

Avant!

TCAD Business Unit



IMPORTANT NOTICE

For Users of TSUPREM-4 Version 2000.2

Enhancements contained in the v2000.2 release of the TSUPREM-4 program are briefly described in Appendix C. Please read Appendix C before using TSUPREM-4 2000.2 to assure efficient use of this upgrade. This Appendix describes the extent of the version differences for users of the previous version of the code.

Reader Comments: *TSUPREM-4 Version 2000.2 User's Manual*

Avant! TCAD welcomes your comments and suggestions concerning this manual. Please mail this form (Attn.: Technical Publications Department) to the address on the reverse side of this sheet, or fax a copy to (510) 413-7743, or e-mail your comments to tcad-pubs@avanticorp.com. Attach additional pages if needed

What model of computer are you using?_____Operating system?_____

Did you find any errors in this manual? If so, please list the page number and describe the error.

Have you encountered program features that need to be better described in this manual?

What additional information should be included?

How can this document be improved?

Other comments and suggestions:

Fold here and tape

To:
Avant! Corporation
TCAD Business Unit, Technical Publications
46871 Bayside Parkway
Fremont, CA 94538
USA

Fold here and tape

From:
(Optional Information)

Name, Position:

Company

Address:

.....

.....

Phone, fax, or e-mail

Avant!

TSUPREM-4

Two-Dimensional Process Simulation Program

Version 2000.2

User's Manual

July 2000



Avant! Corporation, TCAD Business Unit

Fremont, California

TSUPREM-4™ User's Manual, Release 2000.2 First Printing: July 2000
 Copyright © 2000Avant! Corporation and Avant! subsidiary. All rights reserved.
 Unpublished—rights reserved under the copyright laws of the United States.

Avant! software TSUPREM-4™ v2000.2 Copyright © 2000 Avant! Corporation and Avant! subsidiary.
 All rights reserved.
 Unpublished—rights reserved under the copyright laws of the United States.

Use of copyright notices is precautionary and does not imply publication or disclosure. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013

Disclaimer

AVANT! RESERVES THE RIGHT TO MAKE CHANGES TO ANY PRODUCTS HEREIN WITHOUT FURTHER NOTICE. AVANT! MAKES NO WARRANTY, REPRESENTATION, OR GUARANTEE REGARDING THE FITNESS OF ITS PRODUCTS FOR ANY PARTICULAR PURPOSE AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY AND ANY WARRANTY OF NON-INFRINGEMENT. AVANT! DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT, AND SPECIFICALLY DISCLAIMS ANY AND ALL LIABILITY, INCLUDING WITHOUT LIMITATION, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES. AVANT!'S LIABILITY ARISING OUT OF THE MANUFACTURE, SALE OR SUPPLYING OF THE PRODUCTS OR THEIR USE OR DISPOSITION, WHETHER BASED UPON WARRANTY, CONTRACT, TORT OR OTHERWISE, SHALL NOT EXCEED THE ACTUAL LICENSE FEE PAID BY CUSTOMER.

Proprietary Rights Notice

This document contains information of a proprietary nature. No part of this manual may be copied or distributed without the prior written consent of *Avant!* corporation. This document and the software described herein is only provided under a written license agreement or a type of written non-disclosure agreement with *Avant!* corporation or its subsidiaries. ALL INFORMATION CONTAINED HEREIN SHALL BE KEPT IN CONFIDENCE AND USED STRICTLY IN ACCORDANCE WITH THE TERMS OF THE WRITTEN NON-DISCLOSURE AGREEMENT OR WRITTEN LICENSE AGREEMENT WITH AVANT! CORPORATION OR ITS SUBSIDIARIES.

Trademark/Service-Mark Notice

Apollo, Apollo-GA, Aurora, ASIC Synthesizer, AvanTestchip, AvanWaves, ChipPlanner, Columbia, Columbia-CE, Cyclelink, Davinci, DFM Workbench, Discovery, Driveline, Dynamic Model Switcher, Electrically Aware, Evaccess, Hercules, Hercules-Explorer, HotPlace, HSPICE, HSPICE-LINK, LTL, Libra-Passport, Lynx, Lynx-LB, Lynx-VHDL, Mars, Mars-Rail, Mars-Xtalk, MASTER Toolbox, Medici, Michelangelo, Milkyway, Optimum Silicon, Passport, Pathfinder, Planet, Planet-PL, Planet-RTL, Polaris, Polaris-CBS, Polaris-MT, Progen, Prospector, Raphael, Raphael-NES, Saturn, Sirius, Silicon Blueprint, Smart Extraction, Solar, Solar II, Star, Star-Sim, Star-Hspice, Star-HspiceLink, Star-DC, Star-RC, Star-Power, Star-Time, Star-MTB, Star-XP, Taurus, Taurus-Device, Taurus Layout, Taurus-Lithography, Taurus-OPC, Taurus-Process, Taurus-Topography, Taurus Visual, Taurus Workbench, TimeSlice, and TSUPREM-4 are trademarks of *Avant!* Corporation. *Avant!*, *Avant!* logo, AvanLabs, and avanticorp are trademarks and service-marks of *Avant!* Corporation. All other trademarks and service-marks are the property of their respective owners.

Subsidiaries

Anagram, Inc., ArcSys, Inc., Frontline Design Automation, Inc., Galax!, ISS, Inc., Meta-Software, Inc., NexSyn, Inc., and Technology Modeling Associates, Inc. are subsidiaries of *Avant!* Corporation.

Contacting *Avant!* Corporation:

| | |
|---|---------------------------|
| Telephone: (510) 413-8000 | <i>Avant!</i> Corporation |
| (800) 369-0080 | TCAD Business Unit |
| FAX: (510) 413-7743 | 46871 Bayside Parkway |
| e-mail tcad-app@avanticorp.com | Fremont, CA 94538 |
| Internet: http://www.avanticorp.com/ | |

Table of Contents

| | |
|--|-------------|
| List of Figures | xxix |
| <hr/> | |
| Introduction to TSUPREM-4 | xxxv |
| Program Overview | xxxv |
| Processing Steps | xxxv |
| Simulation Structure | xxxv |
| Additional Features | xxxvi |
| Manual Overview | xxxvi |
| Typeface Conventions | xxxvii |
| Related Publications | xxxvii |
| Reference Materials | xxxvii |
| Problems and Troubleshooting | xxxvii |
| <hr/> | |
| Chapter 1 Using TSUPREM-4 | 1-1 |
| Introduction | 1-1 |
| Program Execution and Output | 1-1 |
| Starting TSUPREM-4 | 1-1 |
| Program Output | 1-2 |
| Printed Output | 1-2 |
| Graphical Output | 1-2 |
| Errors, Warnings, and Syntax | 1-3 |
| File Specification | 1-3 |
| File Types | 1-3 |
| Default File Names | 1-3 |
| Environment Variables | 1-3 |
| Input Files | 1-4 |
| Command Input Files | 1-4 |
| Mask Data Files | 1-5 |

| | |
|--|------|
| Profile Files | 1-5 |
| Other Input Files | 1-5 |
| Output Files. | 1-5 |
| Terminal Output. | 1-5 |
| Output Listing Files | 1-6 |
| Standard Output File— <i>s4out</i> | 1-6 |
| Informational Output File— <i>s4inf</i> | 1-6 |
| Diagnostic Output File— <i>s4dia</i> | 1-6 |
| Saved Structure Files | 1-6 |
| TSUPREM-4. | 1-7 |
| TIF | 1-7 |
| Taurus-Lithography. | 1-7 |
| Medici | 1-7 |
| MINIMOS 5 | 1-7 |
| Wave. | 1-7 |
| Graphical Output | 1-8 |
| Extract Output Files | 1-8 |
| Electrical Data Output Files. | 1-8 |
| Library Files | 1-8 |
| Initialization Input File— <i>s4init</i> | 1-9 |
| Ion Implant Data File— <i>s4imp0</i> | 1-9 |
| Plot Device Definition File— <i>s4pcap</i> | 1-9 |
| Key Files— <i>s4fky0</i> and <i>s4uky0</i> | 1-10 |
| Authorization File— <i>s4auth</i> | 1-10 |

Chapter 2 TSUPREM-4 Models 2-1

| | |
|--|-----|
| Introduction. | 2-1 |
| Simulation Structure | 2-1 |
| Coordinates | 2-1 |
| Initial Structure | 2-2 |
| Regions and Materials | 2-2 |
| Grid Structure | 2-2 |
| Mesh, Triangular Elements, and Nodes. | 2-2 |
| Defining Grid Structure | 2-3 |
| Explicit Specification of Grid Structure. | 2-3 |
| The LINE Statement. | 2-3 |
| Generated Grid Lines | 2-3 |
| Eliminating Grid Lines. | 2-4 |
| Automatic Grid Generation | 2-5 |
| Automatic Grid Generation in the X Direction | 2-5 |
| X Grid from WIDTH Parameter | 2-5 |
| X Grid from MASK Statement | 2-5 |
| Column Elimination | 2-6 |
| Automatic Grid Generation in the Y Direction | 2-6 |

Changes to the Mesh During Processing 2-7

DEPOSITION and **EPITAXY** 2-7

 Structure Extension 2-7

ETCH and **DEVELOP** 2-8

 Oxidation and Silicidation 2-8

 Removal of Nodes in Consumed Silicon 2-8

 Addition of Nodes in a Growing Layer 2-8

 Nodes in Regions Where Oxide is Deforming 2-9

 Numerical Integrity 2-9

Adaptive Gridding 2-10

 Refinement 2-10

 Unrefinement 2-10

 Usage 2-11

One-Dimensional Simulation of Simple Structures 2-12

Initial Impurity Concentration 2-12

Diffusion 2-13

DIFFUSION Statement 2-13

 Temperature 2-14

 Ambient Gas Pressure 2-14

 Ambient Gas Characteristics 2-14

 Ambients and Oxidation of Materials 2-14

 Default Ambients 2-15

 Chlorine 2-15

 Example 2-16

 Coefficient Tables 2-16

 Chemical Predeposition 2-16

 Solution of Diffusion Equations 2-16

Diffusion of Impurities 2-17

 Impurity Fluxes 2-17

 Modeling of Diffusivities 2-18

 Mobile Impurities and Ion Pairing 2-18

 Electric Field 2-19

 Diffusivities 2-20

 Polysilicon Enhancement 2-21

 Point Defect Enhancement 2-21

PD.FERMI Model 2-21

PD.TRANS and **PD.FULL** Models 2-21

 Paired Fractions of Dopant Atoms 2-22

 Reaction Rate Constants 2-23

Activation of Impurities 2-24

 Physical Mechanisms 2-24

 Activation Models 2-25

 Model Details 2-25

 Solid Solubility Model 2-26

 Solid Solubility Tables 2-26

 Dopant Clustering Model 2-27

 Dopant-Defect Clustering Model 2-27

| | |
|---|------|
| Segregation of Impurities..... | 2-28 |
| Segregation Flux..... | 2-28 |
| Transport Coefficient..... | 2-28 |
| Segregation Coefficient..... | 2-29 |
| Moving-Boundary Flux..... | 2-29 |
| Interface Trap Model..... | 2-29 |
| Using the Interface Trap Model..... | 2-32 |
| Diffusion of Point Defects..... | 2-32 |
| Equilibrium Concentrations..... | 2-32 |
| Charge State Fractions..... | 2-33 |
| Point Defect Diffusion Equations..... | 2-33 |
| Interstitial and Vacancy Diffusivities..... | 2-34 |
| Reaction of Pairs with Point Defects..... | 2-35 |
| Recombination of Interstitials with Vacancies..... | 2-35 |
| Absorption by Traps, Clusters, and Dislocation Loops..... | 2-37 |
| Injection and Recombination of Point Defects at Interfaces..... | 2-37 |
| Surface Recombination Velocity Models..... | 2-38 |
| V.MAXOX Model..... | 2-38 |
| V.INITOX Model..... | 2-39 |
| V.NORM Model..... | 2-39 |
| Trapped Nitrogen Dependent Surface Recombination..... | 2-39 |
| Injection Rate..... | 2-39 |
| Moving-Boundary Flux..... | 2-41 |
| Interstitial Traps..... | 2-41 |
| Enabling, Disabling, and Initialization..... | 2-42 |
| Interstitial Clustering Model..... | 2-42 |
| Model Equations..... | 2-42 |
| Choosing Model Parameters..... | 2-43 |
| Using the Model..... | 2-44 |
| Small Clusters of Point Defects..... | 2-44 |
| Concentration of Defects in Small Clusters..... | 2-44 |
| Recombination of Defects in Small Clusters..... | 2-45 |
| Oxidation..... | 2-46 |
| Theory of Oxidation..... | 2-46 |
| Analytical Oxidation Models..... | 2-48 |
| Overview..... | 2-48 |
| Oxide Growth Rate..... | 2-48 |
| Thin Regime..... | 2-48 |
| Linear Rate..... | 2-48 |
| Parabolic Rate..... | 2-49 |
| Usage..... | 2-50 |
| The ERFC Model..... | 2-51 |
| Recommended Usage..... | 2-51 |
| The ERF1 , ERF2 , and ERFG Models..... | 2-51 |
| Parameters..... | 2-51 |
| Initial Structure..... | 2-51 |
| ERF1 Model..... | 2-52 |

| | |
|---|------|
| ERF2 Model | 2-52 |
| ERFG Model | 2-53 |
| Recommended Usage | 2-53 |
| Numerical Oxidation Models | 2-54 |
| Oxide Growth Rate | 2-54 |
| Concentration Dependence | 2-55 |
| Thin Regime | 2-56 |
| Usage | 2-56 |
| The VERTICAL Model | 2-56 |
| Recommended Usage | 2-56 |
| COMPRESS Model | 2-57 |
| Compressible Viscous Flow | 2-57 |
| Boundary Conditions | 2-58 |
| Model Parameters | 2-58 |
| COMPRESS Model: Recommended Usage | 2-58 |
| VISCOUS Model | 2-58 |
| Incompressible Viscous Flow | 2-58 |
| Stress Dependence | 2-59 |
| Recommended Usage | 2-60 |
| VISCOELA Model | 2-60 |
| Viscoelastic Flow | 2-60 |
| Model Parameters | 2-62 |
| Recommended Usage | 2-62 |
| Polysilicon Oxidation | 2-63 |
| Surface Tension and Reflow | 2-63 |
| N ₂ O Oxidation | 2-63 |
| Nitrogen Trap and Generation | 2-63 |
| Surface Reaction Rate | 2-64 |
| Thin Oxidation Rate | 2-64 |
| Boron Diffusion Enhancement in Oxides | 2-65 |
| Diffusion Enhancement in Thin Oxides | 2-65 |
| Diffusion Enhancement Due to Fluorine | 2-67 |
| Silicide Models | 2-68 |
| TiSi ₂ Growth Kinetics | 2-68 |
| Reaction at TiSi ₂ /Si Interface | 2-68 |
| Diffusion of Silicon | 2-68 |
| Reaction at TiSi ₂ /Si Interface | 2-69 |
| Initialization | 2-69 |
| Material Flow | 2-69 |
| Impurities and Point Defects | 2-69 |
| Specifying Silicide Models and Parameters | 2-69 |
| Materials | 2-70 |
| Impurities | 2-70 |
| Reactions | 2-70 |
| Impurities | 2-72 |
| Tungsten and Cobalt Silicide Models | 2-72 |
| Other Silicides | 2-72 |

| | |
|--|------|
| Stress Models | 2-73 |
| Stress History Model | 2-73 |
| Thermal Stress Model Equations | 2-73 |
| Boundary Conditions | 2-73 |
| Initial Conditions | 2-73 |
| Intrinsic Stress in Deposited Layers | 2-74 |
| Effect of Etching on Stress | 2-74 |
| Using the Stress History Model | 2-74 |
| Limitations | 2-75 |
| Modeling Stress with the STRESS Statement | 2-75 |
| Boundary Conditions | 2-76 |
| Ion Implantation | 2-77 |
| Analytic Ion Implant Models | 2-77 |
| Implanted Impurity Distributions | 2-77 |
| Implant Moment Tables | 2-78 |
| Gaussian Distribution | 2-80 |
| Pearson Distribution | 2-80 |
| Dual Pearson Distribution | 2-81 |
| Dose-dependent Implant Profiles | 2-81 |
| Tilt and Rotation Tables | 2-84 |
| Multilayer Implants | 2-84 |
| Effective Range Model | 2-84 |
| Dose Matching | 2-84 |
| Lateral Distribution | 2-85 |
| Depth-Dependent Lateral Distribution | 2-85 |
| Wafer Tilt and Rotation | 2-85 |
| BF ₂ Implant | 2-86 |
| Analytic Damage Model | 2-86 |
| Damage Distribution Calculations | 2-86 |
| Recommended Usage and Limitations | 2-87 |
| Monte Carlo Ion Implant Model | 2-87 |
| Binary Scattering Theory | 2-87 |
| Energy Loss | 2-88 |
| Scattering Angle | 2-88 |
| Dimensionless Form | 2-88 |
| Coulomb Potential | 2-89 |
| Universal Potential | 2-90 |
| Amorphous Implant Calculation | 2-91 |
| Nuclear Stopping | 2-92 |
| Electronic Stopping | 2-92 |
| Electronic Stopping at High Energies | 2-93 |
| Total Energy Loss and Ion Deflection | 2-93 |
| Ion Beam Width | 2-94 |
| Crystalline Implant Model | 2-94 |
| Channeling | 2-94 |
| Lattice Temperature | 2-96 |
| Lattice Damage | 2-96 |

| | |
|---|-------|
| Damage Dechanneling | 2-96 |
| Damage Annealing | 2-97 |
| Number of Ions | 2-97 |
| BF ₂ Implantation. | 2-97 |
| Implant Damage Model | 2-98 |
| Damage Produced During Implant. | 2-98 |
| +1 and Effective +n Damage Models | 2-99 |
| Cumulative Damage Model | 2-100 |
| Old Damage Model | 2-101 |
| Conservation of Total Defect Concentrations | 2-102 |
| Using the Implant Damage Model | 2-102 |
| Boundary Conditions for Ion Implantation | 2-103 |
| Polysilicon Monte Carlo Implant Model | 2-103 |
| Ion Implantation into Silicon Carbide | 2-104 |
| Epitaxial Growth. | 2-104 |
| Layer Thickness | 2-105 |
| Incorporation of Impurities | 2-105 |
| Diffusion of Impurities. | 2-105 |
| Deposition. | 2-105 |
| Layer Thickness | 2-106 |
| Incorporation of Impurities | 2-106 |
| Photoresist Type. | 2-106 |
| Polycrystalline Materials | 2-106 |
| Deposition with Taurus-Topography | 2-106 |
| Masking, Exposure, and Development of Photoresist | 2-107 |
| Etching | 2-107 |
| Defining the Etch Region. | 2-107 |
| Removal of Material | 2-108 |
| Trapezoidal Etch Model. | 2-108 |
| Parameters | 2-109 |
| Etch Steps | 2-109 |
| Etch Examples | 2-109 |
| Simple Structure | 2-109 |
| Structure with Overhangs | 2-110 |
| Complex Structures. | 2-111 |
| Etching with Taurus-Topography. | 2-111 |
| Modeling Polycrystalline Materials | 2-112 |
| Diffusion | 2-112 |
| Diffusion in Grain Interiors | 2-112 |
| Grain Boundary Structure. | 2-113 |
| Diffusion Along Grain Boundaries | 2-113 |
| Anisotropic Diffusion | 2-113 |
| Segregation Between Grain Interior and Boundaries | 2-114 |
| Grain Size Model | 2-115 |
| Initial Grain Size. | 2-115 |
| Grain Growth | 2-116 |
| Concentration Dependence. | 2-116 |

| | |
|---|-------|
| Grain Surface Energy | 2-117 |
| Segregation Drag | 2-117 |
| Interface Oxide Break-up and Epitaxial Regrowth | 2-117 |
| Oxide Break-Up | 2-117 |
| Epitaxial Regrowth | 2-118 |
| Using the Polycrystalline Model | 2-119 |
| Electrical Calculations | 2-120 |
| Automatic Regrid | 2-120 |
| Poisson's Equation | 2-120 |
| Boltzmann and Fermi-Dirac Statistics | 2-121 |
| Ionization of Impurities | 2-121 |
| Solution Methods | 2-122 |
| Carrier Mobility | 2-123 |
| Tabular Form | 2-123 |
| Arora Mobility Model | 2-123 |
| Caughey Mobility Model | 2-124 |
| Quantum Mechanical Model for MOSFET | 2-125 |
| Capacitance Calculation | 2-125 |
| DC Method | 2-126 |
| MOS Capacitances | 2-126 |
| Extended Defects AAM | 2-126 |
| Dislocation Loop Model | 2-126 |
| Equations for Dislocation Loop Model | 2-127 |
| Loop Density Specified by L.DENS | 2-128 |
| Loop Density Specified by L.THRESH | 2-128 |
| Evolution of Loops | 2-128 |
| Effects of Dislocation Loops | 2-129 |
| Transient Clustering Model | 2-129 |
| References | 2-131 |

Chapter 3 Input Statement Descriptions 3-1

| | |
|---|-----|
| Introduction | 3-1 |
| Input Statements | 3-1 |
| Format | 3-2 |
| Syntax | 3-2 |
| Specifying Materials and Impurities | 3-2 |
| Parameters | 3-3 |
| Logical | 3-3 |
| Numerical | 3-3 |
| Character | 3-3 |
| Statement Description Format | 3-4 |
| Parameter Definition Table | 3-4 |
| Syntax of Parameter Lists | 3-4 |
| Parameter Types < > | 3-4 |

Parameter Groups 3-5

Optional Parameters []. 3-5

Choices { }, |. 3-5

Group Hierarchy () 3-5

3.1 Documentation and Control. 3-7

COMMENT 3-8

 Description 3-8

 Examples 3-8

 Notes 3-8

SOURCE 3-9

 Description 3-9

 Reusing Combinations of Statements 3-9

 Generating Templates 3-9

 Examples 3-9

RETURN 3-10

 Description 3-10

 Returning from Batch Mode 3-10

 Exiting Interactive Input Mode 3-10

 Example 3-11

INTERACTIVE 3-12

 Description 3-12

 Interactive Input Mode 3-12

 Example 3-13

PAUSE 3-14

 Description 3-14

 Example 3-14

STOP 3-15

 Description 3-15

 Example 3-15

FOREACH/END 3-16

 Description 3-16

 Examples 3-16

 Notes 3-17

LOOP/L.END. 3-18

 Description 3-19

 Termination of Optimization Looping 3-19

 Parameter Sensitivity 3-19

 Dependence and Variability 3-20

 Example 3-20

 Advantages 3-21

L.MODIFY 3-22

 Description 3-22

IF/ELSEIF/ELSE/IF.END 3-23

 Description 3-23

 Conditional Operators 3-24

 Expression for Condition 3-24

ASSIGN 3-25

| | |
|---|-------------|
| Description | 3-28 |
| Varying During Statement Looping | 3-29 |
| ASSIGN with Mathematical Expressions | 3-29 |
| ASSIGN and Optimization | 3-30 |
| Expansion of ASSIGNED Variable | 3-30 |
| Reading the External Data File | 3-31 |
| Reading the Array from a String | 3-31 |
| INTERMEDIATE | 3-32 |
| Advantages of INTERMEDIATES | 3-32 |
| Value Type | 3-33 |
| Array Values | 3-34 |
| Getting Array Values | 3-35 |
| Redefinition of Array Elements | 3-35 |
| Effect Due to Redefinition | 3-35 |
| Comparison to ASSIGN | 3-36 |
| ECHO | 3-38 |
| Description | 3-38 |
| Examples | 3-38 |
| OPTION | 3-39 |
| Selecting a Graphics Device | 3-40 |
| Redirecting Graphics Output | 3-40 |
| Printed Output | 3-40 |
| Informational and Diagnostic Output | 3-40 |
| Echoing and Execution of Input Statements | 3-40 |
| Version Compatibility | 3-41 |
| Examples | 3-41 |
| DEFINE | 3-42 |
| Description | 3-42 |
| Format and Syntax | 3-42 |
| Examples | 3-42 |
| Usage Notes | 3-43 |
| UNDEFINE | 3-45 |
| Description | 3-45 |
| Example | 3-45 |
| Redefined Parameter Names | 3-45 |
| CPULOG | 3-46 |
| Description | 3-46 |
| Examples | 3-46 |
| Limitations | 3-46 |
| HELP | 3-47 |
| Description | 3-47 |
| Example | 3-47 |
| Notes | 3-47 |
| 3.2 Device Structure Specification | 3-49 |
| MESH | 3-50 |
| Description | 3-51 |
| Grid Creation Methods | 3-51 |

Horizontal Grid Generation 3-52

Vertical Grid Generation 3-52

Scaling the Grid Spacing 3-53

1D Mode 3-53

Examples 3-54

LINE 3-55

 Description 3-55

 Placing Grid Lines 3-55

 Example 3-56

 Additional Notes 3-56

 Structure Depth and Point Defect Models 3-56

 Maximum Number of Nodes and Grid Lines 3-56

 Default Regions and Boundaries 3-56

ELIMINATE 3-57

 Description 3-57

 Reducing Grid Nodes 3-58

 Overlapping Regions 3-58

 Examples 3-58

BOUNDARY 3-60

 Description 3-60

 Limitations 3-61

 Example 3-61

REGION 3-62

 Description 3-63

 Example 3-63

INITIALIZE 3-64

 Description 3-66

 Mesh Generation 3-66

 Previously Saved Structure Files 3-66

 Crystalline Orientation 3-66

 Specifying Initial Doping 3-67

 Examples 3-67

LOADFILE 3-68

 Description 3-68

 TSUPREM-4 Files 3-68

 Older Versions 3-69

 User-Defined Materials and Impurities 3-69

 Taurus-Lithography Files 3-69

 Examples 3-69

SAVEFILE 3-71

 Description 3-74

 TSUPREM-4 Files 3-74

 Older Versions 3-74

 TIF Files 3-74

 Medici Files 3-75

 Taurus-Lithography Files 3-75

MINIMOS 3-76

| | |
|--|-------------|
| Temperature | 3-76 |
| Examples | 3-76 |
| STRUCTURE | 3-77 |
| Description | 3-79 |
| Order of Operations | 3-79 |
| Usage | 3-80 |
| TSUPREM-4 Version Compatibility | 3-80 |
| Examples | 3-80 |
| MASK | 3-82 |
| Description | 3-82 |
| Examples | 3-83 |
| PROFILE | 3-84 |
| Description | 3-84 |
| OFFSET Parameter | 3-85 |
| Interpolation | 3-85 |
| IMPURITY Parameter | 3-86 |
| Example | 3-86 |
| ELECTRODE | 3-87 |
| Description | 3-87 |
| Examples | 3-88 |
| Additional ELECTRODE Notes | 3-88 |
| 3.3 Process Steps | 3-89 |
| DEPOSITION | 3-90 |
| Description | 3-92 |
| Polycrystalline Materials | 3-93 |
| Photoresist | 3-93 |
| Deposition with Taurus-Topography | 3-93 |
| Examples | 3-94 |
| Additional DEPOSITION Notes | 3-94 |
| EXPOSE | 3-95 |
| Description | 3-95 |
| Example | 3-96 |
| DEVELOP | 3-97 |
| Description | 3-97 |
| Example | 3-97 |
| ETCH | 3-98 |
| Description | 3-100 |
| Removing Regions | 3-100 |
| Etching with Taurus-Topography | 3-101 |
| Examples | 3-102 |
| IMPLANT | 3-103 |
| Description | 3-108 |
| Gaussian and Pearson Distributions | 3-109 |
| Table of Range Statistics | 3-109 |
| Monte Carlo Implant Model | 3-110 |
| Point Defect Generation | 3-111 |
| Extended Defects | 3-111 |

Channeling Effects 3-111

Boundary Conditions 3-111

TSUPREM-4 Version Considerations 3-112

Examples 3-112

DIFFUSION 3-115

 Description 3-118

 Ambient Gas 3-118

 Ambient Gas Parameters 3-118

 Oxidation Limitations 3-119

 Reflow 3-119

 Examples 3-120

EPITAXY 3-121

 Description 3-123

 Example 3-123

STRESS 3-124

 Description 3-124

 Printing and Plotting of Stresses and Displacements 3-124

 Reflecting Boundary Limitations 3-125

 Example 3-125

3.4 Output **3-127**

SELECT 3-128

 Description 3-128

 Solution Values 3-128

 Mathematical Operations and Functions 3-130

 Examples 3-131

PRINT .1D 3-132

 Description 3-134

 Layers 3-134

 Interface Values 3-134

 Saving Profiles in a File 3-134

 Examples 3-135

PLOT .1D 3-136

 Description 3-141

 Line Type and Color 3-142

IN.FILE Parameter 3-142

 Examples 3-143

PLOT .2D 3-144

 Description 3-146

 Line Type and Color 3-147

 Examples 3-147

CONTOUR 3-148

 Description 3-148

 Line Type and Color 3-149

 Example 3-149

 Additional **CONTOUR** Notes 3-149

COLOR 3-150

 Description 3-151

| | |
|--|--------------|
| Plot Device Selection | 3-151 |
| Examples | 3-151 |
| PLOT . 3D | 3-152 |
| Description | 3-153 |
| Line Type and Color | 3-154 |
| Examples | 3-154 |
| Additional PLOT . 3D Notes | 3-154 |
| LABEL | 3-155 |
| Description | 3-158 |
| Label Placement | 3-158 |
| Line, Symbol, and Rectangle | 3-158 |
| Color | 3-159 |
| Examples | 3-159 |
| EXTRACT | 3-160 |
| Description | 3-165 |
| Solution Variables | 3-166 |
| Extraction Procedure | 3-166 |
| Targets for Optimization | 3-168 |
| File Formats | 3-168 |
| Error Calculation | 3-168 |
| Examples | 3-169 |
| Optimization Examples | 3-171 |
| ELECTRICAL | 3-174 |
| Description | 3-179 |
| Files and Plotting | 3-179 |
| Examples | 3-180 |
| Optimization Examples | 3-181 |
| Quantum Effect in CV Plot | 3-181 |
| Additional ELECTRICAL Notes | 3-183 |
| VIEWPORT | 3-184 |
| Description | 3-184 |
| Scaling Plot Size | 3-184 |
| Examples | 3-185 |
| 3.5 Models and Coefficients | 3-187 |
| METHOD | 3-188 |
| Description | 3-197 |
| Oxidation Models | 3-197 |
| Grid Spacing in Growing Oxide | 3-197 |
| Rigid vs. Viscous Substrate | 3-198 |
| Point Defect Modeling | 3-198 |
| PD . FERMI Model | 3-198 |
| PD . TRANS Model | 3-199 |
| PD . FULL Model | 3-199 |
| Customizing the Point Defect Models | 3-199 |
| Enable/Disable User-Specified Models | 3-199 |
| Adaptive Gridding | 3-200 |
| Fine Control | 3-201 |

| | |
|--|-------|
| Initial Time Step | 3-201 |
| Internal Solution Methods | 3-201 |
| Time Integration | 3-201 |
| System Solutions | 3-202 |
| Minimum-Fill Reordering | 3-202 |
| Block Solution | 3-202 |
| Solution Method | 3-202 |
| Matrix Structure | 3-202 |
| Matrix Refactoring | 3-203 |
| Error Tolerances | 3-203 |
| Examples | 3-203 |
| EQUATION | 3-204 |
| Description | 3-205 |
| Additional Information | 3-206 |
| AMBIENT | 3-207 |
| Description | 3-217 |
| Oxidation Models | 3-217 |
| ERFC Model | 3-218 |
| ERFG Model | 3-218 |
| VERTICAL Model | 3-218 |
| COMPRESS Model | 3-218 |
| VISCOELA Model | 3-218 |
| VISCOUS Model | 3-219 |
| Stress Dependence | 3-219 |
| Coefficients | 3-220 |
| Chlorine | 3-220 |
| Examples | 3-220 |
| Parameter Dependencies | 3-221 |
| Orientation | 3-221 |
| Oxidizing Species | 3-221 |
| Specified Material | 3-222 |
| Specified Units | 3-222 |
| Examples | 3-222 |
| Additional AMBIENT Notes | 3-222 |
| MOMENT | 3-224 |
| Description | 3-226 |
| Optional and Required Model Parameters | 3-226 |
| Using the MOMENT Statement | 3-226 |
| Examples | 3-227 |
| Additional Note | 3-227 |
| MATERIAL | 3-228 |
| Description | 3-236 |
| Viscosity and Compressibility | 3-237 |
| Stress Dependence | 3-237 |
| Examples | 3-237 |
| IMPURITY | 3-239 |
| Description | 3-251 |

| | |
|--|--------------|
| Impurity Type | 3-251 |
| Solution Options | 3-251 |
| Other Parameters | 3-251 |
| Multiplication to Diffusivity | 3-252 |
| Further Reading | 3-252 |
| Examples | 3-252 |
| REACTION | 3-254 |
| Description | 3-256 |
| Defining and Deleting | 3-256 |
| Insertion of Native Layers | 3-256 |
| Reaction Equation | 3-257 |
| Parameters | 3-257 |
| Effects | 3-258 |
| MOBILITY | 3-259 |
| Description | 3-262 |
| Tables and Analytic Models | 3-262 |
| Analytic Models | 3-262 |
| Tables or Model Selection | 3-263 |
| Example | 3-263 |
| INTERSTITIAL | 3-265 |
| Description | 3-274 |
| Bulk and Interface Parameters | 3-274 |
| Examples | 3-275 |
| Additional INTERSTITIAL Notes | 3-275 |
| VACANCY | 3-276 |
| Description | 3-282 |
| Bulk and Interface Parameters | 3-282 |
| Examples | 3-283 |
| Additional VACANCY Notes | 3-283 |
| ANTIMONY | 3-284 |
| Description | 3-288 |
| Examples | 3-289 |
| Additional ANTIMONY Notes | 3-289 |
| ARSENIC | 3-290 |
| Description | 3-294 |
| Examples | 3-295 |
| Additional ARSENIC Notes | 3-295 |
| BORON | 3-296 |
| Description | 3-300 |
| Examples | 3-301 |
| Additional BORON Notes | 3-301 |
| PHOSPHORUS | 3-302 |
| Description | 3-306 |
| Examples | 3-307 |
| Additional PHOSPHORUS Notes | 3-307 |
| 3.6 Summary | 3-309 |

| | | |
|--|--------------------------|------------|
| Chapter 4 | Tutorial Examples | 4-1 |
| Overview | | 4-1 |
| Input File Syntax and Format | | 4-1 |
| One-Dimensional Bipolar Example | | 4-2 |
| TSUPREM-4 Input File Sequence | | 4-2 |
| Initial Active Region Simulation | | 4-2 |
| Mesh Generation | | 4-2 |
| Automatic Mesh Generation | | 4-3 |
| Adaptive Gridding | | 4-4 |
| Model Selection | | 4-4 |
| Oxidation Model | | 4-4 |
| Point Defect Model | | 4-4 |
| Processing Steps | | 4-5 |
| Buried Layer Masking Oxide | | 4-5 |
| Buried Layer | | 4-5 |
| Epitaxial Layer | | 4-5 |
| Pad Oxide and Nitride Mask | | 4-6 |
| Saving the Structure | | 4-6 |
| Plotting the Results | | 4-6 |
| Specifying a Graphics Device | | 4-6 |
| The SELECT Statement | | 4-6 |
| The PLOT.1D Statement | | 4-7 |
| Labels | | 4-7 |
| Printing Layer Information | | 4-8 |
| The PRINT.1D Statement | | 4-8 |
| Using PRINT.1D LAYERS | | 4-8 |
| Completing the Active Region Simulation | | 4-9 |
| Reading a Saved Structure | | 4-9 |
| Field Oxidation | | 4-9 |
| Final Structure | | 4-11 |
| Local Oxidation | | 4-12 |
| Calculation of Oxide Shape | | 4-12 |
| Mesh Generation | | 4-13 |
| Pad Oxide and Nitride Layers | | 4-14 |
| Plotting the Mesh | | 4-14 |
| Model Selection | | 4-14 |
| Plotting the Results | | 4-15 |
| Plotting Stresses | | 4-16 |
| Contour Plots | | 4-17 |
| Two-Dimensional Diffusion with Point Defects | | 4-18 |
| Automatic Grid Generation | | 4-19 |
| Field Implant | | 4-19 |
| Oxidation | | 4-19 |
| Grid Plot | | 4-20 |
| Contour of Boron Concentration | | 4-20 |

| | |
|---|------|
| Using the FOREACH Statement | 4-23 |
| Vertical Distribution of Point Defects | 4-24 |
| Lateral Distribution of Point Defects | 4-25 |
| Shaded Contours of Interstitial Concentration | 4-25 |
| Local Oxidation Summation | 4-26 |
| Point Defect Models | 4-27 |
| Creating the Test Structure | 4-29 |
| Automatic Grid Generation | 4-29 |
| Outline of Example. | 4-29 |
| Oxidation and Plotting of Impurity Profiles | 4-29 |
| Simulation Procedure | 4-29 |
| PD.FERMI and PD.TRANS Models | 4-30 |
| PD.FULL Model | 4-30 |
| Printing Junction Depth | 4-31 |
| Doping and Layer Information. | 4-31 |
| Point Defect Profiles | 4-31 |
| Commentary. | 4-32 |
| Choosing a Point Defect Model | 4-32 |

Chapter 5 Advanced Examples **5-1**

| | |
|---|------|
| Overview. | 5-1 |
| NMOS LDD Process. | 5-2 |
| Creating the Initial Structure | 5-2 |
| Setting the Grid Density | 5-3 |
| Adaptive Gridding | 5-4 |
| Masking Information | 5-4 |
| Field Isolation Simulation | 5-4 |
| Displaying the Plot | 5-5 |
| Active Region Simulation | 5-7 |
| Modeling Polysilicon | 5-7 |
| LDD Implant | 5-8 |
| Oxide Spacer and Source/Drain Implant | 5-9 |
| Source/Drain Contacts | 5-9 |
| Plots | 5-11 |
| Formation of the Complete NMOS Transistor | 5-12 |
| Electrical Extraction. | 5-14 |
| Threshold Voltage | 5-14 |
| MOS Capacitance. | 5-15 |
| Source/Drain Junction Capacitance | 5-16 |
| Plotting Results of Electrical Extraction | 5-17 |
| Trench Implant Simulation | 5-18 |
| Structure Generation | 5-18 |
| Analytic Implant | 5-19 |
| Plotting the Results of the Analytic Method | 5-20 |

| | |
|---|------|
| Monte Carlo Implant | 5-23 |
| Overview | 5-23 |
| Using the Monte Carlo Model | 5-23 |
| Plotting the Results of the Monte Carlo Method | 5-23 |
| Boron Contours | 5-23 |
| Vertical Profiles | 5-24 |
| Sidewall Profiles. | 5-26 |
| Summary | 5-26 |
| Poly-Buffered LOCOS | 5-27 |
| Structure Generation | 5-27 |
| Using the VISCOEL Model | 5-28 |
| Plotting the Results | 5-28 |
| CMOS Process | 5-30 |
| Main Loop | 5-32 |
| Mesh Generation | 5-32 |
| CMOS Processing | 5-34 |
| Models | 5-34 |
| Channel Doping Plot | 5-34 |
| Lightly Doped Drain Structure. | 5-34 |
| Contacts | 5-34 |
| Saving the Structure | 5-35 |
| End of Main Loop | 5-35 |
| Plotting the Results | 5-35 |
| 0.8 Micron Device | 5-37 |
| Final Mesh | 5-37 |
| Arsenic Profiles in Gate | 5-37 |
| 1.2 Micron Device | 5-38 |
| DMOS Power Transistor. | 5-39 |
| Mesh Generation | 5-40 |
| Processing the DMOS Power Transistor | 5-40 |
| Gate Processing | 5-42 |
| Source Processing. | 5-42 |
| Summary | 5-44 |
| SOI MOSFET | 5-45 |
| Mesh Generation | 5-45 |
| Depositing a Layer with Nonuniform Grid Spacing | 5-46 |
| Process Simulation. | 5-48 |
| MOSFET with Self-Aligned Silicides | 5-50 |
| Preparation for Silicidation | 5-50 |
| Silicidation | 5-51 |
| Polysilicon Emitter Study | 5-54 |
| Process Simulation. | 5-54 |
| Models | 5-54 |
| Processing. | 5-54 |
| Plotting the Results | 5-56 |
| After Implant | 5-56 |
| Doping and Grain Size | 5-57 |

| | |
|------------------------------|------|
| Doping vs. Stripe Width..... | 5-58 |
|------------------------------|------|

| | |
|--|-------------|
| Chapter 6 User-Specified Equation Interface-USEIT | 6-1 |
| Overview..... | 6-1 |
| 6.1 Physical Model and Equation Interface | 6-3 |
| Syntax..... | 6-3 |
| Flat Structure Syntax..... | 6-5 |
| PMEI Commands..... | 6-6 |
| Equation | 6-6 |
| Model (alias INTERMEDIATE)..... | 6-7 |
| Expression | 6-8 |
| Parameter | 6-9 |
| Array | 6-9 |
| Loop | 6-10 |
| InitialCondition | 6-11 |
| BoundaryCondition | 6-11 |
| InterfaceCondition | 6-12 |
| ImplantModel | 6-13 |
| Selector | 6-14 |
| Functions and Operators..... | 6-14 |
| PMEI Functions..... | 6-15 |
| Sum | 6-15 |
| Condition | 6-16 |
| Exist | 6-16 |
| OnMat1 and OnMat2 | 6-16 |
| Proposed Language Additions..... | 6-16 |
| Parameterized Functions..... | 6-16 |
| Physical Functions..... | 6-17 |
| 6.2 Using USEIT | 6-19 |
| EQUATION | 6-20 |
| New Solution Variable..... | 6-20 |
| Solving Method..... | 6-20 |
| Initialization..... | 6-20 |
| Adding Expressions to Equations..... | 6-21 |
| Flux at Interfaces..... | 6-22 |
| Diffusion Along Boundaries..... | 6-23 |
| Update Solution..... | 6-23 |
| INTERMEDIATE | 6-24 |
| Advantages of Intermediates..... | 6-24 |
| Application of User-Specified Models..... | 6-24 |
| Modification of Built-in Functions..... | 6-25 |
| ACTIVE | 6-25 |
| MOBILE | 6-26 |
| Built-in Keywords..... | 6-27 |

| | |
|---|-------------|
| Operators | 6-27 |
| Mathematical Functions | 6-28 |
| Physical Functions | 6-29 |
| Interface Functions | 6-31 |
| Miscellaneous Functions. | 6-31 |
| 6.3 Examples | 6-33 |
| Interstitial Clustering Model | 6-34 |
| Transient Clustering Model | 6-35 |
| Dislocation-Loop Model. | 6-36 |
| Diffusion in PD. TRANS | 6-38 |
| Interface Trap Model | 6-39 |
| Boron-Interstitial Clustering Model with Small Clusters | 6-41 |
| Diffusion Enhancement in Thinner Oxides | 6-45 |
| Surface Recombination Rate Depending On Nitrogen. | 6-47 |
| 5-Stream Diffusion Model | 6-48 |
| Bulk Equations | 6-48 |
| Boundary Conditions | 6-49 |
| Implementation. | 6-49 |
| Point defect parameter | 6-49 |
| Dopant parameter | 6-51 |
| Thermal parameter for point defects. | 6-51 |
| OED - Interstitial generation due to oxidation | 6-52 |
| Thermal parameter for the dopant. | 6-52 |
| Point defect solutions | 6-52 |
| Pair solutions. | 6-53 |
| Substitutional boron | 6-53 |
| Intermediates depending on point defects. | 6-53 |
| Kick-out and Frank-Turnbull reacions | 6-53 |
| Boron diffusion equation | 6-53 |
| Pair diffusion equations | 6-54 |
| Point defect equations. | 6-54 |
| Coupling with the small clusters | 6-55 |
| Interstitial small clusters - I2. | 6-55 |
| BI2 and B3I. | 6-56 |
| Modified initialization of interstitial. | 6-56 |
| Comparison of the initialization | 6-56 |
| SiGe Model | 6-58 |

| | |
|---|------------|
| Appendix A: Default Coefficients | A-1 |
| Default Coefficient Values | A-1 |
| Impurity Parameters. | A-2 |
| Impurity Diffusion Coefficients. | A-2 |
| Segregation and Transport Coefficients. | A-4 |
| Polysilicon Grain Segregation | A-6 |
| Clustering and Solid Solubility | A-7 |

| | |
|--|------|
| Point Defect Parameters | A-8 |
| Oxidation | A-10 |
| Diffusion Enhancement in Oxides | A-15 |
| Silicidation | A-15 |
| Analytical Implant | A-16 |
| Monte Carlo Implant | A-17 |
| Electrical Parameters | A-18 |
| Material Coefficients | A-21 |
| Numerical Methods | A-24 |
| Automatic Grid Generation Parameters | A-25 |
| Adaptive Grid Parameters | A-25 |
| Default Coefficient References | A-27 |
| Default Coefficient Notes | A-32 |

Appendix B: Graphics Devices B-1

| | |
|-------------------------------------|-----|
| Determining the Device | B-1 |
| Output | B-1 |
| Supported Devices | B-2 |
| X (Window) | B-2 |
| I/X | B-3 |
| X/BW | B-3 |
| POSTSCRIPT (PS,PS-P) | B-3 |
| L/POSTSCRIPT (PS-L) | B-3 |
| PS-INSERT | B-3 |
| C/POSTSCRIPT (PS-C,PS-CP) | B-3 |
| CL/POSTSCRIPT (PS-CL) | B-3 |
| C/PS-INSERT | B-3 |
| REPLOT | B-3 |
| HP2648 (2648) | B-4 |
| HP2623 (2623) | B-4 |
| TEK4100 (4100) | B-4 |
| TEKBW | B-4 |
| TEK4010 (4010) | B-4 |
| XTERM | B-4 |
| REGIS (VT240, VT241) | B-4 |
| HP7550 (7550) | B-4 |
| HP7550-P (7550-P) | B-4 |
| PRINTRONIX | B-4 |
| SELANAR | B-4 |
| SUN (SUNVIEW) | B-4 |
| I/SUN | B-5 |
| APOLLO | B-5 |
| I/APOLLO | B-5 |
| Unsupported Devices | B-5 |
| TEK4510 (4510) | B-6 |

| | |
|--------------------------------|-----|
| HPJET (THINKJET) | B-6 |
| HPDESK (HP2671G, HP2673) | B-6 |
| HPLP (LP2563) | B-6 |
| IMAGEN | B-6 |
| DITROFF | B-6 |
| TGPLOT | B-6 |
| TGPLOT-P | B-6 |
| The Default Device | B-6 |
| DEFAULT | B-6 |
| Modifying s4pcap | B-6 |
| PIXX and PIXY | B-7 |
| PUNX and PUNY | B-7 |
| FILE | B-7 |
| LIKE | B-7 |
| PEN and AREA | B-7 |
| BFSZ | B-7 |

Appendix C: Version 2000.2 Enhancements **C-1**

| | |
|--|------|
| Improved Physical Models | C-1 |
| Cobalt Silicide Model | C-1 |
| Improved Effective +n Implant Damage Model | C-1 |
| Dislocation Loops in Amorphized Regions | C-2 |
| Trapped Nitrogen Dependent Surface Recombination | C-2 |
| New Applications Using USEIT | C-3 |
| Five Stream Diffusion Model | C-3 |
| SiGe Model for Boron | C-4 |
| Carbon-Doped Silicon Model | C-5 |
| Improved Accuracy, Flexibility, and Ease of Use | C-5 |
| Mesh Unrefinement | C-5 |
| Automatic Unrefinement | C-5 |
| Manual Unrefinement | C-6 |
| Improved Mesh Refinement | C-6 |
| More Flexible MOMENT Statement | C-6 |
| Ion Implantation Enhancements | C-7 |
| Miscellaneous Improvements and Other Changes | C-8 |
| Error Corrections | C-8 |
| Compatibility with TSUPREM-4 Version 1999.4 | C-9 |
| Accuracy Issues | C-10 |
| Performance Issues | C-11 |

