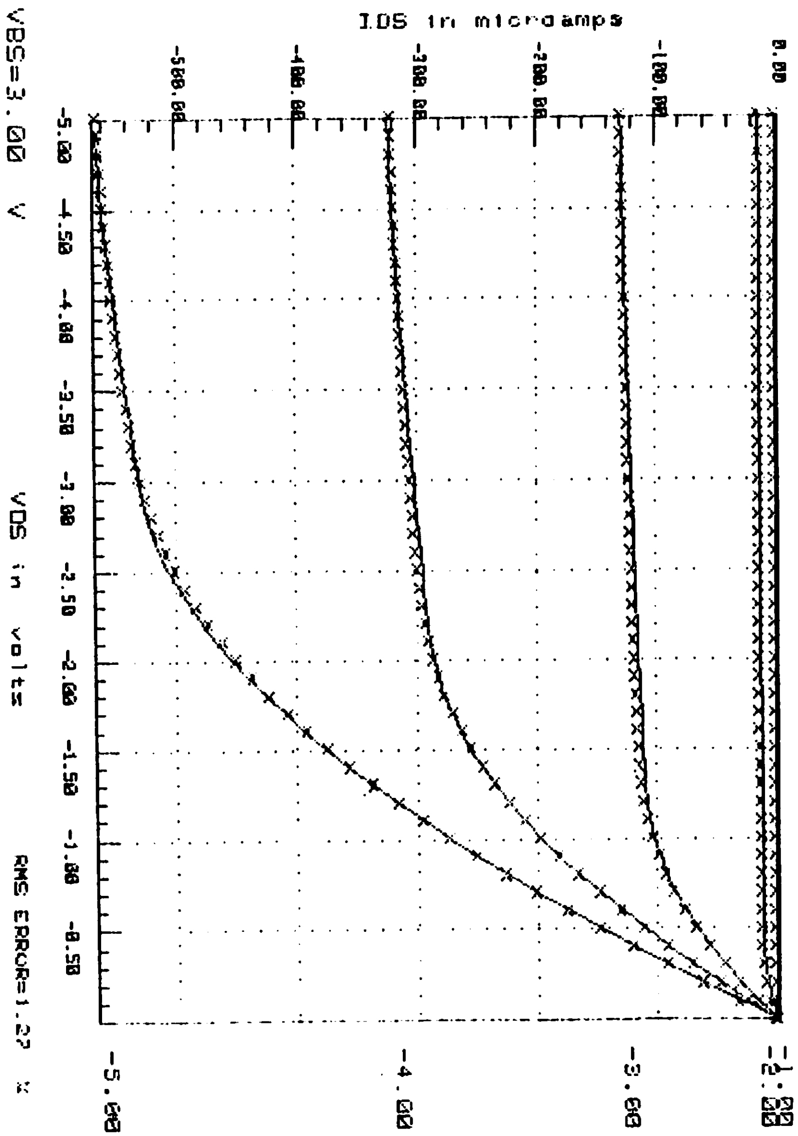


Figure 20
XEROX PARC W=20U L=4U

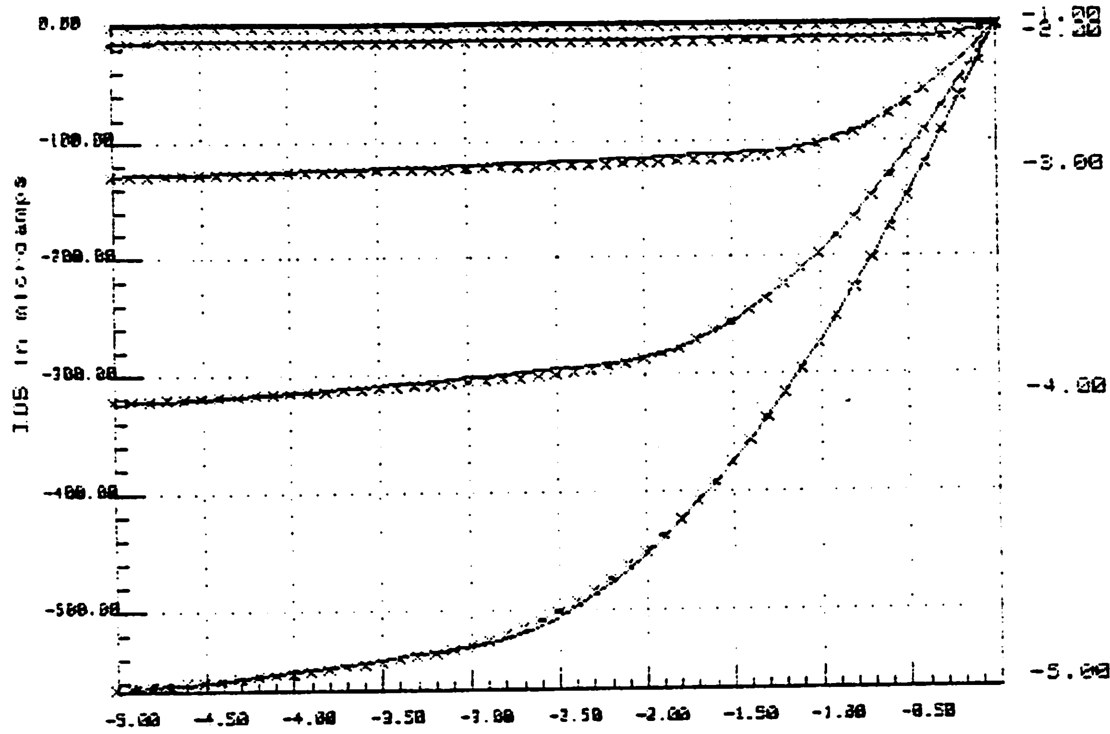


Figure 20
XEROX PARC W=20U L=4U

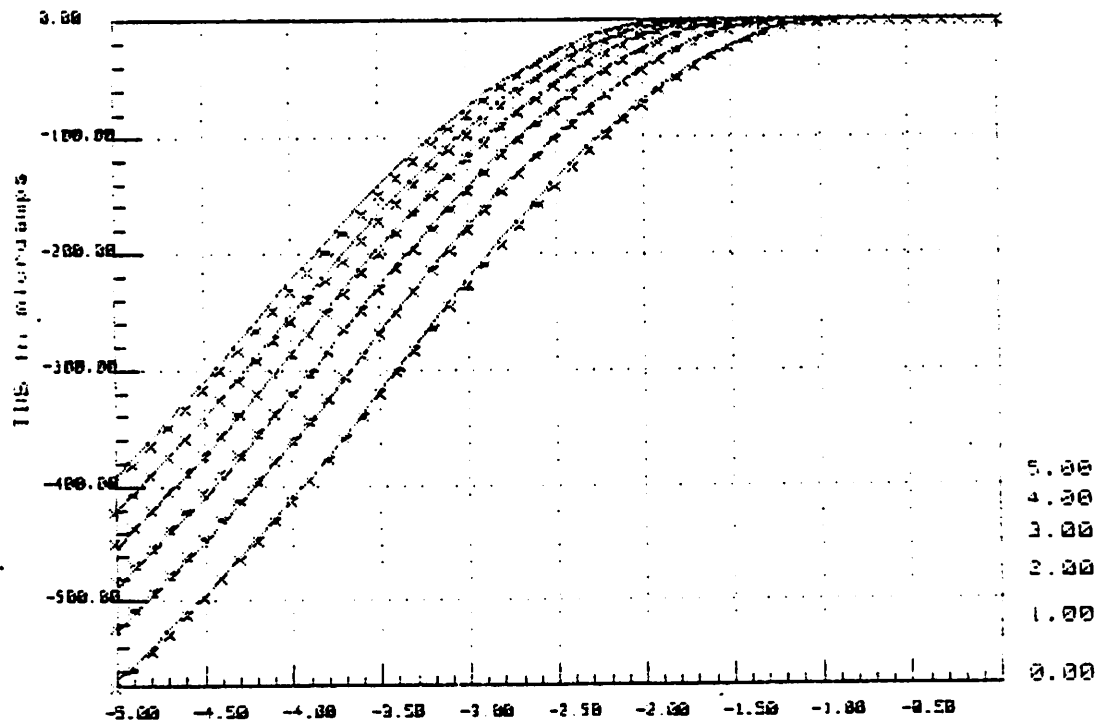


Figure 21
XEROX PARC W=20U L=4U

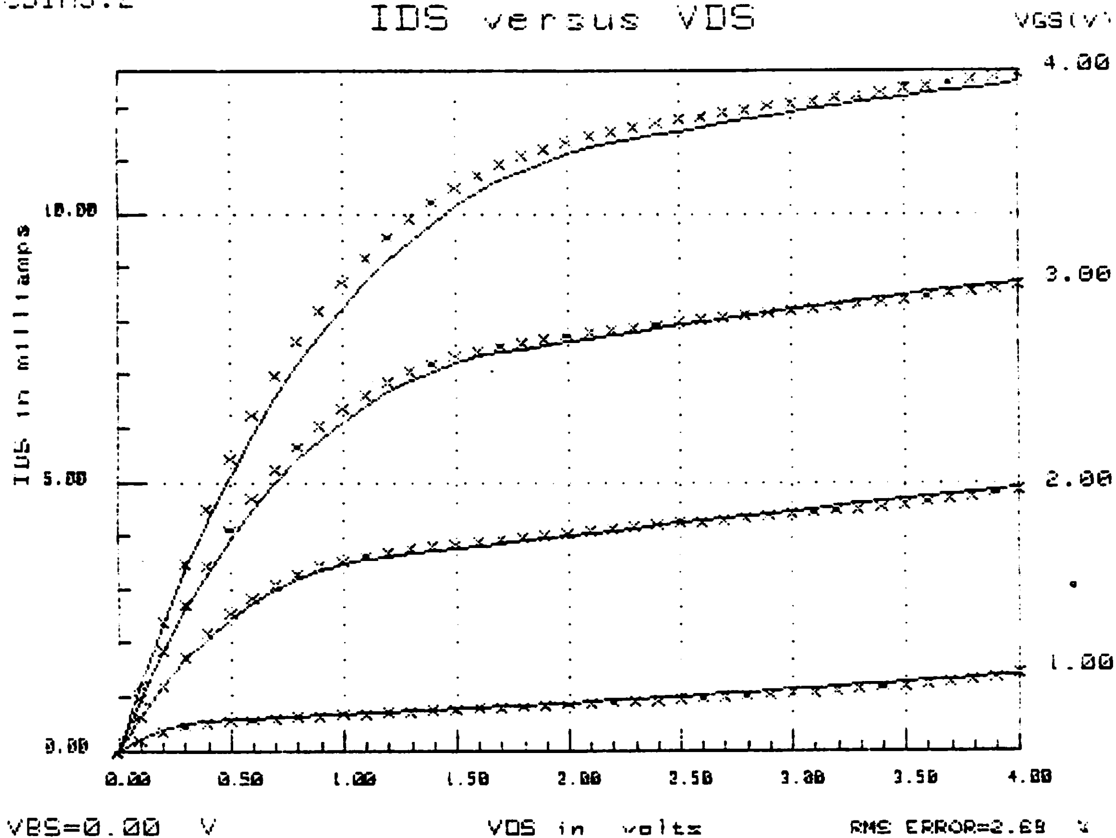


Figure 22
BELL LABS W=50U L=1U

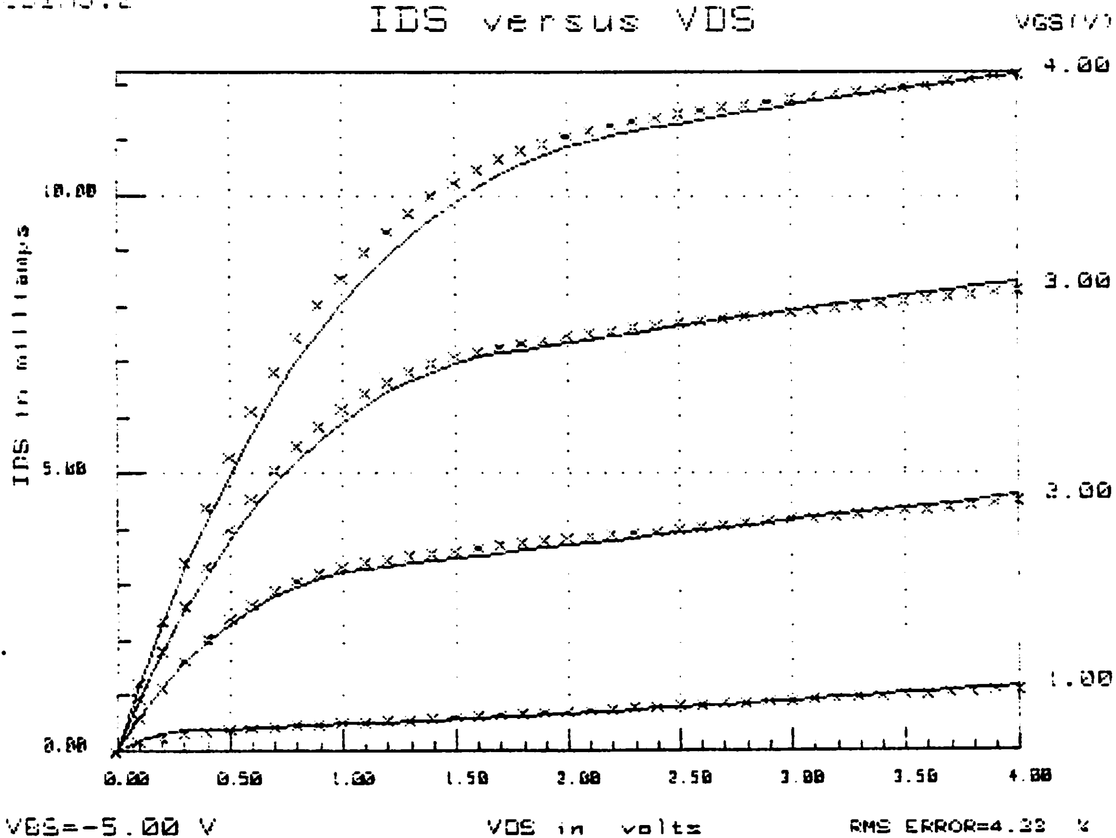


Figure 23
BELL LABS W=50U L=1U