Appendix D: Format of Mask Data Files **D-1**

| | |
|--------------------|-----|
| Introduction | D-1 |
| Format | D-1 |

| | |
|---|------------|
| Appendix E: Data Format of Saved Structure Files | E-1 |
| Introduction | E-1 |
| TSUPREM-4 Structure File Format | E-1 |
| Current Version | E-2 |
| Program Identification | E-2 |
| Coordinates | E-2 |
| Edges | E-2 |
| Regions | E-2 |
| Triangles | E-3 |
| Model Parameters | E-3 |
| Solution Values | E-5 |
| Nodes | E-5 |
| Example | E-5 |
| Older Versions of TSUPREM-4 | E-6 |
| Regions | E-7 |
| Model Parameters | E-7 |
| Solution Values | E-7 |
| Nodes | E-8 |
| Medici Structure File Format | E-8 |
| Example | E-10 |

| | |
|--|------------|
| Appendix F: Using the MINIMOS 5 Interface | F-1 |
| Overview of the TSUPREM-4 Interface to MINIMOS 5 | F-1 |
| Step 1: Directing TSUPREM-4 to Generate a Formatted File | F-2 |
| Defining the MINIMOS 5 Simulation Region | F-2 |
| Notes on the Size of the MINIMOS 5 Simulation Region | F-5 |
| Nonplanar Oxide Regions in MINIMOS 5 | F-5 |
| Step 2: Converting the Formatted File to FORTRAN Binary | F-6 |
| Step 3: Running MINIMOS 5 | F-6 |
| Notes on Using MINIMOS 5 | F-8 |
| Interpreting Error Messages | F-8 |

| | |
|-----------------|-------------------|
| Glossary | Glossary-1 |
|-----------------|-------------------|

| | |
|--------------|----------------|
| Index | Index-1 |
|--------------|----------------|

List of Figures

| | |
|---|------------|
| TSUPREM-4 Models | 2-1 |
| Figure 2-1 BF2 implant profile. | 2-83 |
| Figure 2-2 Examples of the trapezoidal etch model. | 2-110 |

| | |
|--|------------|
| Input Statement Descriptions | 3-1 |
| Figure 3-1 Example of sensitivity plot for target with multiple data. | 3-21 |
| Figure 3-2 Quantum effect in MOS capacitance | 3-182 |
| Figure 3-3 USEIT working diagram | 3-205 |

| | |
|--|------------|
| Tutorial Examples. | 4-1 |
| Figure 4-1 Input file <i>s4ex1a.inp</i> , for simulating the buried layer and epitaxial deposition for a bipolar transistor structure. | 4-3 |
| Figure 4-2 Impurity distributions in bipolar structure at end of input file <i>s4ex1a.inp</i> | 4-8 |
| Figure 4-3 Output listing from PRINT .1D command in file <i>s4ex1a.inp</i> | 4-9 |
| Figure 4-4 Listing of input file <i>s4ex1b.inp</i> , showing statements for simulating the field oxide, base, and emitter region processing for a bipolar transistor. | 4-10 |
| Figure 4-5 Final profiles produced by input file <i>s4ex1b.inp</i> | 4-11 |
| Figure 4-6 Output listing from PRINT .1D command in file <i>s4ex1b.inp</i> | 4-11 |
| Figure 4-7 First part of input file <i>s4ex2a.inp</i> , for determining LOCOS shape | 4-12 |
| Figure 4-8 Mesh used for oxidation simulation. Produced by PLOT .2D GRID statement in input file <i>s4ex2a.inp</i> | 4-13 |
| Figure 4-9 Second part of statement input file <i>s4ex2a.inp</i> , showing statements for plotting results of | |

| | | |
|------------------------------------|---|------------|
| | LOCOS process. | 4-15 |
| Figure 4-10 | Plot produced by the PLOT . 2D FLOW statement in input file <i>s4ex2a.inp</i> | 4-16 |
| Figure 4-11 | Plot produced by the PLOT . 2D STRESS statement in input file <i>s4ex2a.inp</i> | 4-17 |
| Figure 4-12 | Contours of hydrostatic pressure plotted by statements in input file <i>s4ex2a.inp</i> | 4-18 |
| Figure 4-13 | First part of input file <i>s4ex2b.inp</i> , showing processing steps | 4-19 |
| Figure 4-14 | Grid plot produced by first PLOT . 2D statement in input file <i>s4ex2b.inp</i> | 4-20 |
| Figure 4-15 | Second part of input file <i>s4ex2b.inp</i> , showing statements for plotting the results of the diffusion simulation. | 4-22 |
| Figure 4-16 | Contours of boron concentration produced by input file <i>s4ex2b.inp</i> | 4-23 |
| Figure 4-17 | Concentration of point defects vs. depth, as plotted by input file <i>s4ex2b.inp</i> | 4-24 |
| Figure 4-18 | Concentration of point defects vs. width, as plotted by input file <i>s4ex2b.inp</i> | 4-25 |
| Figure 4-19 | Contours of interstitial concentration, as plotted by input file <i>s4ex2b.inp</i> | 4-26 |
| Figure 4-20 | First part of input file <i>s4ex3.inp</i> , showing processing and plotting using the PD . FERMI point defect model. | 4-27 |
| Figure 4-21 | Second part of input file <i>s4ex3.inp</i> , using the full two-dimensional point defect model. | 4-28 |
| Figure 4-22 | Profiles with PD . FERMI and PD . FULL models, from <i>s4ex3.inp</i> | 4-30 |
| Figure 4-23 | Output produced by PRINT . 1D statement in input file <i>s4ex3.inp</i> | 4-31 |
| Figure 4-24 | Point defect profiles plotted by <i>s4ex3.inp</i> | 4-32 |
| Advanced Examples | | 5-1 |
| Figure 5-1 | First part of input file <i>s4ex4a.inp</i> : Setting up the grid for simulating an NMOS process. | 5-3 |
| Figure 5-2 | Listing of mask information read from file <i>s4ex4m.tl1</i> | 5-4 |
| Figure 5-3 | Second part of input file <i>s4ex4a.inp</i> , for simulating an NMOS process | 5-6 |
| Figure 5-4 | Grid after formation of isolation region, plotted by <i>s4ex4a.inp</i> | 5-6 |
| Figure 5-5 | Structure with contours of boron concentration, after formation of isolation region, as plotted by file <i>s4ex4a.inp</i> | 5-7 |
| Figure 5-6 | First part of input file <i>s4ex4b.inp</i> , showing polysilicon gate formation. | 5-8 |
| Figure 5-7 | NMOS structure after LDD implant, as plotted | |

by file *s4ex4b.inp*. 5-9

Figure 5-8 Second part of input file *s4ex4b.inp*, showing source/drain processing and metallization. 5-10

Figure 5-9 Final grid for LDD NMOS example, produced by input files *s4ex4a.inp* and *s4ex4b.inp*. 5-11

Figure 5-10 Final NMOS structure, as plotted by file *s4ex4b.inp* 5-12

Figure 5-11 Input file *s4ex4c.inp*, for plotting the final LDD NMOS structure. 5-13

Figure 5-12 Complete NMOS structure, plotted by input file *s4ex4c.inp*. 5-13

Figure 5-13 First part of input file *s4ex4d.inp*, showing the threshold voltage extraction. 5-15

Figure 5-14 Second part of input file *s4ex4d.inp*, showing the MOS capacitance extraction. 5-16

Figure 5-15 Third part of input file *s4ex4d.inp*, showing the junction capacitance extraction. 5-17

Figure 5-16 Electrical characteristics, plotted by input file *s4ex4d.inp*. 5-17

Figure 5-17 First part of input file *s4ex5.inp*, showing grid setup 5-18

Figure 5-18 Grid for trench implant example. 5-19

Figure 5-19 Second part of input file *s4ex5.inp*, showing tilted implantation using analytic implant model. 5-20

Figure 5-20 Contours of boron after analytic implant 5-21

Figure 5-21 Vertical profiles produced by analytic implant. 5-22

Figure 5-22 Sidewall profiles produced by analytic implant 5-22

Figure 5-23 Third part of file *s4ex5.inp*, using the Monte Carlo implantation model. 5-24

Figure 5-24 Contours of boron after Monte Carlo implant 5-25

Figure 5-25 Vertical profiles after Monte Carlo implant 5-25

Figure 5-26 Sidewall profiles after Monte Carlo implant 5-26

Figure 5-27 First part of input file *s4ex6.inp*: Poly-buffered LOCOS process. 5-27

Figure 5-28 Grid for poly-buffered LOCOS application 5-28

Figure 5-29 Second part of *s4ex6.inp*: Plotting final poly-buffered LOCOS structure. 5-29

Figure 5-30 Contours of hydrostatic pressure in final poly-buffered LOCOS structure. 5-29

Figure 5-31 First part of input file *s4ex7a.inp*, to set up grid for simulating a CMOS process. 5-30

Figure 5-32 Second part of input file *s4ex7a.inp*, showing statements for simulating a CMOS process. 5-31

Figure 5-33 Third part of input file *s4ex7a.inp*, for simulating a CMOS process. 5-32

Figure 5-34 Initial grid for the 0.8 micron NMOS transistor, produced by *s4ex7a.inp*. 5-33

Figure 5-35 Channel doping profile for NMOS transistor 5-35

Figure 5-36 Input file *s4ex7b.inp*, for plotting results 5-36

| | | |
|-------------|--|------|
| Figure 5-37 | Final 0.8 micron structure, NMOS structure, plotted by <i>s4ex7b.inp</i> (left) and final mesh for 0.8 micron NMOS structure (right). | 5-37 |
| Figure 5-38 | Profiles of active and total arsenic concentration through the poly gate. | 5-38 |
| Figure 5-39 | Final 1.2 micron NMOS structure, plotted by input file <i>s4ex7c.inp</i> | 5-38 |
| Figure 5-40 | Mesh generation for DMOS power transistor, from input file <i>s4ex8.inp</i> | 5-39 |
| Figure 5-41 | Initial grid for simulating DMOS power transistor. | 5-40 |
| Figure 5-42 | Second part of file <i>s4ex8.inp</i> : Processing of DMOS power transistor, through body diffusion. | 5-41 |
| Figure 5-43 | Structure with contours of boron concentration, after first p-well diffusion. | 5-41 |
| Figure 5-44 | DMOS power transistor after p-type body diffusion | 5-42 |
| Figure 5-45 | Third part of <i>s4ex8.inp</i> : Final processing and plotting | 5-43 |
| Figure 5-46 | Final mesh for DMOS simulation (left), showing the result of adaptive gridding, and Final DMOS power transistor structure (right), produced by input file <i>s4ex8.inp</i> | 5-44 |
| Figure 5-47 | Mesh generation for SOI MOSFET, from input file <i>s4ex9.inp</i> | 5-45 |
| Figure 5-48 | Initial grid for simulating SOI MOSFET | 5-46 |
| Figure 5-49 | Processing of SOI MOSFET, from input file <i>s4ex9.inp</i> | 5-47 |
| Figure 5-50 | Final structure, showing contours of net doping for SOI MOSFET (left) and final grid for SOI MOSFET (right). | 5-48 |
| Figure 5-51 | Channel and source/drain doping profiles for SOI MOSFET. | 5-49 |
| Figure 5-52 | First part of input file <i>s4ex10.inp</i> : NMOS transistor processing. | 5-50 |
| Figure 5-53 | Second part of input file <i>s4ex10.inp</i> : Silicide growth. | 5-51 |
| Figure 5-54 | Input file <i>se4ex10p.inp</i> : Plotting results | 5-52 |
| Figure 5-55 | Structure immediately before silicide growth step | 5-52 |
| Figure 5-56 | Structure after silicide growth step | 5-53 |
| Figure 5-57 | Final structure, after removal of remaining titanium | 5-53 |
| Figure 5-58 | Listing of input file <i>s4ex11a.inp</i> for simulating the bipolar emitter structure. | 5-55 |
| Figure 5-59 | Bipolar emitter structure and as-implanted arsenic profiles, as plotted using <i>s4ex11c.inp</i> | 5-56 |
| Figure 5-60 | First part of <i>s4ex11c.inp</i> , for plotting the structure and contours of as-implanted arsenic concentration. | 5-57 |
| Figure 5-61 | Contours of total arsenic concentration and poly grain size after RTA. | 5-58 |
| Figure 5-62 | Contours of net doping for 1-micron and 2-micron emitter stripes. | 5-58 |

| | | |
|---|---|------------|
| User-Specified Equation Interface-USEIT..... | | 6-1 |
| Figure 6-1 | USEIT working diagram | 6-2 |
| Figure 6-2 | Flux at interfaces..... | 6-22 |
| Figure 6-3 | Initialization in the built-in 3-stream diffusion model . | 6-57 |
| Figure 6-4 | Initialization in the USEIT 5-stream diffusion model . | 6-57 |

| | | |
|---|-----------------------------------|------------|
| Appendix D: Format of Mask Data Files..... | | D-1 |
| Figure D-1 | Example of a mask data file | D-2 |

| | | |
|---|--------------------------------|------------|
| Appendix E: Data Format of Saved Structure Files | | E-1 |
| Figure E-1 | TSUPREM-4 structure file | E-6 |
| Figure E-2 | Medici structure file..... | E-10 |

| | | |
|---|---|------------|
| Appendix F: Using the MINIMOS 5 Interface..... | | F-1 |
| Figure F-1 | NMOS structure to be transferred to MINIMOS 5..... | F-3 |
| Figure F-2 | Listing of MINIMOS 5 command file <i>EX2D.INP</i> | F-7 |
| Figure F-3 | Listing of MINIMOS 5 command file <i>EX2D.INP</i> , modified to read doping profiles produced by TSUPREM-4..... | F-7 |

Introduction to TSUPREM-4

Program Overview

TSUPREM-4 is a computer program for simulating the processing steps used in the manufacture of silicon integrated circuits and discrete devices. TSUPREM-4 simulates the incorporation and redistribution of impurities in a two-dimensional device cross-section perpendicular to the surface of the silicon wafer. The output information provided by the program includes:

- Boundaries of the various layers of materials in the structure
- Distribution of impurities within each layer
- Stresses produced by oxidation, thermal cycling, or film deposition

Processing Steps

The types of processing steps modeled by the current version of the program include:

- Ion implantation
- Inert ambient drive-in
- Silicon and polysilicon oxidation and silicidation
- Epitaxial growth
- Low temperature deposition and etching of various materials

Simulation Structure

A TSUPREM-4 simulated structure consists of a number of regions, each of which is composed of one of a number of materials. Each material can be doped with multiple impurities. The materials available in TSUPREM-4 are single-crystal silicon, polycrystalline silicon, silicon dioxide, silicon nitride, silicon oxyni-

tride, titanium, titanium silicide, tungsten, tungsten silicide, photoresist, aluminum, and user-defined materials. The available impurities are boron, phosphorus, arsenic, antimony, and user-defined impurities.

Additional Features

TSUPREM-4 also simulates the distribution of point defects (interstitials and vacancies) in silicon layers and their effects on the diffusion of impurities. The distribution of the oxidizing species in silicon dioxide layers is simulated to calculate oxidation rates.

Manual Overview

This manual is organized as follows:

- Chapter 1** Discusses the execution of TSUPREM-4, the required input files, the output files generated, and other files required to run the program.
- Chapter 2** Describes the physical models for the physical processes simulated by TSUPREM-4 and discusses some of the numerical methods used during the simulation.
- Chapter 3** Contains detailed descriptions of the input statements recognized by TSUPREM-4. The description of each statement includes a summary of the statement syntax, descriptions of the statement parameters, and a discussion of the use of the statement, with examples.
- Chapter 4** Presents simple examples illustrating the use of the program.
- Chapter 5** Presents more complicated examples illustrating the use of the program for simulating complete processes.
- Chapter 6** Describes the USEIT Option, Physical Model and Equation Interface language, and details examples.
- Appendix A** Lists the default simulation coefficient values and the literature references from which they were derived.
- Appendix B** Describes the plot device definition file *s4pcap*. This file contains information that describes the available graphical output devices.
- Appendix C** Describes the program enhancements implemented in the latest version of TSUPREM-4.
- Appendix D** Describes the data format used by mask data files.

[Appendix E](#) Describes the data formats files created with the **SAVEFILE** statement.

[Appendix F](#) Contains a detailed description of the interface to the MINIMOS 5 device simulation program.

Typeface Conventions

The following typeface conventions are used in this manual:

| Typeface | Used for |
|-------------------------|---|
| STATEMENT | Commands or keyboard information that you type appears in this bold, fixed width typeface. SILICON is an example of a parameter in this typeface. |
| output text | Text output by TSUPREM-4 or your system appears in this typeface. Listings of output file contents are shown in this typeface. |
| <i><pathname></i> | Variable information you type, which must be replaced with specific text, is indicated in italics enclosed by angle brackets (<>). The plot device definition file <i><mdpdev></i> is an example of this convention. Do <i>not</i> type the angle brackets when entering your text. |

Related Publications

This manual covers all aspects of the TSUPREM-4 2D process simulation program. For information on TSUPREM-4 installation procedures, see the *TCAD Products and Utilities Installation Manual*.

Reference Materials

This manual uses many references from the changing body of industry literature. Where appropriate, you are directed to source material. References are included in [Chapter 2](#), beginning on [page 2-131](#), and in [Appendix A](#), beginning on [page A-27](#).

Problems and Troubleshooting

If you have problems or questions regarding TSUPREM-4 operation, first check the UNIX window from which you started TSUPREM-4 for warning or error messages:

- For help in resolving UNIX system errors (cannot create *<file>*: Permission denied, and others), please see your UNIX systems administrator.
- For TSUPREM-4-specific problems, please see the person who installed this product or associated *Avant!* TCAD product in your company. Usually this is your UNIX systems administrator or the CAD manager.

For further help, please contact *Avant!* TCAD or *Avant!* TCAD's representative in your area.

Using TSUPREM-4

Introduction

This chapter discusses starting TSUPREM-4, required input files, output files generated, and miscellaneous files required to execute the program. The chapter includes discussions of the following:

- Starting TSUPREM-4
- Program output
- File specification
- Output files generated
- Miscellaneous files required to execute the program

Program Execution and Output

This section describes execution of TSUPREM-4 and program output.

Starting TSUPREM-4

The execution of TSUPREM-4 is initiated with the command

```
tsuprem4 <input filename>
```

where the optional command line argument, <input filename>, specifies the name of a TSUPREM-4 command input file.

If <input filename> is specified, TSUPREM-4 executes the statements contained in the input file. If the file specification is blank, the program responds by printing a header identifying the program version on the user's terminal. The user is then prompted for the file specification of a command input file.



Note:

The file specification must conform to conventions in the operating system; it may not contain more than 80 characters.

If no file is specified in response to the prompt, the program enters interactive input mode, described in “3.1 Documentation and Control” on page 3-7. In this case, the input statements must be entered from the user's terminal.

Program Output

Commands entered from the user's terminal or from an input file are treated identically. In either case, the command is executed, and the results are displayed as soon as the command is read. (See also, “Errors, Warnings, and Syntax” on page 1-3.)

TSUPREM-4 generates both printed and graphical outputs that describe the simulation results. All outputs generated before the termination of program execution are made available to the user. The locations of these outputs are described in “Output Files” on page 1-5.

Printed Output

The following printed output can be obtained:

- Solution information (e.g., impurity concentrations) along vertical or horizontal lines through the structure or along material interfaces (**PRINT . 1D** statement)
- Results produced by the **EXTRACT** statement
- Extracted electrical characteristics (e.g., sheet resistance) produced by the **ELECTRICAL** statement
- Summary of the mask information for each mask level (**MASK** statement)
- Summary of ion implantation parameters (**IMPLANT** statement)
- Informational and error messages to indicate the progress of the simulation

Graphical Output

The following graphical output can be obtained:

- Plots of solution values along a line through the structure or along a material interface (**PLOT . 1D** statement)
- Two-dimensional plots of the structure, showing material boundaries, simulation grid, contours of impurity or point defect concentrations, or growth velocity and stress vectors (**PLOT . 2D** statement)
- Three-dimensional (“bird's-eye view” or “surface projection”) plots of solution values (**PLOT . 3D** statement)
- Plots of electrical parameters such as capacitance or channel conductance vs. bias voltage (**PLOT . 1D** statement)
- Plots of user-specified data (e.g., for comparing measured and simulated profiles)

Solutions can also be saved for later analysis with graphical post-processing programs.

Errors, Warnings, and Syntax

If an error is detected in processing a command, a message is printed; if commands are being read from a file, no-execute mode is set and the remainder of the statements in the input file are checked for syntax but not executed.

The program also prints warning messages. Warnings are not fatal and serve only to indicate potential problems that you should be aware of. Warning messages normally indicate any corrective action taken automatically by the program.

File Specification

In this manual, file names are highlighted by printing them in an *italic* font. Lowercase names are used for input files, library files, and plot files, while uppercase names are used for saved structure files.

File Types

Files used by TSUPREM-4 can be grouped into two categories:

- Files known to the program (library files)
- Files specified by the user

Files known to the program (e.g., *s4init*, and *s4pcap*) have names assigned by *Avant!* TCAD. These names can be changed by the system administrator who installs the program at a user's site and by the user when the program is executed.

Files specified by the user include command input files, plot output files, and saved solution files. Any names can be used for these files, provided that they conform to the file naming conventions of the operating system.

Default File Names

The default names for output listing files are derived from the name of the command input file, if one was specified on the **tsuprem4** command line or in response to the file name prompt. This allows multiple copies of TSUPREM-4 to be executed simultaneously (using different command input files) in a single directory without encountering naming conflicts among the output files. The output file names are derived by removing the extension (the last "." and any following characters in the file name), if any, from the input file name and adding the extensions *.out*, *.inf*, and *.dia* for the output, informational, and diagnostic output files, respectively. If no input file name was specified (i.e., TSUPREM-4 is being run in interactive mode), the default names *s4out*, *s4inf*, and *s4dia* are used.

Environment Variables

Environment variables can be used to override the default values for library file names, standard file identifiers, and graphics output device names

(see [Appendix B](#)). A unique environment variable is associated with each file. These environment variables are described in the remainder of this chapter in the descriptions of the file identifiers they control.

The following environment variables are used in TSUPREM-4:

| | |
|----------------|--|
| <i>S4OUT</i> | standard output file identifier on page 1-6 |
| <i>S4INF</i> | informational output file identifier on page 1-6 |
| <i>S4DIA</i> | diagnostic output file identifier on page 1-6 |
| <i>S4FKY0</i> | formatted key file identifier on page 1-10 |
| <i>S4UKY0</i> | unformatted key file identifier on page 1-10 |
| <i>S4INIT</i> | initialization input file identifier on page 1-9 |
| <i>S4IMPO</i> | ion implant data file identifier on page 1-9 |
| <i>S4PCAP</i> | plot device definition file identifier on page 1-9 |
| <i>S4AUTH</i> | authorization file identifier on page 1-10 |
| <i>DEFPDEV</i> | graphics output device name on page 1-8 and Appendix B |

Input Files

TSUPREM-4 requires several types of input files. The user usually supplies one or more command input files. In addition, the program can read solution files generated by previous TSUPREM-4, Taurus-Lithography simulations, mask data files, and library files. TSUPREM-4 can also read TIF (Technology Interchange Format) files for use with Taurus WorkBench, Michelangelo, and Taurus-Topography.

Command Input Files

Command input files contain statements that direct the TSUPREM-4 simulation. These are text files that can be created and modified using any text editor. In interactive applications, the user's terminal serves as the primary command input file, while secondary command input files are specified with the **SOURCE** statement.

Descriptive names can be used for specialized command input files—process description files, coefficient files, and simulation control files are examples of special-purpose command input files. A detailed description of the valid input statements and their proper format is provided in [Chapter 3](#).

For convenience when using the STUDIO visualization program, end command input file names with the extension *.inp*.

Mask Data Files

Mask data files are created by Taurus Layout. These files contain the mask information from a cross section of a mask layout. Mask data files are formatted text files; the format of these files is described in [Appendix D](#). By convention, the extension *.tll* is used for the mask layout files used by TSUPREM-4.

Profile Files

Profile files contain doping profile information to be read by the **PROFILE** statement. Profile data can also be plotted with the **PLOT .1D** statement or used as a target for optimization. The format of these files is described in “PROFILE” on page 3-84.

Other Input Files

The following files are also read by TSUPREM-4; they are described elsewhere in this chapter:

- TSUPREM-4, Taurus-Lithography, and Taurus-Topography (formerly, Terrain) structure files contain saved solution information. See “[Program Output](#)” on page 1-2.
- The *s4init* library file contains commands to initialize the model coefficients used by TSUPREM-4. See [Appendix A](#).
- The *s4compat64* file contains commands to modify the model coefficients to make TSUPREM-4 version 6.5 give the same results as version 6.4.
- The *s4imp0* library file contains ion implant range statistics for use by the **IMPLANT** statement. See “[Ion Implant Data File—*s4imp0*”](#) on page 1-9.
- The *s4pcap* library file defines the characteristics of the various graphical output devices available to TSUPREM-4. See “[Plot Device Definition File—*s4pcap*”](#) on page 1-9 and [Appendix B](#).
- The *s4auth* library file contains information on the computer systems for which TSUPREM-4 has been licensed. See “[Authorization File—*s4auth*”](#) on page 1-10.

Output Files

TSUPREM-4 produces a variety of printed and graphical output and data files describing the simulation results. The various types of output are described in the remainder of this section.

Terminal Output

The standard and error output streams normally appear at the user's terminal, but in some computing environments they can be redirected to a file, or appear both at the user's terminal and in an output file. The standard output consists of output

from the **PRINT . 1D** command as well as informational, error, and warning messages generated by many commands. If **ECHO** is set with the **OPTION** command (it is set by default), the input statements are also printed on the standard output.

The error output usually will be interspersed with the standard output. The error output receives error messages generated by the program. In interactive mode, input prompts are also sent to the error output. (See also [“Errors, Warnings, and Syntax” on page 1-3.](#))

Output Listing Files

This section describes the various output listing files.

Standard Output File—*s4out*

A record of each TSUPREM-4 execution is sent to the output listing file. The default name for this file is derived from the name of the input file, but use of the environment variable *S4OUT* overrides this name during program execution (see [“File Specification” on page 1-3.](#)) This text file includes a listing of all input statements, error messages, and printed output produced by the program.

Informational Output File—*s4inf*

Additional information produced by TSUPREM-4 can be sent to the informational output file. The default name for this file is derived from the name of the input file, but the environment variable *S4INF* can be used to override this name during program execution (see [“File Specification” on page 1-3.](#))

This text file can be useful in understanding the operation of the program, but is not normally of interest to the user. Output to the informational file can be enabled or disabled using the **INFORMAT** keyword on the **OPTION** statement; by default, it is disabled.

Diagnostic Output File—*s4dia*

Diagnostic information produced by TSUPREM-4 can be sent to the diagnostic output file. The default name for this file is derived from the name of the input file, but the environment variable *S4DIA* can be used to override this name during program execution (see [“File Specification” on page 1-3.](#))

This text file receives diagnostic information on the internal operation of the program, and is not normally of interest to the user. Output to the diagnostic file can be enabled or disabled using the **DIAGNOST** keyword on the **OPTION** statement; by default, it is disabled.

Saved Structure Files

The structure and impurity distributions can be saved in a number of formats. The name of a saved structure file is specified by the **OUT . FILE** parameter on the **SAVEFILE** statement. All files are written in text format, so they can be transferred easily between hardware platforms. Some structure files can both be read and written by TSUPREM-4. The structure file formats used by TSUPREM-4 are described in the following sections.

- TSUPREM-4** This is the primary file format for saving structures and solution information for later use in TSUPREM-4. This is the default format produced by the **SAVEFILE** statement. TSUPREM-4 structure files contain complete structure and solution information plus some model specifications. They can be read with the **INITIALIZE** and **LOADFILE** statements and used as a basis for further simulations. The format of TSUPREM-4 structure files is described in [Appendix E](#).
- TIF** TIF (Technology Interchange Format) files are used to communicate with Taurus WorkBench, Michelangelo, Taurus-Topography, and Taurus Visual. TIF files are produced by specifying the **TIF** parameter on the **SAVEFILE** statement. TIF files contain complete structure and solution information plus some model specifications. They can be read by TSUPREM-4 with the **INITIALIZE** and **LOADFILE** statements and used as a basis for further simulations. TIF files can also be used to pass structures to Medici.
- Taurus-Lithography** Taurus-Lithography (formerly Depict) structure files are used for communication with *Avant!* TCAD's Taurus-Lithography programs. There are two Taurus-Lithography structure file formats:
- Written by TSUPREM-4 and read by Taurus-Lithography
 - Written by Taurus-Lithography and read by TSUPREM-4
- Unlike TSUPREM-4 and TIF structure files, Taurus-Lithography structure files only contain structure boundary information. Thus, it is necessary to save a file in TSUPREM-4 format in addition to the Taurus-Lithography file to capture the doping profiles. When reading a Taurus-Lithography-format file, the TSUPREM-4 file should be read first.
- Taurus-Lithography structure files are written or read with the **DEPICT** parameter on the **SAVEFILE** or **LOADFILE** statements.
- Medici** Medici structure files are used to communicate with *Avant!* TCAD's Medici device simulation program. They contain the full physical structure plus net and total doping concentrations. Medici structure files can also be read by many versions of PISCES. The format of Medici structure files is described in [Appendix E](#). TIF files can also be used to pass structures to Medici.
- MINIMOS 5** Structure files can also be created with a format that can be read into the MINIMOS 5 device simulation program from the Technical University of Vienna. These files contain the structure and doping information needed by MINIMOS 5. Use of MINIMOS 5 structure files is described in [Appendix F](#).
- Wave** Solution data can be saved in *wave* format for later graphical display using Wavefront Technologies' Data Visualizer program. The format of these files is described in *The Data Visualizer Version 2.0 Programming Guide* from Wavefront Technologies.

Graphical Output

Graphical output is sent to the device determined by the **DEVICE** parameter on the **OPTION** statement or to a default output device (see “**OPTION**” on page 3-39). This is typically the user's terminal, but some graphics devices use output files specified in the *s4pcap* file (see “Plot Device Definition File—*s4pcap*” on page 1-9). The device names that can be specified are defined in the *s4pcap* file. Drivers are available for a variety of devices, including graphics terminals, pen plotters, and laser printers. A list of available devices is given in [Appendix B](#).

TSUPREM-4 selects a graphics output device by the following process:

1. If a valid **DEVICE** parameter has been specified on an **OPTION** statement, its value is used as the device name.
2. If the *DEFPDEV* environment variable specifies a valid device name, that device is used.
3. If the *TERM* environment variable specifies a valid device name, that device is used.
4. If none of the above steps produces a valid device name, the *default* device in the *s4pcap* file is used. The *default* device can be linked to any device in *s4pcap*; in the *s4pcap* file shipped by Avant! TCAD, the *default* device is equivalent to *ps*, which produces files in PostScript format.

This selection process occurs the first time that plotting is requested in a job.

Extract Output Files

The **EXTRACT** statement (see “**EXTRACT**” on page 3-160) allows printing of arbitrary device structure information such as layer thicknesses and impurity concentrations. Extracted information is sent to the file specified by the **OUT.FILE** parameter on the **EXTRACT** statement. The formatting features of the **EXTRACT** statement allows the flexible combination of text and data in the output file.

Electrical Data Output Files

The **ELECTRICAL** statement saves extracted electrical characteristics in the file specified by the **OUT.FILE** parameter. (See “**ELECTRICAL**” on page 3-174.)

Library Files

The following files are used for specific purposes by TSUPREM-4 and most users will not need to reference or modify these files directly. The names of these files are predefined, but can be overridden by specifying appropriate environment variables (see “[Environment Variables](#)” on page 1-3). These files are typically installed in a library common to all users of the program, but users can use their own customized version if they wish.

**CAUTION**

TSUPREM-4 will not run correctly if these files are missing or inaccessible.

Initialization Input File—*s4init*

The initialization input file contains simulation input statements that specify the default coefficients for the materials, impurities, and numerical models in TSUPREM-4. This file is read automatically each time TSUPREM-4 is executed. The default name for this file is *s4init*, but the environment variable *S4INIT* can be used to override this name during program execution. Default values of coefficients and model parameters can be changed by modifying this file. It is a text file and can be modified with any text editor or with the Studio TSUPREM-4 Command Editor.

**Note:**

*Normally, all files read by TSUPREM-4 must be accessible by the user who runs the program. An exception is made for *s4init*: If the set-user-ID (or set-group-ID) mode is set for the TSUPREM-4 executable file (and set-user-ID execution is allowed by the file system containing the executable), then *s4init* may be owned by the user (or group) that owns the TSUPREM-4 executable. Thus, it is possible for TSUPREM-4 to use an *s4init* file that cannot otherwise be read by the user.*

Ion Implant Data File—*s4imp0*

The ion implant data file defines the range statistics for the ion implantation of impurities in various materials. The default name for this file is *s4imp0*, but the environment variable *S4IMP0* can be used to override this name during program execution. This file is in text format and can be modified by any standard text editor. The data in the file is formatted in a manner that is defined in the file. Range statistics data for any number of materials can be represented. For each material, data for several impurity ions is present, with range statistics listed for a sequence of implantation energies for each ion.

Plot Device Definition File—*s4pcap*

The plot device definition file contains the information required to set up and use various graphics output devices. The default name for this file is *s4pcap*, but the environment variable *S4PCAP* can be used to override this name during program execution. The file is in standard text format and can be modified by any text editor. The format is not intended to be self-explanatory. More information on this file is included in [Appendix B](#).

Key Files—*s4fky0* and *s4uky0*

The files *s4fky0* and *s4uky0* define the statement names, parameter names, and default values used by TSUPREM-4, and are used to check the syntax of the command input file. They contain identical information in different formats: The file *s4fky0* is in text format and can be modified by any standard text editor. The file *s4uky0* is in binary format, and can be used more efficiently than *s4fky0* during the syntax check. *s4uky0* is generated from *s4fky0* during the initial installation of TSUPREM-4. Only the unformatted key file *s4uky0* needs to be available when the program is executed. Note that statement and parameter names must agree with names coded in the program source and cannot be changed simply by modifying the key files. The environment variables *S4FKY0* and *S4UKY0* can be used to override the default names of these files during program execution.

Authorization File—*s4auth*

The authorization file *s4auth* contains authorization values that enable the execution of TSUPREM-4. If this file contains invalid authorization values, an error will be displayed indicating that the program is not authorized for execution on the machine. *Avant!* TCAD must be contacted for assistance in correcting this problem. The file *s4auth* is a text file. The environment variable *S4AUTH* can be used to override the default name of this file during program execution. The *s4auth* file is not required by versions of TSUPREM-4 that use a license management program.

TSUPREM-4 Models

Introduction

This chapter describes the modeling capabilities of TSUPREM-4 for the analysis of fabrication processing steps. Discussions include the following:

- Simulation structure and numerical discretization grid
- Processing capabilities that can be simulated by the program. Equations used to model the physical processes. The equations are discussed in terms of the statement parameters documented in [Chapter 3](#).

For a more detailed discussion of the physical basis of many of the models described in this chapter, refer to the Stanford University SUPREM-III Technical Report [1].

Simulation Structure

A TSUPREM-4 simulation represents a two-dimensional cross-section of a portion of a semiconductor wafer.

Coordinates

Usually, the x coordinate represents distance parallel to the surface of the wafer, and the y coordinate corresponds to depth into the wafer. In plots of the structure, x increases from left to right, and y increases from top to bottom. In specialized applications, the y coordinate also lies parallel to the surface of the wafer, giving a simulation space in the plane of the wafer surface.

The coordinate system is defined by the user and is fixed relative to the initial structure, meaning that the coordinate system is tied to the substrate, not to the wafer surface.

Initial Structure

The initial structure is defined as a rectangular region of arbitrary (user-defined) width and depth. By default, the top of the structure is exposed, and reflecting boundary conditions are applied to the sides and bottom. However, this can be changed using the **BOUNDARY** statement (see [page 3-60](#)). Deposition, etching, impurity predeposition, oxidation, silicidation, reflow, and out-diffusion occur at exposed surfaces, while photolithographic exposure and ion implantation always occur at the top surface.



Note:

*There are restrictions on these processes when surfaces other than the top surface are exposed; see the description of the **BOUNDARY** statement ([page 3-60](#)) for details.*

Regions and Materials

The structure is composed of from one to forty regions of arbitrary shape. Each region consists of a single material. By definition, adjacent regions (i.e., regions that meet along an edge) contain different materials. The same material can be present in multiple (non-adjacent) regions.

Grid Structure

The continuous physical processes modeled by TSUPREM-4 are approximated numerically using finite difference (for diffusion) and finite element (for oxide flow) solution techniques.

Mesh, Triangular Elements, and Nodes

Each region of the structure is divided into a mesh of nonoverlapping triangular elements. There can be up to 80,000 triangles in a TSUPREM-4 mesh. Solution values are calculated at the mesh nodes at the corners of the triangular elements. At points where two or more materials meet, there are multiple solution values (multiple nodes), one for each material at the meeting point. On an exposed boundary, there is also an extra node at each point, which represents concentrations in the ambient gas.

The total number of nodes in a structure is calculated by adding the number of mesh points in each material, plus the number of mesh points along exposed boundaries. There can be up to 40,000 nodes in a TSUPREM-4 mesh. Additional nodes may be required on a temporary basis when simulating process steps that modify the structure. The total of the user-defined nodes plus temporary nodes must not exceed the maximum of 40,000.

Defining Grid Structure

A grid structure must be defined before process simulation can start. You define an initial grid structure by explicitly specifying the locations and spacing of grid lines or, automatically, by generating a grid given the width and (optionally) the locations of mask edges. Once an initial grid has been defined, it is adjusted automatically as various process steps are simulated.

The statements that generate and modify the grid have parameters to control the grid spacing. All grid spacing specifications are multiplied by the value of the **GRID.FAC** parameter on the **MESH** statement.

Explicit Specification of Grid Structure

You can specify a nonuniform rectangular grid that is modified by removing portions of some grid lines. The modified rectangular grid is then converted to a triangular grid by adding diagonals. The nonuniform rectangular grid is specified by means of the **LINE**, **BOUNDARY**, and **REGION** statements and modified with the **ELIMINATE** statement. The triangular grid is produced by the **INITIALIZE** statement (see [Chapter 3, 3.2 Device Structure Specification on page 3-49](#) for complete descriptions of these statements).

The **LINE** Statement

The **LINE** statement is used to specify a series of grid lines. The location of each line is given by the **LOCATION** parameter, and the spacing is specified with the optional **SPACING** parameter. Grid lines must be specified in order of increasing **LOCATION**. The result is a set of locations

$$x_1, x_2 \dots, x_m \tag{Equation 2-1}$$

and spacings

$$h_1, h_2 \dots, h_m \tag{Equation 2-2}$$

for the user-specified grid lines. If a **SPACING** is specified for line *i* on the **LINE** statement, then h_i is given by

$$h_i = \text{SPACING} \times \text{GRID.FAC} \tag{Equation 2-3}$$

otherwise, h_i is taken as

$$h_i = \min(x_{i+1} - x_i, x_i - x_{i+1}) \times \text{GRID.FAC} \tag{Equation 2-4}$$

Generated Grid Lines

Grid lines are added between the user-specified lines based on the locations and spacings of the user-specified lines and on the value of the **RATIO** parameter on the **INITIALIZE** statement. The goal is to choose a ratio *r* and number of spaces *n* that satisfy the equations

$$x_{i+1} - x_i = \frac{r^n - 1}{r - 1} h_i \tag{Equation 2-5}$$

$$h_{i+1} = h_i r^{n-1} \tag{Equation 2-6}$$

subject to the constraints

$$n \text{ is an integer} \tag{Equation 2-7}$$

$$\frac{1}{\mathbf{RATIO}} \leq r \leq \mathbf{RATIO} \tag{Equation 2-8}$$

A total of $n - 1$ grid lines is added with spacings

$$h_i, h_i r, h_i r^2, \dots, h_i r^{n-2} \tag{Equation 2-9}$$

To satisfy the constraints, it is usually necessary to modify the values of h_i and h_{i+1} . Equation 2-8 is satisfied by reducing the larger of h_i and h_{i+1} , as needed. Equation 2-7 is then satisfied by increasing or decreasing both h_i and h_{i+1} by the same factor. The factor is chosen so that the final value of n is the nearest integer to the value obtained by solving Equations 2-5 and 2-6 exactly, with the constraint Equation 2-8 on r . Note that the final spacings h_i and h_{i+1} may be slightly greater than their user-specified values, and the ratio r may be slightly larger than **RATIO** or smaller than $1/\mathbf{RATIO}$ to satisfy the constraint that n be an integer.

Two special cases should be noted: If h_i and h_{i+1} are both greater than or equal to the distance $x_{i+1} - x_i$, then no grid lines are added between x_i and x_{i+1} ; and if h_i and h_{i+1} are equal, then uniformly spaced lines are added.

The x and y grids are generated independently, using the same algorithms.

Eliminating Grid Lines

The **ELIMINATE** statement (see page 3-57) can be used to thin out the grid in user-specified portions of the structure. The user specifies a rectangular subregion of the structure and whether vertical grid lines (**COLUMNS**) or horizontal grid lines (**ROWS**) are to be eliminated. The program then removes every other grid line in the specified direction within the specified region. Each additional **ELIMINATE** statement that specifies the region removes half of the remaining grid lines.

In some instances, it is not possible to eliminate grid lines when the specified region overlaps the region specified on a previous **ELIMINATE** statement. In these cases, a warning is printed and the elimination is not performed. The **ELIMINATE** statement is guaranteed to work when the specified region is the same as, totally included in, or totally separate from the regions specified on previous **ELIMINATE** statements.

The **ELIMINATE** statement is used during initialization to create a structure. It cannot be used to remove grid lines after initialization. The only way to remove grid lines after structure initialization is with **STRUCTURE UNREFINE**.

Automatic Grid Generation

If no **LINE X** statements are specified, grid lines in the x direction are generated automatically. Similarly, if no **LINE Y** statements are specified, the vertical grid is generated automatically. Automatic grid generation in either direction can be combined with manual grid generation in the other direction.

Automatic grid generation is controlled by parameters on the **MESH** statement and by the **WIDTH** and **DX** parameters on the **INITIALIZE** statement. Automatic grid generation in the x direction also depends on the location of mask edges obtained from mask information read with the **MASK** statement.

Automatic grid generation is intended for applications in which fine manual control over the grid is not needed. It is especially useful when mask information from Taurus Layout—IC Layout Interface is available. Note that specifications for automatic grid generation can be put in the *s4init* file, allowing advanced users of TSUPREM-4 to set process-specific defaults for use by less experienced users.

Automatic Grid Generation in the X Direction

A grid in the x direction (i.e., a set of vertical grid lines) is generated automatically if an **INITIALIZE** statement without an **IN.FILE** is processed and no **LINE X** statements have been specified since the last **INITIALIZE** statement.

Placement of grid lines in the x direction is controlled by the **DX.MIN**, **DX.MAX**, and **DX.RATIO** parameters on the **MESH** statement and the **WIDTH** and **DX** parameters on the **INITIALIZE** statement. Automatic elimination of vertical grid lines is controlled by the **LY.SURF** and **LY.ACTIV** parameters on the **MESH** statement.

X Grid from WIDTH Parameter

The **WIDTH** parameter on the **INITIALIZE** statement specifies the width of the device. The grid spacing in the x direction is specified by the **DX** parameter on the **INITIALIZE** statement or by **DX.MAX** on the **MASK** statement if **DX** is not specified.

X Grid from MASK Statement

If the **WIDTH** parameter is not specified, but a **MASK** statement has been used to read mask information, the locations of mask edges are used to guide generation of the grid. A line is placed at each mask edge, with spacing given by

$$h_{min} = \text{DX.MIN} \times \text{GRID.FAC} \quad \text{Equation 2-10}$$

To either side of the mask edge, the grid spacing increases by a factor of **DX.RATIO** until a spacing of

$$h_{max} = \text{DX.MAX} \times \text{GRID.FAC} \quad \text{Equation 2-11}$$

is reached, or until a point halfway between two mask edges is reached. A spacing of h_{max} is used far from mask edges.

If the **GRID** parameter is used on the **MASK** statement, only edges on the specified mask levels are used for grid generation. If the **G.EXTENT** parameter is used, the

spacing h_{min} is used for the specified distance under the mask line before increasing towards h_{max} .

Column Elimination

If no **ELIMINATE COLUMNS** statements are specified, a default set of eliminate operations is performed on the vertical grid lines. One elimination is done between

$$y = \frac{1}{2}(\mathbf{LY.SURF} + \mathbf{LY.ACTIV}) \quad \text{Equation 2-12}$$

and the bottom of the structure. Eight eliminations are done between **LY.ACTIV** and the bottom of the structure. These eliminations are intended to remove all but two grid lines below **LY.ACTIV**.

If the **WIDTH** parameter is not specified and if no mask information has been read with a **MASK** statement, vertical grid lines are placed at $x=0$ and $x=1$ micron producing a grid for one-dimensional simulations.

Automatic Grid Generation in the Y Direction

A grid in the y direction (i.e., a set of horizontal grid lines) is generated automatically if an **INITIALIZE** statement without an **IN.FILE** is processed and if no **LINE Y** statements have been specified since the last **INITIALIZE** statement.

Automatic grid generation in the y direction is controlled by the **LY.SURF**, **DY.SURF**, **LY.ACTIV**, **DY.ACTIV**, **LY.BOT**, **DY.BOT**, and **DY.RATIO** parameters on the **MESH** statement.

For grid generation, the structure is divided into three regions.

- The surface region extends from $y=0$ down to **LY.SURF**, and has grid spacing **DY.SURF** \times **GRID.FAC**. The surface region has the finest grid spacing. It normally contains shallow implants and the channels of MOS transistors.
- The active region extends from **LY.SURF** down to **LY.ACTIV** and has a grid spacing that varies smoothly from **DY.SURF** \times **GRID.FAC** to **DY.ACTIV** \times **GRID.FAC**. The active region extends to below the deepest junctions in the structure.
- The substrate region extends from **LY.ACTIV** to **LY.BOT** and has a grid spacing that starts at **DY.ACTIV** \times **GRID.FAC**. It increases by factors of **DY.RATIO** until it reaches a spacing of **DY.BOT** \times **GRID.FAC**. The substrate region is very deep, but has few vertical grid lines. It provides for accurate modeling of point defect recombination.



Note:

The default values for the automatic grid generation parameters are typical of those required for simulating small-geometry MOS processes. Especially when using bipolar or power processes, you are encouraged to customize the default values for your needs.

Changes to the Mesh During Processing

The initial grid applies to the structure, not to the space containing the structure. Thus, processing steps that change the device structure must necessarily change the mesh structure as well. The processing steps that change the grid are deposition, epitaxy, etching, photoresist development, oxidation, and silicidation. In addition, if adaptive gridding is enabled, the grid may be modified during ion implantation and diffusion. The structure can also be truncated, reflected, or extended using the **STRUCTURE** statement.

DEPOSITION and EPITAXY

The **DEPOSITION** and **EPITAXY** statements deposit a conformal layer on the exposed surface of the structure (see “**DEPOSITION**” on page 3-90 and “**EPITAXY**” on page 3-121). The grid distribution normal to the exposed surface is controlled by four parameters:

- Layer thickness (**THICKNES**)
- Nominal grid spacing (**DY**)
- Location of the nominal grid spacing relative to the top surface of the layer (**YDY**)
- Number of grid spaces in the layer (**SPACES**)

The thickness must always be specified. Effects produced by various combinations of parameters are as follows:

- If none of the other parameters is specified, a single grid space is placed in the layer.
- If **SPACES** or **DY** (but not both) is specified, a uniform grid spacing of **THICKNES/SPACES** or **DY** (multiplied by **GRID.FAC**) is used.
- If both **SPACES** and **DY** are specified, a nonuniform grid spacing is used, with a spacing of **DY × GRID.FAC** at a depth **YDY** below the surface and spacing that increases or decreases by a constant ratio for a total of **SPACES/GRID.FAC** grid spaces. The grid spacing varies linearly from the value at **YDY** to the top and bottom of the layer, increasing or decreasing at the same rate on either side of **YDY** to produce the required number of grid spaces. Not all combinations of **SPACES**, **DY**, and **YDY** will produce a valid grid; in particular, the average grid spacing can never be less than half of **DY**.

Normally, the grid spacing parallel to the exposed surface is the same as the spacing along the original surface. However, when the grid spacing perpendicular to the surface is large compared to the parallel spacing, the parallel spacing is increased to approximately half of the perpendicular spacing. Exposed corners on the original surface produce arcs at the surface of the deposited layer. These arcs are approximated by straight segments of length **ARC.SPAC × GRID.FAC** or smaller.

Structure Extension

The **STRUCTURE** statement with the **EXTEND** parameter works similarly to deposition, except that grid is added horizontally to one edge of the structure; the parameters **WIDTH**, **DX**, and **XDX** are used in place of **THICKNES**, **DY**, and **YDY**.

Note that **XDX** in this case is the absolute x coordinate at which the grid spacing specified by **DX** applies.

ETCH and DEVELOP

The **ETCH** and **DEVELOP** statements remove a specified portion of the structure. The grid is modified in two steps:

1. Nodes are added along the etch boundary. Nodes are placed at points where the etch boundary intersects the boundary of the material to be removed. Within the material being removed, the spacing of the added nodes is set to the smaller of the grid spacing of the points defining the etch boundary and the grid spacing in the material being removed.
2. Mesh elements inside the etch boundary are removed

Oxidation and Silicidation

The moving silicon/oxide interface and the material expansion produced during oxidation require continuous modifications to the simulation mesh. Three kinds of grid manipulation are required:

- Removal of nodes in consumed layers
- Addition of nodes in a growing layer (e.g., during oxidation)
- Removal or rearrangement of nodes in regions where oxide is deforming

The following descriptions are in terms of oxidation, but apply equally to silicidation and other reactions. The only difference is that **DY . DEFAU** is used for the growing material instead of **DY . OXIDE** and **GRID . OXI**. This difference has no effect for any processes other than oxidation.

Removal of Nodes in Consumed Silicon

As oxidation occurs, the silicon/oxide interface advances into the silicon, while the growing oxide expands away from the silicon. Nodes (both silicon and oxide) on the interface move with the interface; nodes in the silicon interior remain fixed, while nodes in the oxide move with the oxide flow. As a consequence, interface nodes are continuously moving towards silicon nodes, which must be removed to allow the interface to advance.

Addition of Nodes in a Growing Layer

On the oxide side of the interface, the triangular mesh elements are expanding. To maintain solution accuracy in the oxide (for example, calculating the diffusion of oxidant in the oxide), it is necessary to add nodes to the oxide. The addition of nodes to the oxide is controlled by the **DY . OXIDE**, **DY . EXACT**, **DY . LOCAL**, and **GRID . OXI** parameters on the **METHOD** statement (see [page 3-188](#)).

Precise grid spacing is obtained by adding new nodes in a growing layer at the distance specified by $\text{DY . OXIDE} \times \text{GRID . FAC}$ from an existing node in the layer. Because only one node can be added at each point on an interface during a simulation time step, the size of the time steps may need to be reduced to achieve the desired spacing. This reduction in the time step can be disabled by specifying **^DY . EXACT** on the **METHOD** statement. By default, **DY . EXACT** is set to allow cutting of the time steps to control the grid spacing. The algorithm will not allow

grid points to be added at spacings less than 1Å and control of the spacing may not be precise for spacings less than 2Å.

The grid control algorithms and parameters apply to the entire structure. To avoid adding very fine grid in field regions when growing gate oxides, an option has been added to allow the grid spacing to vary with the oxide growth rate. When **DY . LOCAL** is specified on the **METHOD** statement, the grid spacing to be used at each point in the growing oxide is

$$h = \frac{v_{max}}{v} \text{DY . OXIDE} \times \text{GRID . FAC} \quad \text{Equation 2-13}$$

where v is the growth rate at a point in the structure and v_{max} is the maximum growth rate at all interfaces of the same type in the structure. (**DY . DEFAU** is used instead of **DY . OXIDE** for reactions other than oxidation.) **DY . LOCAL** is set true by default.

If **DY . OXIDE** is zero, an extra node is added when the spacing in the oxide is approximately equal to the spacing in the silicon multiplied by the value of **GRID . OXI**. For **GRID . OXI** = 2.2 (the silicon to silicon/dioxide volumetric expansion ratio) you might expect the number of nodes added to the oxide to equal the number of nodes removed from the silicon. Actually, more nodes are added to the oxide, because the apparent spacing of nodes in the silicon is reduced by the movement of the interface. The **DY . OXIDE** parameter is the preferred means of controlling grid spacing in the oxide. The **GRID . OXI** parameter is considered obsolete, and is retained only for compatibility with older versions of the program.

Nodes in Regions Where Oxide is Deforming

The flow of oxide may not be uniform when masking layers are present. At the edge of a nitride layer, for example, two corners of a triangular mesh element may be constrained by the presence of the nitride, while the third is being carried by the flow of oxide out from under the mask. In such situations, mesh triangles can become severely distorted, and could prevent further oxidation unless appropriate remedies are applied, such as *flipping* triangles and removing nodes or triangles in congested portions of the mesh. (In flipping triangles, the line common to two adjacent triangles is removed to form a quadrilateral, then a line is added to divide the quadrilateral into two triangles using the opposite two corners.)

In some cases, it may be necessary to eliminate a very thin triangle of one material that protrudes into another material. To avoid creating a hole in the structure, the material type of the triangle is changed to that of the surrounding structure.

Numerical Integrity

After any modification to the mesh, a check is made for triangles that might lead to numerical difficulties (i.e., loss of accuracy or poor convergence). Where possible, these triangles are eliminated by adjusting the triangularization or by adding nodes. Where such adjustment is not possible or would lead to large numerical errors, the discretization of the triangle is modified to avoid numerical instability.

Adaptive Gridding

To reduce the effort required to set up an initial grid and to improve simulation accuracy, TSUPREM-4 can perform adaptive mesh refinement during ion implantation, diffusion, and oxidation. It can perform adaptive mesh unrefinement during diffusion and oxidation.

Refinement

Adaptive mesh refinement in TSUPREM-4 consists of splitting an edge of a triangle by adding a node at its midpoint if the estimated error at the midpoint is greater than the user-specified limit:

$$\epsilon > \mathbf{REL. ADAP} \times \mathbf{ERR. FAC} \times C + \mathbf{ABS. ADAP} \quad \text{Equation 2-14}$$

where **REL. ADAP**, **ABS. ADAP**, and **ERR. FAC** are parameters on the **METHOD** statement and C is the value of a solution variable.

For adaptive gridding during ion implantation (with the analytical models):

- C is the concentration of the impurity being implanted
- ϵ is calculated directly from the ion implantation model

No mesh refinement takes place during implants with the Monte Carlo model.

For adaptive gridding during diffusion:

- The calculation is done for each mobile impurity species, using values of ϵ based on estimates of the curvature of the impurity profile.

For adaptive gridding during oxidation:

- The calculation is done for the oxidizing species, using values of ϵ based on estimates of the curvature of the profile of the oxidant.

Separate values of **REL. ADAP** and **ABS. ADAP** are specified for each solution value in each material. **ERR. FAC** is a single value that scales the relative errors specified by **REL. ADAP**. The minimum grid spacing produced during adaptive refinement is specified for each solution value and material by the **MIN. SPAC** parameter on the **METHOD** statement.

Unrefinement

Adaptive mesh unrefinement in TSUPREM-4 consists of removing mesh points that are not required to maintain the accuracy of the solution. The criterion for point removal is:

$$\epsilon < \mathbf{UNREFINE} \times (\mathbf{REL. ADAP} \times \mathbf{ERR. FAC} \times C + \mathbf{ABS. ADAP}) \quad \text{Equation 2-15}$$

where **UNREFINE** is a parameter on the **METHOD** statement. Unrefinement is done twice in each diffusion step, about one quarter and three quarters of the way through the step. The maximum grid spacing produced by unrefinement is specified for each material by the **MIN. SPAC** parameter on the **METHOD** statement.

You can also request mesh unrefinement with the **UNREFINE** parameter on the **STRUCTURE** statement. The **REPEAT** parameter specifies the maximum number of unrefinement passes. Normally unrefinement does not remove points in the undoped (or background doped) substrate, but if **ROWS** or **COLUMNS** is specified on the **STRUCTURE** statement, these points can be removed. This can be useful in preparing a structure for device simulation. The unrefinement can be limited to a portion of the structure with the **X.MIN**, **X.MAX**, **Y.MIN**, and **Y.MAX** parameters.

Usage

The **IMP.ADAP** parameter on the **METHOD** statement enables or disables adaptive gridding during ion implantation. The **DIF.ADAP** parameter enables or disables adaptive gridding (both refinement and unrefinement) during diffusion. The **OX.ADAP** parameter enables or disables adaptive gridding in oxide based on oxidant concentration. Refinement is disabled for impurity/material combinations for which **REL.ADAP** is zero. No unrefinement will occur in materials for which **MIN.SPAC** is zero, and no unrefinement will occur if **UNREFINE** is zero.

Adaptive gridding reduces the impact of the initial grid on the speed and accuracy of the simulation but does not eliminate it entirely. Meshes that rely heavily on adaptive gridding tend to be less regular than manually specified meshes of similar density. Results using these irregular meshes may appear less accurate because of the irregular variation of the numerical error from point to point. Regular, manually generated meshes may be preferable when precise comparison of similar solutions is required.

Even with adaptive gridding, you must be sure the following requirements are met to ensure accurate simulations:

1. The initial grid spacing in layers that receive implants must be smaller than the standard deviation of the implanted profile.
2. When an epitaxial layer is grown on a substrate with a high surface impurity concentration, the first grid space in the epi layer must be small compared to the expected depth of out-diffusion into the epi layer.

There may be cases where you want to modify or disable adaptive gridding:

1. Adaptive refinement during diffusion following a Monte Carlo implant may cause a large number of nodes to be added in response to statistical variations in the implanted profile. It is often a good idea to disable **DIF.ADAP** during the first few minutes of diffusion following a Monte Carlo implant.
2. When oxidizing through a poly layer you may want to disable **OX.ADAPT** between the time when the oxidation first breaks through the poly and the time when the poly layer has been consumed across its entire width. This avoids adding unnecessary extra nodes in the pad oxide under the poly.
3. In situations where less accuracy is needed, you may want to set **ERR.FAC** to a value greater than 1. Similarly, where more accuracy is needed you may want to use a value less than 1.

4. For large structures that undergo long high-temperature diffusion cycles, you may want to be more aggressive in removing unneeded mesh points. It is usually safe to use values of **UNREFINE** as high as 1.0.
5. There may be cases where unrefinement is removing mesh points you will soon be needing. (For example, nodes near the surface may be removed prior to an implant that needs them.) In this case you can disable unrefinement by setting **UNREFINE** to 0 until the points are used.
6. You may want to reduce the value of **MAX.SPAC** in silicon for small geometry devices and increase the value for power devices.

One-Dimensional Simulation of Simple Structures

In many simulations, the geometry and doping vary in the y direction only. Such structures are represented internally as one-dimensional structures, with considerable savings in simulation time and memory requirements.

One-dimensional structures are automatically converted to two-dimensional structures whenever an etch or expose step destroys the uniformity of the structure in the x direction. One-dimensional structures are also converted to two dimensions for display (using the **PLOT.2D** statement) or saving to a file. Such conversions are temporary, and one-dimensional simulation resumes after plotting or saving.

All saved structures are stored as two-dimensional structures; structures that are uniform in the x direction are converted to one-dimensional structures when they are read from a file. Full two-dimensional simulation of one-dimensional structures can be forced by specifying **^FAST** on the **MESH** statement.

Initial Impurity Concentration

The initial impurity concentration in a structure can be specified directly or by specifying the resistivity of the material. In either case, one of two styles can be used:

- In the old-style specification, **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHOR** parameters are used to set the impurity concentration or resistivity of antimony, arsenic, boron, and phosphorus, respectively; **CONCENTR** or **RESISTIV** is used to determine whether the concentration or resistivity is specified.
- In the new-style specification, impurity is specified by name with the **IMPURITY** parameter and the concentration or resistivity is specified with the **I.CONC** or **I.RESIST** parameters.

The old-style and new-style specifications can be mixed when specifying impurity concentrations directly, i.e., the **IMPURITY** and **I.CONC** parameters can be used on the same statement with the **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHOR** parameters. Only one impurity can be specified when **RESISTIV** or **I.RESIST** is specified.

When the resistivity is specified, the concentration is given by

$$C = \begin{cases} \frac{1}{q\mu_n\rho} & \text{donor impurity} \\ \frac{1}{q\mu_p\rho} & \text{acceptor impurity} \end{cases} \quad \text{Equation 2-16}$$

where μ_n and μ_p are the electron and hole mobilities, respectively, and ρ is the resistivity. The carrier mobilities are contained in three internal tables—one for hole mobility in p-type silicon, one for electron mobility in arsenic- and anti-mony-doped silicon, and one for electron mobility in silicon doped with other donor impurities. The tables are from Masetti, et al. [2].



Note:

*The mobility tables used for determining the initial concentration are not the same as those used for calculating sheet resistance with the **ELECTRICAL** statement. Thus, the extracted sheet resistance is not identical to the specified resistivity of the starting material.*

The same calculation is used for all materials, even though it is only meaningful for silicon and polysilicon; a warning is printed if the resistivity is specified for other materials.

Specification of the initial impurity concentration is the same for the **INITIALIZE**, **DEPOSITION**, and **EPITAXY** statements.

Diffusion

The most fundamental process simulated by TSUPREM-4 is diffusion, which, when performed in an oxidizing ambient or in the presence of an appropriate metal, also produces oxidation or silicidation. In addition to simulating the diffusion of the dopant atoms, TSUPREM-4 models the diffusion of point defects (i.e., self-interstitials and vacancies) and, in some cases, an oxidizing species (assumed to be O₂ or H₂O). For silicidation, diffusion of metal and/or silicon atoms through silicide is modeled. The models used by the **DIFFUSION** statement are described in the following sections.

DIFFUSION Statement

The **DIFFUSION** statement (see [page 3-115](#)) is used to model the diffusion of impurities under oxidizing and nonoxidizing conditions. The duration of the diffusion step (in minutes) is specified with the **TIME** parameter.

Temperature

The initial temperature of the step (in °C) is given by the **TEMPERAT** parameter. Linear variation of the temperature over the step can be specified with the **T.RATE** or **T.FINAL** parameters. If **T.RATE** is specified, the temperature varies as

$$T_c = \text{TEMPERAT} + \text{T.RATE} \times t \quad \text{Equation 2-17}$$

where t is the time since the start of the step and T_c is the diffusion temperature (in °C) at time t . If **T.FINAL** is specified, the temperature varies as

$$T_c = \text{TEMPERAT} + \frac{(\text{T.FINAL} - \text{TEMPERAT})}{\text{TIME}} t \quad \text{Equation 2-18}$$

If neither **T.RATE** nor **T.FINAL** is specified, the temperature is constant. The physical coefficients that depend on temperature are presumed to be valid in the range 800 to 1250°C, but temperatures outside this range are allowed.

Ambient Gas Pressure

The pressure of the ambient gas during the step can vary linearly with time and is specified with the **PRESSURE** parameter and either the **P.RATE** or **P.FINAL** parameter. The **PRESSURE** parameter specifies the initial pressure. If neither **P.RATE** nor **P.FINAL** is specified, the pressure is constant. If **P.RATE** is specified, the pressure varies as

$$P = \text{PRESSURE} + \text{P.RATE} \times t \quad \text{Equation 2-19}$$

where t is the time since the start of the step and P is the pressure at time t . If **P.FINAL** is specified, the pressure varies as

$$P = \text{PRESSURE} + \frac{(\text{P.FINAL} - \text{PRESSURE})}{\text{TIME}} t \quad \text{Equation 2-20}$$

These values must be chosen to yield positive, nonzero pressures throughout the step.

Ambient Gas Characteristics

The characteristics of the ambient gas can be specified in one of two ways:

1. Specify a previously defined ambient with one of the parameters **DRYO2**, **WETO2**, **STEAM**, **INERT**, **AMB.1**, **AMB.2**, **AMB.3**, **AMB.4**, or **AMB.5**.
2. Define the ambient gas directly by specifying the flows of the oxidizing and nonoxidizing species with the parameters **F.O2**, **F.H2O**, **F.H2**, **F.N2**, and **F.HCL** or **HCL**.

Ambients and Oxidation of Materials

The characteristics of defined ambients and the physical coefficients describing the oxidation of materials are specified with the **AMBIENT** statement (see [page 3-207](#)). The flows of the oxidizing and nonoxidizing species associated with the ambient are specified with the **F.O2**, **F.H2O**, **F.H2**, **F.N2**, and **F.HCL** parameters. The default gas pressure can be specified for the ambient with the

PRESSURE parameter, and the default chlorine percentage can be specified with **HCL**.

If flows of both O₂ and H₂ are present in the ambient, these gases are assumed to undergo a complete pyrogenic reaction to form H₂O as given by



The final flows of O₂, H₂, and H₂O after the pyrogenic reaction are given by

$$F_{\text{O}_2} = \mathbf{F} \cdot \mathbf{O}_2 - \min\left(\mathbf{F} \cdot \mathbf{O}_2, \frac{\mathbf{F} \cdot \mathbf{H}_2}{2}\right) \quad \text{Equation 2-22}$$

$$F_{\text{H}_2} = \mathbf{F} \cdot \mathbf{H}_2 - \min(2 \times \mathbf{F} \cdot \mathbf{O}_2, \mathbf{F} \cdot \mathbf{H}_2) \quad \text{Equation 2-23}$$

$$F_{\text{H}_2\text{O}} = \mathbf{F} \cdot \mathbf{H}_2\text{O} + \min(2 \times \mathbf{F} \cdot \mathbf{O}_2, \mathbf{F} \cdot \mathbf{H}_2) \quad \text{Equation 2-24}$$

The partial pressures of the oxidizing species are given by

$$P_{\text{O}_2} = \frac{F_{\text{O}_2}}{F_{\text{O}_2} + F_{\text{H}_2\text{O}} + F_{\text{H}_2} + \mathbf{F} \cdot \mathbf{N}_2 + \mathbf{F} \cdot \mathbf{HCL}} P \quad \text{Equation 2-25}$$

$$P_{\text{H}_2\text{O}} = \frac{F_{\text{H}_2\text{O}}}{F_{\text{O}_2} + F_{\text{H}_2\text{O}} + F_{\text{H}_2} + \mathbf{F} \cdot \mathbf{N}_2 + \mathbf{F} \cdot \mathbf{HCL}} P \quad \text{Equation 2-26}$$

If the resulting ambient contains both O₂ and H₂, the oxidation rate is based on the partial pressure of H₂O.

Default Ambients

The following default ambients are defined:

- **DRYO2**: The dry oxygen ambient contains 100% O₂.
- **WETO2**: The wet oxygen ambient contains 92% H₂O and 8% N₂. This reflects evidence that wet O₂ (oxygen bubbled through H₂O at 95°C) is equivalent to pyrogenic steam with O₂ and H₂ flow rates of 1.175 and 2.0 liters/minute, respectively [3]. (The actual ambient contains 8% O₂, but because TSUPREM-4 cannot model simultaneous oxidation by H₂O and O₂, nitrogen is substituted for the oxygen in the simulation.)
- **STEAM**: The steam ambient contains 100% H₂O. This represents formation of H₂O by a complete pyrogenic reaction of O₂ and H₂, without excess O₂ or H₂.
- **INERT**: The inert ambient contains 100% N₂ (or other inert gasses).

The default **PRESSURE** for each of these ambients is 1 atmosphere.

Chlorine

The inclusion of chlorine in the ambient is specified with either the **HCL** or **F.HCL** parameter. These parameters are related by

$$\text{HCL} = 100 \frac{\mathbf{F} \cdot \text{HCL}}{F_{\text{O}_2} + F_{\text{H}_2\text{O}} + F_{\text{H}_2} + \mathbf{F} \cdot \text{N}_2 + \mathbf{F} \cdot \text{HCL}} \quad \text{Equation 2-27}$$

where F_{O_2} , F_{H_2} , and $F_{\text{H}_2\text{O}}$ are the final flows of O_2 , H_2 , and H_2O , respectively, after the pyrogenic reaction of O_2 and H_2 to form H_2O . The chlorine percentage is defined as 100 times the mole fraction of atomic chlorine relative to the total ambient gas.

Example

For example, in dry oxygen 1% chlorine represents the presence of one chlorine atom for every 99 O_2 molecules. Either **HCL** or **F.HCL** can be used when defining an ambient on the **AMBIENT** or **DIFFUSION** statement. If **HCL** is specified along with a predefined ambient on the **DIFFUSION** statement, the specified chlorine percentage is used instead of the percentage or flow rate defined for the ambient.

Coefficient Tables

The effects of chlorine in the ambient gas on the oxidation rate of silicon are represented by tables of coefficients that modify the linear oxidation rate. The tables are specified with the **AMBIENT** statement (see page 3-207). The tables are two-dimensional, with the rows representing chlorine percentages and the columns representing diffusion temperatures. Linear interpolation is used to obtain values for temperatures or percentages between the values in the table. For temperatures or percentages outside the range of values present in the tables, the values in the first or last rows or columns, as appropriate, are used. The use of the coefficients in these tables is described in “Analytical Oxidation Models” on page 2-48.

Chemical Predeposition

A chemical predeposition step can be modeled with the **DIFFUSION** statement by specifying the concentrations of one or more impurities in the ambient gas. The impurity concentrations are specified explicitly with the parameters **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHOR** or with the **IMPURITY** and **I.CONC** parameters. Impurities can also be included in a thermal oxidation step, but the resulting oxide is always modeled as a high-quality thermal oxide; the effects of high impurity concentrations on the oxide characteristics are not simulated.

Solution of Diffusion Equations

The diffusion equations are nonlinear because of the dependence of the diffusion coefficients and electric field on the impurity, point defect, and carrier concentrations. These dependencies also couple the equations for multiple impurities and point defects. An accurate solution to this coupled nonlinear system is obtained by dividing the total diffusion time specified by the **TIME** parameter into a series of smaller time steps represented by Δt . The initial time step is set by the **INIT.TIM** parameter on the **METHOD** statement (see page 3-188). For each time step thereafter, the value of Δt is made as large as possible while holding the estimated error in the solution to acceptably small values. The time step Δt is chosen to satisfy [10]

$$\sqrt{\frac{1}{n} \sum_{i=1}^n \left(\sum_{j=1}^m \left| \frac{\Delta C_{ij}}{\text{REL} \cdot \text{ERR}_j \cdot C_{ij} + \text{ABS} \cdot \text{ERR}_j} \right| \right)^2} \leq 1 \quad \text{Equation 2-28}$$

where

- n is the number of nodes in the structure
- m is the number of diffusing species (impurities and point defects) at each node
- C_{ij} is the concentration of species j at node i
- ΔC_{ij} is the estimated error in C_{ij}
- **REL . ERR** and **ABS . ERR** are the relative and absolute error for each species, specified with the **METHOD** statement

The time step may be also be reduced during oxidation to avoid numerical difficulties or to prevent mesh tangling.

Diffusion of Impurities

This section describes the equations that model the diffusion of dopant atoms in the device structure. Diffusion of dopants is simulated in all materials.

The diffusion equation solved for each impurity present in the structure is

$$\frac{\partial C}{\partial t} = -\vec{\nabla} \cdot (\vec{J}_m + \vec{J}_n) \quad \text{Equation 2-29}$$

where

- C is the chemical impurity concentration
- $\vec{\nabla} \cdot$ is the divergence operator

Impurity Fluxes

The impurity fluxes in the interior of a material region are given by

$$\vec{J}_m = -D_m \left[\vec{\nabla} \left(C_m \frac{M}{M'} \right) - z_s \left(C_m \frac{M}{M'} \right) \frac{q\vec{E}}{kT} \right] \quad \text{Equation 2-30}$$

$$\vec{J}_n = -D_n \left[\vec{\nabla} \left(C_m \frac{N}{N'} \right) - z_s \left(C_m \frac{N}{N'} \right) \frac{q\vec{E}}{kT} \right] \quad \text{Equation 2-31}$$

where

- \vec{J}_m and D_m are the flux and diffusivity of impurities diffusing with (or as) interstitials
- \vec{J}_n and D_n are the flux and diffusivity of impurities diffusing with vacancies
- z_s is the charge of the ionized impurity (+1 for donors and -1 for acceptors)
- q is the electronic charge
- k is Boltzmann's constant
- T is the absolute temperature

- C_m is the mobile impurity concentration
- $\vec{\nabla}$ is the gradient operator
- M/M' and N/N' model the enhancement (or retardation) of diffusion due to nonequilibrium point defect concentrations

Modeling of Diffusivities

The diffusivities D_m and D_n can be multiplied by the parameters **DI.FAC** and **DV.FAC** in the **IMPURITY** statement. Parameters values can be numbers or formulas. It provides the method for more complicated modeling of diffusivity.

$$\vec{J}_m = -D_m \cdot \mathbf{DI.FAC} \cdot \left(\vec{\nabla} \left(C_m \frac{M}{M'} \right) - z_s \left(C_m \frac{M}{M'} \right) \frac{q\vec{E}}{kT} \right) \quad \text{Equation 2-32}$$

$$\vec{J}_m = -D_n \cdot \mathbf{DV.FAC} \cdot \left(\vec{\nabla} \left(C_m \frac{N}{N'} \right) - z_s \left(C_m \frac{N}{N'} \right) \frac{q\vec{E}}{kT} \right) \quad \text{Equation 2-33}$$

Mobile Impurities and Ion Pairing

TSUPREM-4 includes a model for the pairing of positively and negatively charged dopant ions, see [4], [5], and [6]. This model reduces the concentration of mobile dopant atoms C_m according to:

$$C_m = F_{pd} C_a \quad (\text{donors}) \quad \text{Equation 2-34}$$

$$C_m = F_{pa} C_a \quad (\text{acceptors}) \quad \text{Equation 2-35}$$

where C_a is the electrically active dopant concentration (“[Activation of Impurities](#)” on page 2-24) and F_{pa} and F_{pd} are the ion pairing factors for donors and acceptors, respectively. Ion pairing reduces the diffusivity of dopants where the concentration of dopants of the opposite type is large.

The ion pairing model assumes that positively charged donors can bind with negatively charged acceptors to form neutral pairs:



The forward reaction rate is proportional to the number of unpaired donor and acceptor ions, while the reverse reaction rate is proportional to the number of pairs. In equilibrium:

$$(N_d - N_p)(N_a - N_p) = \Omega N_p \quad \text{Equation 2-37}$$

Where N_d and N_a are the total concentrations of electrically active donors and acceptors, respectively, N_p is the concentration of ion pairs, and Ω is a proportionality factor. [Equation 2-37](#) can be solved for the number of ion pairs, giving:

$$N_p = \frac{1}{2} \left[(N_d + N_a + \Omega) - \sqrt{(N_d + N_a + \Omega)^2 - 4N_d N_a} \right] \quad \text{Equation 2-38}$$

The pairing factors are then given by

$$F_{pd} = \left(1 - \frac{N_p}{N_d} \right) \quad (\text{donors}) \quad \text{Equation 2-39}$$

$$F_{pa} = \left(1 - \frac{N_p}{N_a} \right) \quad (\text{acceptors}) \quad \text{Equation 2-40}$$

The parameter Ω is given by

$$\Omega = \text{IP.OMEGA } n_i \quad \text{Equation 2-41}$$

where **IP.OMEGA** is a parameter on the **MATERIAL** statement; the default value for silicon and polysilicon is 6.0 [5]. The ion pairing model is enabled or disabled for each material by the **ION.PAIR** parameter on the **MATERIAL** statement; by default, it is enabled for silicon and polysilicon, but disabled for all other materials, including new user-defined materials.

The ion pairing model is significant because it allows the dependence of the impurity diffusivity to be modeled in both n-type and p-type materials. In particular, it may reduce the diffusivity of boron in n-type materials without introducing a strong increase in diffusivity at high p-type concentrations.

Electric Field

In insulator and conductor materials, the electric field vector \vec{E} is zero. In semiconductor materials, the electric field is given by

$$\begin{aligned} \vec{E} &= -\vec{\nabla}\psi \\ &= -\frac{kT}{q} \frac{1}{n} \vec{\nabla}n \end{aligned} \quad \text{Equation 2-42}$$

where n is the electron concentration. By assuming local charge neutrality, the electron concentration is written as

$$n = \frac{N_d - N_a}{2} + \sqrt{\left(\frac{N_d - N_a}{2} \right)^2 + n_i^2} \quad \text{Equation 2-43}$$

where

- N_d and N_a are the sums of the electrically active donor and acceptor impurity concentrations, respectively
- n_i is the intrinsic carrier concentration given by

$$n_i = \mathbf{NI.0} \cdot \exp\left(\frac{-\mathbf{NI.E}}{kT}\right) T^{\mathbf{NI.F}} \quad \text{Equation 2-44}$$

where **NI.0**, **NI.E**, and **NI.F** are specified in the **MATERIAL** statement. Presence of the electric field term produces couplings among the diffusion equations for the different impurities.

The electrically active and mobile impurity concentrations (C_a and C_m , respectively) are assumed to be the same. The model for calculating these values is described in “[Activation of Impurities](#)” on page 2-24.

Diffusivities

It is assumed that impurities diffuse in semiconductor materials as dopant-defect pairs. The diffusion coefficients D_m and D_n are sums of the effective diffusivities of impurities due to pairing with defects in various charge states:

$$D_m = \mathbf{FGB} \times \sum_{k=-6}^6 \left(D_{mk} \eta^{-k} \right) \quad D_n = \mathbf{FGB} \times \sum_{k=-6}^6 \left(D_{nk} \eta^{-k} \right) \quad \text{Equation 2-45}$$

where $\eta \equiv n/n_i$ is the normalized electron concentration. The components of diffusivity are given by

$$D_{m0} = \mathbf{DIX.0} \cdot \exp\left(\frac{-\mathbf{DIX.E}}{kT}\right) \quad D_{n0} = \mathbf{DVX.0} \cdot \exp\left(\frac{-\mathbf{DVX.E}}{kT}\right) \quad \text{Equation 2-46}$$

$$D_{m1} = \mathbf{DIP.0} \cdot \exp\left(\frac{-\mathbf{DIP.E}}{kT}\right) \quad D_{n1} = \mathbf{DVP.0} \cdot \exp\left(\frac{-\mathbf{DVP.E}}{kT}\right) \quad \text{Equation 2-47}$$

$$D_{m(-1)} = \mathbf{DIM.0} \cdot \exp\left(\frac{-\mathbf{DIM.E}}{kT}\right) \quad D_{n(-1)} = \mathbf{DVM.0} \cdot \exp\left(\frac{-\mathbf{DVM.E}}{kT}\right) \quad \text{Equation 2-48}$$

$$D_{m(-2)} = \mathbf{DIMM.0} \cdot \exp\left(\frac{-\mathbf{DIMM.E}}{kT}\right) \quad D_{n(-2)} = \mathbf{DVMM.0} \cdot \exp\left(\frac{-\mathbf{DVMM.E}}{kT}\right) \quad \text{Equation 2-49}$$

where the parameters **DIX.0**, **DIX.E**, **DIP.0**, **DIP.E**, **DIM.0**, **DIM.E**, **DIMM.0**, **DIMM.E**, **DVX.0**, **DVX.E**, **DVP.0**, **DVP.E**, **DVM.0**, **DVM.E**, **DVMM.0**, and **DVMM.E** are specified in the **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHORUS** statements (see “[3.5 Models and Coefficients](#)” on page 3-187). Diffusivities for arbitrary charge states are given by

Equation 2-50

$$D_{mk} = \text{DIC} \cdot 0 \cdot \exp\left(\frac{-\text{DIC} \cdot \mathbf{E}}{kT}\right) \quad D_{nk} = \text{DVC} \cdot 0 \cdot \exp\left(\frac{-\text{DVC} \cdot \mathbf{E}}{kT}\right)$$

where the parameters **DIC . 0**, **DIC . E**, **DVC . 0**, and **DVC . E** for the charge state $k = \mathbf{C} \cdot \mathbf{STATE}$ are specified on the **IMPURITY** statement (see page 3-239).

Polysilicon Enhancement

The factor **FGB** is applied only for materials for which the polycrystalline model is disabled, and only if the value specified is nonzero. It compensates for the omission of the grain-boundary diffusion flux in polycrystalline materials when the polycrystalline diffusion models are not used.

Point Defect Enhancement

The definition of the point defect enhancement factors M/M' and N/N' depends on the **NSTREAMS**, **PAIR . SAT**, and **PAIR . REC** parameters specified on the **METHOD** statement. For **NSTREAMS** = 1 (e.g., if **PD . FERMI** is set on the **METHOD** statement) and in materials other than silicon, the point defect concentrations are assumed to be at their thermal equilibrium values so that

PD . FERMI Model

$$\frac{M}{M'} = \frac{N}{N'} = 1$$

Equation 2-51

and Equations 2-29, 2-30, and 2-31 reduce to

$$\frac{\partial C}{\partial t} = -\vec{\nabla} \cdot \left[-(D_m + D_n) \left(\vec{\nabla} C_m - z_s C_a \frac{q\vec{E}}{kT} \right) \right]$$

Equation 2-52

PD . TRANS and PD . FULL Models

For **NSTREAMS** = 3 (e.g., if **PD . TRANS** or **PD . FULL** is set on the **METHOD** statement), use the full equations for the enhancement factors:

$$\frac{M}{M'} = \frac{\frac{I}{I^*} \left(\frac{K_m + K_{mv} \frac{I}{I^*}}{K_m + K_{mv} \frac{V}{V^*}} \right)}{1 + \alpha_m \frac{I}{I^*} \left(\frac{K_m + K_{mv} \frac{I}{I^*}}{K_m + K_{mv} \frac{V}{V^*}} \right) + \alpha_n \frac{V}{V^*} \left(\frac{K_n + K_{ni} \frac{V}{V^*}}{K_n + K_{ni} \frac{I}{I^*}} \right)}$$

Equation 2-53

$$\frac{N}{N'} = \frac{\frac{V}{V^*} \left(\frac{K_n + K_{ni} \frac{V^*}{V}}{K_n + K_{ni} \frac{I}{I^*}} \right)}{1 + \alpha_m \frac{I}{I^*} \left(\frac{K_m + K_{mv} \frac{I^*}{V}}{K_m + K_{mv} \frac{V}{V^*}} \right) + \alpha_n \frac{V}{V^*} \left(\frac{K_n + K_{ni} \frac{V^*}{V}}{K_n + K_{ni} \frac{I}{I^*}} \right)} \quad \text{Equation 2-54}$$

α_m and α_n are the ratios of the dopant-defect pair concentrations to the substitutional dopant concentration under equilibrium conditions. They are used to define M' and N' :

$$M' \equiv \alpha_m C \quad \text{Equation 2-55}$$

$$N' \equiv \alpha_n C \quad \text{Equation 2-56}$$

The values of α_m and α_n are calculated from

$$\alpha_m = \frac{D_m}{D_M} \quad \text{and} \quad \alpha_n = \frac{D_n}{D_N} \quad \text{Equation 2-57}$$

where

$$D_M = \text{DIPAIR.0} \cdot \exp\left(-\frac{\text{DIPAIR.E}}{kT}\right) \quad \text{Equation 2-58}$$

$$D_N = \text{DVPAIR.0} \cdot \exp\left(-\frac{\text{DVPAIR.E}}{kT}\right) \quad \text{Equation 2-59}$$

and **DIPAIR.0**, **DIPAIR.E**, **DVPAIR.0**, and **DVPAIR.E** are parameters on the **IMPURITY**, **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHORUS** statements. Physically, D_M and D_N are the diffusivities (assumed to be independent of charge state) of the dopant-defect pairs; they are expected to have values comparable to the diffusivities of interstitials and vacancies, respectively.

Paired Fractions of Dopant Atoms

The values of α_m and α_n vary with the Fermi level, but do not otherwise depend on the dopant or defect concentrations. Physically, they represent the fractions of dopant atoms that are coupled with interstitials and vacancies. The α_m and α_n terms prevent the pair concentrations from exceeding the total dopant concentrations when the concentrations of point defects are very high [8]. These terms are calculated only if **PAIR.SAT** is true (set on the **METHOD** statement, directly or by specifying **PD.FULL**); otherwise, they are set to 0.

Reaction Rate Constants

The K_m and K_n terms arise from the competition between the kick-out and Frank-Turnable mechanisms of pair formation [9].

- K_m and K_n are the reaction rate constants for generation of dopant-interstitial and dopant-vacancy pairs
- K_{mv} is the rate constant for the reaction of dopant-interstitial pairs with vacancies
- K_{ni} is the rate constant for the reaction of dopant-vacancy pairs with interstitials

Thus, K_{mv} and K_{ni} are the reaction rate constants for dopant-assisted recombination of interstitials and vacancies. The values of the reaction rate constants are computed from

$$K_m = \sum_{k=-2}^1 \left(g_m \phi_{Ik} \eta^{-k} \right) \quad \text{Equation 2-60}$$

$$K_n = \sum_{k=-2}^1 \left(g_n \phi_{Vk} \eta^k \right) \quad \text{Equation 2-61}$$

$$K_{mv} = \sum_{k=-2}^1 \left(r_{mv} \frac{D_{mk}}{D_M} \phi_{V(-k)} \right) \quad \text{Equation 2-62}$$

$$K_{ni} = \sum_{k=-2}^1 \left(r_{ni} \frac{D_{nk}}{D_N} \phi_{I(-k)} \right) \quad \text{Equation 2-63}$$

where g_m , g_n , r_{mv} , and r_{ni} are given by

$$g_m = \mathbf{R.I.S} \cdot 4\pi D_I I_i^* \exp\left(-\frac{\mathbf{E.I.S}}{kT}\right) \quad \text{Equation 2-64}$$

$$g_n = \mathbf{R.V.S} \cdot 4\pi D_V V_i^* \exp\left(-\frac{\mathbf{E.V.S}}{kT}\right) \quad \text{Equation 2-65}$$

$$r_{mv} = \mathbf{R.IP.V} \cdot 4\pi(D_M + D_V) V_i^* \exp\left(-\frac{\mathbf{E.IP.V}}{kT}\right) \quad \text{Equation 2-66}$$

$$r_{ni} = \mathbf{R.VP.I} \cdot 4\pi(D_N + D_I) I_i^* \exp\left(-\frac{\mathbf{E.VP.I}}{kT}\right) \quad \text{Equation 2-67}$$

The capture radii $\mathbf{R.I.S}$, $\mathbf{R.V.S}$, $\mathbf{R.IP.V}$, and $\mathbf{R.VP.I}$ and barrier energies $\mathbf{E.I.S}$, $\mathbf{E.V.S}$, $\mathbf{E.IP.V}$, and $\mathbf{E.VP.I}$ are specified by parameters on the **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHORUS** statements. The interstitial

and vacancy diffusivities (D_I and D_V) and charge fractions (ϕ_{Ik} and ϕ_{Vk}) are described in “Diffusion of Point Defects” on page 2-32.

The dopant-assisted recombination factors are calculated only if **PAIR.REC** is true (set on the **METHOD** statement, directly or by specifying **PD.FULL**); otherwise, they are set to 1.

If neither **PAIR.SAT** nor **PAIR.REC** is set, Equations 2-53 and 2-54 reduce to

$$\frac{M}{M'} = \frac{I}{I^*} \quad \text{Equation 2-68}$$

$$\frac{N}{N'} = \frac{V}{V^*} \quad \text{Equation 2-69}$$

This is the approximation used in many other process simulators.

If the spatial variation of M/M' and N/N' is small, you can approximate Equations 2-29, 2-30, and 2-31 by

$$\frac{\partial C}{\partial t} = -\vec{\nabla} \cdot \left[-\left(D_m \frac{M}{M'} + D_n \frac{N}{N'} \right) \left(\vec{\nabla} C_m - z_s C_a \frac{q\vec{E}}{kT} \right) \right] \quad \text{Equation 2-70}$$

This form of the equation (with the approximation of Equations 2-68 and 2-69) is used for the **TWO.DIM** model in TSUPREM-4 prior to version 6.0. It is used in versions 6.0 and later (without the approximation of Equations 2-68 and 2-69) if **PAIR.GRA** and **PD.PFLUX** are false (reset on the **METHOD** statement, directly or by specifying **PD.TRANS**). (Equations 2-29, 2-30, and 2-31 are always used if **PD.PFLUX** is true to avoid numerical difficulties.)

Activation of Impurities

The electrically active concentration of an impurity may be less than its chemical concentration.

Physical Mechanisms

Four physical mechanisms are considered in calculating the active concentration:

- Solid solubility limits: These are presumed to be the result of precipitation at high concentrations and are present at equilibrium. They are modeled by assuming that the active concentration is in equilibrium with the precipitated dopant.
- Dopant clustering: At high concentrations, some dopants form inactive clusters that are stable at equilibrium. These clusters are modeled by assuming that the active concentration is in equilibrium with the precipitated dopant.
- Dopant-defect clusters (e.g., boron-interstitial clusters or BICs): These clusters incorporate point defects as well as dopant atoms. Their formation and dissolution rates depend on the concentration of point defects, which in turn is

affected by the formation and dissolution of the clusters. These clusters often form following ion implantation and can persist for minutes or hours at low temperatures but are not stable in equilibrium. They are modeled by following their transient behavior as they form and dissolve.

- **Small dopant-defect clusters:** These are small, short-lived precursors to the larger dopant-defect clusters described above. They are modeled by assuming that they are in equilibrium with the instantaneous concentrations of dopants and point defects.

Activation Models

These physical mechanisms are included in the following TSUPREM-4 activation models:

- **ACT.EQUI:** Solid solubility limits and dopant clustering are considered. Both are treated as equilibrium phenomena (i.e., the activation is a function of the instantaneous temperature and does not depend on the processing history). This is the simplest model; it is best suited for long anneals where transient effects are insignificant. If you include the final temperature ramp-down cycle when using this model you will probably obtain an unrealistically low activation level. This can be remedied by specifying the final anneal temperature with the **TEMPERAT** parameter on the **SELECT** or **SAVEFILE** statement.
- **ACT.FULL:** All four of the physical mechanisms described above are included. A transient solution is used for the dopant-defect clusters, while the other mechanisms are treated as equilibrium phenomena.
- **ACT.TRAN:** Solid solubility limits and dopant clustering are considered, but the activation occurs over time rather than instantaneously. Implanted dopants are considered to be initially active only to some minimum activation level. For more on the **ACT.TRAN** model see [“Transient Clustering Model” on page 2-129](#).

The activation model is specified on the **METHOD** statement.

Model Details

The active dopant concentration is found by adjusting the total dopant concentration C to account for grain boundaries (in polycrystalline materials), dopant-defect clusters, small dopant-defect clusters, solid solubility limits, and dopant clustering, in that order. The concentration after adjusting for grain boundaries and dopant-defect clusters is

$$C_{a1} = C - w_{gb} - C_{dd} \tag{Equation 2-71}$$

where w_{gb} is the dopant concentration in polycrystalline grain boundaries (see [“Modeling Polycrystalline Materials” on page 2-112](#)) and C_{dd} is the concentration in dopant-defect clusters ([“Dopant-Defect Clustering Model” on page 2-27](#)). The subtraction of C_{dd} is only done when the **ACT.FULL** model is used. C_{a1} is adjusted to account for small dopant-defect clusters by:

$$C_{a2} + \text{DDCS} \cdot 0 \cdot \exp\left(\frac{-\text{DDCS} \cdot \mathbf{E}}{kT}\right) \left[C_{a2} \left(\frac{I}{I_i^*}\right)^{\text{IFRACS}} \right]^{\text{DDCS} \cdot \mathbf{N}} = C_{a1} \tag{Equation 2-72}$$

where **DDCS . 0**, **DDCS . E**, **IFRACS**, and **DDCS . N** are parameters on the **IMPURITY** statement. **DDCS . 0** and **DDCS . E** specify the fraction of dopants that are clustered in equilibrium and **IFRACS** specifies the number of interstitials associated with each dopant atom. **DDCS . N** is the number of dopant atoms in the cluster, and must be either 1 or 2 (any other value is ignored). The number of interstitials stored in small dopant-defect clusters is given by

$$I_{dds} = \text{IFRACS} \cdot (C_{a1} - C_{a2}) \tag{Equation 2-73}$$

These calculations are only done when the **ACT . FULL** model is used; when **ACT . EQUI** or **ACT . TRAN** is used, $C_{a2} = C_{a1}$ and $I_{dds} = 0$.

Solid Solubility Model

At high doping concentrations, the active concentration is limited by solid solubility to:

$$C_{as} = \begin{cases} C_{a2} , & C_{a2} \leq 0.9C_{ss} \\ C_{ss} - \frac{(C_{a2} - 1.1C_{ss})^2}{0.4C_{ss}} , & 0.9C_{ss} \leq C_{a2} \leq 1.1C_{ss} \\ C_{ss} , & C_{a2} \geq 1.1C_{ss} \end{cases} \tag{Equation 2-74}$$

where C_{ss} is the solid solubility. In version 6.4 and earlier, the active concentration is simply

$$C_{as} = \min(C_{a2}, C_{ss});$$

this form is still used if **v . COMPAT=6.4** is specified on the **OPTION** statement.

Solid Solubility Tables

The solid solubilities of impurities are represented by tables of values that are specified with the **IMPURITY** statement (page 3-239). Each table is one dimensional, with up to 100 rows corresponding to the diffusion temperatures. Pairs of temperatures and concentrations are specified using the **SS . TEMP** and **SS . CONC** parameters; a separate **IMPURITY** statement must be used for each table entry. The table can be cleared by specifying the **SS . CLEAR** parameter.

The material to which the table applies is specified with the **MATERIAL** parameter. The solid solubility data for boron, phosphorus, and antimony can also be specified on the **BORON**, **PHOSPHORUS**, and **ANTIMONY** statements, respectively. The solubility data for silicon and polysilicon is obtained from published polynomial approximations [10].

Logarithmic interpolation is used to obtain values of solid solubility for temperatures between the values in a table. For temperatures outside the range of values present in the table, the value is extrapolated using the first two or last two values in the table. In the default tables, the first two values in each table are the same, so the extrapolation results in the first value being used at temperatures below the lowest temperature in the table. Similarly, the last two values of each table are the

same, so that the last value is used for temperatures above the highest temperature in the table.

Dopant Clustering Model

In the clustering model, the electrically active impurity concentration C_a is obtained by solving

$$C_{as} = C_a + \left[\text{CTN} \cdot 0 \cdot \exp\left(\frac{-\text{CTN} \cdot \text{E}}{kT}\right) C_a \right]^{\text{CTN} \cdot \text{F}} \quad \text{Equation 2-75}$$

where the parameters **CTN . 0**, **CTN . E**, and **CTN . F** are specified in the **IMPURITY** statement (see [page 3-239](#)). The clustering parameters for arsenic can also be specified on the **ARSENIC** statement.

Both the solid solubility and clustering models are used to calculate the active concentration C_a , although usually the parameters are chosen so that only one of them affects C_a . If no solid solubility is specified, then $C_{as} = C_{a2}$, while if **CTN . 0** is zero, then $C_a = C_{as}$.

Dopant-Defect Clustering Model

In some cases, the activation of impurities is strongly dependent on the past sequence of processing conditions to which a wafer has been subjected. An important example is the transient clustering of boron following an ion implantation and annealing at about 900°C or less. This clustering can be modeled by

$$\frac{\partial C_{dd}}{\partial t} = \frac{1}{\tau_{dd}} \left[K_{ddF} n_i \left(\frac{C_a}{n_i}\right)^{\text{DDCF} \cdot \text{D} \cdot \text{N}} \eta^{\text{DDCF} \cdot \text{N} \cdot \text{N}} \left(\frac{I}{I^*}\right)^{\text{DDCF} \cdot \text{I} \cdot \text{N}} - C_{dd} \eta^{\text{DDCR} \cdot \text{N} \cdot \text{N}} \left(\frac{I}{I^*}\right)^{\text{DDCR} \cdot \text{I} \cdot \text{N}} \right] \quad \text{Equation 2-76}$$

$$\tau_{dd} = \text{DDC} \cdot \text{T} \cdot 0 \cdot \exp\left(-\frac{\text{DDC} \cdot \text{T} \cdot \text{E}}{kT}\right) \quad \text{Equation 2-77}$$

$$K_{ddF} = \text{DDC} \cdot \text{F} \cdot 0 \cdot \exp\left(-\frac{\text{DDC} \cdot \text{F} \cdot \text{E}}{kT}\right) \quad \text{Equation 2-78}$$

where C_{dd} is the concentration of dopant atoms in transient dopant-defect clusters and **DDC . T . 0**, **DDC . T . E**, **DDC . F . 0**, **DDC . F . E**, **DDCF . D . N**, **DDCF . N . N**, **DDCF . I . N**, **DDCR . N . N**, and **DDCR . I . N** are parameters on the **IMPURITY** statement.

This equation models dopant-defect clusters with **DDCF . D . N** dopant atoms per cluster, **DDCF . I . N** interstitials required to form a cluster, **DDCR . I . N** interstitial atoms required to dissolve a cluster, and **DDCF . I . N** – **DDCR . I . N** interstitials in the cluster. **DDC . F . 0** and **DDC . F . E** control the amount of clustering while **DDC . T . 0** and **DDC . T . E** control the rate of clustering and declustering. Because

C_{dd} is subtracted from the total concentration before computing C_a , the dopant in transient clusters is inactive and immobile.