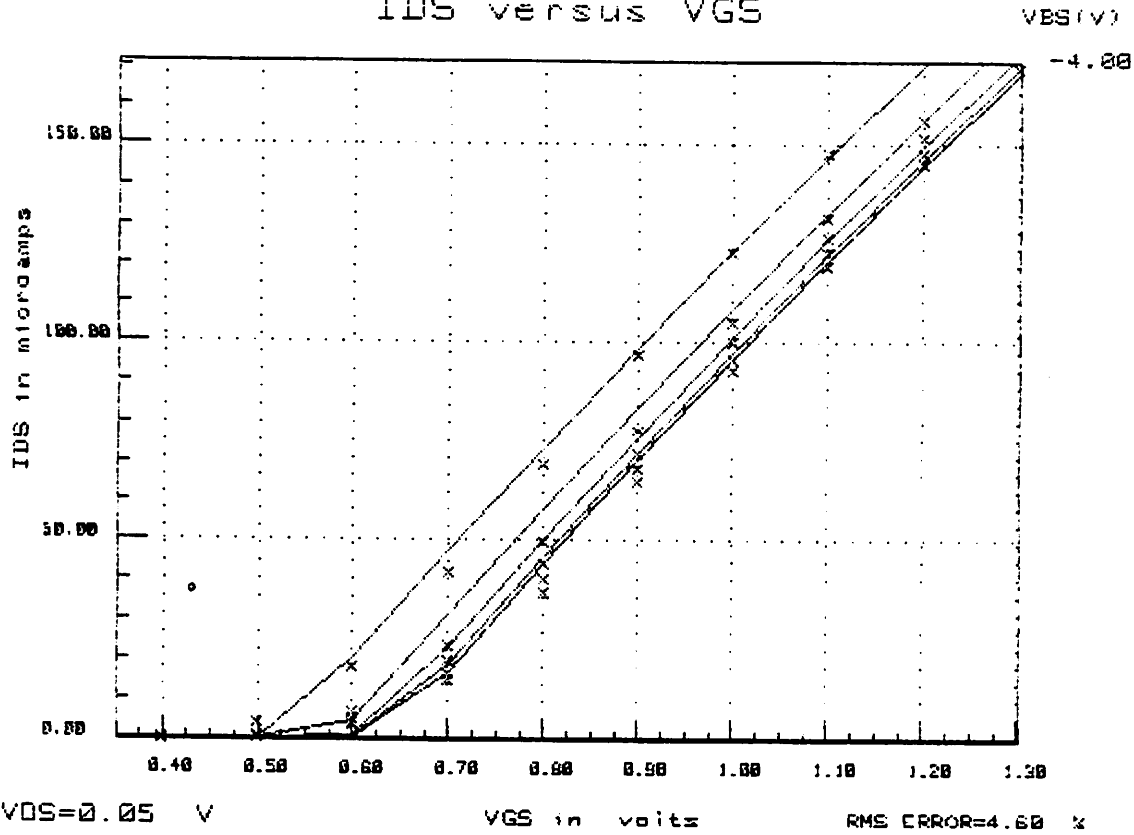


Figure 24
BELL LABS W=50U L=1U

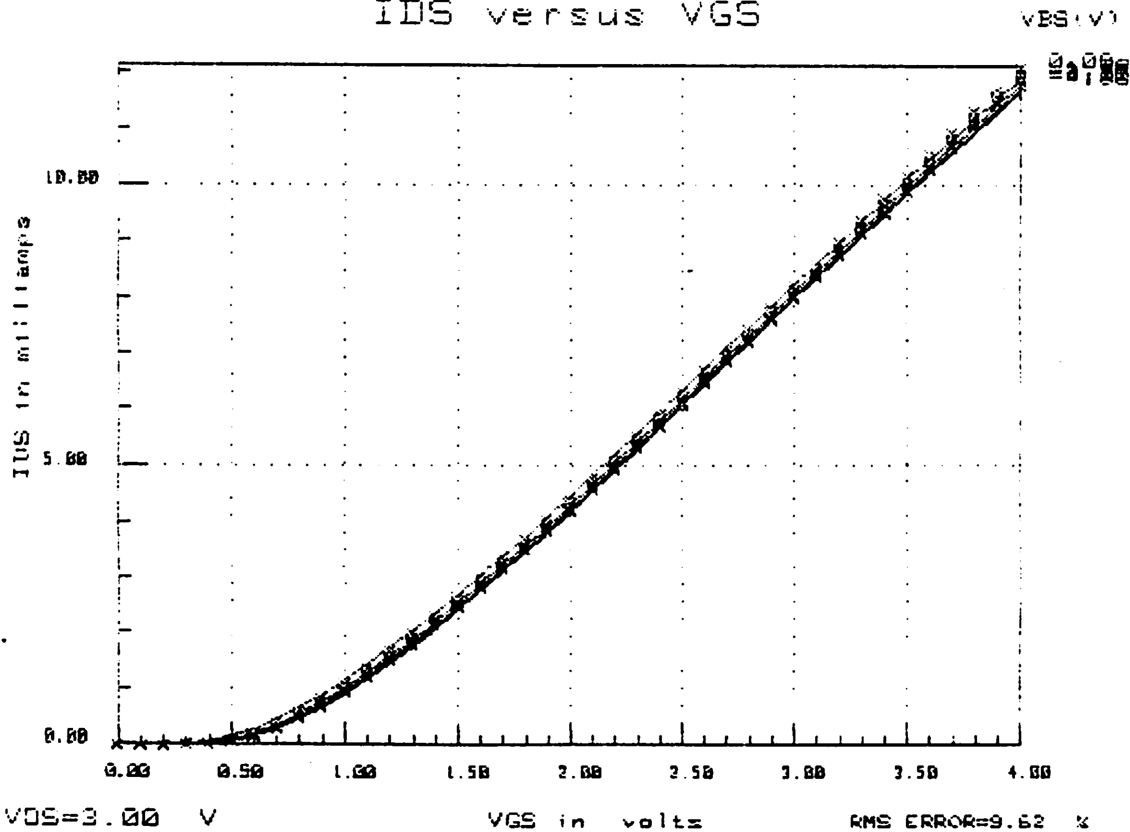


Figure 25
BELL LABS W=50U L=1U

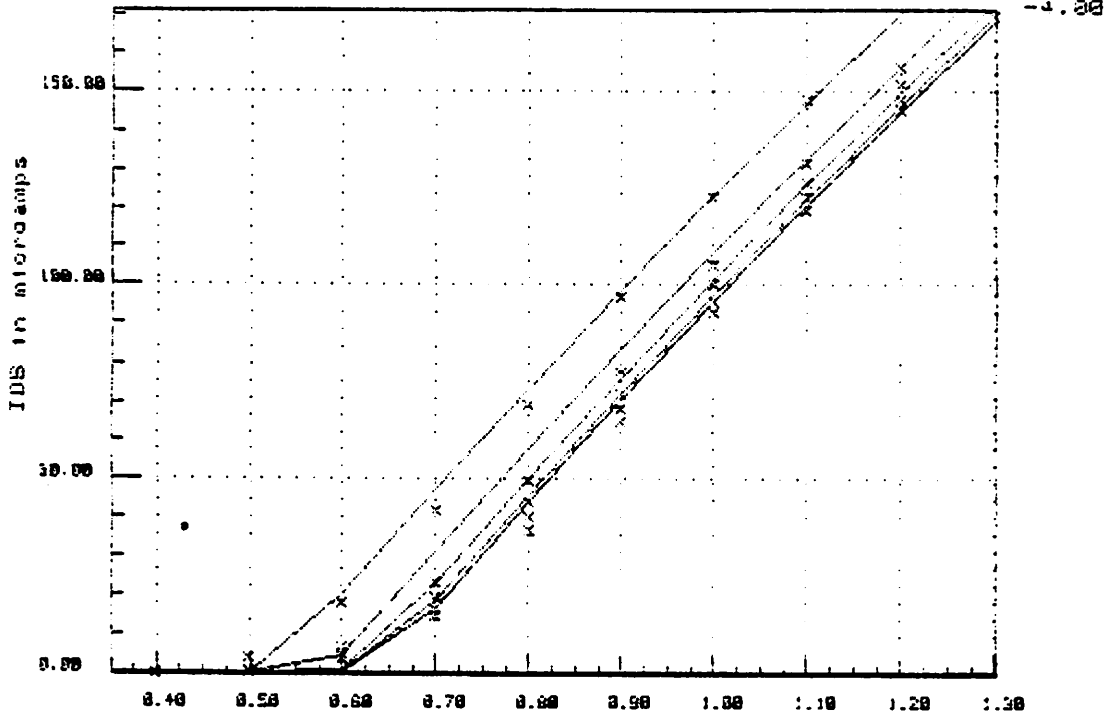


Figure 24
BELL LABS W=50U L=1U

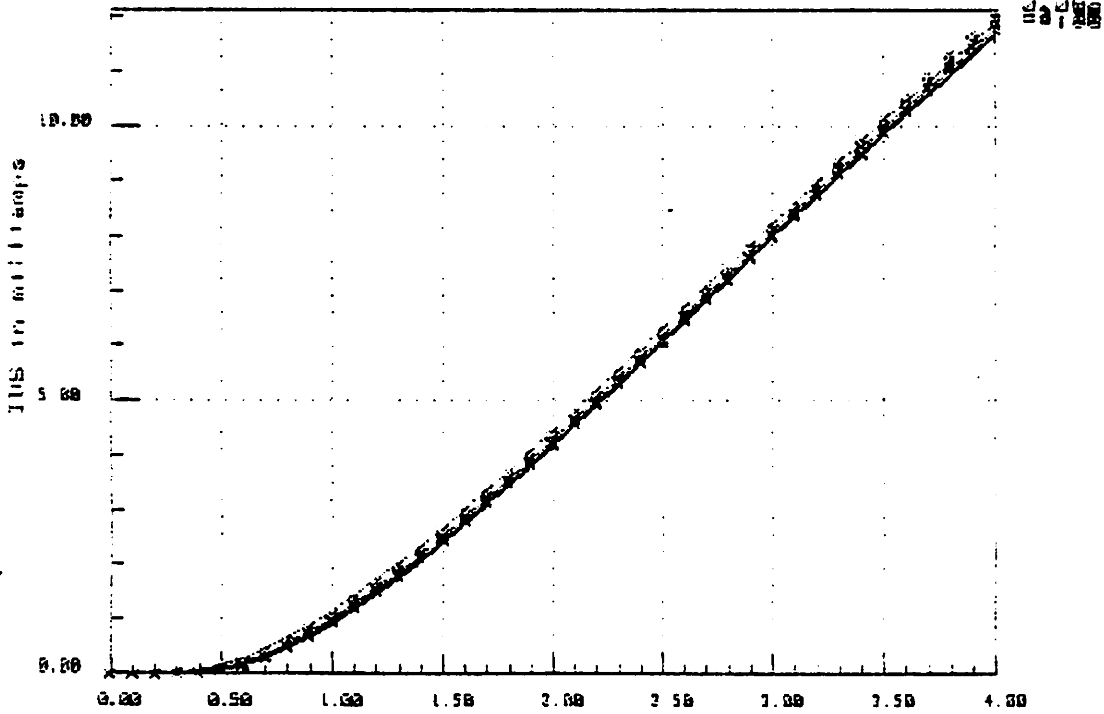


Figure 25
BELL LABS W=50U L=1U

6. POSSIBLE FUTURE PROGRAM ENHANCEMENTS

Future enhancements to the entire CSIM model and to the automatic parameter extraction system are presently under development. These include modeling of device capacitance, interconnect capacitance and resistance, and subthreshold device characteristics. Extensive analysis of large numbers of devices must be done to verify that the geometry dependent modeling presently in use can be universally applied. In conjunction with this analysis, statistical data bases for integrated circuit processes can be developed, and parameter fluctuations with processing variations can be analyzed.

A CSIM device capacitance model has been developed by Bing Sheu[5] to model the dynamic charge distribution in a MOS device. This model is compact, and all of the capacitances can be derived from the present CSIM device parameters. Bing Sheu is also developing a version of CSIM which will model the subthreshold region, and this will require the construction of additional automatic parameter extraction algorithms and their software implementation. In the case of subthreshold parameter extraction, the CSIM program, in the process of determining an approximate threshold voltage, already measures data points which may be applicable to full subthreshold analysis. The alternative, under consideration, is to enter an optional subthreshold analysis subprogram which will use the threshold voltage extracted in the basic extraction routines to determine which data points to measure for subthreshold analysis.