The net capture rate of interstitials by clusters is given by

$$R_{dd} = \mathbf{IFRACM} \frac{\partial C_{dd}}{\partial t} \quad \text{Equation 2-79}$$

where **IFRACM** is a parameter on the **IMPURITY** statement (see “[Point Defect Diffusion Equations](#)” on page 2-33). Ideally, **IFRACM** should be equal to $(\mathbf{DDCF} \cdot \mathbf{I} \cdot \mathbf{N} - \mathbf{DDCR} \cdot \mathbf{I} \cdot \mathbf{N}) / \mathbf{DDCF} \cdot \mathbf{D} \cdot \mathbf{N}$, but it is available as a separate parameter for flexibility in modeling.

The dopant-defect clustering model is enabled by specifying **ACT.FULL** on the **METHOD** statement.

Segregation of Impurities

The segregation of impurities at material interfaces is treated as a nonequilibrium process by TSUPREM-4.

Segregation Flux

At an interface between materials i and j , the impurity flux J from material i to material j (normal to the interface) is given by

$$J = h \left(\frac{C_i}{m} - C_j \right) \quad \text{Equation 2-80}$$

where

- C_i and C_j are the concentrations in materials i and j , respectively
- h is the interface transport coefficient
- m is the equilibrium interface segregation coefficient

This expression represents the impurity flux in the diffusion equation at interfaces between different materials and between the ambient gas and the exposed surfaces of the simulation structure.

The segregation is based on the chemical impurity concentrations when the **ACT.EQUI** model is used and the active concentrations when **ACT.TRAN** is used. (When **V.COMPAT** \leq 6.4, the chemical concentrations are used in all cases.) For materials using the polycrystalline model, segregation at material interfaces is based on the concentration in the grain interior.

Transport Coefficient

The transport coefficient is given by

$$h = \text{TRANS} . 0 \cdot \exp\left(\frac{-\text{TRANS} . \text{E}}{kT}\right) \quad \text{Equation 2-81}$$

Segregation Coefficient

The segregation coefficient defines the ratio C_i/C_j in equilibrium, when the interface impurity flux vanishes. The segregation coefficient is given by

$$m = \text{SEG} . 0 \cdot \exp\left(\frac{-\text{SEG} . \text{E}}{kT}\right) \quad \text{Equation 2-82}$$

The coefficients **TRANS . 0**, **TRANS . E**, **SEG . 0**, and **SEG . E** for each impurity and pair of materials are defined on the coefficient statements for impurities (“**IMPURITY**” on page 3-239, “**ANTIMONY**” on page 3-284, “**ARSENIC**” on page 3-290, “**BORON**” on page 3-296, and “**PHOSPHORUS**” on page 3-302). For an interface between materials i and j , material i is specified by the **MATERIAL** parameter and material j is specified with the **/MATERIA** parameter.

Moving-Boundary Flux

There is an additional flux at oxidizing interfaces due to the consumption of silicon containing impurities. The flux is from the silicon into the oxide and is of magnitude vC_{Si} , where C_{Si} is the impurity concentration on the silicon side of the interface and v is the velocity of the interface.

Interface Trap Model

Assuming that there are trap sites at the interface between two adjacent materials, the dopant diffusing through the interface can be trapped into the trap site [11]. The model is activated by specifying the **ITRAP** parameter on the **METHOD** statement.

$$\frac{\partial \sigma}{\partial t} = -\frac{\partial}{\partial l}\left(D\frac{\partial \sigma}{\partial l}\right) + F_i + F_j \quad \text{Equation 2-83}$$

where σ is the areal density of occupied trap sites and l is the length along the boundary and, F_i and F_j are the dopant flux to the interface trap sites from materials i and j , respectively.

$$F_i = h_i\left(C_i\left(f + r_i\frac{\sigma}{\sigma_{max}}\right) - \kappa_i\sigma\right) \quad \text{Equation 2-84}$$

$$F_j = h_j\left(C_j\left(f + r_j\frac{\sigma}{\sigma_{max}}\right) - \kappa_j\sigma\right) \quad \text{Equation 2-85}$$

where C_i and C_j are the concentrations in materials i and j , respectively. The active concentration is used for the surface concentrations of silicon and polysilicon materials unless the **IT . ACT** parameter on the **METHOD** statement is turned *off*. If the transient clustering model is turned *on*, the active concentration is used regardless of the specification of the **IT . ACT** parameter. The interface trap exists on the boundary of either material i or material j . The material that contains the traps is specified by the **MATERIAL** parameter, while the adjacent material is specified by

the **/MATERIA** parameter on the **IMPURITY** statement. f is the fraction of unfilled trap sites. If the **IT.CPL** parameter on the **METHOD** statement is specified,

$$f = 1 - \sum \frac{\sigma}{\sigma_{max}} \tag{Equation 2-86}$$

where the sum is taken over all the trapped dopant species present in the interface. Otherwise, as default, f is given by

$$f = 1 - \frac{\sigma}{\sigma_{max}} \tag{Equation 2-87}$$

D is the diffusivity of trapped dopant moving along the interface, and σ_{max} denotes the maximum trap density dependent on the property of the interface for each dopant species.

$$\sigma_{max} = \mathbf{Q.MAX.0} \cdot \exp\left(\frac{-\mathbf{Q.MAX.E}}{kT}\right) \tag{Equation 2-88}$$

$$D = \mathbf{DIX.0} \cdot \exp\left(\frac{-\mathbf{DIX.E}}{kT}\right) \tag{Equation 2-89}$$

When the interface is formed by deposition, epitaxy or oxidation, the initial density, σ_{ini} , of trapped dopant is set as one of following:

$$\sigma_{ini} = 0 \tag{Equation 2-90}$$

$$\sigma_{ini} = \mathbf{Q.INI.0} \cdot \exp\left(\frac{-\mathbf{Q.INI.E}}{kT}\right) \tag{Equation 2-91}$$

$$\sigma_{ini} = \frac{\sigma_{max} f(h_i C_i + h_j C_j)}{h_i C_i (1 - r_i) + h_j C_j (1 - r_j) + \sigma_{max} (h_i \kappa_i + h_j \kappa_j)} \tag{Equation 2-92}$$

The initial density of trapped dopant is set by specifying one of parameters, **IT.ZERO** (Equation 2-90), **IT.THERM** (Equation 2-91), or **IT.STEAD** (Equation 2-92) on the **METHOD** statement. Equation 2-92 satisfies the steady state, $F_i + F_j = 0$.



Note:

If the impurity is first introduced by implantation, the initial value of the occupied trap density of the impurity is set to zero regardless of the above specification.

Assuming that the interface trap exists on the boundary of material i , hereinafter, the transport coefficients h_i and h_j are given by

$$h_i = \text{TRANS} \cdot 0 \cdot \exp\left(\frac{-\text{TRANS} \cdot \text{E}}{kT}\right) \quad \text{Equation 2-93}$$

$$h_j = \text{/TRANS} \cdot 0 \cdot \exp\left(\frac{-\text{/TRANS} \cdot \text{E}}{kT}\right) \quad \text{Equation 2-94}$$

The r_i and r_j denote the ratio of detrapping rate to trapping rate at the interface with materials i and j , respectively, which are given by

$$r_i = \text{RATIO} \cdot 0 \cdot \exp\left(\frac{-\text{RATIO} \cdot \text{E}}{kT}\right) \quad \text{Equation 2-95}$$

$$r_j = \text{/RATIO} \cdot 0 \cdot \exp\left(\frac{-\text{/RATIO} \cdot \text{E}}{kT}\right) \quad \text{Equation 2-96}$$

The detrapping of trapped dopant from an interface is determined by how many more dopants can be accepted into the material, as well as how many trapped dopants exist. For concentrations over critical concentration, no more detrapping occurs. **SEG.SS** and **/SEG.SS** specify that the critical concentrations are the same as the solid solubilities of dopant in materials i and j , respectively. **SEG.SS** and **/SEG.SS** may be applied only to the silicon or polysilicon material. The κ_i and κ_j are then given by

$$\kappa_i = r_i \frac{C_{ss,i}}{\sigma_{max}}, \quad \kappa_j = r_j \frac{C_{ss,j}}{\sigma_{max}} \quad \text{Equation 2-97}$$



Note:

For a dopant for which solid solubility is not known, solid solubility is calculated from the clustering model by setting the total concentration to the concentration of atoms in the material.

Also, the κ_i and κ_j can be explicitly given by

$$\kappa_i = \text{SEG} \cdot 0 \cdot \exp\left(\frac{-\text{SEG} \cdot \text{E}}{kT}\right) \quad \text{Equation 2-98}$$

$$\kappa_j = \text{/SEG} \cdot 0 \cdot \exp\left(\frac{-\text{/SEG} \cdot \text{E}}{kT}\right) \quad \text{Equation 2-99}$$

The specification of either **SEG.EQ2** or **/SEG.EQ2** makes the segregation in equilibrium equal to the two-phase segregation in equilibrium. **SEG.EQ2** specifies that the κ_i is calculated with κ_j and the segregation in the two-phase system instead of with Equation 2-98. For **/SEG.EQ2**, κ_j is calculated in the same manner.

The interface trap model is applied only when the impurity and adjacent materials are well defined. Otherwise, the normal two-phase segregation model is used.

When the interface trap model is applied, the two-phase segregation flux is excluded unless the **TWO.PHAS** parameter is specified.

The parameters, **TRANS.0**, **TRANS.E**, **RATIO.0**, **RATIO.E**, **/RATIO.0**, **/RATIO.E**, **SEG.0**, **SEG.E**, **/SEG.0**, **/SEG.E**, **SEG.SEG.SS**, **SEG.EQ2**, **/SEG.EQ2**, **Q.MAX.0**, **Q.MAX.E**, **Q.INI.0**, **Q.INI.E**, **DIX.0**, **DIX.E**, and **TWO.PHAS** are specified on the **IMPURITY** statement.

Using the Interface Trap Model

The trapped component is stored as a separate impurity. The name of the trapped impurity is created from the name of the base impurity by prepending **I_** to the name. Thus, you set the diffusivities of the trapped component of boron with

```
IMPURITY IMP=I_BORON MAT=OXIDE /MAT=SILICON DIX.0=...
```

The interface trap model for the specified trapped impurity works only at the interface between the materials specified with the **MATERIAL** and **/MATERIA** parameters. Note that the trapped impurity exists on the surface node of the material specified by **MATERIAL**. Thus, you get the occupied trap density with

```
SELECT Z=I_BORON
EXTRACT OXIDE /SILICON X=0
```

Diffusion of Point Defects

This section describes the equations that model the diffusion of interstitials and vacancies in silicon. The modeling of point defects depends on the **NSTREAMS**, **PD.PFLUX**, **PD.PTIME**, and **PD.PREC** parameters on the **METHOD** statement (see [page 3-188](#)). For **NSTREAMS** = 1 (e.g., if **PD.FERMI** is set on the **METHOD** statement), the interstitial and vacancy concentrations depend only on the Fermi level:

$$I = I^* \quad \text{and} \quad V = V^* \tag{Equation 2-100}$$

The point defect enhancement factors M/M' and N/N' in the equation for diffusion of impurities are unity (see [“Diffusion of Impurities” on page 2-17](#)), and there is no enhancement (or retardation) of impurity diffusion due to oxidation.

Equilibrium Concentrations

The equilibrium concentrations I^* and V^* are given by

$$I^* = I_i^* \sum_{k=-2}^2 \left(\phi_{Ik} \eta^{-k} \right) \quad V^* = V_i^* \sum_{k=-2}^2 \left(\phi_{Vk} \eta^{-k} \right) \tag{Equation 2-101}$$

$$I_i^* \quad \text{and} \quad V_i^* = \text{CEQUIL.0} \cdot \exp\left(\frac{-\text{CEQUIL.E}}{kT}\right) \tag{Equation 2-102}$$

where **CEQUIL.0** and **CEQUIL.E** are parameters on the **INTERSTITIAL** and **VACANCY** statements.

Charge State Fractions

The charge state fractions ϕ_{Ik} and ϕ_{Vk} are given by

$$\phi_{Ik} = \frac{\phi'_{Ik}}{\sum_{k=-6} \phi'_{Ik}} \quad \text{and} \quad \phi_{Vk} = \frac{\phi'_{Vk}}{\sum_{k=-6} \phi'_{Vk}} \quad \text{Equation 2-103}$$

$$\phi'_{I(-2)} \quad \text{and} \quad \phi'_{V(-2)} = \mathbf{DNEG.0} \cdot \exp\left(\frac{-\mathbf{DNEG.E}}{kT}\right) \quad \text{Equation 2-104}$$

$$\phi'_{I(-1)} \quad \text{and} \quad \phi'_{V(-1)} = \mathbf{NEG.0} \cdot \exp\left(\frac{-\mathbf{NEG.E}}{kT}\right) \quad \text{Equation 2-105}$$

$$\phi'_{I0} \quad \text{and} \quad \phi'_{V0} = \mathbf{NEU.0} \cdot \exp\left(\frac{-\mathbf{NEU.E}}{kT}\right) \quad \text{Equation 2-106}$$

$$\phi'_{I1} \quad \text{and} \quad \phi'_{V1} = \mathbf{POS.0} \cdot \exp\left(\frac{-\mathbf{POS.E}}{kT}\right) \quad \text{Equation 2-107}$$

$$\phi'_{I2} \quad \text{and} \quad \phi'_{V2} = \mathbf{DPOS.0} \cdot \exp\left(\frac{-\mathbf{DPOS.E}}{kT}\right) \quad \text{Equation 2-108}$$

$$\phi'_{Ik} \quad \text{and} \quad \phi'_{Vk} = \mathbf{FRAC.0} \cdot \exp\left(\frac{-\mathbf{FRAC.E}}{kT}\right) \quad \text{Equation 2-109}$$

The last form gives the fractions for arbitrary charge states $-6 \leq k \leq 6$. The parameters for interstitials are specified on the **INTERSTITIAL** statement (see page 3-265), while the parameters for vacancies are specified on the **VACANCY** statement (see page 3-276); the charge state k is specified by the **C.STATE** parameter. Note that the defect concentrations are not actually calculated when **NSTREAMS=1**; the values I^* and V^* are used when the interstitial and vacancy concentrations are printed or plotted.

Point Defect Diffusion Equations

For **NSTREAMS = 3** (e.g., if **PD.TRANS** or **PD.FULL** is set on the **METHOD** statement), the equations for point defect diffusion are solved:

$$\frac{\partial(I + I_c + I_{dds})}{\partial t} + \sum \frac{\partial M}{\partial t} = -\vec{\nabla} \cdot \left[-D_I I^* \vec{\nabla} \left(\frac{I}{I^*} \right) + \sum \vec{J}_m \right] \quad \text{Equation 2-110}$$

$$+ G_p - R_b - R_t - R_c - R_l - R_{dd}$$

$$\frac{\partial(V + V_c)}{\partial t} + \sum \frac{\partial N}{\partial t} = -\vec{\nabla} \cdot \left[-D_V V^* \vec{\nabla} \left(\frac{V}{V^*} \right) + \sum \vec{J}_n \right] + G_p - R_b \quad \text{Equation 2-111}$$

$$G_p = \sum \left[\frac{K_{mv} \frac{V}{V^*}}{K_m + K_{mv} \frac{V}{V^*}} \left(\frac{\partial M}{\partial t} + \vec{\nabla} \cdot \vec{J}_m \right) + \frac{K_{ni} \frac{I}{I^*}}{K_n + K_{ni} \frac{I}{I^*}} \left(\frac{\partial N}{\partial t} + \vec{\nabla} \cdot \vec{J}_n \right) \right] \quad \text{Equation 2-112}$$

where

- M and N are the concentrations of dopant-interstitial and dopant-vacancy pairs defined in “Diffusion of Impurities” on page 2-17
- \vec{J}_m and \vec{J}_n are the corresponding fluxes
- I_c and V_c are the concentrations of interstitials and vacancies in small clusters (see “Small Clusters of Point Defects” on page 2-44)
- I_{dds} is the concentration of interstitials in small dopant-defect clusters (see “Activation of Impurities” on page 2-24)
- R_{dd} , R_b , R_t , R_c , and R_l model the loss of free interstitials to dopant-defect clusters (“Dopant-Defect Clustering Model” on page 2-27), bulk recombination (“Recombination of Interstitials with Vacancies” on page 2-35), interstitial traps (“Interstitial Traps” on page 2-41), {311} defects (“Interstitial Clustering Model” on page 2-42), and dislocation loops (“Dislocation Loop Model” on page 2-126), respectively.

The sums are taken over all dopant species present in the structure. The M and N terms are included only if **PD.PTIME** is true (set on the **METHOD** statement, directly or by specifying **PD.FULL**); otherwise, they are set to 0. Likewise, the \vec{J}_m and \vec{J}_n terms are included only if **PD.PFLUX** is true (set on the **METHOD** statement, directly or by specifying **PD.FULL**).

Interstitial and Vacancy Diffusivities

D_I and D_V are the diffusivities of interstitials and vacancies, respectively, given by

$$D_I = \frac{\sum_k D_{Ik} \phi_{Ik} \eta^{-k}}{\sum_k \phi_{Ik} \eta^{-k}}, \quad D_V = \frac{\sum_k D_{Vk} \phi_{Vk} \eta^{-k}}{\sum_k \phi_{Vk} \eta^{-k}} \quad \text{Equation 2-113}$$

where D_{Ik} and D_{Vk} are specified by

$$D_{Ik} \text{ and } D_{Vk} = \mathbf{D} \cdot \mathbf{0} \cdot \exp\left(\frac{-\mathbf{D} \cdot \mathbf{E}}{kT}\right) \times \mathbf{DC} \cdot \mathbf{0} \cdot \exp\left(\frac{-\mathbf{DC} \cdot \mathbf{E}}{kT}\right) \quad \text{Equation 2-114}$$

where **DC . 0** and **DC . E** are specified for **C . STATE = k . D . 0, D . E, DC . 0, DC . E**, and **C . STATE** for D_I and D_V are specified on the **INTERSTITIAL** (page 3-265) and **VACANCY** (page 3-276) statements, respectively. **DC . 0** and **DC . E** can also be set for all charge states by specifying **C . ALL**.

Two approaches can be used to specify the diffusivity of point defects:

1. **D . 0** (in cm²/sec or microns²/min) and **D . E** can be used to specify the basic diffusivity while **DC . 0** (unitless) and **DC . E** specify relative diffusivities of the various charge states
2. **DC . 0** (in cm²/sec or microns²/min) and **DC . E** can be used to specify the diffusivities of the various charge states while **DC . 0** (unitless) and **DC . E** specify a scale factor for adjusting the total diffusivity.



Note:

*While theoretical calculations suggest that the point defect diffusivities may depend on the Fermi level, this dependence has not been experimentally characterized. By default, all of the **DC . 0** are set to 1.0 and all of the **DC . E** are set to zero, so the diffusivity (determined by the values of **D . 0** and **D . E** using approach 1 above) is independent of the Fermi level.*

Reaction of Pairs with Point Defects

The G_p terms are the result of dopant-defect pairs reacting with dopants. Normally, when dopant-interstitial pairs break up they produce an interstitial, which is accounted for by the M and J_m terms in Equation 2-110. However, when dopant-assisted recombination is dominant, dopant-interstitial pairs are destroyed by absorbing a vacancy rather than producing an interstitial. Under these circumstances, the G_p term cancels the M and J_m terms in Equation 2-110 and subtracts them from Equation 2-111 instead. Similarly, dopant-vacancy pairs can either produce a vacancy or absorb an interstitial when they dissolve.

Recombination of Interstitials with Vacancies

The net recombination rate of interstitials with vacancies in the bulk silicon is given by

$$R_b = K_b(F_{IV} + F_{cV} + F_{Ic}) \left(\frac{I}{I^*} \frac{V}{V^*} - 1 \right) \tag{Equation 2-115}$$

$$+ \sum \left[\left(\frac{K_m K_{mv}}{K_m + K_{mv} \frac{V}{V^*}} + \frac{K_n K_{ni}}{K_n + K_{ni} \frac{I}{I^*}} \right) S \right] \left(\frac{I}{I^*} \frac{V}{V^*} - 1 \right)$$

where

- $S = C - M - N$ is the substitutional dopant concentration
- K_m, K_n, K_{mv} and K_{ni} are described in "Diffusion of Impurities" on page 2-17

- F_{Ic} and F_{cV} are described in “Small Clusters of Point Defects” on page 2-44.

Again, the sums are taken over all dopant species. The K_{mv} and K_{ni} terms model dopant-assisted recombination; they are included only if **PD.PREC** is true (set on the **METHOD** statement, directly or by specifying **PD.FULL**); otherwise, they are set to zero.

The bulk recombination factor K_b is specified by

$$K_b = \mathbf{KB.0} \cdot \exp\left(\frac{-\mathbf{KB.E}}{kT}\right) \tag{Equation 2-116}$$

where **KB.0** and **KB.E** are parameters on the **INTERSTITIAL** statement.

The factor F_{IV} describes the dependence of the recombination rate on the charge states of the point defects:

$$F_{IV} = I_i^* V_i^* \sum_{j,k} \left(K_{IV,jk} \phi_{Ij} \eta^{-j} \phi_{Vk} \eta^{-k} \right) / \alpha \tag{Equation 2-117}$$

$$K_{IV,jk} = \mathbf{KIV.0} \cdot \exp\left(\frac{-\mathbf{KIV.E}}{kT}\right) \tag{Equation 2-118}$$

where **KIV.0** and **KIV.E** are specified on the **INTERSTITIAL** statement with **C.I** = j and **C.V** = k . The normalizing factor α is depends on whether **KIV.NORM** (set on the **INTERSTITIAL** statement) is true or false:

$$\alpha = \begin{cases} \sum_{j,k} \left(K_{IV,jk} \phi_{Ij} \phi_{Vk} \right) & \mathbf{KIV.NORM} \text{ true} \\ 1 & \mathbf{KIV.NORM} \text{ false} \end{cases} \tag{Equation 2-119}$$

When **KIV.NORM** is true, $F_{IV} = I_i^* V_i^*$ under intrinsic doping conditions.

The values of $K_{IV,jk}$ are unknown, but TSUPREM-4 provides parameters on the **INTERSTITIAL** statement for setting them based on plausible assumptions:

- **KB.HIGH** sets $K_{IV,jk} = 1$ for all j and k ; this assumes that the rate of recombination between interstitials and vacancies is independent of their charge states; $\alpha = 1$ regardless of the setting of **KIV.NORM**.
- **KB.MED** sets $K_{IV,jk} = 1$ for $j = -k$ or $jk = 0$ and $K_{IV,jk} = 0$ otherwise; this assumes that a point defect can recombine only with a neutral or oppositely charged defect.
- **KB.LOW** sets $K_{IV,jk} = 1$ only for $j = -k$ and $K_{IV,jk} = 0$ otherwise; this assumes that a point defect can recombine only with an oppositely charged defect.

The three bulk recombination models reflect differing assumptions about the reaction rates between interstitials and vacancies in various charge states. The

KB .LOW model assumes that charged interstitials recombine only with oppositely charged vacancies. This model, with **KIV .NORM** set, is the default; it is equivalent to the model used in older versions of TSUPREM-4. The **KB .HIGH** model assumes that any interstitial is equally likely to recombine with any vacancy, regardless of their charge states. This is the model used in many other process simulators, giving:

$$R_b = K_b (I V - I^* V^*) \quad \text{Equation 2-120}$$

The **KB .MED** model assumes that reactions involving an uncharged point defect and reactions involving oppositely charged point defects are equally likely, while reactions between point defects of like charge do not occur (because of Coulomb repulsion).

It is expected that for a given value of K_b the **KB .LOW** model underestimates the recombination rate at high doping levels because it neglects the recombination of neutral defects with charged defects. Similarly, the **KB .HIGH** model overestimates the recombination because it includes recombination of similarly charged defects. The **KB .MED** should work the best, although it ignores reactions between defects having opposite charge of different magnitudes (e.g., doubly negative vacancies and singly positive interstitials). Note that the values of K_b and (perhaps) other parameters may need to be recalibrated if the model is changed.



Note:

*Versions of the program prior to 1999.2 did not use the **KIV .0**, **KIV .E**, or **KIV .NORM** parameters, but some of them did use **KB .LOW**, **KB .MED**, and **KB .HIGH**. If compatibility with those versions is required, use **KIV .NORM** when **KB .LOW** is used and **^KIV .NORM** when **KB .MED** is used, or use **V .COMPAT=1998 .4** on the **OPTION** statement.*

Absorption by Traps, Clusters, and Dislocation Loops

R_t is the rate of absorption of interstitials at stationary interstitial trapping sites; R_t is given by Equation 2-141 in “Interstitial Traps” on page 2-41. R_c is the rate of absorption by interstitial clusters (see “Interstitial Clustering Model” on page 2-42). R_l is the rate of absorption of interstitials by dislocation loops (see “Dislocation Loop Model” on page 2-126).

Injection and Recombination of Point Defects at Interfaces

Recombination of interstitials and vacancies at interfaces with other materials is modeled as [14]

$$R_s = K_s (C - C^*) \quad \text{Equation 2-121}$$

where

- C is the local concentration of interstitials or vacancies
- C^* is the equilibrium concentration of interstitials or vacancies

- K_s is the surface recombination rate

Surface Recombination Velocity Models

There are three models for the surface recombination velocity, specified by the **V.MAXOX**, **V.INITOX**, and **V.NORM** parameters on the **INTERSTITIAL** and **VACANCY** statements. In each case, the surface recombination rate depends on the motion of the interface due to oxidation:

Equation 2-122

$$K_s = K_{surf} + (K_{svel} - K_{surf}) \left(\frac{v}{v_{max}} \right)^{K_{pow}} \quad (\mathbf{V.MAXOX} \text{ model})$$

Equation 2-123

$$K_s = K_{surf} \left[1 + K_{srat} \left(\frac{v}{v_{init}} \right)^{K_{pow}} \right] \quad (\mathbf{V.INITOX} \text{ model})$$

Equation 2-124

$$K_s = K_{surf} + K_{svel} \left(\frac{v}{v_{norm}} \right)^{K_{pow}} \quad (\mathbf{V.NORM} \text{ model})$$

where v is the local velocity of the interface and K_{surf} , K_{svel} and K_{srat} are specified by

$$K_{surf} = \mathbf{KSURF.0} \cdot \exp\left(\frac{-\mathbf{KSURF.E}}{kT}\right) \quad \text{Equation 2-125}$$

$$K_{svel} = \mathbf{KSVEL.0} \cdot \exp\left(\frac{-\mathbf{KSVEL.E}}{kT}\right) \quad \text{Equation 2-126}$$

$$K_{srat} = \mathbf{KSRAT.0} \cdot \exp\left(\frac{-\mathbf{KSRAT.E}}{kT}\right) \quad \text{Equation 2-127}$$

$$K_{pow} = \mathbf{KPOW.0} \cdot \exp\left(\frac{-\mathbf{KPOW.E}}{kT}\right) \quad \text{Equation 2-128}$$

where **KSURF.0**, **KSURF.E**, **KSVEL.0**, **KSVEL.E**, **KSRAT.0**, **KSRAT.E**, **KPOW.0**, and **KPOW.E** are parameters on the **INTERSTITIAL** and **VACANCY** statements.

The three models differ primarily in the normalizing factor for the interface velocity. This section describes the differences between the models, and their advantages and disadvantages.

V.MAXOX Model

The **V.MAXOX** model (the model used in releases of TSUPREM-4 prior to version 6.0) uses v_{max} , the maximum y component of interface velocity in the structure. The disadvantage of this model is that the normalization factor varies with

time and oxidation conditions so that the peak recombination velocity does not depend on the oxidation rate.

V . INITOX Model

The **V . INITOX** model (the model used in *SUPREM-IV* from Stanford University and the University of Florida) corrects the time dependence by using a normalization factor

$$v_{init} = \frac{B}{A} + r_{thin} \tag{Equation 2-129}$$

where $\frac{B}{A}$ and r_{thin} are the linear and thin regime oxidation rates defined in “Oxidation” on page 2-46. Thus v_{init} is the initial oxidation rate of a bare silicon surface. This normalizing factor gives a good time dependence, but the initial surface recombination velocity is still independent of the oxidation conditions such as wet or dry ambient, pressure, or presence of HCl.

V . NORM Model

The **V . NORM** model provides both the time dependence and the dependence on the oxidation conditions by using a constant normalizing factor specified by you:

$$v_{norm} = \text{VNORM} . 0 \cdot \exp\left(\frac{-\text{VNORM} . E}{kT}\right) \tag{Equation 2-130}$$

where **VNORM . 0** and **VNORM . E** are parameters on the **INTERSTITIAL** and **VACANCY** statements.

Trapped Nitrogen Dependent Surface Recombination

The point defect recombination at interfaces depends on trapped nitrogen atoms. Since the recombination rate at interfaces is proportional to the number of kink sites which is supposed to be reduced by trapping nitrogen, the recombination of interstitials and vacancies at interfaces is given by;

$$R = K_s \left(1 - \frac{\sigma_N}{\text{N} . \text{MAX} . 0 \exp\left(\frac{\text{N} . \text{MAX} . E}{kT}\right)} \right) (C - C^*) \tag{Equation 2-131}$$

where **N . MAX . 0** and **N . MAX . E** are parameters on the **INTERSTITIAL** and **VACANCY** statements. The model is taken into account by specifying the **PD . NREC** parameter on the **METHOD** statement.

Injection Rate

At moving interfaces there can be injection of interstitials and/or vacancies into the silicon. The injection rate is given either as a function of the interface velocity or by an analytical function of time, depending on whether the **GROWTH** parameter has been specified on the **INTERSTITIAL** or **VACANCY** statement. The injection rate also depends on which of the injection/recombination models (**V . MAXOX**, **V . INITOX**, or **V . NORM**) is specified.

When **GROWTH** is true, the injection G_s is calculated based on the local interface velocity:

$$G_s = \text{VMOLE} \cdot \theta v^{K_{pow}} \quad (\text{V.MAXOX model}) \quad \text{Equation 2-132}$$

$$G_s = \text{VMOLE} \cdot \theta v \left(\frac{v}{v_{init}} \right)^{G_{pow}} \quad (\text{V.INITOX model}) \quad \text{Equation 2-133}$$

$$G_s = \text{VMOLE} \cdot \theta v \left(\frac{v}{v_{norm}} \right)^{G_{pow}} \quad (\text{V.NORM model}) \quad \text{Equation 2-134}$$

where

- **VMOLE** is the number of silicon atoms per cubic centimeter
- θ is the fraction of silicon atoms injected
- v is the local interface velocity
- K_{pow} , v_{init} , and v_{norm} are given by [Equations 2-128](#), [2-129](#), and [2-130](#), respectively

The values of θ and G_{pow} are specified as

$$\theta = \text{THETA.0} \cdot \exp\left(\frac{-\text{THETA.E}}{kT}\right) \quad \text{Equation 2-135}$$

$$G_{pow} = \text{GPOW.0} \cdot \exp\left(\frac{-\text{GPOW.E}}{kT}\right) \quad \text{Equation 2-136}$$

where **THETA.0**, **THETA.E**, **GPOW.0**, and **GPOW.E** are parameters on the **INTERSTITIAL** and **VACANCY** statements.

If **GROWTH** is false, the analytical model is used:

$$G_s = \frac{A}{(T_0 + t)^{K_{pow}}} \quad (\text{V.MAXOX model}) \quad \text{Equation 2-137}$$

$$(G_s = A(T_0 + t)^{G_{pow}}) \quad (\text{V.INITOX and V.NORM models}) \quad \text{Equation 2-138}$$

where t is the time into the oxidation step and A and T_0 are given by

$$A = \text{A.0} \cdot \exp\left(\frac{-\text{A.E}}{kT}\right) \quad \text{Equation 2-139}$$

$$T_0 = \text{T0.0} \cdot \exp\left(\frac{-\text{T0.E}}{kT}\right) \quad \text{Equation 2-140}$$

where **A.0**, **A.E**, **T0.0**, and **T0.E** are parameters on the **INTERSTITIAL** and **VACANCY** statements. Note that this model must be calibrated for a particular starting structure and growth conditions.

The **GROWTH** model is normally used for injection of interstitials at an oxide/silicon interface. The analytical model is used with **A.0** = 0 to disable injection of interstitials at other interfaces and injection of vacancies at all interfaces.

Moving-Boundary Flux

There is an additional flux at oxidizing interfaces due to the consumption of silicon containing point defects. The flux is directed out of the silicon and is of magnitude vC_{Si} , where C_{Si} is the interstitial or vacancy concentration in the silicon and v is the velocity of the interface. Point defects are not modeled in materials other than silicon, so point defects removed in this fashion have no further effect on the simulation.

Interstitial Traps

The rate of absorption of interstitials at stationary trapping sites is given by [16] and [18],

$$R_t = \frac{\partial T_F}{\partial t} = k_f T_E I - k_r T_F \tag{Equation 2-141}$$

where

- T_F is the concentration of filled interstitial traps
- T_E is the concentration of empty traps
- k_f and k_r are the forward and reverse rates for the trap-filling reaction

In equilibrium, the forward and reverse reactions proceed at equal rates giving

$$k_r = \frac{k_f T_E^* I^*}{T_F^*} \tag{Equation 2-142}$$

where T_F^* and T_E^* are the equilibrium concentrations of filled and empty traps, respectively. Thus Equation 2-141 becomes

$$R_t = \frac{\partial T_F}{\partial t} = k_f \left[T_E I - \frac{T_E^* I^*}{T_F^*} T_F \right] \tag{Equation 2-143}$$

The total number of traps is given by

$$T_T = T_F + T_E = T_F^* + T_E^* = \mathbf{TRAP.CON} \tag{Equation 2-144}$$

where **TRAP.CON** is a parameter on the **INTERSTITIAL** statement. The reaction rate and equilibrium concentration of empty traps are given by

$$k_f = \mathbf{K.TRAP.O} \cdot \exp\left(-\frac{\mathbf{K.TRAP.E}}{kT}\right) \tag{Equation 2-145}$$

$$T_E^* = \mathbf{F.TRAP.O} \cdot \exp\left(-\frac{\mathbf{F.TRAP.E}}{kT}\right) \times \mathbf{TRAP.CON} \tag{Equation 2-146}$$

where **K.TRAP.O**, **K.TRAP.E**, **F.TRAP.O**, and **F.TRAP.E** are parameters on the **INTERSTITIAL** statement.

Enabling, Disabling, and Initialization

The interstitial trap model is enabled by setting **TRAP.CON** to a nonzero value and disabled by setting **TRAP.CON** to zero. The empty trap concentration T_E is initialized to its equilibrium value T_E^* at the start of the first diffusion step (**DIFFUSION** or **EPITAXY** statement) after the trap model is first enabled. In some cases it may be desirable to initialize the empty trap concentration to a value smaller than its equilibrium value. This can be accomplished by first setting T_E^* to the desired initial concentration of empty traps, doing a short diffusion (with **TRAP.CON** set nonzero to enable the trap model), then setting T_E to the actual equilibrium concentration of empty traps.

Interstitial Clustering Model

TSUPREM-4 includes a model for the formation and dissolution of interstitial clusters ($\{311\}$ or $\{113\}$ defects). These clusters play an important part in transient-enhanced diffusion (TED) of impurities following ion implantation. The main effect of the model is to delay the onset of TED at low temperatures and to distribute the diffusion enhancement over a longer period of time. This eliminates the excessive diffusion at low temperatures that is predicted by older versions of TSUPREM-4.

Model Equations

The kinetics of $\{311\}$ formation and dissolution are not well understood, and several models have been proposed (e.g., [12]). TSUPREM-4 therefore uses a generalized model that includes many of the proposed models as subsets. This is accomplished by including two terms describing the clustering of interstitials and one describing the declustering:

$$R_c = \frac{\partial C}{\partial t} = K_{fi} \frac{I^{\mathbf{CL.IFI}}}{I^{*\mathbf{CL.ISFI}}} + K_{fc} \frac{I^{\mathbf{CL.IFC}}}{I^{*\mathbf{CL.ISFC}}} (C + \alpha I)^{\mathbf{CL.CF}} - K_r C^{\mathbf{CL.CR}} \tag{Equation 2-147}$$

where C is the concentration of clustered interstitials, I is the concentration of unclustered interstitials, and I^* is the equilibrium concentration of interstitials. The reactions constants are given by

$$K_{fi} = \mathbf{CL.KFI.O} \cdot \exp\left(-\frac{\mathbf{CL.KFI.E}}{kT}\right) \tag{Equation 2-148}$$

$$K_{fc} = \text{CL.KFC.0} \cdot \exp\left(-\frac{\text{CL.KFC.E}}{kT}\right) \quad \text{Equation 2-149}$$

$$K_r = \text{CL.KR.0} \cdot \exp\left(-\frac{\text{CL.KR.E}}{kT}\right) \quad \text{Equation 2-150}$$

and $\alpha = \text{CL.KFCI}$, CL.IFI , CL.ISFI , CL.IFC , CL.ISFC , CL.CF , CL.CR , CL.KFI.0 , CL.KFI.E , CL.KFC.0 , CL.KFC.E , CL.KR.0 , and CL.KR.E are parameters specified on the **INTERSTITIAL** statement.

All changes in C are accompanied by corresponding inverse changes in I . Thus clustering reduces the number of free interstitials, while the dissolution of clusters releases interstitials.

The clustered interstitial concentration C is initialized to the value specified for **ABS.ERR** ($10^9/\text{cm}^3$ in silicon) when the point defect models are first activated. After that, the concentration is determined by [Equation 2-147](#).

Choosing Model Parameters

The interstitial clustering model is designed to work automatically, but for correct operation suitable parameter values must be chosen. Parameters for the model are set on the **INTERSTITIAL** statement specifying **MATERIAL=SILICON** as the material. Parameters should be chosen so that:

1. Clusters form rapidly when the interstitial concentration is very high (i.e., following an implant).
2. Clusters decay at a suitable rate when the interstitial concentration is low.
3. There is a small but nonzero concentration of clustered interstitials at equilibrium. This is required for numerical reasons, and may be needed to start the clustering process following a subsequent implant.
4. The clustering model does not interfere with simulations of high-concentration diffusion, oxidation-enhanced diffusion, or other situations in which non-equilibrium interstitial concentrations may be present.
5. The clustering model is numerically well behaved.

To obtain clustering following an implant, either the K_{fi} or K_{fc} term (or both) must be nonzero. For rapid clustering in response to an excess of interstitials, **CL.IFI**, **CL.IFC**, and/or **CL.CF** must be set appropriately; larger values increase the dependence of the clustering rate on the interstitial concentration. Use **CL.ISFI** and **CL.ISFC** to modify the clustering rate at high doping concentrations.

Dissolution of clusters and the release of interstitials requires a nonzero K_r term. The parameter **CL.CR** must be greater than zero to avoid numerical difficulties (a value of 1.0 is typical). The forward reaction terms are also active during dissolution, so the net dissolution rate depends on the K_{fi} and K_{fc} terms as well as on the K_r term.

The equilibrium concentration of clustered interstitials depends on the balance between the K_{fi} and K_{fc} terms and the K_r term. It can be found by setting $\partial C/\partial t = 0$ in Equation 2-147 and solving for C .

The clustering model should also be examined to determine its impact on high-concentration diffusion (with **PD.FULL**) or OED. In either of these cases, excess interstitials could produce clustering, which can affect the diffusion of impurities.

The interstitial clustering model has been designed to be numerically well behaved when used with “reasonable” parameter values. In general this means that the clustering and dissolution rates must not be too large and the concentration of clustered interstitials must approach a reasonable value in equilibrium. In practice, it has been found that parameter values that give physically reasonable results are numerically well behaved.

The default parameter values use only the K_{fc} and K_r terms. They have been chosen to fit the clustering data of Poate, et al. [13] when used with the default values of the point defect parameters.

Using the Model

The model is enabled by using the **L.MODEL** parameter on the **INTERSTITIAL** statement:

```
INTERST MAT=SILICON CL.MODEL
```

The model is enabled by default. To disable the model use:

```
INTERST MAT=SILICON ^CL.MODEL
```

The concentration of clustered interstitials is obtained by specifying the name *cl_interst* in the **SELECT** statement:

```
SELECT Z=LOG10(CL_INTERST)
```

In *Avant!* TCAD's graphical post-processing programs and in *TIF* files, the name *clInterst* is used.

Small Clusters of Point Defects

At high concentrations, point defects will tend to form small, loosely-bound clusters, reducing the number of free defects available to enhance diffusion of dopants.

Concentration of Defects in Small Clusters

These are modeled in TSUPREM-4 by assuming that the number of defects in small clusters in equilibrium with the number of free defects:

$$I_c = \text{ECLUST.N} \sum_i \left(K_{Ii} \eta^{-i} \right) \left(\frac{I}{I^*} \right)^{\text{ECLUST.N}}$$

$$V_c = \text{ECLUST.N} \sum_j \left(K_{Vj} \eta^{-j} \right) \left(\frac{V}{V^*} \right)^{\text{ECLUST.N}}$$

Equation 2-151

$$K_{Ii} = \mathbf{ECLUST.0} \cdot \exp\left(\frac{-\mathbf{ECLUST.E}}{kT}\right)$$

$$K_{Vj} = \mathbf{ECLUST.0} \cdot \exp\left(\frac{-\mathbf{ECLUST.E}}{kT}\right)$$

Equation 2-152

where **ECLUST.0**, **ECLUST.E**, and **ECLUST.N** for interstitials of charge state **C.STATE = i** and vacancies of charge state **C.STATE = j** are specified on the **INTERSTITIAL** and **VACANCY** statements, respectively. **ECLUST.N** denotes the number of defects in the cluster (typically 2–4) while **ECLUST.0** and **ECLUST.E** specify the concentration of defects in small clusters when the free point defect concentrations are in thermal equilibrium.

Recombination of Defects in Small Clusters

Vacancies can react with small clusters of interstitials to reduce the total number of interstitials and vacancies:

$$F_{cV} = \left(\frac{I}{I^*}\right)^{\mathbf{ECLUST.N}-1} V_i^* \sum_{i,j} \left(K_{cV,ij} K_{Ii} \eta^{-i} \phi_{Vj} \eta^{-j} \right)$$

Equation 2-153

$$K_{cV,ij} = \mathbf{KCV.0} \cdot \exp\left(\frac{-\mathbf{KCV.E}}{kT}\right)$$

Equation 2-154

where **KCV.0** and **KCV.E** (parameters on the **INTERSTITIAL** statement) specify the relative reaction rate between small interstitial clusters in charge state **C.I = i** and vacancies in charge state **C.V = j**. Similarly, interstitials can react with small clusters of vacancies to reduce the total number of interstitials and vacancies:

$$F_{Ic} = I_i^* \left(\frac{V}{V^*}\right)^{\mathbf{ECLUST.N}-1} \sum_{i,j} \left(K_{Ic,ij} \phi_{Ii} \eta^{-i} K_{Vj} \eta^{-j} \right)$$

Equation 2-155

$$K_{Ic,ij} = \mathbf{KIC.0} \cdot \exp\left(\frac{-\mathbf{KIC.E}}{kT}\right)$$

Equation 2-156

where **KIC.0** and **KIC.E** (parameters on the **VACANCY** statement) specify the relative reaction rate between interstitials in charge state **C.I = i** and small vacancy clusters in charge state **C.V = j**. **KCV.0**, **KCV.E**, **KIC.0**, and **KIC.E** for all charge state combinations can be set using **C.ALL**.

The number of point defects in small clusters can be significant following an implant with damage, especially at low temperatures. These clusters can reduce the number of free point defects and the amount of diffusion enhancement of dopants.

Oxidation

Oxidation occurs whenever a **DIFFUSION** statement specifies an oxidizing ambient and either exposed silicon or polysilicon or exposed oxide adjacent to silicon or polysilicon is present in the structure. An oxidizing ambient is one containing O₂, H₂O, or N₂O. The flows of O₂, H₂O, or N₂O can be specified directly on the **DIFFUSION** statement or a predefined ambient containing O₂, H₂O, or N₂O (i.e., **DRYO2**, **WETO2**, **STEAM**, or **N2O**) can be specified (see “Diffusion” on page 2-13, and on page 3-115, and “AMBIENT” on page 3-207). If an ambient contains O₂ or N₂O with H₂O, the oxidation rate is based on the partial pressure of H₂O.

Oxidation occurs at points in the structure where an oxide region is adjacent to a silicon or polysilicon region. If a structure contains an exposed silicon or polysilicon surface at the start of an oxidation step, a native oxide of thickness **INITIAL** (specified on the **AMBIENT** statement) is deposited on this surface before proceeding with the oxidation.

Five oxidation models are available. All are based on the one-dimensional theory of Deal and Grove [17]. The differences are in the way they extend the Deal-Grove model to two dimensions. “Theory of Oxidation” on page 2-46 outlines the basic theory of oxidation, while the sections “Analytical Oxidation Models” on page 2-48 through “VISCOELA Model” on page 2-60 describe how the theory is applied by the models available in TSUPREM-4.

Theory of Oxidation

Oxidation in TSUPREM-4 is based on the theory of Deal and Grove [17], which is briefly outlined. The flux of oxidant (assumed to be O₂ or H₂O) entering the oxide from the ambient gas is given by

$$\vec{F} = h(C^* - C_o)\vec{n}_s \quad \text{Equation 2-157}$$

where

- h is the gas-phase mass-transfer coefficient
- C_o is the concentration of oxidant in the oxide at the surface
- $C^* = HP_{ox}$ where
 H is the Henry's law coefficient for the oxidant in oxide and
 P_{ox} is the partial pressure of oxidant in the ambient
- \vec{n}_s is the unit vector normal to the oxide surface, pointing towards the oxide

The flux of oxidant in the oxide is

$$\vec{F} = D\vec{\nabla}C \quad \text{Equation 2-158}$$

where

- D is the diffusivity of oxidant in the oxide

- C is the local concentration of oxidant
- $\vec{\nabla}$ is the gradient operator

The rate of oxidant consumption at the oxidizing interface is

$$\vec{F} = k_s C \vec{n}_i \tag{Equation 2-159}$$

where

- k_s is the surface reaction rate
- C is the oxidant concentration at the interface
- \vec{n}_i is the unit vector normal to the interface pointing away from the oxide

In steady state, the divergence of the fluxes is zero:

$$\vec{\nabla} \cdot \vec{F} = 0 \tag{Equation 2-160}$$

The oxide growth rate is given by

$$\frac{d\vec{Y}}{dt} = \frac{\vec{F}}{N_1} + r_{thin} \tag{Equation 2-161}$$

where

- $d\vec{Y}/dt$ represents the interface velocity relative to the oxide
- N_1 is the number of oxidant molecules needed to form each cubic centimeter of oxide

The r_{thin} term models the rapid growth that is seen during the initial stages of oxidation; the calculation of this term depends on whether an analytical or numerical model is used.

In one dimension, [Equations 2-157](#) through [2-161](#) can be solved to give

$$\frac{dy}{dt} = \frac{B}{A + 2y} \tag{Equation 2-162}$$

where y is the oxide thickness and A and B are given by

$$A = 2D \left(\frac{1}{k_s} + \frac{1}{h} \right) \tag{Equation 2-163}$$

$$B = \frac{2DC^*}{N_1} \tag{Equation 2-164}$$

In TSUPREM-4, A and B are specified in terms of the linear and parabolic rate constants B/A and B , respectively.

The analytical models are based on Equation 2-162, while the numerical models use Equations 2-157 through 2-161. Values of D and k_s for the numerical models are derived from Equations 2-163 and 2-164, using the user-specified linear and parabolic rate constants.

Analytical Oxidation Models

There are two analytical models, **ERFC** and **ERFG** (**ERF1** and **ERF2** are subsets of **ERFG**). They differ in how the growth rate depends on the x coordinate near the mask edge.

Overview

The analytical oxidation models in TSUPREM-4 are designed for fast simulation of simple structures. They are limited to structures consisting of a planar silicon substrate covered by an optional initial oxide layer; masking layers over the oxide are ignored. Oxidation occurs to the right of an assumed mask location specified by **MASK.EDG** on the **AMBIENT** statement. Any actual mask layers on the structure move vertically with the surface of the growing oxide, but do not affect the oxide shape. Furthermore, the **ERF1**, **ERF2**, and **ERFG** models assume that the initial silicon surface is planar, at $y = 0$.

Oxide Growth Rate

The analytical oxidation models are based on Equation 2-162, with an added term to model thin oxide growth. Far to the right of **MASK.EDG** the oxidation rate is

$$\frac{dy_\infty}{dt} = \frac{B}{A + 2y_\infty} + r_{thin} \quad \text{Equation 2-165}$$

where

- y_∞ is the unmasked (one-dimensional) oxide thickness
- B/A , B , and r_{thin} are the linear, parabolic, and thin regime oxidation rate constants, respectively, described below.

Thin Regime

The thin regime oxidation rate constant is given by [18]

$$r_{thin} = \text{THINOX.0} \cdot \exp\left(\frac{-\text{THINOX.E}}{kT}\right) \exp\left(\frac{-y_\infty}{\text{THINOX.L}}\right) \quad \text{Equation 2-166}$$

where the parameters **THINOX.0**, **THINOX.E**, and **THINOX.L** are specified in the **AMBIENT** statement (see page 3-207) for each of the oxidizing species. Different values of **THINOX.0**, **THINOX.E**, and **THINOX.L** can be defined for each of the three available silicon orientations and for polysilicon by specifying the **<111>**, **<110>**, **<100>**, or **POLYSILI** parameters in the **AMBIENT** statement.

Linear Rate

The linear oxidation rate constant is given by [3], [19], [20], [21]

$$B/A = l_0 l_P l_C l_{Cl} \tag{Equation 2-167}$$

where l_0 represents the intrinsic linear oxidation rate and l_P , l_C , and l_{Cl} represent the dependence on partial pressure, carrier concentration, and chlorine concentration, respectively. The intrinsic linear oxidation rate is given by

$$l_0 = \begin{cases} \mathbf{L.LIN.O} \cdot \exp\left(\frac{-\mathbf{L.LIN.E}}{kT}\right) & T_c < \mathbf{LIN.BREA} \\ \mathbf{H.LIN.O} \cdot \exp\left(\frac{-\mathbf{H.LIN.E}}{kT}\right) & T_c \geq \mathbf{LIN.BREA} \end{cases} \tag{Equation 2-168}$$

where T_c is the diffusion temperature in °C, and the parameters **L.LIN.O**, **L.LIN.E**, **H.LIN.O**, **H.LIN.E**, and **LIN.BREA** are specified in the **AMBIENT** statement for each of the oxidizing species. Different values of **L.LIN.O**, **L.LIN.E**, **H.LIN.O**, and **H.LIN.E** can be defined for each of the three available silicon orientations and for polysilicon by specifying the **<111>**, **<110>**, **<100>**, or **POLYSILI** parameters in the **AMBIENT** statement.

The partial pressure dependence of the linear oxidation rate is given by

$$l_P = \left(\frac{P}{1 \text{ atmosphere}}\right)^{\mathbf{LIN.PDEP}} \tag{Equation 2-169}$$

where the parameter **LIN.PDEP** is specified in the **AMBIENT** statement for each of the oxidizing species.

The concentration dependence l_C is only used with the numeric models; it is described in [“Numerical Oxidation Models” on page 2-54](#).

The chlorine dependence of the linear oxidation rate is obtained by interpolating within a table of values depending on the chlorine percentage and the diffusion temperature (see [“DIFFUSION Statement” on page 2-13](#)). The values in the table are specified with the **LIN.CLDE**, **COLUMN**, **TEMPERAT**, **LIN.PCT**, and **TABLE** parameters in the **AMBIENT** statement for each of the oxidizing species. Values of l_{Cl} are specified with the **LIN.CLDE** and **TABLE** parameters for the column of the table defined by the **COLUMN** parameter. The chlorine percentages associated with the rows of the table are defined with the **LIN.PCT** and **TABLE** parameters. The temperature associated with each column of the table is defined with the **TEMPERAT** parameter.

Parabolic Rate

The parabolic oxidation rate constant is given by [3], [19], [20], [21]

$$B = p_0 p_P p_{Cl} \tag{Equation 2-170}$$

where p_0 represents the intrinsic parabolic oxidation rate and p_P and p_{Cl} represent the dependence on partial pressure and chlorine concentration, respectively. The intrinsic parabolic oxidation rate is given by

$$p_0 = \begin{cases} \mathbf{L.PAR.0} \cdot \exp\left(\frac{-\mathbf{L.PAR.E}}{kT}\right) & T_c < \mathbf{PAR.BREA} \\ \mathbf{H.PAR.0} \cdot \exp\left(\frac{-\mathbf{H.PAR.E}}{kT}\right) & T_c \geq \mathbf{PAR.BREA} \end{cases} \quad \text{Equation 2-171}$$

where T_c is the diffusion temperature in °C, and the parameters **L.PAR.0**, **L.PAR.E**, **H.PAR.0**, **H.PAR.E**, and **PAR.BREA** are specified in the **AMBIENT** statement for each of the oxidizing species.

The partial pressure dependence of the parabolic oxidation rate is given by

$$p_P = \left(\frac{P}{1 \text{ atmosphere}}\right)^{\mathbf{PAR.PDEP}} \quad \text{Equation 2-172}$$

where the parameter **PAR.PDEP** is specified in the **AMBIENT** statement for each of the oxidizing species.

The chlorine dependence of the parabolic oxidation rate is obtained by interpolating within a table of values depending on the chlorine percentage and the diffusion temperature (see “**DIFFUSION** Statement” on page 2-13).

The values in the table are specified with the **PAR.CLDE**, **COLUMN**, **TEMPERAT**, **PAR.PCT**, and **TABLE** parameters in the **AMBIENT** statement for each of the oxidizing species. Values of p_{Cl} are specified with the **PAR.CLDE** and **TABLE** parameters for the column of the table defined by the **COLUMN** parameter.

The chlorine percentages associated with the rows of the table are defined with the **PAR.PCT** and **TABLE** parameters. The temperature associated with each column of the table is defined with the **TEMPERAT** parameter.

Usage

Oxide growth is vertical, with the oxide interface moving in the +y direction and everything above it moving in the -y direction. (Actually, all silicon nodes, except for those at the oxide interface, remain stationary, while all nonsilicon nodes, except for oxide nodes at the interface, move with the oxide surface. This is the origin of the restrictions on the device structure for the analytical models.)

The analytical models are appropriate for oxidation of planar or near-planar surfaces; they are not appropriate for very nonplanar structures, or structures containing non-silicon layers below the top layer of oxide (e.g., silicon-on-insulator structures). Likewise, the analytical models cannot be used to simulate oxidation of polysilicon.

The unmasked thickness y_∞ is calculated by numerically integrating Equation 2-165, starting with the initial value specified by the **INITIAL** parameter on the **AMBIENT** statement. The analytical models are only correct when the value of the **INITIAL** parameter is equal to the actual initial oxide thickness. Because the **INITIAL** parameter also specifies the thickness of the native oxide

deposited on exposed silicon surfaces prior to oxidation, the results are accurate when oxidizing bare silicon.

The ERF C Model

The **ERFC** model is the simplest oxidation model available in TSUPREM-4. It is selected by specifying the **ERFC** parameter on the **METHOD** statement (see [page 3-188](#)). The **ERFC** model uses one parameter, **SPREAD**, in addition to the **INITIAL** and **MASK.EDG** parameters. All three parameters are specified on the **AMBIENT** statement (see [page 3-207](#)).

The oxidation rate as a function of x is given by

$$\frac{dy(x)}{dt} = \frac{1}{2} \operatorname{erfc}\left(\frac{\sqrt{2}}{\text{SPREAD}} \cdot \frac{x_o - x}{y_\infty - \text{INITIAL}}\right) \frac{dy_\infty}{dt} \quad \text{Equation 2-173}$$

where

- $x_o = \text{MASK.EDG}$ is the position of the mask edge
- y_∞ is the unmasked oxide thickness

The **SPREAD** parameter controls the width of the “bird’s beak” relative to the unmasked oxide thickness. From [Equation 2-173](#), you can see that the growth rate at x_o is half the unmasked growth rate; thus **MASK.EDG** actually represents the half-thickness point, not necessarily the true location of the mask edge.

Recommended Usage

The **ERFC** model is accurate for one-dimensional simulations, provided that the coefficients are accurate, **INITIAL** is correctly set or a bare silicon surface is being oxidized, and the dependence of the oxidation rate on doping can be neglected. The **ERFC** model can also be used for simulating local oxidation of planar or near-planar structures, if accurately calibrated. It is the fastest of the oxidation models, but this speed advantage is rarely significant. The **ERFC** model does not simulate the oxidation of polysilicon, nor does it take into account masking layers. Because of its limitations, the **ERFC** is not often used.

The ERF1, ERF2, and ERF G Models

The **ERFG** model is a more complex analytical model for oxidation of silicon under a nitride mask. It is based on the work of Guillemot, et al. [22] and contains two models, selected by the **ERF1** and **ERF2** parameters on the **METHOD** statement (see [page 3-188](#)). If **ERFG** is specified, **ERF1** or **ERF2** is selected by the program based on the initial pad oxide and nitride mask thicknesses.

Parameters

The **ERF1** model uses three parameters, **NIT.THIC**, **ERF.H**, and **ERF.LBB** in addition to the **INITIAL** and **MASK.EDG** parameters. The **ERF2** model uses two additional parameters, **ERF.Q** and **ERF.DELT**. All seven parameters are specified on the **AMBIENT** statement (see [page 3-207](#)).

Initial Structure

The **ERF1**, **ERF2**, and **ERFG** models assume that the initial structure consists of a planar silicon surface at $y = 0$, with a pad oxide of thickness **INITIAL** and a

nitride mask of thickness **NIT.THIC** to the left of **MASK.EDG**. The nitride mask need not be present in the simulated structure.

ERF1 Model

The **ERF1** model is used when the thickness of the nitride mask is small compared to the pad oxide thickness. The oxidation rate as a function of x for the **ERF1** model is given by

$$\frac{dy(x)}{dt} = \frac{1}{2} \operatorname{erfc} \left[\frac{x_o - x}{\mathbf{ERF.LBB}} \left(\sqrt{c^2 + \ln(10)} - c \right) + c \right] \frac{dy_\infty}{dt} \quad \text{Equation 2-174}$$

where

$$c = \frac{\sqrt{\pi}}{2} \left(1 - \frac{2H'}{1 - 0.44} \right) \quad \text{Equation 2-175}$$

and $x_o = \mathbf{MASK.EDG}$ is the position of the mask edge, $H' = \mathbf{ERF.H} / (1 - 0.44)$, and **ERF.LBB** and **ERF.H** are user-supplied empirical expressions. The default values of **ERF.LBB** and **ERF.H** are

$$\mathbf{ERF.LBB} = 8.25 \times 10^{-3} (1580.3 - T_{ox}) y_\infty^{0.67} \mathbf{INITIAL}^{0.3} \quad \text{Equation 2-176}$$

$$\times \exp \left(- \frac{(\mathbf{NIT.THIC} - 0.08)^2}{0.06} \right)$$

$$\mathbf{ERF.H} = 402 \times (0.445 - 1.75 \times \mathbf{NIT.THIC}) e^{-T_{ox}/200} \quad \text{Equation 2-177}$$

where

- T_{ox} is the oxidation temperature (in degrees Kelvin)
- y_∞ is the unmasked oxide thickness
- **INITIAL** and **NIT.THIC** are the pad oxide and nitride mask thicknesses, respectively

ERF2 Model

The **ERF2** model is used when the nitride is thick compared to the pad oxide thickness. The oxidation rate as a function of x for the **ERF2** model is given by

$$\frac{dy(x)}{dt} = \begin{cases} H' \operatorname{erfc}\left(\frac{x_o - \delta - x}{\mathbf{ERF.LBB} \sqrt{\ln 10}}\right) & x < x_o - \delta \\ \frac{x - (x_o - \delta) + \frac{H' \mathbf{ERF.Q}}{1 - H'}}{x - (x_o - \delta) + \frac{\mathbf{ERF.Q}}{1 - H'}} & x \geq x_o - \delta \end{cases} \quad \text{Equation 2-178}$$

where

- **ERF.LBB** is a parameter on the **AMBIENT** statement (with a default value of 0.05 microns)
- H' , x_o , and **ERF.LBB** have the same meanings and values as for the **ERF1** model

The value used for δ depends on the y coordinate of the point under consideration:

$$\delta = \begin{cases} 0 & y \leq -10^{-5} \text{ microns} \\ \mathbf{ERF.DELT} & y > -10^{-5} \text{ microns} \end{cases} \quad \text{Equation 2-179}$$

where **ERF.DELT** is a parameter on the **AMBIENT** statement (with a default value of 0.04 microns). Thus the breakpoint in Equation 2-178 occurs at **MASK.EDG** when calculating the position of the top surface of the oxide ($y \leq -10^{-5}$ microns), but is offset by **ERF.DELT** when calculating the position of the silicon/oxide interface ($y > -10^{-5}$ microns).

ERFG Model

The **ERFG** model selects either **ERF1** or **ERF2** depending on the values of **INITIAL**, and **NIT.THIC**. **ERF1** is used if $\mathbf{NIT.THIC} \leq 2.5 \times \mathbf{INITIAL} + 0.035 \mu\text{m}$, and **ERF2** is used otherwise.

Recommended Usage

The **ERFG** models provide a fast, analytical simulation of local oxidation of planar surfaces. The accuracy of this model has not been determined. (Note that these models differ slightly from those proposed by Guillemot et al. Guillemot computes the final oxide shape as a function of the final field oxide thickness, while TSUPREM-4 uses the same equations to calculate the oxide growth rate as a function of the field oxide thickness at each time point in the simulation.)

The **ERFG** shares all the limitations of the **ERFC** model. In addition, it places more restrictions on the initial structure and has more parameters that need to be determined. The **ERFC** model is rarely used in practical simulations.

Numerical Oxidation Models

There are four numerical models of oxidation, **VERTICAL**, **COMPRESS**, **VISCOUS**, and **VISCOELA** in TSUPREM-4. These models are designed for accurate simulation of arbitrary structures, using the masking layers present in the simulated structure. They work by solving [Equations 2-157 through 2-161](#) directly to obtain the growth rate at each point on the oxide/silicon interface. The models differ in the way in which the oxide flow caused by volume expansion is calculated. This section describes the calculation of the oxide growth rate. Sections “[The VERTICAL Model](#)” on page 2-56, “[COMPRESS Model](#)” on page 2-57, “[The VISCOUS Model](#)” on page 2-58 and “[VISCOELA Model](#)” on page 2-60 discuss how the four models simulate the oxide flow.

Oxide Growth Rate

The oxide growth rate is calculated using [Equations 2-157 through 2-161](#). [Equations 2-157](#) gives the flux of oxidant molecules entering the oxide from the ambient gas. The gas-phase mass transfer coefficient h is given by

$$h = \text{TRANS} \cdot 0 \cdot \exp\left(\frac{-\text{TRANS} \cdot E}{kT}\right) \quad \text{Equation 2-180}$$

where **TRANS . 0** and **TRANS . E** are specified on the **AMBIENT** statement with the **OXIDE** and **/AMBIENT** parameters (see [page 3-207](#)). Henry's law coefficient H is specified for each oxidizing species (O_2 or H_2O) with the **HENRY . CO** parameter on the **AMBIENT** statement.

Diffusion of oxidant through the oxide is modeled by [Equation 2-158](#). The diffusivity is calculated from the parabolic oxidation rate B using [Equations 2-163 and 2-164](#):

$$D = B \frac{N_1}{2C^*} \quad \text{Equation 2-181}$$

where N_1 is given by

$$N_1 = \begin{cases} \text{THETA} & \text{for } O_2 \\ 2 \times \text{THETA} & \text{for } H_2O \end{cases} \quad \text{Equation 2-182}$$

and **THETA** is a parameter on the **AMBIENT** statement. The parabolic oxidation rate B is given by [Equation 2-170](#) in “[Analytical Oxidation Models](#)” on page 2-48. The diffusivity in other materials is given by

$$D = \text{D} \cdot 0 \cdot \exp\left(\frac{-\text{D} \cdot E}{kT}\right) \quad \text{Equation 2-183}$$

where **D . 0** and **D . E** are parameters on the **AMBIENT** statement.

The rate at which the oxidant molecules react with silicon at the oxide/silicon interface is given by Equation 2-159. The reaction rate k_s is derived from Equation 2-163:

$$k_s = \frac{BN_1}{AC^*} \tag{Equation 2-184}$$

where it is assumed that $k \ll h$. The value of B/A is the linear growth rate given by Equation 2-167 in "Analytical Oxidation Models" on page 2-48.

Concentration Dependence

The electron concentration dependence l_C of the linear oxidation rate is given by [23], [24], [25]

$$l_C = \begin{cases} 1 & \text{analytical oxidation models} \\ 1 + \gamma_V(C_V - 1) & \text{numerical oxidation models} \end{cases} \tag{Equation 2-185}$$

where

$$\gamma_V = \text{GAMMA.0} \cdot \exp\left(\frac{-\text{GAMMA.E}}{kT}\right) \tag{Equation 2-186}$$

$$C_V = \frac{1 + C^+ \left(\frac{n_i}{n}\right) + C^- \left(\frac{n}{n_i}\right) + C^= \left(\frac{n}{n_i}\right)^2}{1 + C^+ + C^- + C^=} \tag{Equation 2-187}$$

The parameters **GAMMA.0** and **GAMMA.E** are specified in the **AMBIENT** statement for each of the oxidizing species; n is the electron concentration in the silicon at the oxidizing interface. The terms C^+ , C^- , and $C^=$ are the normalized intrinsic concentrations of positive, negative, and double negative vacancies, respectively, given by

$$C^+ = \exp\left(\frac{E^+ - E_i}{kT}\right) \tag{Equation 2-188}$$

$$C^- = \exp\left(\frac{E_i - E^-}{kT}\right) \tag{Equation 2-189}$$

$$C^= = \exp\left(\frac{2E_i - E^- - E^=}{kT}\right) \tag{Equation 2-190}$$

$$E^+ = 0.35 \text{ eV} \tag{Equation 2-191}$$

$$E^- = E_g - 0.57 \text{ eV} \tag{Equation 2-192}$$

$$E^= = E_g - 0.12 \text{ eV} \tag{Equation 2-193}$$

$$E_i = \frac{E_g}{2} + 0.75 \ln(0.719) kT \text{ eV} \tag{Equation 2-194}$$

$$E_g = 1.17 - \frac{4.73 \times 10^{-4} T^2}{T + 636} \text{ eV} \tag{Equation 2-195}$$

The dependence on carrier concentration is a function of the location along the oxidizing interface, and is only used with the numerical oxidation models. The surface reaction rate depends on the local crystal orientation of the interface, which is derived from the user-specified substrate orientation and rotation.

Thin Regime The thin regime oxidation rate constant in [Equation 2-161](#) is given by [18]

$$r_{thin} = \text{THINOX} \cdot \mathbf{0} \cdot \exp\left(\frac{-\text{THINOX} \cdot \mathbf{E}}{kT}\right) \exp\left(\frac{-y_{eff}}{\text{THINOX} \cdot \mathbf{L}}\right) \tag{Equation 2-196}$$

where the parameters **THINOX . 0**, **THINOX . E**, and **THINOX . L** are specified in the **AMBIENT** statement (see [page 3-207](#)) for each of the oxidizing species. Different values of **THINOX . 0**, **THINOX . E**, and **THINOX . L** can be defined for each of the three available silicon orientations and for polysilicon by specifying the **<111>**, **<110>**, **<100>**, or **POLYSILI** parameters in the **AMBIENT** statement. The effective thickness y_{eff} at each point on the oxidizing interface is calculated as the oxide thickness required to produce the observed oxidant concentration in a one-dimensional solution to the Deal-Grove equations.

Usage The numerical models have no restrictions regarding initial oxide thickness. The value of **INITIAL** need not correspond to the oxide thickness in the starting structure, but is still used for the thickness of the native oxide to be deposited on bare silicon or polysilicon surfaces prior to oxidation.

The VERTICAL Model The **VERTICAL** model is the simplest of the numerical oxidation models in TSUPREM-4. In this model, the oxide/silicon interface is constrained to move in the +y direction while the expansion of the oxide occurs in the -y direction. The implementation assumes that there is a single active oxide/silicon interface, with the oxide on top. All layers above the interface move with the oxide surface, and all layers below the interface remain fixed. Because the growth is only in the y direction, the growth rate depends only on the orientation of the silicon substrate (and not on the orientation of the interface).

Recommended Usage The **VERTICAL** model is the fastest of the numerical oxidation models. It is useful for uniform oxidation with arbitrary initial oxide thickness (**INITIAL** need

not be set), and for local oxidation when the initial structure is approximately planar. It is not appropriate for fully recessed oxides, trenches, or other nonplanar structures. The **VERTICAL** model can be used to model silicon-on-insulator structures, provided that only one silicon/oxide interface is being oxidized. There must be no path for oxidant to diffuse to underlying oxide layers. The **VERTICAL** model does not simulate oxidation of polysilicon.

COMPRESS Model

The **COMPRESS** model simulates viscous flow of the oxide during oxidation. The movement of the oxide/silicon interface is two dimensional (movement is perpendicular to the interface), and variation of crystal orientation along the interface is taken into account when calculating the oxidation rate. The viscous flow of the oxide is calculated in two dimensions using linear (3-node) finite elements.

The name **COMPRESS** comes from the fact that a small amount of compressibility must be allowed to ensure that the model remains numerically well-behaved. Because of the simplicity of the model and the small amount of compressibility, the model cannot be used to calculate accurate values of stress, and does not include the effects of stress on the oxidation process.

Compressible Viscous Flow

The equations governing oxide flow were derived by Chin, et al. [26]. The oxide flow is described by a creeping-flow equation:

$$\mu \nabla^2 \vec{V} = \vec{\nabla} P \tag{Equation 2-197}$$

where

- μ is the shear viscosity of oxide
- \vec{V} is the local velocity
- P is the hydrostatic pressure

The incompressibility condition can be written

$$\vec{\nabla} \cdot \vec{V} = 0 \tag{Equation 2-198}$$

To solve these equations with the 3-node element, the incompressibility condition is modified to relate the divergence of velocity to the pressure:

$$\vec{\nabla} \cdot \vec{V} = - \left(\frac{1-2\nu}{\mu} \right) P \tag{Equation 2-199}$$

where ν , specified by **POISS.R** on the **MATERIAL** statement (page 3-228), is a parameter analogous to Poisson's ratio. The viscosity is specified by analogy to an elastic model:

$$\mu = \frac{E}{2(1+\nu)} \tag{Equation 2-200}$$

where E is specified by the **YOUNG.M** parameter on the **MATERIAL** statement.

Equations 2-197, 2-199, and 2-200 are applied to all materials except single-crystal silicon, which remains fixed.

Boundary Conditions

The stress in the direction normal to a material interface is continuous across the interface, while the stress normal to a free surface is zero. There is no slippage between materials at an interface.

Model Parameters

Default values of **YOUNG.M** and **POISS.R** are provided for each material, except for aluminum and photoresist, which should not be present in the structure during oxidation; users must provide values for these parameters for user-defined materials that are present in the structure during oxidation with the **COMPRESS** model.

COMPRESS Model: Recommended Usage

The **COMPRESS** model is recommended for simulating the oxidation of general nonplanar structures and structures containing polysilicon when stress calculations are not required. Because it does not include the effects of stress on the oxidation process, the **COMPRESS** model should be used only when one or more of the following conditions is satisfied:

- Structure is planar
- Amount of oxide grown is small
- Exact details of the shape of the oxide are not critical

When none of these conditions is satisfied, the **VISCOELA** or **VISCOUS** model should be used. The **COMPRESS** mode is slower than the **VERTICAL** model and has somewhat larger memory requirements.

VISCOUS Model

The **VISCOUS** model simulates viscous flow of the oxide during oxidation. The movement of the oxide/silicon interface is two-dimensional (movement is perpendicular to the interface), and variation of crystal orientation along the interface is taken into account when calculating the oxidation rate. The viscous flow of the oxide is calculated in two dimensions using 7-node finite elements, which allow accurate values of stress to be computed [27].

Incompressible Viscous Flow

The equations and boundary conditions governing oxide flow are the same as for the **COMPRESS** model (Equations 2-197 through 2-200), except that surface tension is included (refer to “Surface Tension and Reflow” on page 2-63) and a different set of parameters is used. The viscosity is specified for each material by

$$\mu_o = \text{VISC.0} \cdot \exp\left(\frac{-\text{VISC.E}}{kT}\right) \quad \text{Equation 2-201}$$

where **VISC.0** and **VISC.E** are parameters on the **MATERIAL** statement (see page 3-228). The parameter ν , which determines the degree of compressibility, is specified as **VISC.X** on the **MATERIAL** statement. The default value of ν for all materials (except aluminum and photoresist) is 0.499, which produces a negligible amount of compressibility; the user must provide values for these parameters for

user-defined materials that are present in the structure during oxidation with the **VISCOUS** model.

The stresses are calculated from the constitutive equations for each material:

$$\sigma_{xx} + \sigma_{yy} = \frac{\mu}{\frac{1}{2} - \nu} \left(\frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} \right) \quad \text{Equation 2-202}$$

$$\sigma_{xx} - \sigma_{yy} = 2\mu \left(\frac{\partial u_x}{\partial x} - \frac{\partial u_y}{\partial y} \right) \quad \text{Equation 2-203}$$

$$\sigma_{xy} = \mu \left(\frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \right) \quad \text{Equation 2-204}$$

where u_x and u_y are the x and y components of velocity, and σ_{xx} , σ_{yy} , and σ_{xy} are the components of the stress tensor. The stresses in a purely viscous model are proportional to the flow velocities, and thus fall to zero when oxidation stops. The stresses reported by TSUPREM-4 after an oxidation step are the values calculated at the end of the step, before the flow is stopped. The stresses calculated by the **VISCOUS** model replace any stresses that may have been previously calculated using the **ST.HISTO** model or the **STRESS** statement.

Stress Dependence

When **STRESS.D** is set true on the **AMBIENT** statement, the surface reaction rate k_s , diffusivity of oxidant in oxide D , and the oxide viscosity μ are modified to reflect their dependence on the stresses in the oxide:

$$k'_s = k_s \exp\left(-\frac{\sigma_n \mathbf{VR}}{kT}\right) \exp\left(-\frac{\sigma_t \mathbf{VT}}{kT}\right) \quad \text{Equation 2-205}$$

$$D' = D \exp\left(-\frac{p \mathbf{VD}}{kT}\right) \quad \text{Equation 2-206}$$

$$\mu = \mu_o \frac{\sigma_s \mathbf{VC} / 2kT}{\sinh(\sigma_s \mathbf{VC} / 2kT)} \quad \text{Equation 2-207}$$

where k'_s , D' , and μ are the stress-dependent reaction rate, diffusivity, and viscosity, respectively, and **VR**, **VT**, **VD**, and **VC** are parameters on the **AMBIENT** statement. The exponential in Equation 2-206 is limited for positive arguments (i.e., negative values of p) to the value of **VDLIM** (a parameter on the **AMBIENT** statement) to prevent unrealistic enhancement of the diffusivity. The surface reaction rate depends both on the stress normal to the interface

$$\sigma_n = -(\sigma_{xx} n_x^2 + \sigma_{yy} n_y^2 + 2\sigma_{xy} n_x n_y) \quad \text{Equation 2-208}$$

and on the stress in the plane perpendicular to the interface

$$\sigma_t = - (\sigma_{xx}n_y^2 + \sigma_{yy}n_x^2 + 2\sigma_{xy}n_xn_y) \tag{Equation 2-209}$$

where n_x and n_y are the components of the unit vector normal to the interface. The oxidant diffusivity depends on the hydrostatic pressure p defined by

$$p = - \frac{1}{2}(\sigma_{xx} + \sigma_{yy}) \tag{Equation 2-210}$$

while the oxide viscosity depends on the total shear stress σ_s :

$$\sigma_s = \sqrt{\frac{1}{4}(\sigma_{xx} - \sigma_{yy})^2 + \sigma_{xy}^2} \tag{Equation 2-211}$$

Newton-Raphson iteration is used to solve the nonlinear equations produced by the stress dependences. Many iterations are usually required for convergence, and in some cases full convergence is not be obtained. Thus, the time required to simulate stress-dependent oxidation may be 20-200 times that required without the stress dependences.

Recommended Usage

The **VISCOUS** model has been made largely obsolete by the **VISCOELA** model. It is more accurate than the **VISCOELA** model when the viscosity is much smaller than Young's modulus, but is much slower when stress dependence is used. It is occasionally useful in verifying the results of the **VISCOELA** model, but it is often difficult to determine whether the difference between the two models is due to the differences in the physical model or to differences in numerical approach.

VISCOELA Model

The **VISCOELA** model simulates viscoelastic flow of the oxide during oxidation. The movement of the oxide/silicon interface is two-dimensional (movement is perpendicular to the interface), and variation of crystal orientation along the interface is taken into account when calculating the oxidation rate. The viscoelastic flow of the oxide is calculated in two dimensions by using three-node finite elements, which use numerical techniques that allow approximate values of stress to be computed. The model is similar to that developed by Senez, et al., [28] and in Reference [42] in Appendix A.

Viscoelastic Flow

The **VISCOELA** model adds an elastic component to the equations used by the **COMPRESS** and **VISCOUS** models. In one dimension, the stress σ is related to the strain ϵ by

$$\frac{\partial \sigma}{\partial t} = 2G \frac{\partial \epsilon}{\partial t} - \frac{\sigma}{\tau} \tag{Equation 2-212}$$

where

- G is the modulus of elasticity in shear and
- $\tau = \mu/G$ is the stress relaxation time (μ is the viscosity)

This states that the change in stress is proportional to the change in strain, minus a relaxation term that is proportional to the stress. In three dimensions, the equations are written most simply in terms of volumetric (dilatory) and shear components:

$$\frac{\partial \sigma_v}{\partial t} = 3K \frac{\partial \epsilon_v}{\partial t} - \frac{\sigma_v}{\tau_v}, \quad \tau_v = \frac{\mu_v}{K}$$

$$\frac{\partial \vec{\sigma}'}{\partial t} = 2G \frac{\partial \vec{\epsilon}'}{\partial t} - \frac{\vec{\sigma}'}{\tau'}, \quad \tau' = \frac{\mu}{G}$$

Equation 2-213

where

$$\sigma_v \equiv \frac{1}{3}(\sigma_{xx} + \sigma_{yy} + \sigma_{zz}) \quad \text{and} \quad \epsilon_v \equiv \frac{1}{3}(\epsilon_{xx} + \epsilon_{yy} + \epsilon_{zz})$$

Equation 2-214

are the volumetric components of the stress and strain, respectively, and

$$\vec{\sigma}' \equiv \begin{bmatrix} \sigma_{xx} - \sigma_v \\ \sigma_{yy} - \sigma_v \\ \sigma_{zz} - \sigma_v \\ \sigma_{xy} \end{bmatrix} \quad \text{and} \quad \vec{\epsilon}' \equiv \begin{bmatrix} \epsilon_{xx} - \epsilon_v \\ \epsilon_{yy} - \epsilon_v \\ \epsilon_{zz} - \epsilon_v \\ \epsilon_{xy} \end{bmatrix}$$

Equation 2-215

are the shear components.

The strains are defined by

$$\epsilon_{xx} \equiv \frac{\partial u_x}{\partial x} \quad \epsilon_{yy} \equiv \frac{\partial u_y}{\partial y} \quad \epsilon_{zz} \equiv \frac{\partial u_z}{\partial z} \quad \epsilon_{xy} \equiv \frac{1}{2} \left(\frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \right)$$

Equation 2-216

where u_x , u_y , and u_z are the components of the flow displacements. Note that u_z and ϵ_{zz} are zero for the two-dimensional case analyzed in TSUPREM-4.

The stresses must satisfy the force balance equations

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} = 0$$

$$\frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{xy}}{\partial x} = 0$$

Equation 2-217

These equations can be combined with [Equation 2-213](#) and solved for the flow velocities. The stress in the direction normal to a material interface is continuous across the interface, while the stress normal to a free surface is zero (unless surface tension is included). (See [“Surface Tension and Reflow”](#) on page 2-63.)

There is no slippage between materials at an interface. The equations are solved in silicon only if **^SKIP.SIL** has been specified on the **METHOD** statement.

Model Parameters

The **VISCOELA** model uses the same elasticity parameters as the **COMPRESS** model and the same viscosity and stress dependence parameters as the **VISCOUS** model. The bulk modulus K and shear modulus G are given by

$$K = \frac{E}{3(1-2\nu)} \quad , \quad G = \frac{E}{2(1+\nu)} \quad \text{Equation 2-218}$$

where $E = \mathbf{YOUNG.M}$ and $\nu = \mathbf{POISS.R}$ are specified on the **MATERIAL** statement.

The bulk viscosity μ_v and shear viscosity μ are given by

$$\mu_v = \frac{2\mu(1+\nu_v)}{3(1-2\nu_v)} \quad , \quad \mu = \mathbf{VISC.O} \cdot \exp\left(\frac{-\mathbf{VISC.E}}{kT}\right) \quad \text{Equation 2-219}$$

where $\mathbf{VISC.O}$, $\mathbf{VISC.E}$, and $\nu_v = \mathbf{VISC.X}$ are specified on the **MATERIAL** statement. Note that ν_v is normally very slightly less than 0.5, so μ_v is much larger than K and τ_v is much larger than typical oxidation times. Thus the relaxation in volumetric stress is negligible.

The oxidant diffusivity, interface reaction rate, and material viscosities depend on stress in the same way as in the **VISCOUS** model. Stress dependence is enabled by default; it can be disabled by specifying **^STRESS.D** on the **AMBIENT** statement.

The linear elements used in the **VISCOELA** model produce stresses that are constant across each element and discontinuous between elements. In order to calculate the stress dependences, these discontinuous stresses must be smoothed. The amount of smoothing is controlled by the **VE.SMOOT** parameter on the **METHOD** statement. **VE.SMOOT** can be varied between 0.0 (minimum smoothing) and 1.0 (maximum smoothing). With smaller amounts of smoothing, the stress contours become rougher; larger amounts give smoother contours, but may lose some detail in the solution.

Recommended Usage

The **VISCOELA** model is recommended for simulating 2D structures when details of the resulting oxide shape are important or when stress values are required. It is designed to be used with **STRESS.D** enabled. (It is slower than the **COMPRESS** model, and without stress dependence it is not significantly more accurate.) It is much faster than the **VISCOUS** model, especially when stress dependence is considered. The model is designed to give a good approximation to the shape of the oxide in a minimum simulation time. While stress and flow rate information are available, these quantities are provided only as rough estimates.

The **VISCOELA** model must be used when comprehensive stress history modeling (using **ST.HISTO**) is required; see [“Stress History Model” on page 2-73](#). When **ST.HISTO** is used with the **VISCOELA** model, intrinsic and thermal mis-

match stresses are included in the oxidation model and the stress model is enabled during inert anneals to model relaxation of stresses in the structure.

Polysilicon Oxidation

The oxidation of polysilicon is treated using the models for the oxidation of silicon. Only the **COMPRESS**, **VISCOUS**, and **VISCOELA** models allow oxidation of polysilicon. Values of **THINOX.0**, **THINOX.E**, **THINOX.L**, **L.LIN.0**, **L.LIN.E**, **H.LIN.0**, and **H.LIN.E** may be defined for polysilicon by specifying the **POLYSILI** parameter on the **AMBIENT** statement. The ratio of polysilicon thickness consumed to oxide thickness grown can be defined with the **ALPHA** parameter on the **AMBIENT** statement.

Surface Tension and Reflow

The **VISCOUS** and **VISCOELA** models include the effects of surface tension. The surface tension is modeled as a tensile force along the surface of magnitude specified by the **SURF.TEN** parameter on the **MATERIAL** statement. On planar surfaces, the forces on either side of any point of the surface cancel, giving no net force. But on curved surfaces there is a net force in the direction of the curvature with magnitude inversely proportional to the radius of curvature. The resulting force tends to round off corners and smooth irregularities in the surface. With appropriate values of **SURF.TEN** and material viscosities, the surface tension model can be used to simulate reflow processes.

The surface tension model is active during oxidation with the **VISCOUS** and **VISCOELA** model and whenever stress history modeling is active (i.e., when both **VISCOELA** and **ST.HISTO** are used).

N₂O Oxidation

In N₂O oxidation or oxynitridation, nitrogens are trapped at Si/SiO₂ interfaces so that the number of oxidizing sites, for this reason the oxidation rate is reduced. N₂O oxidation is specified by the **N2O** parameter in the **DIFFUSION** statement.

Nitrogen Trap and Generation

It is assumed that N₂O oxidants reach to Si/SiO₂ interfaces then break up there so that nitrogens are generated. Some of nitrogen are trapped at the interfaces. The trapping flux of nitrogen is described with the interface trap model by ignoring the detrapping flux.

$$\vec{F}_{trap} = hC_i \left(1 - \frac{\sigma_N}{\sigma_{max}} \right) \vec{n}_i \quad \text{Equation 2-220}$$

where h is the trapping rate, C_i is the nitrogen concentration at the interface. σ_N and σ_{max} are the trapped density and the maximum trap density, respectively. The coefficients h and σ_{max} are specified in the **IMPURITY** statement (See “Segrega-

tion Flux” on page 2-28). The total nitrogen flux at interfaces is the sum of the trapping flux into interfaces and the 2-phase segregation flux into adjacent materials. The generation rate of nitrogen is dependent on the boundary velocity and is given by:

$$G = \mathbf{N.G.0} \exp\left(-\frac{\mathbf{N.G.E}}{kT}\right) v \left(\frac{v}{\mathbf{N.V.0} \exp\left(-\frac{\mathbf{N.V.E}}{kT}\right)} \right)^{gpow} \quad \text{Equation 2-221}$$

$$gpow = \mathbf{N.GPOW.0} \exp\left(-\frac{\mathbf{N.GPOW.E}}{kT}\right) \quad \text{Equation 2-222}$$

The **N.G.0**, **N.G.E**, **N.V.0**, **N.V.E**, **N.GPOW.0**, and **N.GPOW.E** are specified in the **AMBIENT** statement.



Note:

N₂O oxidation invokes the interface trap model for nitrogen regardless of the specification of the ITRAP parameter in the METHOD statement.

Surface Reaction Rate

Since the surface reaction rate is proportional to the number of available oxidizing sites, the rate of oxidant consumption at the oxidizing interface is given by:

$$\vec{F} = k_s C_i \left(1 - \frac{\sigma_N}{\mathbf{N.0} \exp\left(-\frac{\mathbf{N.E}}{kT}\right)} \right) \vec{n}_i \quad \text{Equation 2-223}$$

where:

- k_s is the surface reaction rate
- C_i is the oxidant concentration at the interface
- σ_N is the nitrogen density trapped at the interface
- Different values of **N.0** and **N.E** can be defined for each of the three available silicon orientations and for polysilicon by specifying the **<111>**, **<110>**, **<100>**, or **POLYSILI** parameters in the **AMBIENT** statement.

Thin Oxidation Rate

For the thin oxidation regime, the thin oxidation rate is given by:

$$r_{thin} = \text{THINOX} \cdot 0 \exp\left(-\frac{\text{THINOX} \cdot \mathbf{E}}{kT}\right) \exp\left(\frac{-y_{eff}}{\text{THINOX} \cdot \mathbf{L}}\right) \text{ Equation 2-224}$$

$$\times \left(1 - \frac{\sigma_N}{\mathbf{N} \cdot \text{THIN} \cdot 0 \exp\left(-\frac{\mathbf{N} \cdot \text{THIN} \cdot \mathbf{E}}{kT}\right)} \right)$$

Different values of **N.THIN.0** and **N.THIN.E** can be defined for each of the three available silicon orientations and for polysilicon by specifying the **<111>**, **<110>**, **<100>**, or **POLYSILI** parameters in the **AMBIENT** statement.

Boron Diffusion Enhancement in Oxides

Ultrathin gate oxides allow easier dopant penetration between polysilicon and bulk silicon. Boron penetrates more easily than most other dopants because it has a larger diffusivity than the others do. A nonstoichiometric layer rich in peroxy-linkage-defect forms when oxide is grown on top of the silicon [30]. Dopant diffusion is enhanced in thinner oxides because of the greater ratio of nonstoichiometric layer thickness to the total oxide thickness. Also fluorine increases the number of peroxy-linkage-defects and thus enhances dopant diffusion.

Diffusion Enhancement in Thin Oxides

It is assumed that the formation of silicon-oxide (SiO) due to the reaction between oxygen and silicon atom increases the number of peroxy-linkage-defects. The interstitial segregation at silicon/oxide interfaces introduces silicon atoms into the oxide. The SiO is assumed to be immobile and to react with oxygen to produce SiO₂ to rarely dissolve into SiO and O. The volume expansion due to this SiO₂ formation in bulk is assumed to be negligible. The diffusivity of impurities depends on the distribution of SiO.

$$\frac{\partial C_{Si}}{\partial t} = -\vec{\nabla} \cdot (-D_{Si} \vec{\nabla} C_{Si}) - k_{Si} C_{Si} \sqrt{C_{O_2}} \text{ Equation 2-225}$$

$$\frac{\partial C_{SiO}}{\partial t} = k_{Si} C_{Si} \sqrt{C_{O_2}} - k_{SiO} C_{SiO} \sqrt{C_{O_2}} \text{ Equation 2-226}$$

$$F_{Si} = h_{Si}(C_{Si} - mI) \text{ at silicon/oxide interfaces Equation 2-227}$$

$$F_{Si} = k_i C_{Si} \text{ at oxide surfaces Equation 2-228}$$

C_{Si} , C_{SiO} and I are the concentrations of silicon atom, SiO in oxide, and interstitial in silicon, respectively. The diffusivity D_{Si} of silicon atom in oxide is set to $13.0 \exp(-4.5\text{eV}/kT)$ [31]. m is the segregation value of interstitial at the interface between silicon and oxide. Its value is set to $2.9 \times 10^{-5} \exp(2.19\text{eV}/kT)$ corresponding to the value mC_I^* of Agarwal's work [31]. k_{Si} and k_{SiO} are the

bulk reaction rates of silicon atom and SiO with oxygen, respectively. k_i is the reaction rate of silicon atom with oxygen at oxide surfaces.

The diffusion enhancement in oxide is described as follows;

$$D = D_o \left(1 + f_{SiO} \frac{C_{SiO}}{C_{SiO_2}} \right) \tag{Equation 2-229}$$

D_0 is a diffusivity in pure thick oxides. f_{SiO} is the temperature-dependent factors for the diffusion-enhancement due to SiO. C_{SiO_2} is the constant for normalization ($2.2 \times 10^{22} / \text{cm}^3$).

The activation energy of f_{SiO} is set to -0.28 eV [30]. The experimental data for calibration were taken from the published paper [32]. It is assumed that K_{SiO} has the same value as K_{Si} in order to avoid the redundancy due to the lack of measured data. The new impurity names for silicon atom and SiO are given by IOX and SIO, respectively.

```

IMPURITY MODEL=DEOX IMP=IOX NEW C.INIT=1E5
IMPURITY MODEL=DEOX IMP=SIO NEW C.INIT=1E5
METHOD IMP=IOX PART REL.ERR=0.01 ABS.ERR=1E9
METHOD IMP=SIO NONE REL.ERR=0.01 ABS.ERR=1E9
    
```

The diffusivities of silicon atom and boron in oxide is given by

```

IMPURITY IMP=IOX MAT=OXIDE +
          DIX.0=13.0 DIX.E=4.5 CM.SEC
IMPURITY D.MODEL=DEOX IMP=BORON MAT=OXIDE +
          DI.FAC=1+4.55e-22*exp(0.28/kt)*SIO
    
```

The bulk reactions are given by

```

REACTION MODEL=DEOX +
          NAME=KbIOXnO MAT.BULK=OXIDE +
          IMP.L=IOX /IMP.L=O2 IMP.R=SIO +
          EI.L=1.0 /EI.L=0.5 EI.R=1.0 +
          NI.L=1.0 /NI.L=0.0 NI.R=1.0 +
          RATE.0=7.673E6 RATE.E=4.0 EQUIL.0=0.0
REACTION MODEL=DEOX +
          NAME=KbSiOnO MAT.BULK=OXIDE +
          IMP.L=SiO /IMP.L=O2 +
          EI.L=1.0 /EI.L=0.5 +
          NI.L=1.0 /NI.L=0.0 +
          RATE.0=7.673E6 RATE.E=4.0 EQUIL.0=0.0
    
```

The segregation flux of interstitial is given by

```
REACTION MODEL=DEOX +
      NAME=ISEG +
      /MAT.L=OXIDE MAT.R=SILICON +
      /IMP.L=IOX IMP.R=INTERST +
      RATE.0=0.01 RATE.E=0.0 +
      EQUIL.0=2.9E-5 EQUIL.E=-2.19
```

The surface reaction rate of silicon atom is given by

```
REACTION MODEL=DEOX +
      NAME=KiIOX +
      /MAT.L=OXIDE MAT.R=AMBIENT +
      /IMP.L=IOX IMP.R=O2 +
      /EI.L=1.0 EI.R=1.0 +
      /NI.L=1.0 NI.R=0.0 +
      RATE.0=8.67E7 RATE.E=4.0 EQUIL.0=0.0
```

Since the model DEOX is initially disabled in *s4init* file, it needs to be activated to use the model by

```
METHOD MODEL=DEOX ENABLE
```

The SiO distribution is believed to be mostly concentrated near the Si/SiO₂ interface and thus the dense grid structure is needed near the interface. Usually it is believed that the native oxide thickness must be thinner than the default value 20 angstroms.

```
METHOD DY.OXIDE=0.0005
AMBIENT INITIAL=0.0005
```



Note:

This model is only applicable to dry oxidation. And the accuracy of this model is not verified for the oxidation on low oxygen pressure.

Diffusion Enhancement Due to Fluorine

Fluorine enhances the boron diffusion in oxides. For the diffusion model of fluorine, fluorine is assumed to be electrically neutral. The diffusion enhancement due to fluorine is added to [Equation 2-229](#).

$$D = D_o \left(1 + f_{SiO} \frac{C_{SiO}}{C_{SiO_2}} + f_F \frac{C_F}{C_{SiO_2}} \right) \tag{Equation 2-230}$$

where f_F is an equilibrium constant and C_F is a fluorine concentration. f_F and the diffusivities and segregation values of fluorine were extracted by fitting the published data [33].

```
IMPURITY D.MODEL=DEOX IMP=BORON MAT=OXIDE +
DI.FAC=1+4.55e-22*exp(0.28/kt)*SIO+2.23e-20*Fluorine
```

In order to consider the fluorine effect, the model `FLUORINE` as well as the model `DEOX` must be activated. The following statement enables the `FLUORINE` model.

METHOD MODEL=FLUORINE ENABLE

If the model `FLUORINE` is enabled while the model `DEOX` is disabled, the diffusion enhancement of boron due to fluorine is ignored even though the diffusion equation for fluorine is solved.

Silicide Models

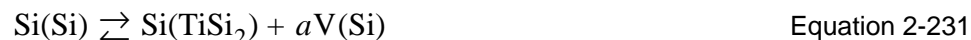
TSUPREM-4 allows you to define models for new materials and reactions. This ability has been used to define models for the growth of titanium and tungsten silicides. The following sections describe the kinetics of TiSi_2 growth, the specification of the model and parameters, and suggestions for how you can model other silicides.

TiSi₂ Growth Kinetics

Titanium silicide is assumed to form when silicon atoms react in the silicide with titanium at the TiSi_2/Ti interface. The consumption of silicon and titanium lead to deformation of the material layers in the structure. Note that while the discussion that follows describes the growth of TiSi_2 on silicon, it also applies to growth of TiSi_2 on polycrystalline silicon.

Reaction at TiSi₂/Si Interface

At the TiSi_2/Si interface you have the reaction



Thus silicon (on the Si side of the interface) reacts to form silicon atoms (on the TiSi_2 side of the interface) plus some number a of vacancies (on the Si side of the interface). The reaction is reversible, allowing the reformation of silicon (if silicon is released by nitridation of TiSi_2 , for example). The forward rate of this reaction depends only on temperature, while the reverse rate is also proportional to the concentration of diffusing silicon atoms in the TiSi_2 . For each silicon atom removed from the silicon side of the interface, the volume of silicon is reduced by

$$\Delta V = \frac{\text{MOL.WT}}{6.022 \times 10^{23} \times \text{DENSITY}} \quad \text{Equation 2-232}$$

where **MOL.WT** and **DENSITY** are the molecular weight and density, respectively, of silicon, as specified on the **MATERIAL** statement.

Diffusion of Silicon

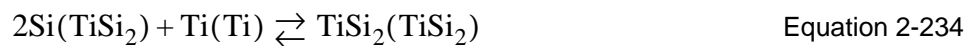
Silicon is transported across the TiSi_2 layer by simple diffusion:

$$\frac{\partial C}{\partial t} = -\vec{\nabla} \cdot (-D\vec{\nabla}C) \quad \text{Equation 2-233}$$

where C is the concentration of diffusion silicon atoms and D is their diffusivity.