Work is underway at Berkeley to lay out standard test structures which can be used to analyze both devices and interconnect characteristics. Accurate models for interconnect resistance and capacitance need to be developed, and the automatic extraction of these parameters should be fairly straightforward. Accurate capacitance measurements and measurements of ring oscillators for speed analysis will require a more elaborate prober and measurement setup.

The subthreshold and device capacitance modeling can be done with the present system.

Many processes need to be analyzed to confirm that the geometry dependent modeling used in the parameter extraction software is general over all device sizes. This work will be done by Joe Pierret and Bing Sheu, under the direction of Dr. Scharfetter in the coming months. Work will begin on developing a statistical base of device and process parameters. A statistical program named STAT2, which was written at the National Bureau of Standards, will likely be used to do statistical analysis of large quantities of data across wafer runs. This data will be used to highlight the affects on device parameters due to process variations, and will enable the development of best, nominal and worst case parameter files to be used by circuit designers.

The subthreshold and device capacitance modeling can be done with the present system.

Many processes need to be analyzed to confirm that the geometry dependent modeling used in the parameter extraction software is general over all device sizes. This work will be done by Joe Pierret and Bing Sheu, under the direction of Dr. Scharfetter in the coming months. Work will begin on developing a statistical base of device and process parameters. A statistical program named STAT2, which was written at the National Bureau of Standards, will likely be used to do statistical analysis of large quantities of data across wafer runs. This data will be used to highlight the affects on device parameters due to process variations, and will enable the development of best, nominal and worst case parameter files to be used by circuit designers.

REFERENCES

[1] H. C. Poon, "A Compact Short-Channel IGFET Model (CSIM)," internal Bell Labs memorandum, April 2,1976

[2] B. J. Sheu, D. L. Scharfetter and H. C. Poon, "Compact Short-Channel IGFET Model," to be published.

[3] R. Liu, "An IC Process-Oriented Circuit Simulator Developed from SPICE 2G.6 using the CSIM MOSFET Model," University of California, Berkeley Master's Report, January 1984.

[4] H.K.J. Ihantola and J.L. Moll, "Design Theory of a Surface Field Effect Transistor," Solid State Electronics, Vol. 7, p. 423-430 (June 1964).

[5] B. J. Sheu, D. L. Scharfetter, C. Hu, and D. O. Pederson, "A New Charge Model for MOSFET Circuit Simulation," to be published.

[6] H. C. Poon, L. D. Yau, R. L. Johnston, and D. Beecham, "DC Model for Short-Channel IGFETs," IEEE IEDM Technical Digest, pp. 156- (1973), and H. C. Poon, Asilomar Conference on Circuits, Systems and Computers, Pacific Grove, Ca, Dec. 1977, pp. 508-512.

APPENDIX 1 PARAMETER EXTRACTION SYSTEM SETUP INSTRUCTIONS

1. 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 light up and diplay a 4145 Menu after it calibrates itself.

2. 2001X PROBER SETUP INSTRUCTIONS

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 tightened down securely.

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. The prober display is in *italic* print, and the answer you should return is in **bold** print.

Type Message Plus Enter=> **Enter Key**
Wait for Pattern Rec I/O Test... **Wait about 30 seconds**
Rom Test? **Y**
Repeat Test? **Enter Key**

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 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 pannel. (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 Enter Key to exit any submenu for the display.

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. If adjustment is to be made, refer to steps I and J for preliminary setup. 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.

3. 9836 SETUP INSTRUCTIONS

A) Insert *Pascal 2.0 Boot Disc* in the right (#3:) Disc Drive. Insert *Pascal 2.0 Sysvol Disc* in the left (#4) Disc Drive. Turn on the 9836 with the switch on the front bottom right of the machine by pressing the switch in.

B) The operating system will now be loaded from the Boot Disc. Enter the date and time when they are requested, or hit *Enter* to skip these inputs.

C) Insert the *Pascal 2.0 Access Disc* in the right (#3:) Disc Drive. The following instructions will load the filer and the editor from the Access Disc into main memory. Enter the Bold values in the following sequence.

P {this selects the permanent load operating system option}
Load what code file? **#3:EDITOR.**
P
Load what code file? **#3:FILER.**

D) Now the main memory volume will be created. This volume is used to store the CSIM program in, and to perform all process file manipulations in. **If you create a process file that you want to save, you must either write it to a FLOPPY DISC, or send it to the VAX before you turn off the machine. If you do not do this, you will loose the file.**

M {**creating a Memory Volume**}
****CREATING A MEMORY VOLUME****
What unit number? #45: How many 512 byte blocks? **1000**
How many entries in directory? **0**
#45: (RAM:) zeroed

E) Now the csim compiled code will be copied into RAM. Place the *CODE:* Disc in the right (#3:) Disc Drive.