Reaction at TiSi₂/Si Interface

At the TiSi₂/Ti interface you have the reaction



This reaction is assumed to be irreversible. The forward reaction rate is proportional to the concentration of diffusing silicon at the TiSi₂ side of the interface. The volumes of Ti and TiSi₂ change according to [Equation 2-232](#).

Initialization

The TiSi₂ growth model is initialized by inserting a thin layer of titanium silicide between layers of titanium and silicon (or polysilicon) wherever they come into contact. This layer is added automatically as needed when titanium is deposited.

Material Flow

Consumption of silicon and titanium and growth of TiSi₂ cause distortion of the layers making up the structure. The flow of material caused by silicidation is calculated using the **VERTICAL**, **COMPRESS**, or **VISCOELA** growth models described in “[The VERTICAL Model](#)” on page 2-56, the “[COMPRESS Model](#)” on page 2-57, and the “[VISCOELA Model](#)” on page 2-60; silicide growth occurs only if one of these models has been specified.

Impurities and Point Defects

Impurities in silicides are modeled in the same way as in other nonsemiconductor materials. Transport within a silicide is governed by simple diffusion (i.e., no electric field effects). Segregation at material interfaces is as described in “[Segregation of Impurities](#)” on page 2-28.

Point defects can participate in reactions at interfaces with silicon. The current titanium silicide model specifies the generation of vacancies by the consumption of silicon ([Equation 2-231](#)). The reaction rate has not yet been adequately calibrated to experimental data, however; it is suggested that the **PD.FERMI** model be used for silicide growth processes.

Specifying Silicide Models and Parameters

The specification of the titanium silicide model requires the following:

1. The new materials (titanium and TiSi₂ in this case) must be defined.
2. Any diffusing species that participates in the growth reactions must be defined.
3. The growth reactions themselves (one at each interface) must be defined.

4. The deposition of initial layers (e.g., the initial TiSi_2 layer between titanium and silicon) must be specified.
5. The diffusion and segregations of impurities in the new materials and at interfaces must be specified.

All of these are accomplished with TSUPREM-4 input statements. Thus no new code is required to implement new silicide models.

Materials

The required materials are specified with the **MATERIAL** statement:

```
MATERIAL NEW MAT=TITANIUM TIF.NAME=TI +
MD.INDEX=-5 DENSITY=4.5 AT.NUM=22.0 +
AT.WT=47.90 MOL.WT=47.90
```

```
MATERIAL NEW MAT=TISI2 TIF.NAME=TISI2 +
MD.INDEX=-5 DENSITY=4.043 AT.NUM=16.67 +
AT.WT=34.68 MOL.WT=104.038 DY.DEFAU=0.025
```

For purposes of defining a silicide growth model, the important parameters are the names of the material (the **MATERIAL** parameter) and the density and molecular weight (**DENSITY** and **MOL.WT**, respectively). The atomic number and atomic weight (**AT.NUM** and **AT.WT**) are used when implanting into the material with the Monte Carlo implant model. The default grid spacing in a growing material is given by **DY.DEFAU** × **GRID.FAC**. A full description of the **MATERIAL** statement is given in “**MATERIAL**” on page 3-228.

Impurities

For the titanium silicide model, you define silicon as a diffusing impurity:

```
IMPURITY NEW IMP=SILICON TIF.NAME=SI STEADY
IMPURITY IMP=SILICON MAT=TISI2 DIX.0=2.0 DIX.E=1.86 +
CM.SEC
```

The important parameters here are the pre-exponential factor and activation energy (**DIX.0** and **DIX.E**, respectively) for the diffusivity of silicon in TiSi_2 . The **STEADY** parameter specifies that the equation for silicon diffusion is solved under steady-state conditions; this is appropriate for fast-diffusing impurities and is recommended for impurities that produce material growth. For a complete description of the **IMPURITY** statement see page 3-239.

Reactions

Two reactions are needed to model the growth of titanium silicide. The first describes the solution of silicon atoms in TiSi_2 :

```
REACTION /MAT.L=TISI2 MAT.R=SILICON NM.R=1.0 +
/IMP.L=SILICON IMP.L=VACANCY NI.L=1E-3 EI.L=0 +
RATE.0=1E-3 EQUIL.0=1E20
```

This statement implements the reverse of the reaction of Equation 2-231:



Parameters ending in “.L” denote reactants (silicon atoms in TiSi₂ and vacancies in silicon) while parameters ending in “.R” denote products (bulk silicon). Parameters can also be classified by which side of the material interface they are associated with. In this case, the vacancies are on the silicon side of the interface (no “/” in the parameter name), while silicon (as an impurity) is on the TiSi₂ side of the interface (with “/” in the parameter name).

By default, one molecule of each impurity and no molecules of each material participate in the reaction. These defaults have been overridden for silicon (NM.R=1.0 implies that one silicon atom participates) and for vacancies (NI.L=1e-3 implies that one vacancy is generated for each 1000 reactions). The forward reaction rate in this case is given by

$$R_f = \text{RATE.0} \exp\left(\frac{-\text{RATE.E}}{kT}\right) [\text{V}]^{\text{EI.L}} [\text{Si}]^{\text{EI.L}} \quad \text{Equation 2-236}$$

where [V] and [Si] denote the vacancy and silicon concentrations, respectively. EI.L=0 has been specified, so there is no dependency on the vacancy concentration; /EI.L defaults to 1.0. The equilibrium concentration of reactant impurities to product impurities is given by

$$[\text{V}]^{\text{EI.L}} [\text{Si}]^{\text{EI.L}} = \text{EQUIL.0} \exp\left(\frac{-\text{EQUIL.E}}{kT}\right) \quad \text{Equation 2-237}$$

Note that there are no product impurities in this case. The equilibrium concentration of silicon diffusing in TiSi₂ has been specified as 1e20.

The reaction at the interface between TiSi₂ and polysilicon is the same as at silicon interfaces except that no vacancies are involved:

```
REACTION /MAT.L=TISI2 MAT.R=POLY NM.R=1.0 +
          /IMP.L=SILICON RATE.0=1E-3 EQUIL.0=1E20
```

The reaction at the Ti/TiSi₂ interface is specified by

```
REACTION /MAT.L=TITANIUM /NM.L=0.5 +
          MAT.R=TISI2 NM.R=0.5 IMP.L=SILICON +
          RATE.0=104 RATE.E=1.0 EQUIL.0=1.0
```

This corresponds directly with Equation 2-234 except that you have divided the quantities of all reactants by two. The forward reaction rate is proportional to the silicon concentration at the Ti/TiSi₂ interface. The equilibrium ratio of reactant impurities to product impurities (= [Si]) has been set to 1.0, a very small value.

The initial TiSi_2 layers to be deposited between titanium layers and exposed silicon and poly silicon layers are also specified with **REACTION** statements:

```
REACTION MAT=TITANIUM /MAT=SILICON +
      MAT.NEW=TISI2 THICK=0.002
REACTION MAT=TITANIUM /MAT=POLY +
      MAT.NEW=TISI2 THICK=0.002
```

These specify that 0.002 microns of TiSi_2 should be deposited on silicon or polysilicon before depositing titanium.

Impurities

Diffusion and segregation of impurities in silicides are specified in the usual way:

```
BORON MAT=TISI2 CM.SEC DIX.0=6.0E-7 DIX.E=2.0
BORON SILICON /MAT=TISI2 SEG.0=0.3 +
      TRANS.0=1E-6 TRANS.E=2.0
BORON MAT=POLY /MAT=TISI2 SEG.0=0.3 +
      TRANS.0=1E-6 TRANS.E=2.0
```

The **MATERIAL** and **/MATERIA** parameters must be used when specifying new material names; they are optional when specifying old built-in materials. These parameters could also be specified with the **IMPURITY** statement:

```
IMPURITY IMP=BORON MAT=TISI2 CM.SEC DIX.0=6.0E-7 +
      DIX.E=2.0
IMPURITY IMP=BORON MAT=SILICON /MAT=TISI2 +
      SEG.0=0.3 . . .
```

Tungsten and Cobalt Silicide Models

The tungsten and cobalt silicide models are identical in form to the titanium silicide model. The parameters of the models are different, however, reflecting the differences between the materials (see [29] to [39] for the tungsten silicide model and [40] to [44] for the cobalt silicide model). The names of the relevant materials are *tungsten* and *WSi2* for tungsten silicide and *cobalt* and *CoSi2* for cobalt silicide.

Other Silicides

Models for other silicides can be defined following the example given above for titanium silicide. The diffusing impurity in the silicide can be metal or silicon or both.

Stress Models

TSUPREM-4 has several models for calculating the stresses produced during the fabrication model. The most complete is the stress history (**ST.HISTO**) model. Older and more limited capabilities are provided by the **STRESS** statement and the **VISCOUS** oxidation model.

Stress History Model

A number of physical phenomena give rise to stress in a structure during fabrication. These include volume changes during oxidation, thermal mismatch between materials, intrinsic strain in deposited layers, and surface tension. These phenomena are simulated by the stress history (**ST.HISTO**) model in TSUPREM-4.

Thermal Stress Model Equations

The effect of thermal expansion during temperature ramping is modeled by adding an additional term to Equation 2-213 in Chapter 2 for the volumetric stress:

$$\frac{\partial \sigma_v}{\partial t} = 3K \left(\frac{\partial \epsilon_v}{\partial t} - \text{LCTE} \frac{dT}{dt} \right) - \frac{\sigma_v}{\tau_v} \quad \text{Equation 2-238}$$

where **LCTE** is the linear coefficient of thermal expansion specified on the **MATERIAL** statement and T is the temperature.

Boundary Conditions

The boundary conditions assume that the thermal expansion is dominated by a thick silicon substrate with equivalent conditions on the front and back surfaces (so there is no curvature of the wafer). Thermal expansion of the substrate produces a constant strain $\epsilon_{zz} = \text{LCTE} \Delta T$ in the z direction. The displacement in the x direction at vertical reflecting boundaries and in silicon when **SKIP.SIL** is set is proportional to x , $u_x = x \text{LCTE} \Delta T$, while the displacement in the y direction at horizontal reflecting boundaries and in silicon when **SKIP.SIL** is set is proportional to y , $u_y = y \text{LCTE} \Delta T$. The value of **LCTE** for silicon is used for these boundary conditions.

Surface tension acts on the surface of the structure; see “[Surface Tension and Reflow](#)” on page 2-63 for details.

Initial Conditions

The final stress from one high-temperature processing step is used as the starting stress for the next; if the temperature changes between the end of one step and the start of the next, a very short ramping step is inserted automatically by the program to compute the change in stress caused by the change in temperature. The stress is set to zero at the first temperature specified after structure initialization; this temperature may be specified as part of a high-temperature processing step (i.e., **DIFFUSION**, **DEPOSITION**, or **EPITAXY**) or on a **SELECT** or

SAVEFILE statement. The stresses may be modified during deposition and etching, as described below. The **STRESS** statement and oxidation with the **VISCOUS** model compute new stresses, ignoring and replacing previously calculated values.

Intrinsic Stress in Deposited Layers

The intrinsic stress in deposited layers can be specified for each material with the **INTRIN.S** parameter on the **MATERIAL** statement. This intrinsic stress is included whenever stress history modeling is enabled. Each deposition step is followed by a stress relaxation calculation to determine the stresses in the structure following the deposition. The model includes the effects of surface tension; the stress is only calculated when **ST.HISTO** and **VISCOELA** models are active.



Note:

*The interpretation of the **INTRIN.S** parameter by the stress history model is different from that used by the **STRESS** statement. Both work by placing an initial stress in the deposited layer then allowing the layer to relax to conform to the boundary conditions (e.g., zero stress normal to free surfaces). The **STRESS** statement uses **INTRIN.S** as the initial stress, while the stress history model uses an initial stress that gives **INTRIN.S** as the *x* (and *z*) component of stress in a uniform, planar layer after relaxation. The difference is summarized by [Table 2-1](#). For the **ST.HISTO** model to generate the same results as the **STRESS** statement, you must multiply **INTRIN.S** by $(1 - 2\nu)/(1 - \nu)$, where ν is the value of **POISS.R**.*

Table 2-1. Model Comparisons After Relaxation

| Model | Initial | After Relaxation |
|-----------------|--|--|
| ST.HISTO | $\sigma_{xx} = \sigma_{yy} = \sigma_{zz} = \text{INTRIN.S} \frac{1 - \nu}{1 - 2\nu}$ | $\sigma_{yy} = 0,$ $\sigma_{xx} = \sigma_{zz} = \text{INTRIN.S}$ |
| STRESS | $\sigma_{xx} = \sigma_{yy} = \sigma_{zz} = \text{INTRIN.S}$ | $\sigma_{yy} = 0,$ $\sigma_{xx} = \sigma_{zz} = \text{INTRIN.S} \frac{1 - 2\nu}{1 - \nu}$ |

Effect of Etching on Stress

When stressed material is removed from a structure, the balance of forces acting on the remaining material changes, as do the stresses in the remaining structure. Thus, each etching step is followed by a stress relaxation calculation to determine the stresses in the structure after the etch. The model includes the effects of surface tension; the stress is only calculated when the **ST.HISTO** and **VISCOELA** models are active.

Using the Stress History Model

The stress history in a structure is simulated when the following conditions are satisfied:

- Stress history simulation has been enabled by the **ST.HISTO** parameter on the **METHOD** statement:

```
METHOD ST.HISTO
```

- The **VISCOELA** oxidation model has been selected

Stresses in silicon are simulated only if **^SKIP.SIL** has been specified on the **METHOD** statement. Thermal mismatch stresses are simulated whenever temperature ramping is specified on the **DIFFUSION** statement. Intrinsic stresses are included during deposition whenever **INTRIN.S** for the deposited material is nonzero. Surface tension is included whenever **SURF.TEN** for an exposed material is nonzero.

Limitations

The thermal stress model has the following limitations:

1. The model assumes that thermal expansion is dominated by a thick silicon substrate with equivalent conditions on the front and back sides of the wafer (i.e., no bending of the wafer).
2. The model is active only when the **VISCOELA** oxidation model is active. Stresses in silicon are calculated only if **^SKIP.SIL** has been specified. The stress history is lost and replaced with newly calculated stresses by the **STRESS** statement and during oxidation with the **VISCOUS** model.
3. While the thermal mismatch model works down to room temperature, the diffusion models do not. At temperatures below about 50°C they may produce arithmetic exceptions or cause the program to hang; below about 600°C they may produce inaccurate results.
4. The stresses caused by phase changes during temperature cycling are not modeled. It may be possible to approximate these stresses by changing the thermal expansion coefficients as a function of time and temperature.

Modeling Stress with the **STRESS** Statement

The **STRESS** statement (see [page 3-124](#)) allows calculation of stresses due to thermal mismatch between materials or due to intrinsic stress in deposited films. It uses a finite-element analysis based on a linear elastic model for the materials involved. The equations that are solved are

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} = 0 \quad \text{Equation 2-239}$$

$$\frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{xy}}{\partial x} = 0 \quad \text{Equation 2-240}$$

Equation 2-241

$$\sigma_{xx} + \sigma_{yy} = \frac{E}{(1+\nu)(1-2\nu)} \left(\frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} \right) - \frac{2E}{1-2\nu} \int_{T_1}^{T_2} \text{LCTE} \, dT + 2\sigma_i$$

$$\sigma_{xx} - \sigma_{yy} = \frac{E}{1+\nu} \left(\frac{\partial u_x}{\partial x} - \frac{\partial u_y}{\partial y} \right) \quad \text{Equation 2-242}$$

$$\sigma_{xy} = \frac{E}{2(1+\nu)} \left(\frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \right) \quad \text{Equation 2-243}$$

Equations 2-239 and 2-240 are the equations of motion for the structure, while Equations 2-241 through 2-243 are the constitutive relations for the materials. Here σ_{xx} , σ_{yy} , and σ_{xy} are the calculated stresses and u_x and u_y are the x and y displacements; u_x and u_y are stored as the x and y velocities (even though they are actually displacements), and can be accessed for printing or plotting by referencing the $x.vel$ and $y.vel$ functions on the **SELECT** statement. The values of Young's modulus E and Poisson's ratio ν for each material are specified on the **MATERIAL** statement by **YOUNG.M** and **POISS.R**, respectively (see page 3-228).

The linear coefficient of thermal expansion for each material, **LCTE**, can be specified as a function of T (absolute temperature) on the **MATERIAL** statement. This function is integrated between temperatures T_1 and T_2 , specified by the **TEMP1** and **TEMP2** parameters on the **STRESS** statement. If **TEMP1** and **TEMP2** are not given, then the **LCTE** term is omitted from the analysis.

Boundary Conditions

The following boundary conditions are used:

- Exposed surfaces: stress normal to the surface is zero.
- Material interfaces: stress normal to the interface and the displacements are continuous across the interface.
- Reflecting boundaries: displacement perpendicular to the boundary is zero. This displacement value produces incorrect results for thermal expansion calculations when there are reflecting boundaries on both the left and right sides of the structure and the **LCTE** of the substrate is nonzero. This is one reason that the stress history model should be used instead of the **STRESS** statement.
- z direction: displacement and stress in the z direction are taken to be zero. The strain in the z direction should be determined by the thermal expansion of the substrate, as it is in the stress history model.

Thin film intrinsic stresses are accounted for by the σ_i terms [89] and [90]. The value of σ_i for each material is given by the **INTRIN.S** parameter on the **MATERIAL** statement.

The calculated values of stresses and displacements replace any values of stress and velocity that may have been calculated by a previous oxidation step or by the stress history model.

The **STRESS** statement has been made obsolete by the stress history (**ST.HISTO**) model.

Ion Implantation

The **IMPLANT** statement (see [page 3-103](#)) is used to model the implantation of ionized impurities into the simulation structure. Two distinct approaches are available for modeling ion implantation.

- The analytic approach models the impurity and point defect distributions using Gaussian or Pearson functions based on distribution moments contained in a data file.
- The Monte Carlo approach calculates the trajectories of implanted ions through the two-dimensional target structure, based on physical models.

These two approaches are described in the following sections.

Analytic Ion Implant Models

The impurity being implanted is selected with one of the parameters **ANTIMONY**, **ARSENIC**, **BORON**, **PHOSPHOR**, or **BF2**. The **DOSE** parameter specifies the total number of impurity ions per square centimeter provided by the ion implanter. The acceleration energy of the ions is specified with the **ENERGY** parameter. Only the portion of the distribution within the simulation region contributes to the dose in the structure. If the **BACKSCAT** parameter is specified, the portion of the distribution above the top of the simulation structure is assumed to be backscattered at the surface. Any portion of the distribution below the bottom of the simulation region is assumed to have passed through the structure.

Implanted Impurity Distributions

Implanted impurity distributions in a two-dimensional structure are derived from distributions calculated along vertical lines through the structure. The one-dimensional procedures described below are used to find the vertical implant distribution along each line.

Each one-dimensional profile is converted to a two-dimensional distribution by multiplying by a function of x . The final profile is determined by integrating the contributions of all the two-dimensional distributions to the doping at each node. If the **TILT** parameter is nonzero, the lines for the one-dimensional calculation are taken at the specified angle from the vertical. The variable u in the discussion that follows then represents the distance along the angled line, while the variable x corresponds to distance perpendicular to the slices.

The vertical distribution along each line is given by

$$I(u) = \text{DOSE} \times f(u) \tag{Equation 2-244}$$

where $u = 0$ occurs at the surface of the top material layer along the line, and $f(u)$ is a normalized Gaussian or Pearson distribution, depending on whether

GAUSSIAN or **PEARSON** is selected on the **IMPLANT** statement. The equations for $f(u)$ are described below.

The vertical distribution function $f(u)$ is calculated from its spatial distribution moments. The first four moments are defined as

$$R_p = \int_{-\infty}^{\infty} uf(u) du \quad \text{Equation 2-245}$$

$$\sigma = \sqrt{\int_{-\infty}^{\infty} (u - R_p)^2 f(u) du} \quad \text{Equation 2-246}$$

$$\gamma = \frac{\int_{-\infty}^{\infty} (u - R_p)^3 f(u) du}{\sigma^3} \quad \text{Equation 2-247}$$

$$\beta = \frac{\int_{-\infty}^{\infty} (u - R_p)^4 f(u) du}{\sigma^4} \quad \text{Equation 2-248}$$

The values of R_p , σ , γ , and β are obtained from the implant data file *s4imp0* or from an alternate implant data file specified with the **IN.FILE** parameter in the **IMPLANT** statement. For each combination of impurity and material, these files contain the distribution moments for a series of acceleration energies in order of increasing energy.

Implant Moment Tables

The implant data file associates distribution moments with each ion and material by using ion and material names present in the file. The material names are the same as those used elsewhere in the program (i.e., **SILICON**, **OXIDE**, **NITRIDE**, **POLYSILI**, **PHOTORES**, and **ALUMINUM**). The data for silicon dioxide is also used for silicon oxynitride.

The data table to be used for an implantation step can be specified explicitly with the **IMPL.TAB** parameter in the **IMPLANT** statement. This allows the implant data file to contain several sets of distribution moments for the same impurity. The implant data file defines the following table names:

- **ANTIMONY**: Original antimony data with extended energy ranges fitted to results of amorphous Monte Carlo calculations (energies: 5–1000 keV). This is the default for antimony implantation.
- **ARSENIC**: Original arsenic data with extended energy ranges fitted to results of amorphous Monte Carlo calculations (energies: 5–11,000 keV).
- **DUAL.ARS**: Dual-Pearson data for arsenic with channeling in silicon (energies: 10-1000 keV) [45]. This is the default for arsenic implantation.

- **TR.ARSENIC:** Dual-Pearson data for arsenic in <100> silicon with full energy, dose, tilt, and rotation dependence (energy: 0.5–180 keV; dose: 10^{13} – 8×10^{15} atoms/cm²; tilt: 0°–10°; rotation: 0°–45°) [46] and [101].
- **BF2:** Data for boron from a BF₂ source (energies: 5–120 keV) [48].
- **DUAL.BF2:** Dual-Pearson data for boron from a BF₂ source with channeling in silicon (energies: 10–200 keV) [45]. This is the default for BF₂ implantation.
- **UT.BF2:** Dual-Pearson data for boron from a BF₂ source with channeling in silicon (energies: 15–120 keV) [46].
- **TR.BF2:** Dual-Pearson data for BF₂ in <100> silicon with full energy, dose, tilt, and rotation dependence (energy: 0.5–65 keV; dose: 10^{13} – 8×10^{15} atoms/cm²; tilt: 0°–10°; rotation: 0°–45°) [46] and [101]. The data for < 5 keV implants were generated by using the Monte Carlo model in Taurus Process & Device, calibrated using Eaton data.
- **BORON:** Original boron data with extended energy ranges fitted to results of amorphous Monte Carlo calculations (energies: 5–4000 keV).
- **LEBORON:** Data for low-energy boron with channeling in silicon (energies: 10–30 keV) [48].
- **CHBORON:** Data for boron with channeling in silicon (energies: 5–2000 keV) [45]. This is the default for boron implantation.
- **UT.BORON:** Dual-Pearson data for boron with channeling in silicon (energies: 15–100 keV) [46].
- **TR.BORON:** Dual-Pearson data for boron in <100> silicon with full energy, dose, tilt, and rotation dependence (energy: 0.5–80 keV; dose: 10^{13} – 8×10^{15} atoms/cm²; tilt: 0°–10°; rotation: 0°–45°) [49] and [101]. The data for < 5 keV implants were generated by using the Monte Carlo model in Taurus Process & Device, calibrated using Eaton data.
- **SCR.BORON:** Dual-Pearson data for boron in <100> silicon with full energy, dose, tilt, rotation, and *screening oxide thickness* dependence. Data from The University of Texas at Austin (energy: 15 keV–80 keV; dose: 10^{13} – 8×10^{15} atoms/cm²; tilt: 0°–10°; rotation: 0°–45°; oxide thickness: 15–400 Angstroms).
- **PHOSPHORUS:** Original phosphorus data with extended energy ranges fitted to results of amorphous Monte Carlo calculations (energies: 5–7000 keV).
- **DUAL.PHO:** Dual-Pearson data for phosphorus with channeling in silicon (energies: 10–200 keV) [45]. This is the default for phosphorus implantation.
- **TR.PHOSPHORUS:** Dual-Pearson data for phosphorus into bare <100> silicon with full energy, dose, tilt, and rotation dependence (energy: 15–180 keV; dose: 10^{13} – 8×10^{15} atoms/cm²; tilt: 0°–10°; rotation: 0°–45°) [47].
- **FLUORINE:** Dual-Pearson data for fluorine fitted to results of Monte Carlo calculations (energies: 2–95 keV).
- **TR.INDIUM:** Dual-Pearson data for indium in <100> silicon with full energy, dose, tilt, and rotation dependence (energy: 1–100 keV; dose: 10^{13} – 8×10^{15} atoms/cm²; tilt: 0°–10°; rotation: 0°–45°). The 200 keV parameters

are based on tilt=0 implants, while 300 keV are based on tilt=7 and rotation=30 implants. All parameters are extracted from the data generated by Monte Carlo simulations.

The energy ranges shown are for implantation into silicon. If no data is present in the implant data file for the specified energy, linear interpolation is used with the available data to determine the distribution moments. No extrapolation is done for energies outside the range of the table. For implants into silicon, an error is printed and the program terminates; for other materials, a warning is printed and the nearest available energy is used. If the first three moments for the specified energy are present in the implant data file, but the value of β is missing or zero, then β is calculated using the expression [50]

$$\beta = 2.91 + 1.56\gamma^2 + 0.59\gamma^4 \tag{Equation 2-249}$$

which yields a reasonable value for β and requires knowledge of only the first three distribution moments.

Moments for up to 20 materials can be included in the implant moments file. Each material in the file has a unique name. Usually, TSUPREM-4 searches the moments file for a material with a name that matches the material name used in TSUPREM-4. For example, data for the material called *photoresist* in the moments file is used for implantation into photoresist. But you can request that a different set of data be used with the **IMPL.TAB** parameter on the **MATERIAL** statement. Thus the statement

MATERIAL MAT=PHOTORESIST IMPL.TAB=AZ-7500

requests that data for the material named *az-7500* in the implant moments file be used for implantation into photoresist.

Gaussian Distribution

A Gaussian distribution requires only the moments R_p and σ and is defined as

$$f(u) = \frac{1}{\sqrt{2\pi} \sigma} \exp\left[\frac{-(u - R_p)^2}{2\sigma^2} \right] \tag{Equation 2-250}$$

where R_p and σ are defined above.

Pearson Distribution

A Pearson distribution requires the moments R_p , σ , γ , and β , and is defined by the differential equation [51]

$$\frac{df(v)}{dv} = \frac{(v - a)f(v)}{b_0 + av + b_2v^2} \tag{Equation 2-251}$$

where

$$v = u - R_p \tag{Equation 2-252}$$

$$a = \frac{-\sigma\gamma(\beta + 3)}{A} \tag{Equation 2-253}$$

$$b_0 = \frac{-\sigma^2(4\beta - 3\gamma^2)}{A} \tag{Equation 2-254}$$

$$b_2 = \frac{-2\beta + 3\gamma^2 + 6}{A} \tag{Equation 2-255}$$

$$A = 10\beta - 12\gamma^2 - 18 \tag{Equation 2-256}$$

Not all combinations of β and γ produce useful profiles. The characteristics of the profile can be determined by examining the denominator of [Equation 2-251](#):

$$p(v) = b_0 + av + b_2v^2 \tag{Equation 2-257}$$

For the profile to have a maximum at $v = a$, $p(a)$ must be negative; for the mean of the profile to be at R_p requires that $b_2 > -1/2$.

TSUPREM-4 checks these conditions and produces an error unless both are satisfied. The fourth moment is equal to β only if $b_2 > -1/5$, which can only occur when $A > 0$. However, the program produces profiles for $-1/2 < b_2 < -1/5$ without printing a warning (provided that $p(a) < 0$).

The nature of the profile depends on the value of b_2 and whether there are real solutions to the equation $p(v) = 0$. In the following, it is assumed that $p(a) < 0$ and the solutions to $p(v) = 0$ (if they exist) are v_1 and v_2 , with $v_1 \leq v_2$. For $b_2 > 0$, the profile is nonzero for $v_1 < v < v_2$; this corresponds to a Pearson type I or II distribution. If $b_2 \leq 0$ and $p(v) = 0$ has no real solutions, then the profile is nonzero for all v , but approaches zero as v approaches $\pm\infty$; this is a Pearson type IV or VII distribution for $b_2 < 0$ and a Gaussian for $b_2 = 0$.

If $b_2 \leq 0$ and $p(v) = 0$ has real solutions, then the profile either goes to zero at v_1 and $-\infty$ (for $a < v_1$) or at v_2 and ∞ (for $a > v_2$); this is Pearson type V or VI distribution for $b_2 < 0$ and a type III distribution for $b_2 = 0$.

Dual Pearson Distribution

In the case of implantation into crystalline silicon, for low to medium dose implants the shape of the resultant profile is relatively insensitive to implant dose. In this range, a single Pearson function can be used to model the range of implant doses at a given energy by simply scaling the magnitude of the profile to match the implant dose. This is the method used when the **PEARSON** distribution is selected and data for a single Pearson function is included in the implant moment data file for the specified impurity name.

Dose-dependent Implant Profiles

At higher doses, substrate damage causes the implant profile to vary significantly with dose. A dual-Pearson approach has been found to work well for modeling such dose-dependent implant profiles [45], [46], and [49]. When the **PEARSON** distribution is selected and data for a dual-Pearson function is included in the

implant moment data file for the specified impurity name, a dual-Pearson function is used.

Following this approach, one Pearson profile models the channeled portion of the implant profile, while the second Pearson profile models the nonchanneled portion resulting from implantation into partially amorphized silicon. The dose dependence is modeled by varying the relative magnitude of the channeled and nonchanneled Pearson functions. The moments of each of the Pearson functions are independent of dose. The composite dual-Pearson profile can be described as:

Equation 2-258

$$I_{composite}(u) = rI_{amorphous}(u, R_{Pa}, \sigma_a, \gamma_a, \beta_a) + (1-r)I_{channeled}(u, R_{Pc}, \sigma_c, \gamma_c, \beta_c)$$

where

- $I_{amorphous}$ and $I_{channeled}$ are the normalized amorphous and channeled Pearson profiles, respectively
- r is the ratio of the dose of the amorphous profile to the total dose
- subscripts a and c refer to amorphous and channeled profiles, respectively
- u is the depth coordinate along the line

The implant data file for dual-Pearson data has two sets of four moments, corresponding to both Pearson profiles, and a table of ratio values as a function of dose. [Figure 2-1](#) shows the variation of profile shape with implantation dose, in the case of an initially crystalline silicon substrate, and compares the dual-Pearson model with data taken from [46]. In [Figure 2-1\(a\)-\(d\)](#), the composite profile is plotted (solid line), along with the channeled and amorphous profile contributions (dashed lines). The ratio of the dose allocated to the amorphous profile relative to the channeled dose is indicated in each figure. A ratio of unity indicates a highly

damaged substrate with little channeling, whereas a ratio of zero indicates a highly channeled profile with little or no implant damage.

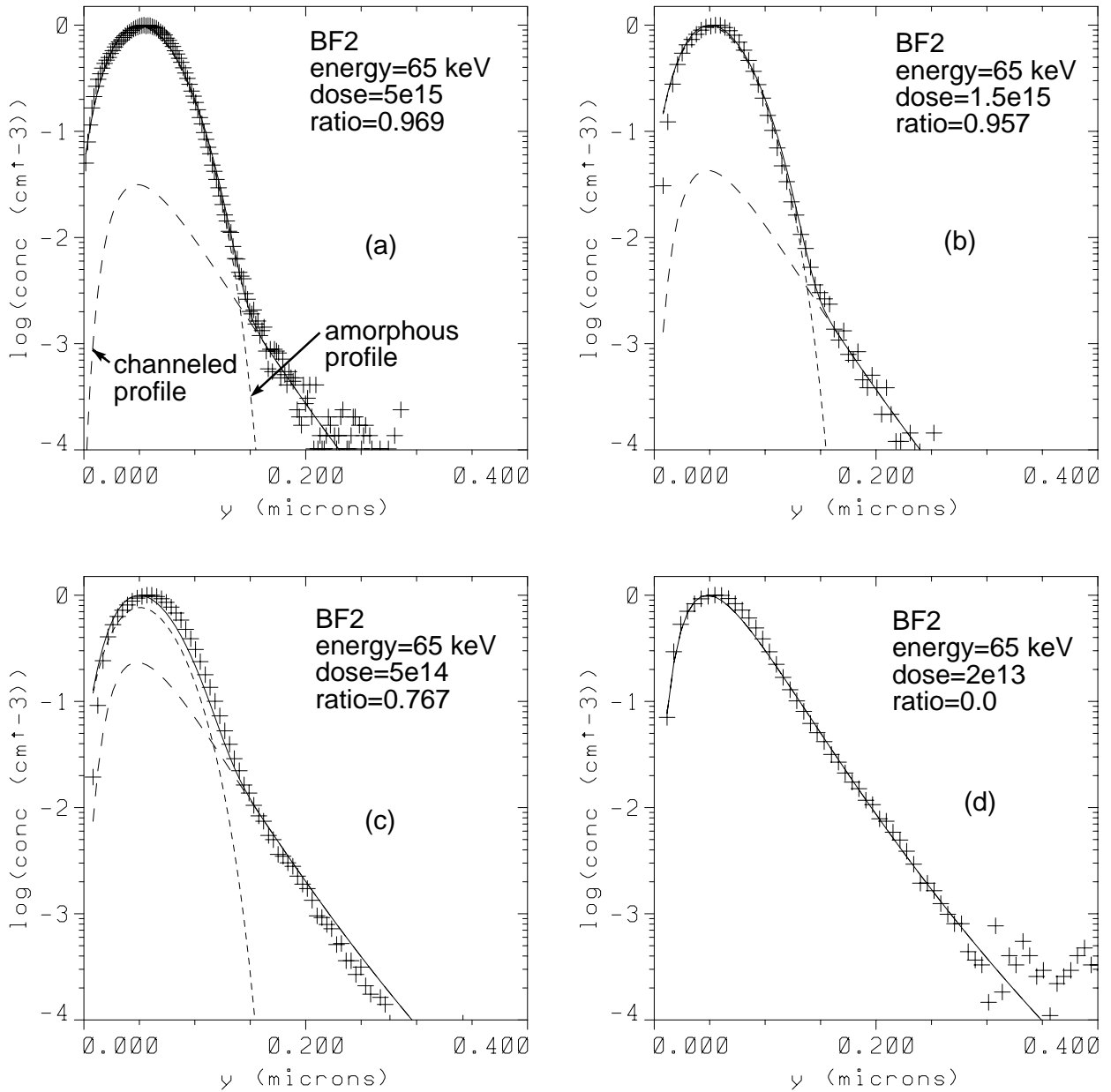


Figure 2-1 BF₂ implant profile

When the **GAUSSIAN** distribution is selected and dual-Pearson data is contained in the implant data file, the first set of moments is used as described above for the calculation of the distribution. If the dose is outside the range of values available in the implant data file, the fraction corresponding to the nearest available dose is used. The dual-Pearson functions are useful for modeling channeling effects, with the primary Pearson function modeling the bulk of the implant and the secondary Pearson function modeling the distribution of channeled ions.

Tilt and Rotation Tables

The *tr.arsenic*, *tr.bf2*, and *tr.boron* tables contain dual-Pearson distributions for a range of energies, doses, wafer tilts, and rotations for implantation into bare, <100> silicon. When one of these tables is specified with the **IMPL.TAB** parameter, the distribution for the specified energy, dose, tilt and rotation are obtained from the table. If the specified implant parameters do not appear in the table, linear interpolation (in four dimensions) between table entries is used. If the specified implant parameters lie outside of the range of the table, a warning message is printed and the nearest available values are used.



Note:

The tilt and rotation tables give accurate results after implantation into bare, <100> silicon, but may not be appropriate under other implant conditions.

Multilayer Implants

A multilayer implant is represented by treating each layer sequentially, starting with the top layer in the structure. The impurity distribution $I(u)$ is determined by first obtaining the moments from the implant data file for the impurity in the material comprising the layer. The distribution $I(u - u_l + u_s)$ is used for the impurity distribution within the layer, where

$$u_l = \sum_i t_i \tag{Equation 2-259}$$

The summation is performed over all previously treated layers of the structure, and t_i is the thickness of layer i . Either of two approaches can be used to determine u_s : effective range model or dose matching.

Effective Range Model

By default, u_s is given by [52]

$$u_s = \sum_i \frac{t_i R_p}{R_{p_i}} \tag{Equation 2-260}$$

where R_{p_i} is the first moment of $f(u)$ in layer i , R_p is the first moment of $f(x)$ in the present layer, and the summation is performed over all previously treated layers of the structure. For layers below the first, the magnitude of the distribution is scaled so that the integral of $I(u)$ from $u = u_s$ to $u = \infty$ plus the total dose placed in all previously treated layers is equal to the specified implant dose. This method is referred to as the *effective range* or *effective thickness* approach.

Dose Matching

If the parameter **RP.EFF** is set as false on the **IMPLANT** statement, u_s is determined such that the integral of $I(u)$ from $u = 0$ to $u = u_s$ equals the total dose placed in all previously treated layers [53]. For the top layer, $u_s = 0$. This method is referred to as the *dose matching* approach.

The effective range approach has proved to be more accurate than the dose matching approach; the dose matching approach is retained only for compatibility with older revisions (8926 and older) of the program.

Lateral Distribution

Each one-dimensional profile is expanded to two dimensions by multiplying by a Gaussian distribution in the direction perpendicular to the line [54]:

$$I(u, v) = I(u) \times \frac{1}{\sqrt{2\pi} \sigma_x} \exp\left(-\frac{v^2}{2\sigma_x^2}\right) \quad \text{Equation 2-261}$$

where v is the distance perpendicular to the line. The quantity σ_x is the lateral standard deviation of the implant profile in the given material and is found by interpolation in the implant data file. The complete implant profile is obtained by summing together the two-dimensional profiles produced by all of the lines.

Depth-Dependent Lateral Distribution

In order to better model the lateral distribution, TSUPREM-4 has two additional parameters, **LSLOPE** (for the first Pearson) and **D.LSLOPE** (for the second Pearson) to allow for the depth dependent lateral straggle:

$$\sigma_1(z) = \text{SIGMA} * (1 + \text{LSLOPE} * (z / \text{RANGE} - 1)),$$

$$\sigma_2(z) = \text{D.SIGMA} * (1 + \text{D.LSLOPE} * (z / \text{D.RANGE} - 1)),$$

where z is the depth. **LSLOPE** and **D.LSLOPE** can be specified on the **MOMENT** statement. TSUPREM-4 versions 1999.4 or later also provide the default parameters for **LSLOPE** and **D.LSLOPE** in all “**TR**” tables. By choosing the parameters **LSLOPE** and **D.LSLOPE** properly, the two-dimensional profiles can be modeled more accurately.

The depth-dependent lateral straggle model will be automatically used if the “**TR**” tables are specified. This model is not used for any other tables.

Please note that starting from TSUPREM-4 version 1999.4, the lateral straggle formula has been changed in such a way that

$$\text{LSLOPE}(\text{new}) = \text{LSLOPE}(\text{old}) / \text{SIGMA},$$

$$\text{D.LSLOPE}(\text{new}) = \text{D.LSLOPE}(\text{old}) / \text{D.SIGMA}$$

Identical results will be obtained from previous input files if **V.COMPAT** is set to versions earlier than 1999.4. On the other hand, if no compatibility mode is set, **LSLOPE** and **D.LSLOPE** should be adjusted accordingly on the **MOMENT** statement.

Wafer Tilt and Rotation

Tilt and rotation of the wafer during implantation are specified by the **TILT** and **ROTATION** parameters, respectively. The various effects of tilting the wafer are simulated as follows:

1. When a nonplanar structure is tilted, shadowing of portions of the structure can occur. The amount of shadowing for a given tilt depends on the rotation—from full shadowing at zero rotation to no shadowing at 90° rotation. Shadowing is simulated by tilting the simulated structure by an appropriate amount.
2. The number of incident ions per square centimeter of the wafer is reduced when the wafer is tilted. This dose reduction is simulated naturally when the

simulation structure is tilted, but an analytical adjustment to the dose is required if the tilt of the simulation structure is less than the specified wafer tilt because of rotation.

3. Tilting the wafer causes the implanted profiles to be foreshortened. As with dose reduction, foreshortening of the profiles is simulated by a combination of tilting the simulation structure and by analytical adjustments to the implant parameters. If the implantation data is derived from measurements based on tilted wafer samples, the tilt of the data also affects foreshortening. For this reason, each table of data in the implantation data file has an associated tilt and rotation value that reflects the conditions under which the implantation data were measured. The tilt value for each table is used in the calculation of foreshortening effects.
4. The tilt and rotation of the wafer affect the number of ions that are channeled along crystal planes in silicon. Channeling effects in bare <100> silicon as a function of tilt and rotation are included in the *tr.arsenic*, *tr.bf2*, and *tr.boron* implant tables.

BF₂ Implant

When the model `FLUORINE` is enabled in `METHOD` statement, i.e.,

```
METHOD MODEL=FLUORINE ENABLE
```

fluorine is automatically implanted during BF₂ implantation. the fluorine of BF₂ is implanted by a fluorine implant with an energy of 0.3893 times the user-specified implant energy. The fluorine dose is two times the BF₂ dose.

Analytic Damage Model

When the `DAMAGE` parameter on the `IMPLANT` statement is set true, an analytic model for the production of point defects during ion implantation is invoked. The interstitial and vacancy distributions created by the implantation are added to any interstitials and vacancies that may have existed in the structure prior to implantation. For more information on how damage is used to model transient-enhanced discussion, see [“Implant Damage Model” on page 2-98](#).

Damage Distribution Calculations

The damage distributions are calculated using the model of Hobler and Selberherr in its one-dimensional form [55]. This model approximates the damage profiles by combinations of Gaussian and exponential functions. The parameters of these functions were chosen to fit damage profiles predicted by Monte Carlo simulations over the range of implant energies between 1 and 300 keV. For a complete description of the equations and parameter values, see [Reference \[55\]](#). The implementation in TSUPREM-4 differs from that in [55] in that both the Gaussian and exponential components are used when implanting arsenic at energies above 170 keV. The model is extended to two dimensions by multiplying by a Gaussian with a standard deviation equal to the value of Hobler and Selberherr's a_3 parameter (the standard deviation of the Gaussian component of the vertical distribution). For BF₂ implants, the model for boron is used and the damage is scaled by 49/11. The scaling is omitted if `V.COMPAT=6.4` is specified on the `OPTION` statement.

Recommended Usage and Limitations

The damage model is designed only for antimony, arsenic, boron, and phosphorus. If **DAMAGE** is specified on a BF₂ implantation, the model for boron is used, with energy reduced by a factor of 0.2215. For other impurities, the damage model for one of the ions listed above (the one having atomic weight closest to that of the implanted ion) is used. There are no user-accessible coefficients associated with the damage model. If the damage calculation is followed by a **DIFFUSION** step, care should be used to ensure that a simulation region with sufficient depth is used to accommodate the rapid diffusion of point defects into the structure.

Monte Carlo Ion Implant Model

You can select a Monte Carlo model for ion implantation by specifying the **MONTECAR** parameter on the **IMPLANT** statement. TSUPREM-4 contains a comprehensive Monte Carlo model that incorporates models for crystalline silicon as well as amorphous models for silicon and other materials. The calculation models the crystal to amorphous transition that occurs during implantation. The model includes the effect of reflected ions and produces damage information in the form of vacancy and interstitial profiles. In addition, TSUPREM-4 contains models for calculating damage self-annealing of silicon substrates.

The Monte Carlo calculation is useful for examining a number of dependencies for which the empirical models are imperfect or incompletely calibrated. Some examples of these are: profile dependence on tilt and rotation angles, dose dependence, implant temperature dependence, and low energy implants. The Monte Carlo model is the only implant model in TSUPREM-4 that can simulate the effects of reflected ions.

The capabilities contained in TSUPREM-4 are a superset of the Monte Carlo ion implant functionality of the one-dimensional process simulator, PEPPER [56]. The calculation used in TSUPREM-4 assumes that ions lose energy through two processes.

- Nuclear scattering, where the nucleus of the ion elastically scatters off the nucleus of an atom in the target. This interaction is based on binary collision theory and is described in the following section.
- Interaction of the ion with the electrons of the target atoms. This mechanism is inelastic and does not alter the direction of the ion's motion. This is described in "[Amorphous Implant Calculation](#)" on page 2-91.

The calculation of damage and damage self-annealing of silicon substrates are described in "[Crystalline Implant Model](#)" on page 2-94.

Binary Scattering Theory

TSUPREM-4 models the nuclear collision energy loss according to classical binary scattering theory. The basic assumption of the nuclear collision energy loss mechanism is that the ion interacts with only one target atom at a time. This assumption allows the use of binary scattering theory from classical mechanics [57]. This section briefly outlines the pertinent results of this theory.

Consider a particle of mass M_1 and kinetic energy E_0 approaching a stationary particle with mass M_2 . The impact parameter, b , is the distance of closest approach if the particle is not deflected and gives a convenient measure of how close the collision is. After collision, the first particle deviates from its original course by an angle θ .

Energy Loss

It can be shown that the first particle loses kinetic energy

$$\frac{\Delta E_n}{E_0} = \frac{4M_1M_2}{(M_1 + M_2)^2} \cos^2(bI) \tag{Equation 2-262}$$

where ΔE_n is the energy lost by particle 1, E_0 is its energy before collision, and I is the integral

$$I = \int_0^{s_{max}} \frac{ds}{\sqrt{1 - \frac{V(s)}{E_r} - b^2 s^2}} \tag{Equation 2-263}$$

where $s = 1/r$ is the inverse separation between the two particles, $V(s)$ is the potential between the two particles (assumed to be repulsive), and

$$E_r = \frac{E_0}{1 + M_1/M_2} \tag{Equation 2-264}$$

is the reduced energy in the center of mass coordinates. The upper limit of the integral, s_{max} , is the inverse distance of closest approach of the two particles, and is given by the solution to the equation

$$1 - \frac{V(s_{max})}{E_r} - b^2 s_{max}^2 = 0 \tag{Equation 2-265}$$

Scattering Angle

The angle θ by which particle 1 is deflected is given by

$$\cos\theta = \frac{1 - 0.5 \left[1 + \frac{M_2}{M_1} \right] \Delta E/E_0}{\sqrt{1 - \Delta E/E_0}} \tag{Equation 2-266}$$

Dimensionless Form

Equations 2-262 through 2-266 are the basic equations for classical two-body scattering. The scattering integral, Equation 2-263, can be cast into a dimensionless form by assuming the potential has the form

$$V(s) = Z_1 Z_2 k_1 s g(a_u s) \tag{Equation 2-267}$$

where Z_1 is the charge on particle 1, Z_2 is the charge on particle 2, k_1 is the constant

$$k_1 = \frac{q^2}{4\pi\epsilon_0} = 14.39495 \times 10^{-7} \text{ keV}\mu\text{m} \quad \text{Equation 2-268}$$

$g(a_u s)$ is an arbitrary function of $a_u s$, to be defined later, and a_u is a unit of length. TSUPREM-4 uses

$$a_u = 0.8854 \times 10^{-4} \frac{0.529}{\left(Z_1^{0.23} + Z_2^{0.23} \right)} \mu\text{m} \quad \text{Equation 2-269}$$

which is the so-called Universal screening length [58] and a dimensionless impact parameter

$$b_n = b/a_u \quad \text{Equation 2-270}$$

and a dimensionless energy

$$\epsilon = \frac{a_u E_r}{Z_1 Z_2 k_1} \quad \text{Equation 2-271}$$

Using Equations 2-269, 2-270, and 2-271 in the scattering integral Equation 2-263, and making the substitution $s' = a_u s$ gives

$$I = \frac{1}{a_u} \int_0^{s'_{max}} \frac{ds'}{\sqrt{1 - s'g(s')/\epsilon - b_n^2 s'^2}} \quad \text{Equation 2-272}$$

From Equation 2-262, the quantity of interest is $\cos^2(bI)$, which becomes

$$\cos^2(bI) = \cos^2 \left[b_n \int_0^{s'_{max}} \frac{ds'}{\sqrt{1 - s'g(s')/\epsilon - b_n^2 s'^2}} \right] \quad \text{Equation 2-273}$$

Thus using Equation 2-273, $\cos^2(bI)$ can be evaluated in terms of the dimensionless variables b_n and ϵ , without reference to a particular particle's charge or mass.

Coulomb Potential

As an example of the above procedure, consider the Coulomb potential between two particles,

$$V(r) = \frac{Z_1 Z_2 k_1}{r} \quad \text{Equation 2-274}$$

or $V(s) = Z_1 Z_2 k_1 s$. In this case, $g(a_u s) = 1$. Then from [Equation 2-273](#)

$$\cos^2(bI) = \cos^2 \left[b_n \int_0^{s'_{max}} \frac{ds'}{\sqrt{1 - s'/\epsilon - b_n^2 s'^2}} \right] \tag{Equation 2-275}$$

with

$$s'_{max} = \frac{\left(\sqrt{1 + 4b_n^2 \epsilon^2} - 1 \right)}{2\epsilon b_n^2} \tag{Equation 2-276}$$

from a solution of [Equation 2-265](#). Then the integral can be evaluated exactly, giving

$$\cos^2(bI) = \frac{1}{1 + 4b_n^2 \epsilon^2} \tag{Equation 2-277}$$

For a given impact parameter b and incident energy E_0 , the dimensionless b_n and ϵ can be obtained from [Equations 2-270](#) and [2-271](#), giving $\cos^2(bI)$ from [Equation 2-277](#). Then the energy loss due to the collision is given by [Equation 2-262](#), and the angle at which particle 1 leaves the collision is given by [Equation 2-266](#).

Universal Potential

For the simple form of the Coulomb potential used in the example above, the scattering integral can be solved analytically. For more realistic inter-atomic potentials, however, the scattering integral cannot be evaluated analytically. For example, the Universal potential [58] that is used in TSUPREM-4 is

Equation 2-278

$$V(r) = \frac{Z_1 Z_2 k}{r} \left[0.18175 e^{-3.1998r/a_u} + 0.50986 e^{-0.94229r/a_u} + 0.28022 e^{-0.4029r/a_u} + 0.028171 e^{-0.20162r/a_u} \right]$$

An analytic solution does not exist since the upper limit of the integral in [Equation 2-263](#) is given by [Equation 2-265](#), which becomes a transcendental equation with this potential.

All Monte Carlo ion implantation codes use a formalism similar to [Equations 2-262](#) through [2-266](#) to treat the nuclear scattering; the difference between codes is in the method of evaluating the scattering integral, [Equation 2-263](#). The code MARLOWE [59] numerically integrates the scattering integral, providing accurate solutions at great computational expense. The program TRIM [60] fits an analytic function of five parameters to the values of the

scattering integral obtained by numerical integration. This technique retained the accuracy of MARLOWE while improving efficiency by an order of magnitude. However, TRIM still requires the evaluation of s_{max} , the inverse distance of closest approach at each collision. This requires solving the nonlinear Equation 2-265. With an initial guess of $s_{max} = 1/b$, Newton's method converges to an answer in about 3 to 5 iterations. If the Universal potential [58] is used, this requires 18 to 30 exponentials to be evaluated at each collision.

A different algorithm is used in TSUPREM-4 [61]. The quantity $\cos^2(bI)$, in its dimensionless form, Equation 2-273, is numerically integrated for a wide range of its parameters b_n and ϵ . These results are stored in tables. Then at each collision, $\cos^2(bI)$ is obtained from these tables. This scheme eliminates the need to find s_{max} for each collision, minimizing the amount of arithmetic performed during the calculation of an ion's trajectory, while retaining accuracy.

Tables for the Universal potential over a wide range of energies and impact parameters are provided for immediate use in TSUPREM-4. They span the normalized energy range of $10^{-5} \leq \epsilon \leq 100$ and the normalized impact parameter range $0 \leq b_n \leq 30$. For $\epsilon > 100$, the Coulomb form Equation 2-278 is used. Values of $\epsilon < 10^{-5}$ are not encountered for ion-atom combinations of interest at energies above the energy at which the ion is assumed to be stopped (10 eV). For values of $b_n > 30$, the ion is assumed to be undeflected.

Amorphous Implant Calculation

This section describes how the binary scattering theory of the previous section is used to calculate ion trajectories in an amorphous solid. TSUPREM-4 calculates a number of ion trajectories that can be specified using the **N. ION** parameter on the **IMPLANT** statement. The implant species can be any impurity (including user-defined impurities) for which the required information (atomic number and weight, electronic stopping powers, and so on) has been specified.

The calculation of ion trajectories proceeds as follows. Assume an ion with kinetic energy E_0 hits a target with an angle θ_0 with respect to the target normal. The surface of the target is assumed to be at $y = 0$, with y increasing vertically into the target. The incident energy E_0 can be set on the **IMPLANT** statement using the **ENERGY** parameter. The incident angle θ_0 can be specified on the **IMPLANT** statement using the **TILT** parameter.

Given the atomic density N_{dens} for the target material, the mean atomic separation between atoms in the target is $1/(N_{dens})^{1/3}$. The ion is assumed to travel a distance

$$L = 1/(N_{dens})^{1/3} \tag{Equation 2-279}$$

between scattering events. As the ion enters the target material, it approaches the first target atom with impact parameter b , defined in the previous section. The probability of finding a target atom between b and $b + \delta b$ is given by

$$w(b)\delta b = 2\pi N_{dens}^{2/3} b \delta b \tag{Equation 2-280}$$

for $b < 1/\sqrt{\pi N_{dens}^{2/3}}$. If R_{rand} is a uniformly distributed random number between 0 and 1, then the probability distribution gives

$$b = \sqrt{\frac{R_{rand}}{\pi N_{dens}^{2/3}}} \tag{Equation 2-281}$$

as described in [60].

Nuclear Stopping

Given the above definitions, the algorithm for calculating the energy loss through nuclear collisions experienced by the ion proceeds as follows:

1. A random number between 0 and 1 is chosen.
2. The normalized impact parameter for this collision is calculated from [Equations 2-270 and 2-281](#)

$$b = \frac{1}{a_u} \sqrt{\frac{R_{rand}}{\pi N_{dens}^{2/3}}} \tag{Equation 2-282}$$

3. The ion energy, E_0 , is normalized to

$$\varepsilon = \frac{a_u E_0}{(1 + M_1/M_2) Z_1 Z_2 k_1} \tag{Equation 2-283}$$

from [Equations 2-264 and 2-271](#).

4. Now the value of $\cos^2(bI)$ can be obtained from the tables, and [Equation 2-262](#) gives the energy loss due to nuclear scattering

$$\Delta E_0 = E_0 \frac{4M_1 M_2}{(M_1 + M_2)^2} \cos^2(bI) \tag{Equation 2-284}$$

This procedure is repeated for each collision event.

Electronic Stopping

The ion also loses energy by inelastic electronic processes, which include both nonlocal and local stopping power. At low energies this is modeled by

$$\Delta E_e = x^{nl} \cdot \Delta E_e^{nl} + x^{loc} \cdot \Delta E_e^{loc} \tag{Equation 2-285}$$

$$\Delta E_e^{nl} = L \cdot N_{dens} \cdot S_e \tag{Equation 2-286}$$

$$\Delta E_e^{loc} = \frac{S_e}{2\pi a^2} \cdot \exp(-p/a) \tag{Equation 2-287}$$

$$x^{nl} = \min(\text{NLOC} \cdot \text{PRE} \cdot \varepsilon^{\text{NLOC} \cdot \text{EXP}}, \text{NLOC} \cdot \text{MAX}) \tag{Equation 2-288}$$

$$x^{nl} + x^{loc} = 1 \tag{Equation 2-289}$$

$$S_e = \text{NLOC.K} \cdot \text{ES.RAND} \cdot E_0^{\text{ES.F.RAN}} \tag{Equation 2-290}$$

$$a = \text{LOC.FAC} \cdot \frac{a_U}{0.3} \tag{Equation 2-291}$$

where L is the free flight path between collisions, p is the impact parameter, ϵ is the scaled dimensionless energy, E_0 is the ion energy, and a_U is the universal screening length. **NLOC.PREN**, **NLOC.EXP**, **NLOC.MAX**, **NLOC.K**, **ES.RAND**, **ES.F.RAN**, and **LOC.FAC** can be defined on the **IMPURITY** statement.

In the absence of a specification using an **IMPURITY** statement, or if the specified values are zero, values for these parameters are obtained from the coefficient file. If no value or a value of zero is specified in the coefficient file for parameters **ES.RAND** and **ES.F.RAN**, the default values are given by [62]

$$\text{ES.RAND} = \frac{1.212 Z_1^{7/6} Z_2}{\left[Z_1^{2/3} + Z_2^{2/3} \right]^{3/2} M_1^{1/2}} \tag{Equation 2-292}$$

$$\text{ES.F.RAN} = 0.5 \tag{Equation 2-293}$$

where Z_1 is the ion atomic number and Z_2 is the composite target atomic number. For boron and phosphorus in silicon, the default values of **ES.RAND** have been set to 2.079 and 2.5, respectively [60]. For light ions ($Z_1 \leq 5$) and α -particles, the value for electronic stopping given by Equation 2-292 is very crude. Experimental values should be specified wherever possible [58], [63], [64], and [66], using **ES.RAND** and **ES.F.RAN** on the impurity statements.

Electronic Stopping at High Energies

At energies above **ES.BREAK**, Equation 2-293 is replaced by

$$\Delta E_e = LN_{dens} \text{ES.RAND} \cdot \text{ES.BREAK}^{\text{ES.F.H}} \left(\frac{E_0}{\text{ES.BREAK}} \right)^{\text{ES.F.H}} \tag{Equation 2-294}$$

where **ES.BREAK** and **ES.F.H** are parameters on the **IMPURITY** statement. Note that the high energy stopping model is used only when **ES.BREAK** is non-zero, i.e., the model can be disabled by setting **ES.BREAK**=0.

Total Energy Loss and Ion Deflection

The total change in energy of the ion after the i^{th} collision is the sum of nuclear energy loss, Equation 2-284, and electronic energy loss, Equation 2-285

$$E_i = E_{i-1} - \Delta E_n - \Delta E_e \tag{Equation 2-295}$$

The corresponding angle through which the ion is scattered is given by

$$\theta = \cos^{-1} \left[\frac{1 - \frac{1}{2}(1 + M_2/M_1)\Delta E_n/E_0}{\sqrt{1 - \Delta E_n/E_0}} \right] \tag{Equation 2-296}$$

Note that for $\Delta E_n/E_0 \ll 1$, θ approaches zero.

Ion Beam Width

The incident angle used in the Monte Carlo calculation given by **TILT** can be varied about its nominal value by specifying the **BEAMWIDT** parameter. For a non-zero value of **BEAMWIDT**, the angle used to calculate the incident velocity of each ion is varied about its nominal value, **TILT**, by the addition of $(\text{BEAMWIDT}) R_{rand}$ where R_{rand} is selected from a uniform distribution of random numbers from -1/2 to 1/2.

Crystalline Implant Model

The calculation described in the previous section is for amorphous layers. When the **CRYSTAL** parameter is set during a Monte Carlo ion implant calculation, silicon layers are treated with a more sophisticated calculation that incorporates the crystal structure of the silicon lattice. This calculation is intended to model channeling, which is the preferential penetration of implanted ions along crystal axes.

The calculation proceeds as in the amorphous case except that rather than selecting the collision of the implanted ion with target atoms based on the density of the target material and a random number, the simulation determines an impact parameter based on the implanted ion's position relative to sites on an idealized lattice. This is accomplished by discretizing the silicon lattice and calculating the impact parameter for each of eight lattice sites within each discretization cell. A single site is then selected for collision, and the energy loss is calculated as in the amorphous case.

Channeling

The effect of the **TILT** parameter is much more pronounced for implants into crystalline silicon than into amorphous silicon. Axial implants (obtained by setting **TILT** to zero) show an enhanced penetration due to channeling.

Channeling occurs naturally due to the inclusion of the structure of the silicon lattice. For a given incident ion energy, the critical angle is a measure of the angle of deviation from a crystal axis that is required to prevent an ion from being channeled along that axis. For channeling along the <100> axial direction, the critical angle is given by

$$\Psi_{100} = \Psi_0 E^{-\text{CRIT.F}} \tag{Equation 2-297}$$

where

$$\Psi_0 = 2.314 \times \left[a_0 Z_1 Z_2 a_{conv}^{-3} \right]^{\text{CRIT.F}} \tag{Equation 2-298}$$

a_{conv} is the conventional cell dimension of 5.431Å for silicon and **CRIT.F** is a parameter on the **IMPLANT** statement. The critical angle for the <110> direction is given by [67]

$$\Psi_{110} = \mathbf{CRIT.110} \times \Psi_{100} \quad \text{Equation 2-299}$$

where **CRIT.110** is a parameter on the **IMPLANT** statement.

Due to the lower electron density in the center of an axial channel, the electronic stopping of ions that are channeled is generally less than that for ions travelling in random directions. TSUPREM-4 allows explicit specification of the electronic stopping along the <100> and <110> directions. The channeled electronic stopping is selected whenever the ion is traveling within an angle with respect to a particular crystal axis given by

$$\Psi_{100} = \mathbf{CRIT.PRE} \cdot \Psi_{100} \quad \text{Equation 2-300}$$

and

$$\Psi_{110} = \mathbf{CRIT.PRE} \cdot \Psi_{100} \quad \text{Equation 2-301}$$

for the <100> and <110> directions, respectively. The factor **CRIT.PRE** is a fraction of the critical angle that can be specified on the **IMPLANT** statement.

The electronic stopping along the <100> and <110> crystal axes can be specified with the parameters **ES.100**, **ES.F.100**, **ES.110**, and **ES.F.110**. When the implanted ion is determined to be travelling along a <100> crystal axis, as described above, the values of **ES.100** and **ES.F.100** are substituted for **ES.RAND** and **ES.F.RAN**, respectively, in Equations 2-292 and 2-294 to determine the energy loss of the ion through electronic stopping.

For nonzero values of **TILT**, the penetration of implanted ions through silicon depends on the crystalline orientation of the x axis of the simulation space and the value of **ROTATION** because of planar channeling. The orientation can be set on the **INITIALIZE** statement with the **X.ORIENT** or **ROT.SUB** parameters.

For the Monte Carlo code, the multiple collision should be handled carefully, because the traditional multiple collision algorithm does not conserve both energy and momentum simultaneously [59], yet increases the computational time significantly due to the calculation of multiple collision partners. In TSUPREM-4, a novel approach is used to simulate the channeling effect. In this new approach, a small scattering angle is identified to be that of channel ions. This angle is further reduced to reflect the nature of multiple collision. So, if the scattering angle is $\theta < \mathbf{CHAN.CRI}$, it is replaced by an effective scattering angle, $\theta_{eff} = \theta / \mathbf{CHAN.FAC}$. The default parameters are **CHAN.CRI** = 11.54, and **CHAN.FAC** = 2.0 for all species. These parameters can also be changed on the **IMPURITY** statement. Increasing **CHAN.CRI** and/or **CHAN.FAC** causes more channeling.

Lattice Temperature

The temperature of the lattice can be specified using the **TEMPERAT** parameter. When the **VIBRATIO** parameter is set, the temperature specified by **TEMPERAT** is used in a Debye calculation to determine the rms vibration amplitude of silicon lattice atoms from their sites. This calculation can be superseded by specifying the rms vibration amplitude explicitly using the **X.RMS** parameter. At each collision site, the displacement of silicon lattice atoms from their idealized sites is selected as $(\mathbf{X.RMS}) R_{norm}$ where R_{norm} is a value selected from a normal distribution of random numbers with unity standard deviation.

Empirical observations of implant profiles produced by TSUPREM-4 have shown that better agreement with measured profiles is achieved by neglecting the dechanneling effect of lattice vibrations for each collision at which the ion's energy is above some limit. This limit can be specified using the **E.LIMIT** parameter. If no value for **E.LIMIT** is specified, TSUPREM-4 neglects lattice vibrations if the ion's energy is greater than 5 keV for boron or 40 species.

Lattice Damage

TSUPREM-4 contains a damage calculation that models the transition from crystalline to amorphous material which occurs in silicon as an implant proceeds. Other workers have modeled the crystal-to-amorphous transition by a critical energy model [68]. When the **DAMAGE** parameter is specified, a calculation is performed to determine the trajectories of silicon lattice atoms that are knocked from their sites in the lattice by collisions with implanted ions. A silicon atom is assumed to be knocked from its site when it absorbs an energy greater than a damage threshold from a collision. The value of this damage threshold can be set using the **THRESHOL** parameter on the **IMPLANT** statement. The silicon atoms freed from the lattice can in turn knock other atoms from their sites so that cascades of damage result [45]. TSUPREM-4 calculates the trajectories of these knock-ions with the same detail as the implanted ions. The program maintains efficiency by calculating only a weighted fraction of these secondaries. The fraction can be specified using the **REC.FRAC** parameter; setting this parameter equal to unity results in the calculation of trajectories for all secondaries as they are generated.

The output from the damage calculation produces information in the form of vacancy and interstitial profiles [69]. A vacancy is assumed to be formed whenever a lattice atom is knocked from its site. An interstitial is assumed to be formed whenever a silicon lattice atom that has been knocked from its site comes to rest. The profiles of interstitials and vacancies that result are retained as an initial condition for subsequent diffusion steps if the **DAMAGE** parameter is specified.

Damage Dechanneling

The accumulated damage has significant effect on the destination of the subsequent ions, thus altering the shape of the impurity profiles. This effect is known as *damage dechanneling*. TSUPREM-4 handles this problem by switching from the crystal model to the amorphous model based on the damage that has accumulated in the substrate. The probability for the selection of the amorphous model is proportional to the local interstitial concentration and a random number call. Silicon is treated as amorphous when

$$R_{rand} < \frac{C_I(x)}{\text{DISP.FAC} \cdot \text{MAX.DAMA}} \quad \text{Equation 2-302}$$

where $C_I(x)$ is the local interstitial concentration, and **MAX.DAMA** is the maximum damage allowed. **MAX.DAMA** can be specified on the **IMPLANT** statement, while **DISP.FAC** can be specified on the **IMPURITY** statement. Increasing **DISP.FAC** makes the profiles more like those implants into amorphous materials.



Note:

The amorphization process is not explicitly simulated by TSUPREM-4. However, for the Monte Carlo model, by common practice, when a critical amount of damage is accumulated in a certain region, a crystal/amorphous phase transition is assumed to occur in this region. For silicon implants, this critical damage is approximately 10% of the lattice density. Thus, if the interstitial concentration reaches more than $5e21 / \text{cm}^3$ for silicon, this region is considered to be amorphized. Using this criteria, TSUPREM-4 can reasonably predict the onset of amorphization and the thicknesses of the amorphous layers for high dose implants.

Damage Annealing

TSUPREM-4 contains a model for the self-annealing of the damage produced during implantation. Experimental measurements of the dose required to amorphize silicon as a function of temperature show an increase with increasing temperature [70]. This is modeled in TSUPREM-4 by assuming that a temperature-dependent fraction of the point defects is self-annealing. A temperature-dependent value for **THRESHOL** has been empirically determined. This compensates for recombination by eliminating calculation of trajectories of ions that eventually recombine [71].

Besides the *in situ* annealing that occurs during ion implantation, the damage which accumulates in the silicon material is annealed during subsequent high temperature processing steps. Until such a step, the damage produced by an implant step serves as an initial condition for subsequent implantations.

Number of Ions

The results of the Monte Carlo calculation are subject to statistical variation due to the finite number of particles that make up the solution. The resulting noise in the solution can be reduced by increasing the value of the **N.ION** parameter above its default value of 1000. The solution time is directly proportional to this value. The solution time can be reduced at the expense of a higher statistical variation in the results by reducing the value of **N.ION**.

BF₂ Implantation

The Monte Carlo model does not model the dissociation of BF₂ ions. If **BF₂** is specified with **MONTECAR** on the **IMPLANT** statement, the BF₂ implant is approximated by a boron implant with an energy of 0.2215 times the user-specified implant energy. The damage is scaled by 49/11 to account for the fluorine ions. (The scaling is omitted if **V.COMPAT**=6.4 is specified on the **OPTION** statement.)

Implant Damage Model

The implant damage model in TSUPREM-4 accounts both for the silicon atoms knocked out of lattice sites and for interstitials produced when silicon atoms are displaced by implanted ions (the “plus one” model). In addition, the effects of amorphization are taken into account, and an analytical model of point defect recombination has been included to speed up subsequent diffusion steps. The implant damage model works with both the analytical and Monte Carlo implantation models.

There are two slightly different versions of the implant damage model in TSUPREM-4. The newer version, the *cumulative damage model*, automatically accumulates the damage produced by successive implants and simulates defect recombination and amorphous regrowth at the start of the first high-temperature step following a series of implants. The *old damage model* simulates defect recombination and amorphous regrowth at every implant step by default.

Damage Produced During Implant

Both models start by calculating the damage produced during each implantation step. The concentrations of interstitials and vacancies produced by the impact of implanted ions (or recoiling silicon atoms) are denoted by I_F and V_F , respectively. When the analytical implant model is used, I_F and V_F are calculated according the model of Hobler and Selberher [55] and are assumed to be equal at every point in the structure. When the Monte Carlo model is used, I_F and V_F may be slightly different because most displaced lattice ions are knocked forward while a few are scattered out of the structure entirely. If the increase in concentration of the implanted species is ΔC , then the number of added interstitials and vacancies due to the implant is

$$\Delta I = \mathbf{D.SCALE} \times I_F + f_{pl} \times \Delta C \tag{Equation 2-303}$$

$$\Delta V = \mathbf{D.SCALE} \times V_F$$

$$f_{pl} = \begin{cases} \mathbf{D.PLUS} & , \text{ if } \mathbf{D.PLUS} \text{ specified} \\ f_{pl\infty} f_{pld} & , \text{ otherwise} \end{cases} \tag{Equation 2-304}$$

$$f_{pl\infty} = 1 + \mathbf{D.PHDF} m^{\mathbf{D.PME}} E^{\lambda_\infty} \tag{Equation 2-305}$$

$$f_{pld} = \mathbf{D.P1} + \mathbf{D.PCONST} m^{\mathbf{D.PDME}} E^{\lambda_d} \left(\frac{\mathbf{D.PNORM}}{d} \right)^{\mathbf{D.PDE}} R_p^{\mathbf{D.PDRE}} \tag{Equation 2-306}$$

where

$$\lambda_\infty = \mathbf{D.PLF} m^{\mathbf{D.PLME}} \tag{Equation 2-307}$$

$$\lambda_d = \mathbf{D.PDLF} m^{\mathbf{D.PDLME}} \tag{Equation 2-308}$$

and:

- d and E are the dose (per cm^2) and energy (in keV) of the implant
- R_p is the projected range (in nm)
- m is the mass of the implanted ion (in atomic mass units)
- **D.PNORM** and **D.PDE** are parameters on the **IMPURITY** and/or **MATERIAL** statements (see below)
- **D.PHDF**, **D.PME**, **D.PCONST**, **D.PDME**, **D.PDRE**, **D.PLF**, **D.PLME**, **D.PDLF**, and **D.PDLME** are parameters on the **MATERIAL** statement
- **D.PLUS** and **D.P1** are parameters on the **IMPLANT** statement.

The parameter **D.SCALE** (on the **IMPLANT** statement) adjusts the Frenkel pair density to account for uncertainties in the damage creation models. You can use it to change the simulated amorphization depth of amorphizing implants.

+1 and Effective +n Damage Models

The ΔC term accounts for the fact that each implanted ion displaces a silicon atom, producing an interstitial. The recombination of vacancies at the surface leaves an additional excess of interstitials, which is accounted for by the factor f_{pl} . This excess can be specified directly by the **D.PLUS** parameter on the **IMPLANT** statement. If **D.PLUS** is not specified, an *effective +n* model [72] is used to determine the number of excess interstitials as a function of energy, ion mass, and projected range. Higher energy or ion mass increases the separation between the generated interstitials and vacancies, increasing the chance that vacancies will recombine at the surface before recombining with an interstitial. Increasing the range of the implant has the opposite effect. The effectiveness of the I-V recombination decreases at low doses [73][74].

The effective +n model is designed for use with the analytical implant model. For Monte Carlo implants the **D.PHDF** and **D.PCONST** terms are ignored. (The Monte Carlo damage model inherently includes the effects modeled by the new analytical model.)

The parameters **D.PNORM** and **D.PDE** can be specified on either the **IMPURITY** or **MATERIAL** statement. If either parameter is given a nonzero value on an **IMPURITY** statement, the values on the **IMPURITY** statement are used for that impurity. If neither value is specified on an **IMPURITY** statement (or if both values are set to zero), the values specified on the **MATERIAL** statement are used.

This model is a superset of several models that have been published in the literature; by setting the parameters appropriately, any of these published models can be used. By default, the parameters are set to give the model of Hobler and Moroz [75]:

```
MATERIAL MAT=SILICON D.PHDF=0.0905 D.PME=0.85 +  
D.PLF=-2 D.PLME=-0.5 D.PCONST=3.25 D.PDME=-0.5 +  
D.PDLF=-2 D.PDLME=-0.5 D.PNORM=3e12 D.PDE=1 +  
D.PDRE=0
```

Another choice of parameters gives the model of Pelaz *et al.* [72] with added dose dependence [73][74], as implemented in TSUPREM-4 1999.4:

```
MATERIAL MAT=SILICON D.PHDF=0 D.PCONST=0.42 +
D.PDME=0.5 D.PDLF=0.5 D.PDLME=0 D.PNORM=1e14 +
D.PDE=0.5 D.PDRE=-0.75
```

The original model of Pelaz *et al.* is obtained with the parameters

```
MATERIAL MAT=SILICON D.PHDF=0 D.PCONST=0.42 +
D.PDME=0.5 D.PDLF=0.5 D.PDLME=0 D.PNORM=0 +
D.PDRE=-0.75
```

For BF₂ implants the mass and energy of the boron ion are used, and the dose is multiplied by three to account for the effect of the fluorine atoms.

You can use **D.P1** on the **IMPLANT** statement to adjust the amount of TED produced by an implant; it defaults to 1.0 if not specified.

Cumulative Damage Model

The cumulative damage model calculates a quantity called *damage* that represents the total damage at any point in the structure from all implants since the last high-temperature processing step. An implant step increases the damage by

$$\Delta D = \min(\Delta I, \Delta V) \quad \text{Equation 2-309}$$

The interstitial, vacancy, and damage concentrations following an implant are then given by

$$\begin{aligned} I &= I_0 + \Delta I - \Delta D \\ V &= V_0 + \Delta V - \Delta D \end{aligned} \quad \text{Equation 2-310}$$

$$D = D_0 + \Delta D \quad \text{Equation 2-311}$$

where I_0 , V_0 , and D_0 are the concentrations of interstitials, vacancies, and damage, respectively, before the implant.

Most of the Frenkel pair damage produced during implantation will recombine during the first few seconds of the first post-implant annealing step. This is modeled by setting $D = 0$ at the start of the first diffusion step following an implant. Incomplete recombination of Frenkel pair damage is modeled with the **D.RECOMB** parameter on the **DIFFUSION** statement:

$$\begin{aligned} I_1 &= I_0 + \mathbf{D.RECOMB} \times D_0 \\ V_1 &= V_0 + \mathbf{D.RECOMB} \times D_0 \end{aligned} \quad \text{Equation 2-312}$$

where I_0 , V_0 , and D_0 are the concentrations of interstitials, vacancies, and damage, respectively, before the start of the diffusion.

Crystalline regrowth of silicon amorphized by implant damage normally occurs during the ramp-up phase of the first post-implant annealing step. This is modeled by setting the interstitial and vacancy concentrations to their equilibrium values in the amorphized region. The transition between the crystalline and amorphized regions is smoothed by

$$s = 0.5 - 0.5 \times \tanh[\mathbf{DAM.GRAD} \times \ln(D_0/\mathbf{MAX.DAMA})] \quad \text{Equation 2-313}$$

$$I = I^* + s(I_1 - I^*) \quad \text{Equation 2-314}$$

$$V = V^* + s(V_1 - V^*)$$

where **MAX.DAMA** and **DAM.GRAD** are parameters specified for silicon on the **MATERIAL** statement. When the **ACT.TRAN** model is used and **^CL.INI.A** has been specified for an implanted dopant in silicon, the same smoothing function is used to smooth the transition from no activation (in the crystalline region) to full activation (in the amorphized and regrown region) of the dopant.

The cumulative damage model is used whenever a nonzero value for **MAX.DAMA** has been specified for silicon on the **MATERIAL** statement. It is the preferred model and is enabled by default.

Old Damage Model

In the old damage model, all point defect recombination and amorphous layer regrowth takes place at the end of an implant step. The model is described by the following equations, where **MAX.DAMA**, and **D.RECOMB** are parameters on the **IMPLANT** statement:

$$\begin{aligned} I_1 &= I_0 + \Delta I \\ V_1 &= V_0 + \Delta V \end{aligned} \quad \text{Equation 2-315}$$

where I_0 and V_0 are the interstitial and vacancy concentrations before the implant. If **D.RECOMB** is set true on the **IMPLANT** statement, recombination of Frenkel pairs and regrowth of amorphized regions are modeled by:

$$R = \min(I_1 - I^*, V_1 - V^*) \quad \text{Equation 2-316}$$

$$I = \begin{cases} I_1 - \max(R, 0), & I_1 \leq \mathbf{MAX.DAMA} \\ I^*, & I_1 > \mathbf{MAX.DAMA} \end{cases} \quad \text{Equation 2-317}$$

$$V = \begin{cases} V_1 - \max(R, 0), & I_1 \leq \mathbf{MAX.DAMA} \\ V^*, & I_1 > \mathbf{MAX.DAMA} \end{cases} \quad \text{Equation 2-318}$$

If **D.RECOMB** is set false on the **IMPLANT** statement, then Frenkel pair recombination and amorphous layer regrowth are not included in the implant step:

$$\begin{aligned} I &= I_1 \\ V &= V_1 \end{aligned} \quad \text{Equation 2-319}$$

The old damage model is selected by setting **MAX.DAMA=0** for silicon on the **MATERIAL** statement. The old damage model should only be used for compatibility with old input files; the cumulative model is more physically realistic and is easier to use.

Conservation of Total Defect Concentrations

Some of the point defects introduced during ion implantation will combine with dopants or other defects to produce pairs or clusters. These point defects are removed from the population of free point defects to ensure that the total number of added defects (free and otherwise) is given by the models described above. Thus the reported number of (free) point defects following an implant may be less than the values calculated by the models above; the remaining defects are contained in dopant-defect pairs, small clusters of defects, or other defect-absorbing complexes that are modeled by the program.

The incorporation of defects into dopant-defect pairs only occurs when **PD.PTIME** is enabled (i.e., when **PD.FULL** is used). Thus the action of the damage model depends on the models in effect at the time of the implant. For correct simulation of TED with the **PD.FULL** model, **PD.FULL** should be specified *before* the implant. In general, parameters that affect the pairing or clustering of point defects should not be changed between an implant with damage and the subsequent annealing step.

Using the Implant Damage Model

The default parameter values are expected to give reasonable results in most cases. However, you might obtain improved results with the following modifications:

- You can simplify the calculation by setting **D.SCALE**=0.0. The result is the scaled plus one model without the effects of amorphization.
- You can modify the amorphization threshold by reducing **MAX.DAMA**. This causes amorphization effects to appear at lower doses. You can also change the amorphization depth for a particular implant by modifying **D.SCALE**.
- **D.SCALE** can be reduced to model the effects of self-annealing during implantation.
- **D.PLUS** can be adjusted to change the amount of transient-enhanced diffusion. This may be necessary for modeling situations in which interstitials are bound in clusters after an implantation or where an imbalance between interstitials and vacancies is produced by recombination at the surface.
- *Old model only:* To model the accumulation of damage in consecutive implants, you should specify **^D.RECOMB** in all implants except the last. This allows the full damage produced by each implant to accumulate, with recombination occurring only after the last one. Damage accumulation is automatic with the cumulative model.
- The abruptness of the amorphous/single-crystal interface can be adjusted with the **DAM.GRAD** parameter on the **MATERIAL** statement. Smaller values give a more extended interface while larger values give a more abrupt interface; typical values are between 1 and 100. **DAM.GRAD** can be used to reduce the sensitivity of the amorphization model to the grid spacing.

The damage model can produce concentrations of point defects that are much greater than those produced by oxidation. Accurate simulation of diffusion in the presence of such high defect concentrations requires that the **PD.FULL** diffusion model be used. Note that very small time steps are required in the initial stages of diffusion following an implant with damage. It is suggested that an initial time

step on the order of 10^{-6} minutes be used. If a larger initial time step is used, there is a delay during the first step while the start of the post-implant transient is simulated. Use of **D.RECOMB** reduces the peak point defect concentration and lessens the need for small time steps and the **PD.FULL** model. If the **PD.FERMI** model is in effect when an implant with **DAMAGE** is specified, the **PD.TRANS** model is automatically enabled.

Boundary Conditions for Ion Implantation

Ion implantation does not obey the reflecting boundary conditions that are usually used at the left and right edges of the structure. Instead, the analytic implantation extends the structure at a reflecting boundary out to infinity, while the Monte Carlo model uses the boundary condition specified by your choice of the parameters, **VACUUM**, **PERIODIC**, or **REFLECT**. Thus there is a loss of accuracy in the implanted profile unless the lateral spread of the implant distribution is small compared to the distance between the edge of the structure and the nearest mask edge. In some cases, you may need to reflect the structure before implantation and truncate it afterwards to ensure the accuracy of the implanted profile.

For Monte Carlo implants, three choices of boundary conditions are available.

1. The default, **PERIODIC**, specifies that ions leaving one side of the structure re-enter on the other side, with the same velocity.
This condition gives accurate answers for one-dimensional structures and for structures where the sequence and thickness of layers are the same at the left and right edges.
2. **REFLECT** specifies that ions hitting the edge of the structure are reflected back into the structure.
This condition is accurate for **TILT=0** implants and for pairs of implants with opposite tilts.
3. **VACUUM** specifies that ions leaving the structure through the sides are lost.
This boundary condition is a poor approximation for most structures.

There are many situations (particularly single-tilted implants into two-dimensional structures) where none of the available boundary conditions is perfectly accurate. As with the analytical implant model, you lose some accuracy unless the lateral spread of the implant distribution is small compared to the distance between the edge of the structure and the nearest mask edge. Again, you may need to reflect the structure before implantation and truncate it afterwards to ensure the accuracy of the implanted profile.

Polysilicon Monte Carlo Implant Model

The polysilicon Monte Carlo implant model works by constantly switching between the crystal model and the amorphous model. The probability of switching from the crystal model to the amorphous model is determined by the accumulative path length (*pathlength*) and polysilicon grain size (**POLY.GSZ**). If:

$pathlength > (POLY.FAC * POLY.GSZ)$

it switches from the crystal model to the amorphous model. Besides, if:

$rand(x) < pathlength / (POLY.FAC * POLY.GSZ)$

it also switches from the crystal model to the amorphous model. After processing a collision for the amorphous model, the *pathlength* is reset to zero, and it starts accumulating the *pathlength* again, and the crystal model is selected. The model used for the next collision is again determined by the same rules. This process is repeated until the simulation is finished. **POLY.GSZ** is specified on the **IMPLANT** statement, while the **POLY.FAC** parameter is specified on the **MATERIAL** statement.

The polysilicon Monte Carlo implant model is selected by specifying the grain size **POLY.GSZ** on the **IMPLANT** statement.

Ion Implantation into Silicon Carbide

Since version 6.6 TSUPREM-4 has extended the capability of the Monte Carlo implantation model to implants into silicon carbide (SiC). SiC implant model can be selected either by the **INITIALIZE** or **REGION** statements. For example,

- **INITIALIZE MATER=sic**
- **REGION MATER=sic XLO=left XHI=right YLO=top YHI=bottom
INITIALIZE**

The **IMPLANT** statement format for SiC is the same as silicon.

Epitaxial Growth

The **EPITAXY** statement (see [page 3-121](#)) is used to model the epitaxial growth of silicon layers. An epitaxial layer can only be grown when the top layer of the structure consists of single crystal silicon. The duration of the epitaxy step (in minutes) is specified with the **TIME** parameter. The initial temperature of the step (in °C) is given by the **TEMPERAT** parameter. Linear variation of the temperature over the step can be specified with the **T.RATE** or **T.FINAL** parameters. If **T.RATE** is specified, the temperature varies as

$$T_c = \text{TEMPERAT} + \text{T.RATE} \times t \quad \text{Equation 2-320}$$

where

- t is the time since the start of the step
- T_c is the diffusion temperature (in °C) at time t

If **T.FINAL** is specified, the temperature varies as

$$T_c = \text{TEMPERAT} + \frac{(\text{T.FINAL} - \text{TEMPERAT})}{\text{TIME}} t \quad \text{Equation 2-321}$$

If neither **T.RATE** nor **T.FINAL** is specified, the temperature is constant. The physical coefficients that depend on temperature are presumed to be valid in the range 800 to 1250°C, but temperatures outside this range may be specified.

Layer Thickness

The thickness of the epitaxial layer must be specified with the **THICKNES** parameter. The grid distribution within the epitaxial layer can be controlled with the **DY**, **YDY**, and **SPACES** parameters, as described in “Changes to the Mesh During Processing” on page 2-7.

Incorporation of Impurities

One or more impurities may be incorporated into the growing layer by using the parameters **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHOR** to indicate the concentrations of impurities present in the ambient gas. The concentrations of multiple impurities can be specified with these parameters. Impurities can also be specified using the **IMPURITY** and **I.CONC** parameters. For a single impurity, the resistivity may be specified in place of the impurity concentration.

Diffusion of Impurities

The diffusion equations are solved for all the mobile species present in the structure during the epitaxial growth step (see “Diffusion of Impurities” on page 2-17 and “Diffusion of Point Defects” on page 2-32). The epitaxial growth is divided into diffusion steps proportional in length to the thicknesses of the grid spacings in the deposited layer. An epitaxial step that specifies five grid spaces in the deposited layer is simulated by five diffusion steps.

Deposition

TSUPREM-4 models the deposition of material layers at temperatures low enough that impurity diffusion can be ignored during the deposition. The **DEPOSITION** statement specifies the deposition of a material on the exposed surfaces of the existing structure (see page 3-90). The material to be deposited is specified by one of the seven logical parameters for identifying a material: **SILICON**, **OXIDE**, **OXYNITRI**, **NITRIDE**, **POLYSILI**, **ALUMINUM**, or **PHOTORES**; or by specifying the name of the material with the **MATERIAL** statement.



Note:

The implementation of the deposition capability assumes that the top surface is exposed, and allows the right side of the structure to either be exposed or on a reflecting boundary. Deposition should not be attempted

when the left or bottom sides of the structure are exposed, or when the top surface is not exposed.

Layer Thickness

The thickness of the deposited layer must be specified with the **THICKNES** parameter. The deposition is conformal, i.e., all points within a distance of **THICKNES** of the exposed surface are included in the new layer. The generation of the mesh in the new layer and the use of the **DY**, **YDY**, **SPACES** and **ARC.SPAC** parameters are described in [“Changes to the Mesh During Processing” on page 2-7](#).

Incorporation of Impurities

The deposited layer can be doped with one or more impurities. The impurity concentrations are specified with the **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHOR** parameters. Impurities can also be specified using the **IMPURITY** and **I.CONC** parameters. For a single impurity, the resistivity may be specified in place of the impurity concentration. If single crystal silicon is deposited, its orientation the same as the substrate orientation specified by the **INITIALIZE** statement.

Photoresist Type

The type of deposited photoresist can be specified as either **POSITIVE** or **NEGATIVE**. All photoresist in the structure is assumed to be of this type. This type of resist is used by the **DEVELOP** statement to determine whether exposed or unexposed resist should be removed (see [“Masking, Exposure, and Development of Photoresist” on page 2-107](#)).

Polycrystalline Materials

Deposition of polycrystalline materials uses the models for initial grain size and orientation described in [“Modeling Polycrystalline Materials” on page 2-112](#). These models depend on the **TEMPERAT** and **GSZ.LIN** parameters on the **DEPOSITION** statement.

Deposition with Taurus-Topography

TSUPREM-4 can call Taurus-Topography to simulate deposition steps using detailed physical models for processes such as physical vapor deposition (PVD), chemical vapor deposition (CVD), plasma-enhanced CVD, high-density plasma deposition, atmospheric pressure CVD, spin-on glass (SOG), and reflow. Deposition using Taurus-Topography is specified with the **TOPOGRAP** parameter on

the **DEPOSITION** statement (see [“Deposition with Taurus-Topography” on page 3-93](#)).

Masking, Exposure, and Development of Photoresist

Masking, exposure, and development of photoresist are used to transfer an image on a mask to a structure on a semiconductor wafer. Masking information is read from a mask file created by Taurus Layout—IC Layout Interface. For each mask level, the starting and ending coordinates of each opaque region are recorded. The **EXPOSE** statement uses these x coordinates to determine which portions of the photoresist in a structure should be marked as exposed (in the photographic sense). The **DEVELOP** statement removes all positive photoresist that has been marked as exposed, or all negative photoresist that has not been marked as exposed.

TSUPREM-4 uses idealized exposure and development models: photoresist lines always have vertical sidewalls, positioned directly beneath mask edges. If accurate physical models of photolithographic processes are needed, a simulator such as Taurus-Lithography should be used.

Etching

TSUPREM-4 allows the removal of material layers or portions of layers using the **ETCH** statement (see [page 3-98](#)). The material to be removed is specified by one of the seven logical parameters for identifying a material: **SILICON**, **OXIDE**, **OXYNITRI**, **NITRIDE**, **POLYSILI**, **ALUMINUM**, or **PHOTORES**; or by specifying the name of the material with the **MATERIAL** parameter. If a material is specified, only regions of that material are subject to removal; if no material is specified, the entire structure is subject to removal.

TSUPREM-4 can also call Taurus-Topography to simulate etch steps using detailed physical models.

Defining the Etch Region

The region to be removed (the *etch region*) can be defined in one of six ways:

1. The **TRAPEZOI** parameter specifies that removal occurs where etchable material appears at the exposed surface of the structure. The shape of the region is specified with the **THICKNES**, **ANGLE**, and **UNDERCUT** parameters. In a planar substrate, the region to be removed is trapezoidal in shape. [“Trapezoidal Etch Model” on page 2-108](#) describes the **TRAPEZOI** model.
2. The **ISOTROPI** parameter specifies that removal occurs in a region within the given **THICKNES** of the exposed surface.

3. The **LEFT** or **RIGHT** parameter specifies that removal occurs to the left or right of a line defined by the coordinates **P1 . X**, **P1 . Y**, **P2 . X**, and **P2 . Y**.
4. The **START**, **CONTINUE**, and **DONE** parameters are used to specify an arbitrary region to be removed. A series of **ETCH** statements are given, each one using the **X** and **Y** parameters to specify the location of one point on a polygon that defines the etch boundary. The first **ETCH** statement in the series uses the **START** parameter, the last uses the **DONE** parameter, and the statements in between use the **CONTINUE** parameter. The polygon defining the region to be removed is closed automatically by connecting the last point to the first. A material specification can appear on any of the **ETCH** statements; if more than one of the **ETCH** statements contains a material specification, all but the last are ignored.
5. The **ALL** parameter specifies that the entire structure is subject to removal. It is only useful with a material specification.
6. The **OLD . DRY** parameter specifies that the region to be removed consists of all points within a vertical distance **THICKNES** of the top surface. This model (called the **DRY** model in TSUPREM-4 versions 5.0 and older) has been superseded by the **TRAPEZOI** model.

If no region is specified, **TRAPEZOI** is assumed.

Removal of Material

Etching proceeds from exposed surfaces through material of the specified type. Thus a point within the structure is removed if it meets all of the following conditions:

- The material at the point is of the specified type (or no material is specified).
- The point lies within the etch region.
- There is a continuous path, through the specified material and within the etch region, from the point to an exposed surface.



Note:

*It is not possible to etch holes in a structure, but it is possible to cut a structure into two or more pieces with the **ETCH** statement. All pieces except the one with the largest area are discarded. A warning is printed for each detached piece of the structure that is removed.*

Trapezoidal Etch Model

The **TRAPEZOI** etch model provides a simple but flexible approximation to a number of real etching processes. The location of the etch is determined by masking layers (i.e., layers of nonetchable material), and thus does not require mask-dependent coordinates to be specified. This means the **TRAPEZOI** model can be used with the photoresist masking, exposure, and development capabilities of

TSUPREM-4 to create process descriptions that are independent of any particular mask layout.

Parameters

The **TRAPEZOI** model uses the three parameters **THICKNES**, **ANGLE**, and **UNDERCUT** to specify the shape of the region to be removed. **THICKNES** specifies the vertical depth (in microns), **ANGLE** specifies the angle (in degrees) of the resulting sidewalls, and **UNDERCUT** specifies the horizontal penetration (in microns) of the etch under the edges of the masking layer.

These parameters can be used to approximate a number of real etching processes, including combinations of vertical and isotropic etches, V-groove etches, and etches that produce retrograde sidewall profiles.

Etch Steps

An etch with the **TRAPEZOI** model is done in three steps:

1. A vertical etch to depth **THICKNES** is performed. This etch does not apply to portions of the surface that are masked by nonetchable materials or shadowed by etchable or nonetchable materials, nor is it used on segments of the surface that form an angle greater than **ANGLE** to the horizontal.
2. A horizontal etch is performed. Surfaces that were exposed at the start of Step 1 are etched horizontally by the distance **UNDERCUT**. Surfaces that were exposed during Step 1 are etched by a distance proportional to the length of time between when they first became exposed and the end of Step 1. Thus a sidewall exposed 3/4 of the way into Step 1 is etched horizontally by 1/4 of **UNDERCUT**. (An exception is made when an **ANGLE** greater than 90° is specified; this case is described below.)
3. Where overhangs of etchable material are present at the end of Step 2, a vertical *upwards* etch (i.e., in the $-y$ direction) is performed. On surfaces that were exposed at the start of Step 2, this etch is to a distance **UNDERCUT**; on surfaces that were first exposed during the course of Step 2, the distance of this etch is reduced in proportion to the time from the start of Step 2. This step approximates the undercutting of the mask due to the isotropic component of the etch.

Etch Examples

Figure 2-2 shows a number of examples of the **TRAPEZOI** etch model. In each case, an etchable layer (light and medium gray) is partially masked by a nonetchable layer (dark gray).

Simple Structure

When the **THICKNES**, **ANGLE**, and **UNDERCUT** parameters satisfy the relationship

$$\frac{\text{THICKNES}}{\text{UNDERCUT}} = \tan(\text{ANGLE}) \quad \text{Equation 2-322}$$

the etch approximates a vertical etch with an isotropic component. This is the case whenever two or fewer of the parameters **THICKNES**, **ANGLE**, and **UNDERCUT**

are specified. Figure 2-2 (a) and (b) show the effect of this type of etch on various starting structures.