F {This loads the filer system}
F {This loads the filecopy program}
Filecopy what file? **#3:csim.CODE**
Filecopy to what? **#45:csim.CODE**

F) Now the RAM Volume will be prefixed. **If this is not done, a program testing many devices will take much longer since the floppy disc which is prefixed has a much slower access time.**

P {this loads the prefix program}
Prefix to what directory? **#45:**
Q {this quits the filer program}

G) Now the CSIM PROGRAM is ready for execution.

X {this selects the execute option}
Execute What File? **csim**
Loading 'csim.CODE'

H) Note: If an I/O error 26 occurs before the main menu is displayed, repeat section G, and the program will execute. This error is due to an intermittent problem with the RS232 interface board.

4. 9836 TO VAX LINK

A) To transfer files to the Vax, the RS232 Data Communication Board must be in the back of the 9836, and it must be connected to a port selector or a modem. Insert the *Pascal 2.0 VTCODE Disc* in the right Disc Drive.

P {this selects the permanent load operating system option}
Load what code file? **#3:NEWKBD** {*This is a new keyboard file needed by VT2*}
X {*This selects the execute option*}
Execute what file? **#3:VT2**

B) When the program is loaded, a menu will appear, and the user must load in the configuration as follows.

Main> 4 {option to create a configuration}
Selection? 1 {VAX/UNIX}
Rate? **9600** {baud rate}

APPENDIX 2 LAYOUT CONSIDERATIONS FOR AUTOMATIC PROBING

Numerous methods for laying out transistors and test structures on integrated circuits have been used by various universities and throughout the industry. The extraction system described here can be used to analyze most MOS transistors, assuming some means of direct contact to the *drain*, *gate*, *source*, and *body* are provided. The full benefit of the extraction capability is derived when the program can be operated in the Automatic Wafer Probing Mode, and this is dependent on the layout of the test devices. Given the present 4145 based test system, only a limited number of test pattern layouts are acceptable for Automatic Wafer Probing.

The extraction system is designed to be used with a four probe wafer *Probe Card* which can provide contact to each of the four terminals of a device. If back-side wafer contact is possible for an NMOS process, only three probes are necessary for operation. The only limitation on metal pad layout for device testing is that the pads occur in some regular pattern which will allow all devices to be tested in some sequential fashion. It is not necessary to designate individual probes for individual device terminals since the automatic prober file can absorb a variation such as a probe being the *source* for one device and the *gate* for another device.

At U.C. Berkeley, a two by ten rectangular pad arrangement has been adopted for use in laying out the next generation of test structures. Various approaches can be used to lay out individual devices on a general two by n pattern depending upon the density and device to device isolation required. A possible layout resulting in total device isolation would have four pads for each device, and would allow five devices for every two by ten array. A much denser layout, which is also compatible with automatic probing, would be to share pads so that a device could be laid out between every row of pads stepping down through a two by ten array. This would allow up to nine devices to be laid out on a two by

ten pattern, assuming each set of four pads in a square of pads was connected to the four device terminals for the device between the pads.

APPENDIX 3 INSTRUCTIONS FOR AUTOMATIC PROBER FILE DEVELOPMENT

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
00000000000000000000
00000000000000000000
00000000000000000000
00000010000010000000
000X0000000000000000    {This is the 20 by 20 array to designate the}
00000000000000000000    {die which are to be tested. All locations}
10000100001000000000    {marked with a '1' will be probed, and the}
00000000000000000000    {location marked with a 'X' is the origin die}
00000000000000000000
00000100000010000000    {The upper left corner of the array represents}
00000000000000000000    {location (1,1), and the x values increase towards}
00000000000000000000    {the right, while the y values increase towards}
00100000001000000000    {the bottom.}
00000000000000000000
00000000000000000000    {In this instance, the origin is (4,6).}
00000000000000000000
00000000000000000000
00000000000000000000
00000000000000000000
**                          {Device to device delimiters}
{Here is where the first device to be tested is described}
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 #2 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 toSMU 2}
ss=4                          {Source of device #1 is connected to SMU 4}
sb=3                          {Body of device #1 is connected toSMU 3}
**                          {Device to device delimiters}
{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.

```
mx, my  {microns}
w, l    {microns}
dt      {-1=PMOS, 1=NMOS}
ed      {-1=depletion, 0=zero-threshold, and 1=enhancement}
sd,sg,ss,sb  {any value from 1 to 4 which is not the value of any of}
             {any one of the remaining three.}
```