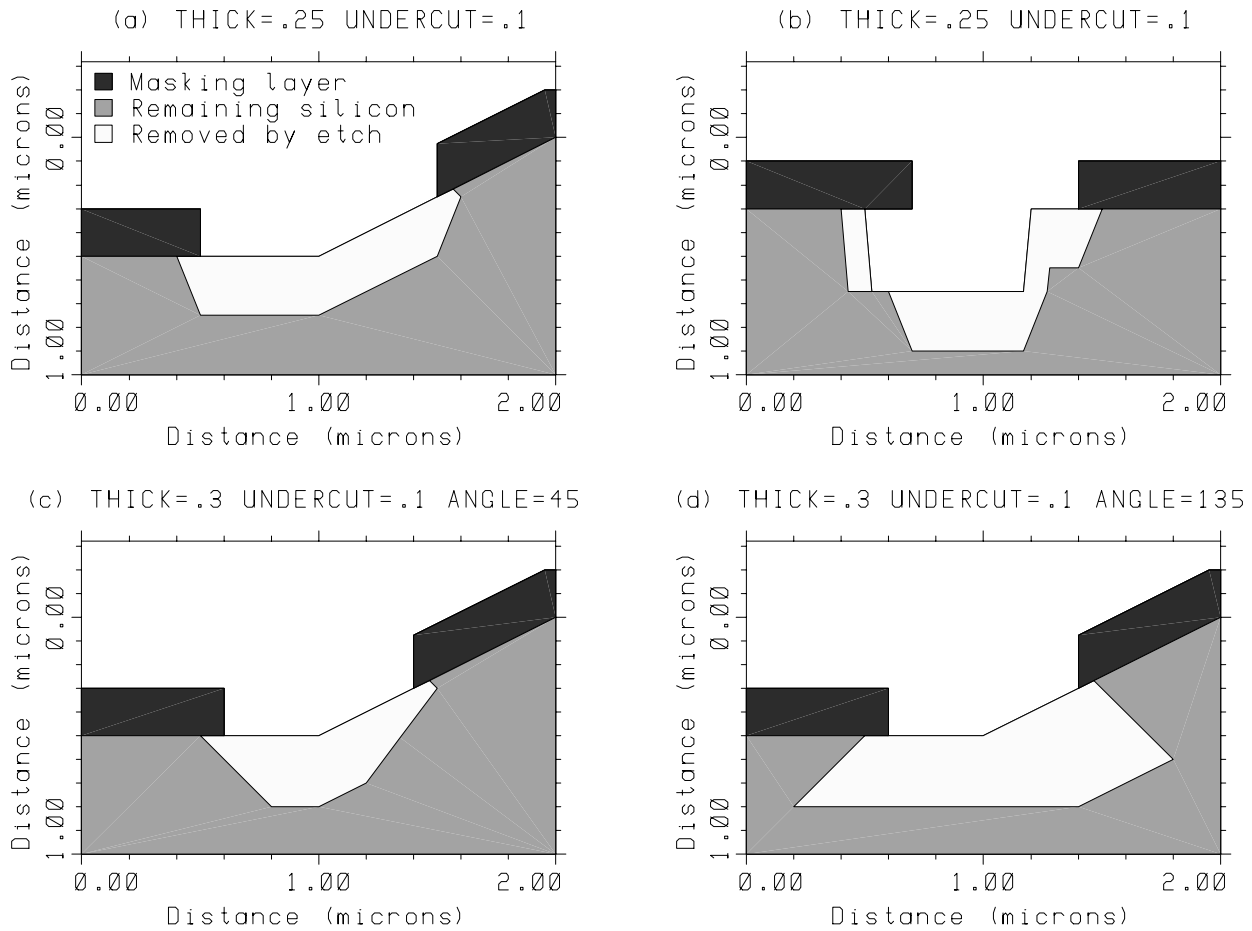


Figure 2-2 Examples of the trapezoidal etch model

The left half of Figure 2-2(a) shows the result when etching a planar substrate: the etch region is a trapezoid of depth **THICKNES**, extending a distance **UNDERCUT** beneath the mask edge, and with a sidewall slope of **ANGLE** degrees. The right half of Figure 2-2(a) shows the result when etching a nonplanar surface: Step 1 of the etch sequence etches the exposed surface vertically to a depth of **THICKNES** microns. Step 2 etches the resulting sidewall in the horizontal direction, producing an undercutting of the mask and the sloped sidewall. In this case Step 3 also has an effect, etching upwards from the undercut region. Thus, the *hook* in the final silicon profile is the result of approximating the isotropic component of the etch. In every case, the intersection between the bottom of the etch region and the sidewall occurs directly under the edge of the mask.

Structure with Overhangs

Figure 2-2 (b) shows the result of etching a more complicated structure. The left side of the figure shows the effects of overhangs. The vertical etch (Step 1) is suppressed where the silicon is shadowed by the masking layer. The horizontal etch (Step 2) applies to the entire structure, including the near-vertical sidewalls shel-

tered under the overhang. The right half of the figure shows what happens to exposed near-vertical sidewalls: Again the horizontal surfaces are etched vertically in Step 1, then both the sidewalls created by the vertical etch and the original near-vertical sidewalls are etched horizontally in Step 2. The effect is that material is removed if it can be etched by *either* the vertical (anisotropic) component or the horizontal (isotropic) component of the etch.

Complex Structures

Figure 2-2 (c) and (d) show what happens when Equation 2-322 is not satisfied. In Figure 2-2 (c) you have $\text{THICKNESS}/\text{UNDERCUT} < \tan(\text{ANGLE})$. In this case the sloped sidewall of the etch extends out under the opening in the mask. The intersection between the bottom of the etch region and the sidewall is no longer directly beneath the edge of the mask. If the mask opening is narrow enough, the bottom of the etch region disappears entirely, resulting in a V-groove etch. To produce this etch shape, Step 1 of the etch process is modified to reduce the depth of the vertical etch near the edges of the mask opening. Note that in this situation, even the tiniest speck of nonetchable material can produce a triangular mound of unetched material in the final structure.

Figure 2-2 (d) shows the case where $\text{ANGLE} > 90$. In this case, the bottom of the etched region is wider than the opening in the masking layer, producing overhanging sidewalls. This etch is accomplished by modifying Step 2 of the etch procedure to etch further horizontally at the bottom of the sidewalls formed by Step 1 than at the top. The apparent etch depth of 0.5 microns at the right side of the mask opening is the result of a 0.3 micron vertical etch of the original sloped surface (Step 1) followed by a 0.4 micron horizontal etch of the sloped “bottom wall” that results from Step 1.

Etching with Taurus-Topography

TSUPREM-4 can call Taurus-Topography to simulate etch steps using detailed physical models for processes such as wet etching, dry etching (RIE), ion milling, and chemical mechanical polishing (CMP). Etching using Taurus-Topography is specified with the **TOPOGRAP** parameter on the **ETCH** statement (see “Etching with Taurus-Topography” on page 3-101).

Modeling Polycrystalline Materials

The behavior of dopants in polycrystalline materials is strongly influenced by the boundaries between crystalline grains. Dopant atoms tend to segregate from the interior of a grain to the boundaries, which provide paths for rapid diffusion. The rate of segregation depends on the rate of grain growth, while the number of diffusion paths along the boundaries depends on the grain size. In addition, the boundaries of the polycrystalline material act like grain boundaries, providing sites for electrically inactive dopant atoms and paths for diffusion.

The diffusion equations for the grain-interior and grain-boundary components of the doping profile are solved separately. The equations are coupled by terms describing the segregation between the grain interiors and grain boundaries. To determine the rate of segregation and the density of grain-boundary diffusion paths, you also solve for the growth in grain size during high-temperature processing. The boundaries of the polycrystalline region are included as explicit grain boundaries in the diffusion and segregation equations.

The poly model has been implemented in TSUPREM-4 in collaboration with GEC Plessey Semiconductors (GPS) in the UK. The physical model has been developed at GPS and GEC-Marconi Materials Technology, Caswell, UK within the collaborative European ESPRIT project STORM. The key contributors to this work were S. K. Jones, C. Hill, and A. G. O'Neill [76] to [79]. Although the model was developed and optimized for polysilicon, it can be applied to other polycrystalline materials (e.g., silicides).

Diffusion

Redistribution of dopants in polycrystalline materials occurs by the parallel diffusion of dopants through the interiors of grains and along grain boundaries.

Diffusion in Grain Interiors

In the grain interiors diffusion of the active dopant is given by

$$\frac{\partial c_g}{\partial t} = -\vec{\nabla} \cdot \left[-D_g \left(\vec{\nabla} c_g - z_s c_g \frac{q\vec{E}_g}{kT} \right) \right] - G \quad \text{Equation 2-323}$$

where c_g is the active concentration in the grain interior and the other symbols are described in [“Diffusion of Impurities” on page 2-17](#). The diffusivity D_g and electric field \vec{E}_g in the grain interior are calculated from the electron concentration n_g , which is in turn calculated from the doping concentrations c_g . G accounts for the segregation of dopant to grain boundaries, as described in [“Segregation Between Grain Interior and Boundaries” on page 2-114](#). (The calculation of the active concentration is described in [“Activation of Impurities” on page 2-24](#), and the calculation of the electron concentration is described in [“Diffusion of Impurities” on page 2-17](#).)

Grain Boundary Structure

Diffusion along grain boundaries is described in terms of the dopant concentration *per unit area* of grain boundary c_{gb} and the average area of grain boundaries per unit volume

$$\rho' = \rho + \delta_{if} \quad \text{Equation 2-324}$$

where ρ is the average area of grain boundaries per unit volume in the bulk of the poly layer and δ_{if} accounts for the dopant at interfaces between poly and other materials (or ambient). ρ is inversely proportional to the average grain size L_g

$$\rho = \frac{\mathbf{G.DENS}}{L_g} \quad \text{Equation 2-325}$$

where **G.DENS** is a geometrical factor specified on the **MATERIAL** statement. δ_{if} is a function of position defined by the fact that its integral over any area is equal to the length L_{if} of the polysilicon interface passing through that area:

$$\int \delta_{if} dA = L_{if} \quad \text{Equation 2-326}$$

The concentration of dopants in the grain boundaries per unit volume of material is then given by

$$w_{gb} = \rho' c_{gb} \quad \text{Equation 2-327}$$

Diffusion Along Grain Boundaries

The diffusion of dopant in the grain boundaries is given by

$$\frac{\partial w_{gb}}{\partial t} = -\vec{\nabla} \cdot \left[-FD_{gb} \left(\vec{\nabla} c_{gb} - z_s c_{gb} \frac{q\vec{E}_{gb}}{kT} \right) \right] + G \quad \text{Equation 2-328}$$

The diffusivity D_{gb} and electric field \vec{E}_{gb} along the grain boundaries are calculated from the electron concentration n_{gb} ; n_{gb} is calculated as in [Equation 2-43](#) except that the net donor and acceptor concentrations are calculated from c_{gb}/K , the equilibrium dopant concentrations in the grain interior near the grain boundary.

Anisotropic Diffusion

F is a tensor that describes the diffusion paths available to dopant in the grain boundaries. It is composed of two parts: $F = F_b + (1 - F_{bu})F_{if}$. F_b describes the available paths within the bulk of the poly layer. For a horizontal poly layer, it is given by

$$F_b = \text{diag} \left(\frac{\mathbf{F11}}{L_g}, \frac{\mathbf{F22}}{L_g} \right) \quad \text{Equation 2-329}$$

where **F11** and **F22** are parameters on the **MATERIAL** statement. Because of the columnar grain structure, **F22** is larger than **F11**, which implies that diffusion through the layer is faster than diffusion parallel to the layer. F_{if} describes the

available paths for diffusion along material interfaces. In the vicinity of a horizontal interface it has the value

$$F_{if} = \text{diag}(\delta_{if}, 0) \quad \text{Equation 2-330}$$

For the interface between polysilicon and silicon, the phenomenon of interfacial break-up accompanied by epitaxial realignment can occur, as described in “[Interface Oxide Break-up and Epitaxial Regrowth](#)” on page 2-117. F_{bu} is the fraction of the polysilicon/silicon interface that has broken up. For layers or interfaces that are not horizontal, F_b and F_{if} are rotated by the angle of the layer or interface, respectively, with respect to the x axis.

Segregation Between Grain Interior and Boundaries

When dopant is initially introduced into a polycrystalline material, some of the dopant occupies sites in the interior of a grain and some occupies sites on a grain boundary. The initial segregation of dopant is given by

$$c_{gb} = \frac{\mathbf{Q.SITES}}{\mathbf{CG.MAX}} \mathbf{GSEG.INI} c_g \quad \text{Equation 2-331}$$

Q.SITES, **CG.MAX**, and **GSEG.INI** are parameters on the **IMPURITY** statement; they represent the density of available sites on grain boundaries and in the grain interiors and the initial segregation entropy, respectively. In the case of ion implantation c_g and c_{gb} describe the additional dopant introduced by the implantation; dopant that is present before the implantation is not redistributed.

Dopant atoms are free to move between sites in the interior of a grain and sites on the grain boundary during high-temperature processing. The rate of segregation is given by

$$G = (\rho q_b + (1 - F_{bu}) \delta_{if} q_{if}) \left(f_{gb} c_g - f_g \frac{c_{gb}}{K} \right) \quad \text{Equation 2-332}$$

The segregation coefficient K is given by [80]

$$K = \frac{\mathbf{Q.SITES}}{\mathbf{CG.MAX}} \mathbf{GSEG.0} \exp\left(\frac{\mathbf{GSEG.E}}{kT}\right) \quad \text{Equation 2-333}$$

where **GSEG.0** and **E.SEG** are parameters on the **IMPURITY** statement. The segregation velocities associated with the bulk of the poly region and the material interfaces are given by

$$q_b = \frac{1}{\mathbf{ALPHA}} \frac{\partial L_g}{\partial t} + \frac{4}{L_g} D_g \quad \text{Equation 2-334}$$

$$q_{if} = \mathbf{VELIF.0} \exp\left(-\frac{\mathbf{VELIF.E}}{kT}\right) \quad \text{Equation 2-335}$$

where **ALPHA** is a parameter on the **MATERIAL** statement, **VELIF.0** and **VELIF.E** are parameters on the **IMPURITY** statement. f_g and f_{gb} are the fractions of unfilled interior and boundary sites:

$$f_g = 1 - \frac{c_g}{\mathbf{CG.MAX}} \quad \text{Equation 2-336}$$

$$f_{gb} = 1 - \sum \frac{c_{gb}}{\mathbf{Q.SITES}} \quad \text{Equation 2-337}$$

where the sum is taken over all the dopant species present in the structure. F_{bu} is the fraction of the polysilicon/silicon interface that has broken up, as described in “[Interface Oxide Break-up and Epitaxial Regrowth](#)” on page 2-117.

When calculating the segregation between poly and another material, the total (active plus inactive) concentration in the grain interior is used for the concentration in poly in [Equation 2-80](#).

Grain Size Model

The grains in the polycrystalline material are assumed to be oriented as columns that extend through the wafer. The structure is characterized by L_g , the average grain size in the lateral direction (i.e., in the plane of the layer), and $\vec{\xi}$, a vector describing the orientation of the columnar grains. The grain size can be examined by specifying the *lgrain* solution variable on the **SELECT** statement:

SELECT Z=LGRAIN

This returns the average grain size in microns or a negative value at nodes where the material is amorphous. (Note that the grain size is reported in centimeters by Michelangelo and Taurus Visual.)

Initial Grain Size

The initial grain size is determined by the temperature of the poly deposition process and the value of the **GSZ.LIN** parameter on the **DEPOSITION** statement. If **GSZ.LIN** is false, the grain size in the layer is constant [81]:

$$L_g = \begin{cases} \max(\mathbf{FRAC.TA} \times t_a, \mathbf{MIN.GRAI}) & T_c \leq \mathbf{TEMP.BRE} \\ \mathbf{GRASZ.0} \exp\left(\frac{-\mathbf{GRASZ.E}}{kT}\right) \mathbf{THICKNES} & T_c > \mathbf{TEMP.BRE} \end{cases} \quad \text{Equation 2-338}$$

where **FRAC.TA**, **MIN.GRAI**, **GRASZ.0**, **GRASZ.E**, and **TEMP.BRE** are parameters on the **MATERIAL** statement, T_c is the deposition temperature (specified on the **DEPOSITION** statement) in °C, and T is the deposition temperature in Kelvins. **THICKNES** is the thickness of the deposited layer, as specified on the **DEPOSITION** statement. t_a is the thickness of the amorphous silicon layer produced by low-temperature deposition. If **GSZ.LIN** is true, the grain size increases linearly from the bottom of the layer to the top:

Equation 2-339

$$L_g = \begin{cases} \max(\mathbf{FRAC.TA} \times t_a, \mathbf{MIN.GRAI}) & T_c \leq \mathbf{TEMP.BRE} \\ \mathbf{MIN.GRAI} + 2\mathbf{GRASZ.0} \exp\left(\frac{-\mathbf{GRASZ.E}}{kT}\right) z & T_c > \mathbf{TEMP.BRE} \end{cases}$$

where z is the distance from the bottom of the layer.

For high-temperature depositions, grain size depends on the thickness specified on the **DEPOSITION** statement. Dividing a deposition into multiple smaller depositions produces different results for the grain size. For low-temperature depositions, the material is assumed to be amorphous (a negative grain size is reported in printing or plotting). The initial grain size (above) is calculated from the actual thickness t_a of the amorphous layer at the beginning of the next diffusion step. Thus, successive amorphous depositions are merged in computing the grain size.

Grain Growth

The growth of the grains during high-temperature processing is given by [81]

Equation 2-340

$$\frac{\partial L_g}{\partial t} = \frac{1}{L_g} \mathbf{GEOM} \times \mathbf{GAMMA.0} \exp\left(\frac{-\mathbf{GAMMA.E}}{kT}\right) \times D_{Si} \times \frac{E_{gb}}{kT} \times F_{seg} \times G_{EA}$$

where **GEOM**, **GAMMA.0**, and **GAMMA.E** are parameters on the **MATERIAL** statement, D_{Si} is the silicon self-diffusivity in the vicinity of a grain boundary, E_{gb} is the surface energy per atom associated with the grain boundary, F_{seg} models the segregation drag effect, and G_{EA} models epitaxial regrowth of the poly layer (see “[Interface Oxide Break-up and Epitaxial Regrowth](#)” on page 2-117).

Concentration Dependence

The silicon self-diffusivity is given by

$$D_{Si} = D_{Si}^x + D_{Si}^+ \left(\frac{n_i}{n_g}\right) + D_{Si}^- \left(\frac{n_g}{n_i}\right) + D_{Si}^{\bar{-}} \left(\frac{n_g}{n_i}\right)^2 \tag{Equation 2-341}$$

where n_i is the intrinsic carrier concentration at the processing temperature and n_g is the electron concentration in the grain interior. The components of D_{Si} are given by

$$D_{Si}^x = \mathbf{DSIX.0} \exp\left(-\frac{\mathbf{DSIX.E}}{kT}\right) \tag{Equation 2-342}$$

$$D_{Si}^+ = \mathbf{DSIP.0} \exp\left(-\frac{\mathbf{DSIP.E}}{kT}\right) \tag{Equation 2-343}$$

$$D_{Si}^{\bar{-}} = \mathbf{DSIM.0} \exp\left(-\frac{\mathbf{DSIM.E}}{kT}\right) \tag{Equation 2-344}$$

$$D_{Si}^- = \text{DSIMM.0} \exp\left(-\frac{\text{DSIMM.E}}{kT}\right) \quad \text{Equation 2-345}$$

where **DSIX.0**, **DSIX.E**, **DSIP.0**, **DSIP.E**, **DSIM.0**, **DSIM.E**, **DSIMM.0**, and **DSIMM.E** are parameters on the **MATERIAL** statement.

Grain Surface Energy

The grain boundary energy is given by [81] to [83]

$$E_{gb} = \frac{\text{GBE.0}}{1 + \text{GBE.H}f_n} + \text{GBE.1} \frac{L_g}{t_{poly}} \quad \text{Equation 2-346}$$

$$f_n = \begin{cases} \frac{L_g}{2t_{poly} - L_g} & L_g < t_{poly} \\ \frac{L_g}{t_{poly}} & L_g \geq t_{poly} \end{cases} \quad \text{Equation 2-347}$$

where **GBE.0**, **GBE.H**, and **GBE.1** are parameters on the **MATERIAL** statement, and t_{poly} is the thickness of the polycrystalline layer. Note that **GBE.0** and **GBE.1** are in units of electron volts per atom; to convert from values given in eV/ μm^2 you must multiply by $6.25 \times 10^{-8} \mu\text{m}^2/\text{atom}$.

Segregation Drag

The segregation drag effect reduces the grain growth rate [84]:

$$F_{seg} = \left(1 + \sum \frac{c_{gb}}{\text{Q.SITES}}\right)^{-\text{NSEG}} \quad \text{Equation 2-348}$$

Q.SITES is a parameter on the **IMPURITY** statement, **NSEG** is a parameter on the **MATERIAL** statement, and the summation is taken over the impurities present in the structure.

Interface Oxide Break-up and Epitaxial Regrowth

A thin interfacial oxide layer is typically present between a deposited polysilicon layer and any underlying single-crystal silicon. This interfacial oxide presents a barrier to epitaxial realignment of the poly layer. With sufficient high-temperature processing, the oxide layer breaks up into a discrete set of small spheres, allowing epitaxial regrowth of the poly to proceed.

Oxide Break-Up

The oxide break-up is modeled by the formation of voids in the interfacial oxide layer [85] to [88]. The radius of the voids R_{void} increases as

$$\frac{dR_{void}}{dt} = \frac{\beta}{t_{ox}^3} \times \exp\left(-\frac{E_{bu}}{kT}\right) \quad \text{Equation 2-349}$$

where β is a constant, t_{ox} is the initial oxide thickness, and E_{bu} is the activation energy of the break-up process. R_{void} is initialized to zero whenever poly is

deposited on exposed silicon. The fraction of the interface that is broken up is given by

$$F_{bu} = 1 - \exp(-\pi N_{EA} R_{void}^2) \quad \text{Equation 2-350}$$

where N_{EA} is the areal density of the voids. The parameters for the model are specified in terms of a characteristic break-up time for the thinnest (5Å) interfacial oxide layers

$$t_{bu} \equiv \frac{5\text{\AA}^3}{\sqrt{\pi N_{EA}} \beta} \times \exp\left(\frac{E_{bu}}{kT}\right) = \mathbf{TBU.0} \cdot \exp\left(-\frac{\mathbf{TBU.E}}{kT}\right) \quad \text{Equation 2-351}$$

where **TBU.0** and **TBU.E** are parameters on the **MATERIAL** statement. [Equation 2-349](#) can be then written

$$\frac{dR_{void}}{dt} = \frac{1}{t_{bu}} \left(\frac{5\text{\AA}}{t_{ox}}\right)^3 \frac{1}{\sqrt{\pi N_{EA}}} \quad \text{Equation 2-352}$$

In the present implementation, you assume that all poly/single-crystal interfaces share a common oxide thickness given by

$$t_{ox} = \mathbf{TOXIDE} \quad \text{Equation 2-353}$$

where **TOXIDE** is a parameter on the **MATERIAL** statement.

Epitaxial Regrowth

Epitaxial regrowth is modeled by increasing the poly grain size to a value much larger than the thickness of the poly layer. This grain growth is described by G_{EA} in [Equation 2-330](#) for the grain size:

$$G_{EA} = D_{L_g} \nabla^2 L_g + F_{bu} v_{EA} \delta_{if} \quad \text{Equation 2-354}$$

The first term models the propagation of the regrowth through the poly layer, while the second term (which is nonzero only at the silicon/polysilicon interface) serves as a driving force for epitaxial regrowth from the interface. Parameters for this model are given by

$$D_{L_g} = \mathbf{DLGX.0} \cdot \exp\left(-\frac{\mathbf{DLGX.E}}{kT}\right) \quad \text{Equation 2-355}$$

$$v_{EA} = \mathbf{EAVEL.0} \cdot \exp\left(-\frac{\mathbf{EAVEL.E}}{kT}\right) \quad \text{Equation 2-356}$$

where **DLGX.0**, **DLGX.E**, **EAVEL.0**, and **EAVEL.E** are parameters on the **MATERIAL** statement.

Using the Polycrystalline Model

For each dopant in a polycrystalline material there are two solution values: one for the total doping concentration at each node in the structure and one for the concentration in grain boundaries. The total concentration is accessed by the name of the impurity, e.g.

```
SELECT Z=LOG10(BORON)
```

The grain-boundary component is accessed with the *gb()* function on the **SELECT** statement:

```
SELECT Z=LOG10(GB(BORON))
```

The grain-boundary component includes the dopant stored at the boundary of the polycrystalline material. Although this dopant is confined to the boundary, it is reported as if it were evenly distributed over the areas associated with the nodes on the boundary. Thus, the concentration at the boundary of the polycrystalline region is discontinuous and depends on the grid spacing at the boundary. While it is not possible to resolve the profile at the boundary, care has been taken that the total dose at and near the boundary is correct.

The grain-interior component can be computed as the difference between the total concentration and the grain-boundary concentration:

```
SELECT Z=LOG10(BORON-GB(BORON))
```

The grain-boundary component is stored as a separate impurity. The name of the grain-boundary impurity is created from the name of the base impurity by prepending **GB_** to the name. Thus, you set the diffusivities of the grain-boundary component of boron with

```
IMPURITY IMP=GB_BORON MAT=POLY DIX.0= . . .
```

The grain size can be examined by specifying the *lgrain* solution variable on the **SELECT** statement:

```
SELECT Z=LGRAIN
```

This returns the average grain size in microns; a negative value is returned for nodes where the material is amorphous. (Note that the grain size is reported in centimeters by Michelangelo and Taurus Visual.)

The polycrystalline models address the need to model polycrystalline silicon, but the implementation allows them to be used for other polycrystalline materials. The polycrystalline model is enabled or disabled for a material by the **POLYCRYS** parameter on the **MATERIAL** statement. When the polycrystalline models are disabled with **^POLYCRYS**, the impurity concentrations in the grain boundaries are set to zero and the corresponding equations are not solved. Thus, the diffusivity reduces to that of the grain interiors. To compensate for the lack of grain boundary solutions, the grain interior diffusivity is multiplied by **FGB** in materials for which the polycrystalline models are disabled. **FGB** is a parameter on the **IMPURITY** statement.

Electrical Calculations

TSUPREM-4 calculates a limited set of electrical characteristics along vertical cut-lines through a simulation structure. This is accomplished by solving the one-dimensional Poisson's equation for specified bias conditions. The electrical information consists of the threshold voltage, the low-frequency, high-frequency, and deep-depletion MOS capacitances, spreading resistance profile and sheet resistances for all diffused regions in all semiconductor layers within the structure. This information can be saved in a file for later plotting by using the **OUT.FILE** parameter in the **ELECTRICAL** statement (see [page 3-174](#)).

TSUPREM-4 extends the structure automatically if the bias voltage expands the depletion region beyond the bottom of a simulation structure. Also, the grid for simulation may not be adequate for solving Poisson's equation; the grid is automatically extended if necessary. The extended structure and the added grid are temporary and are used only for the electrical extraction.

Automatic Regrid

The **E.REGRID** in the **METHOD** statement specifies the automatic regrid for electrical extraction. The automatic regrid follows the two steps.

1. The first step regrids the region in the estimated maximum depth of depletion region. The minimum and maximum grid spacing are determined by the incremental depletion depth calculated for given bias conditions.
2. The second step generates more dense grids near the surface. This regrid results in the accurate calculation specially for the quantum effect.

Placement of grid near the surface is controlled by the thickness of the regrid region (**E.TSURF**), the first grid spacing at the surface (**E.DSURF**) and the incremental ratio of grid spacing (**E.RSURF**) in the **METHOD** statement.

Poisson's Equation

The form of Poisson's equation solved in semiconductor and insulator regions is

$$\frac{\partial}{\partial x} \left(\mathbf{EPSILON} \epsilon_0 \frac{\partial \Psi}{\partial x} \right) = \begin{cases} -q(p - n + N_d^+ - N_a^-) & \text{semiconductor} \\ 0 & \text{insulator} \end{cases} \quad \text{Equation 2-357}$$

where ϵ_0 is the dielectric constant in vacuum, Ψ is the potential, p is the hole concentration, n is the electron concentration, and N_d^+ and N_a^- are the sums of the ionized electrically active donor and acceptor impurity concentrations, respectively. The **EPSILON** parameter is the relative dielectric constant specified in the **MATERIAL** statement (see [page 3-228](#)). The potential in a semiconductor region is defined as the potential of the edge of the conduction band.

Boltzmann and Fermi-Dirac Statistics

Either Boltzmann or Fermi-Dirac statistics may be used to represent the electron and hole concentrations in semiconductor regions. The type of statistics used in a semiconductor material is specified with the **BOLTZMAN** parameter in the **MATERIAL** statement. If Boltzmann statistics are used, the electron and hole concentrations are given by

$$n = \mathbf{N.CONDUC} \left(\frac{T}{300} \right)^{\mathbf{NC.F}} \exp \left[\frac{q(\Psi - \phi_n)}{kT} \right] \quad \text{Equation 2-358}$$

$$p = \mathbf{N.VALENC} \left(\frac{T}{300} \right)^{\mathbf{NV.F}} \exp \left[\frac{q(\phi_p - \Psi) - E_g}{kT} \right] \quad \text{Equation 2-359}$$

$$E_g = \mathbf{BANDGAP} + \mathbf{EGALPH} \cdot \left(\frac{300^2}{300 + \mathbf{EGBETA}} - \frac{T^2}{T + \mathbf{EGBETA}} \right) \quad \text{Equation 2-360}$$

where:

- ϕ_n is the electron quasi-Fermi potential
- ϕ_p is the hole quasi-Fermi potential

If Fermi-Dirac statistics are used, the electron and hole concentrations are given by

$$n = \mathbf{N.CONDUC} \left(\frac{T}{300} \right)^{\mathbf{NC.F}} F_{1/2} \left[\frac{q(\Psi - \phi_n)}{kT} \right] \quad \text{Equation 2-361}$$

$$p = \mathbf{N.VALENC} \left(\frac{T}{300} \right)^{\mathbf{NV.F}} F_{1/2} \left[\frac{q(\phi_p - \Psi) - E_g}{kT} \right] \quad \text{Equation 2-362}$$

where:

- $F_{1/2}$ is the Fermi-Dirac integral of one-half order
- The parameters **N.CONDUC**, **N.VALENC**, **NC.F**, **NV.F**, **BANDGAP**, **EGALPH**, and **EGBETA** are specified in the **MATERIAL** statement
- The temperature T used for the device simulation is specified with the **TEMPERAT** parameter in the **ELECTRICAL** statement

Ionization of Impurities

Either complete or incomplete ionization of impurities may be used to represent the ionized donor and acceptor impurity concentrations in semiconductor regions. The type of impurity ionization used in a semiconductor material is specified with the **IONIZATI** parameter in the **MATERIAL** statement. If complete ionization is used, the ionized donor and acceptor impurity concentrations are given by

$$N_d^+ = N_d \quad \text{Equation 2-363}$$

$$N_a^- = N_a \quad \text{Equation 2-364}$$

If incomplete ionization (**^IONIZATI**) is used, the ionized donor and acceptor impurity concentrations are given by [90]

$$N_d^+ = \frac{N_d}{1 + \mathbf{G.DONOR} \exp\left[\frac{q(\Psi - \phi_n) + \mathbf{E.DONOR}}{kT}\right]} \quad \text{Equation 2-365}$$

$$N_a^- = \frac{N_a}{1 + \mathbf{G.ACCEP}^{-1} \exp\left[\frac{q(\phi - \Psi) - \mathbf{BANDGAP} + \mathbf{E.ACCEP}}{kT}\right]} \quad \text{Equation 2-366}$$

where N_d and N_a are the sums of the electrically active donor and acceptor impurity concentrations, respectively. The parameters **G.DONOR**, **E.DONOR**, **G.ACCEP**, and **E.ACCEP** are specified in the **MATERIAL** statement.

The potential in a conductor region is constant with a value given by

$$\Psi = V_c - \mathbf{WORKFUNC} + \chi \quad \text{Equation 2-367}$$

where V_c is the conductor bias and the **WORKFUNC** parameter is specified in the **MATERIAL** statement. If any semiconductor regions are present in the simulation structure, χ is the electron affinity for the bottommost semiconductor layer in the structure. The value of χ is specified with the **AFFINITY** parameter in the **MATERIAL** statement. If no semiconductor regions are present, χ is set to zero.

Solution Methods

Poisson's equation is solved numerically using a three-point finite difference approach. Newton's method is used to iteratively solve the nonlinear equation by linearizing the equation at each iteration and using Gaussian elimination to solve the resulting tridiagonal system. The iteration is normally terminated when the relative change in the potential between successive iterations is less than the allowed error at each node in the structure. The iteration may also be terminated when the maximum number of allowed Newton iterations has been performed. A minimum number of required iterations are performed before the iteration is terminated. The allowed error, maximum number of iterations, and minimum number of iterations may be specified with the **E.RELERR**, **E.ITMAX**, and **E.ITMIN** parameters, respectively, in the **METHOD** statement.

Carrier Mobility

The mobilities of electrons and holes depend on the impurity concentration, electric field, and temperature. The mobility is given by [92] and [93]

$$\mu_n = \mathbf{GSURFN} \frac{\mu_{0n}}{\sqrt{1 + \left(\frac{|E|}{\mathbf{ECN} \cdot \mathbf{MU}}\right)^2}} \quad \text{Equation 2-368}$$

$$\mu_p = \mathbf{GSURFP} \frac{\mu_{0p}}{\sqrt{1 + \left(\frac{|E|}{\mathbf{ECP} \cdot \mathbf{MU}}\right)^2}} \quad \text{Equation 2-369}$$

where μ_0 is the low-field mobility, E is the electric field, and $\mathbf{ECN} \cdot \mathbf{MU}$ and $\mathbf{ECP} \cdot \mathbf{MU}$ are the critical electric fields. The mobility used for calculating current along semiconductor-insulator interfaces may be reduced by specifying degradation factors \mathbf{GSURFN} and \mathbf{GSURFP} which multiply the low field mobility. It is important to note that the factors \mathbf{GSURFN} and \mathbf{GSURFP} are only applied at interfaces between semiconductor and insulator. Everywhere else, these factors should be considered to have values of unity. $\mathbf{ECN} \cdot \mathbf{MU}$, $\mathbf{ECP} \cdot \mathbf{MU}$, \mathbf{GSURFN} , and \mathbf{GSURFP} can be specified on the **MOBILITY** statement (see page 3-259).

TSUPREM-4 provides three alternatives—tabular form, the Arora model, and the Caughey model—for specifying low field electron and hole mobilities dependent on impurity concentration and temperature.

Tabular Form

The low-field mobilities for the tabular form depend on the electrically active impurity concentration and device temperature. The mobilities are represented by the table of values specified with the **MOBILITY** statement. Two-dimensional interpolation is used to obtain values of low-field mobility for impurity concentration and temperature between the values in a table. The concentration is assumed to vary exponentially and the temperature and mobility are assumed to vary linearly. TSUPREM-4 uses this method by default. The mobility using this table can be selected with the **MOB . TABL** on the **METHOD** statement. The table values may be modified using the **MOBILITY** statement.



Note:

The default mobility tables are the same as those used in Medici. They differ slightly from the tables used for calculating initial impurity concentrations from resistivity.

Arora Mobility Model

Also available is the analytic model based on work by Arora, et al., [94] which takes into account total impurity concentration and temperature. The mobility expressions are

$$\mu_{0n} = \text{MUN1} \left(\frac{T}{300} \right)^{\text{EXN1}} + \frac{\text{MUN2} \left(\frac{T}{300} \right)^{\text{EXN2}}}{1 + \left[\frac{N_{total}}{\text{CN} \left(\frac{T}{300} \right)^{\text{EXN3}}} \right]^{\alpha_n}} \quad \text{Equation 2-370}$$

$$\mu_{0p} = \text{MUP1} \left(\frac{T}{300} \right)^{\text{EXP1}} + \frac{\text{MUP2} \left(\frac{T}{300} \right)^{\text{EXP2}}}{1 + \left[\frac{N_{total}}{\text{CP} \left(\frac{T}{300} \right)^{\text{EXP3}}} \right]^{\alpha_p}} \quad \text{Equation 2-371}$$

where

$$\alpha_n = \text{AN} \left(\frac{T}{300} \right)^{\text{EXN4}} \quad \text{Equation 2-372}$$

$$\alpha_p = \text{AP} \left(\frac{T}{300} \right)^{\text{EXP4}} \quad \text{Equation 2-373}$$

where N_{total} is the local total impurity concentration in atoms/cm³, and T is the temperature in Kelvins. The Arora mobility model is selected with the **MOB.AROR** parameter on the **METHOD** statement. The default parameter values used by the Arora model may be modified with the **MOBILITY** statement.

Caughey Mobility Model

The Caughey mobility model [95] is given by the expressions

$$\mu_{0n} = \text{MUN.MIN} + \frac{\text{MUN.MAX} \left(\frac{T}{300} \right)^{\text{NUN}} - \text{MUN.MIN}}{1 + \left(\frac{T}{300} \right)^{\text{XIN}} \left(\frac{N_{total}}{\text{NREFN}} \right)^{\text{ALPHAN}}} \quad \text{Equation 2-374}$$

$$\mu_{0p} = \text{MUP.MIN} + \frac{\text{MUP.MAX} \left(\frac{T}{300} \right)^{\text{NUP}} - \text{MUP.MIN}}{1 + \left(\frac{T}{300} \right)^{\text{XIP}} \left(\frac{N_{total}}{\text{NREFP}} \right)^{\text{ALPHAP}}} \quad \text{Equation 2-375}$$

where N_{total} is the local total impurity concentration in atoms/cm³, and T is the temperature in Kelvins. The Caughey mobility model is selected with the **MOB.CAUG** parameter on the **METHOD** statement. The default parameter values used by the Caughey model can be modified with the **MOBILITY** statement.

Quantum Mechanical Model for MOSFET

The high electric fields caused by thin gate oxides and high substrate doping quantize electron motion perpendicular to semiconductor/insulator interfaces in MOSFET inversion layers. This affects threshold voltages, inversion layer areal charge densities, and capacitance-voltage characteristics. A robust solution for the quantum effect can be achieved by solving the Schrödinger equation coupled with Poisson's equation. However, it takes considerable CPU time to solve the Schrödinger equation. In TSUPREM-4, the quantum effect is taken into account in an approximate manner by using a method suggested by van Dort, et al [96]. This approach provides a reasonable solution in spite of very short CPU time.

The approximate bandgap-widening effect in the inversion layer due to quantization is:

$$\Delta E_g = \frac{13}{9} \text{QM.BETA} \left(\frac{\epsilon_{Si}}{4kT} \right)^{1/3} |E_n(0)|^{2/3} \quad \text{Equation 2-376}$$

where $E_n(0)$ is the electric field at the interface.

The bandgap-widening effect causes a decrease in the intrinsic carrier concentration in the inversion layer.

$$n_{i,QM} = n_{i,CL} \exp\left(\frac{-\Delta E_g}{2kT}\right) \quad \text{Equation 2-377}$$

in which $n_{i,CL}$ is the classical model for the intrinsic carrier concentration. However, this model should be applied to the region where the electrons are confined to a layer near the interface. The smoothing function $s(a)$ is used to describe the transition:

$$s(a) = 2 \frac{\exp(-a^2)}{(1 + \exp(-2a^2))} \quad \text{Equation 2-378}$$

in which $a = y/\text{QM.YCRIT}$, where y is the distance from the surface of an inversion layer. Thus, the intrinsic carrier concentration becomes:

$$n_i = n_{i,CL}(1 - s(a)) + s(a)n_{i,QM} \quad \text{Equation 2-379}$$

QM.BETA and **QM.YCRIT** are parameters on the **MATERIAL** statement. The quantum mechanical model is used when **QM** is specified on the **ELECTRICAL** statement.

Capacitance Calculation

In order to achieve the accurate capacitance dependent on the frequency and waveform of input small signal, the transient equation must be solved by device simulators.

DC Method

The capacitance calculation in TSUPREM-4 is based on the DC method which extracts the charge quantities at different DC biases. The voltage difference is specified in the **METHOD** statement. If **E.USEAVC** is false (default), the relative ratio **E.RVCAP** to the DC bias step is used.

$$C = \frac{Q(V + \mathbf{E.RVCAP} \cdot \mathbf{VSTEP}) - Q(V - \mathbf{E.RVCAP} \cdot \mathbf{VSTEP})}{2 \cdot \mathbf{E.RVCAP} \cdot \mathbf{VSTEP}} \quad \text{Equation 2-380}$$

Otherwise, the absolute voltage difference **E.AVCAP** is used.

$$C = \frac{Q(V + \mathbf{E.AVCAP}) - Q(V - \mathbf{E.AVCAP})}{2 \cdot \mathbf{E.AVCAP}} \quad \text{Equation 2-381}$$

If only one bias is given so that **VSTEP** is zero, [Equation 2-381](#) is used regardless of **E.USEAVC** status.

**MOS
Capacitances**

There are three kinds of measurement for MOS capacitance.

1. Slow DC sweep and low-frequency AC small signal
2. Fast DC sweep and high-frequency AC small signal
3. Slow DC sweep and high-frequency AC small signal

The parameters **LOW**, **DEEP**, and **HIGH** in the **ELECTRICAL** statement correspond to the above measurements in order. Since TSUPREM-4 does not permit AC analysis, the high-frequency effect is modeled by applying a bias not only to the majority carriers, but also to the minority carriers. This has no effect on the C-V characteristics up to the threshold level, since the charge storage mechanism in this region is due primarily to modulation of majority carriers at the silicon surface. However, above the threshold voltage, setting the majority Fermi-level to a value higher than the maximum applied gate bias, prevents the build-up of minority carriers at the surface of the silicon and results in the high-frequency effect. However, this approximation can cause inaccurate results when a lot of intrinsic carriers are generated at high temperature.

Extended Defects AAM

The Extended Defects Advanced Applications Module is an optional feature of TSUPREM-4. It includes models for dislocation loops and transient clustering of impurities during ion implantation and subsequent annealing. These models are described in the following sections.

Dislocation Loop Model

It has been observed that amorphizing implants give rise to dislocation loops at the edge of the amorphized region following annealing. The size of these loops has been observed to grow (presumably by absorption of interstitials) in oxidizing

ambients and shrink (by emission of interstitials) in inert ambients. The observed reduction in the amount of oxidation-enhanced diffusion (OED) in underlying dopant distributions is consistent with the presumption of growth and shrinkage by absorption and emission of interstitials.

Dislocation loops can also form at the peak of the damage distribution following nonamorphizing implants.

Equations for Dislocation Loop Model

The Extended Defects AAM in TSUPREM-4 implements the model of Huang and Dutton for the interaction between interstitials and dislocation loops [97]. The rate of absorption of interstitials by dislocation loops R_l is given by

$$R_l = 2\pi^2 K_L D_I \rho_v r (I - I_{loop}^*) \tag{Equation 2-382}$$

$$K_L = \mathbf{KLOOP.0} \exp\left(-\frac{\mathbf{KLOOP.E}}{kT}\right) \tag{Equation 2-383}$$

$$I_{loop}^* = I^* \exp\left(\frac{\gamma\Omega}{bkT}\right) \exp\left[\frac{\mu b\Omega}{4\pi r kT(1-\nu)} \ln\left(\frac{8r}{b}\right)\right] \tag{Equation 2-384}$$

where ρ_v is the volume density (number/cm³), r is the loop radius (in cm), and **KLOOP.0** and **KLOOP.E** are parameters on the **INTERSTITIAL** statement. The default value for **KLOOP.0** (29.8047) is the value fit by Huang and Dutton [97]; the value of the activation energy in the expression for **KLOOP.E** (0.4) and the expression for I_{loop}^* are from Hu [98].

The other quantities in the equation above are:

- D_I , diffusivity of interstitials
- I , interstitial concentration
- k , Boltzmann's constant
- T , temperature (Kelvins)
- γ , internal energy associated with a stacking fault in silicon (70×10^{-5} N/cm)
- Ω , volume per silicon atom (2×10^{-23} cm³)
- b , magnitude of the Burger's vector for the dislocation loop (3.14×10^{-8} cm)
- μ , shear modulus of silicon (6.6406×10^6 N/cm²)
- ν , Poisson's ratio for silicon (0.28)

The evolution of the loop radius r is given by

$$\frac{dr}{dt} = \left(\frac{\pi}{N_o}\right) K_L D_I (I - I_{loop}^*) \tag{Equation 2-385}$$

where

- N_o is the {111} planar density of silicon (1.57×10^{15} /cm²)

- I_{loop}^* is the equilibrium concentration of interstitials in the presence of dislocation loops

I_{loop}^* decreases from about 10 times I^* for 100Å loops down to one or two times I^* for loops of 1000Å or more. Loops of radius **RLMIN** or less dissolve quickly during diffusion in an inert ambient. To save simulation time, such loops are discarded. **RLMIN** is specified on the **INTERSTITIAL** statement; it initially defaults to 100×10^{-8} cm (100Å).

The initial loop radius is specified by the **L.RADIUS** parameter on the **IMPLANT** statement. The density of loops can either be set to a fixed value (specified by the **L.DENS** parameter with the location derived from the **L.DMIN** and **L.DMAX** parameters) or the location and density can be specified indirectly with the **L.THRESH** and **L.FRAC** parameters.

Loop Density Specified by L.DENS

End-of-range dislocation loops are created following amorphizing ion implantation steps. The initial volume density ρ_v and radius r are specified by the **L.DENS** and **L.RADIUS** parameters, respectively, on the **IMPLANT** statement. Loops are produced in that portion of the structure where the interstitial concentration (due to implant damage, before recombination) is in the range

$$L.DMIN < I \leq L.DMAX \tag{Equation 2-386}$$

where **L.DMIN** and **L.DMAX** are parameters on the **IMPLANT** statement (default values of $10^{20}/\text{cm}^3$ and $1.15 \times 10^{22}/\text{cm}^3$). The concentration of interstitials corresponding to the edge of the amorphous region is from the work of Cerva and Hobler [4]. The concentration of interstitials is not reduced by the formation of end-of-range loops. Pre-existing dislocation loops in the region where $I > L.DMIN$ are presumed to be destroyed by the implant. Note that the loop density and radius are taken to be constant for a particular implant.

Loop Density Specified by L.THRESH

Dislocation loops can also be created near the peak of the damage profile produced by nonamorphizing implants. These loops are modeled by assuming that a specified fraction **L.FRAC** of the interstitials above some threshold **L.THRESH** are incorporated into dislocation loops of radius **L.RADIUS**, where **L.FRAC**, **L.THRESH**, and **L.RADIUS** are parameters on the **IMPLANT** statement. The loop density is given by

$$\rho_v = \frac{L.FRAC \cdot \max(I - L.THRESH, 0)}{\pi N_o r^2} \tag{Equation 2-387}$$

The concentration of interstitials is reduced by the number of interstitials incorporated into dislocation loops, **L.FRAC** $\max(I - L.THRESH, 0)$.

Evolution of Loops

Following the implant, the loop density remains constant, but the change in radius is a function of the interstitial concentration and thus depends on position. When the radius decreases to less than **RLMIN**, both the radius and density are set to

zero. The density can vary with position as a result of multiple implants. The density and radius can be selected for printing or plotting by specifying *dloop* or *rloop*, respectively, as values of the **Z** parameter on the **SELECT** statement. The loop radius is reported in centimeters.

Effects of Dislocation Loops

Dislocation loops affect the diffusion of impurities by absorbing or emitting interstitials. They tend to reduce the impact of transient-enhanced diffusion effects immediately after an implant, but can produce a long-term diffusivity enhancement as they dissolve; this effect is particularly important for high-dose subamorphizing boron implants. Similarly, they tend to reduce the immediate impact of oxidation on impurity diffusion while producing a smaller long-term enhancement.

Transient Clustering Model

The transient clustering model simulates the activation of dopant atoms following ion implantation. It does this by solving the transient equation for the active concentration C_a :

$$\frac{\partial(C - C_a)}{\partial t} = \frac{C_a - C_a^*}{\tau_a} \tag{Equation 2-388}$$

where C is the chemical concentration of the dopant and $C_a^* = \min(C_{as}, C_{ac})$ is the equilibrium active concentration defined in “[Activation of Impurities](#)” on page 2-24. The time constant for the activation of dopants τ_a is given by

$$\tau_a = \mathbf{T.ACT.0} \times \exp\left(\frac{-\mathbf{T.ACT.E}}{kT}\right) \tag{Equation 2-389}$$

where **T.ACT.0** and **T.ACT.E** are parameters on the **IMPURITY** statement.

After an implant, it is assumed that all implanted dopant atoms are inactive, i.e., ion implantation increases the value of C but does not change the value of C_a ; in amorphized regions, the implanted dopants are assumed to be inactive unless **^CL.INI.A** has been specified for the dopant and material on the **IMPURITY** statement. This assumption is modified by the constraint

$$C_a \geq \min(C_a^*, \mathbf{ACT.MIN} \times n_i) \tag{Equation 2-390}$$

where n_i is the intrinsic carrier concentration and **ACT.MIN** is a parameter on the **IMPURITY** statement. Thus activation to a level comparable to n_i occurs instantaneously, after which [Equation 2-389](#) takes over. The result is that transient-enhanced diffusion can occur in the tail of an implanted profile without significant diffusion near the peak.

The transient clustering model is activated by specifying the **ACT . TRAN** parameter on the **METHOD** statement; it is disabled by specifying the **ACT . EQUI** parameter.

References

- [1] C. P. Ho, S. E. Hansen, and P. M. Fahey. "SUPREM III—A Program for Integrated Circuit Process Modeling and Simulation," Technical Report No. SEL84-001, Stanford Electronics Laboratories, Stanford University, Stanford, California, July 1984.
- [2] G. Masetti, M. Severi, and S. Solmi. "Modeling of Carrier Mobility Against Carrier Concentration in Arsenic-, Phosphorus-, and Boron-Doped Silicon," *IEEE Trans. Electron Devices*, Vol. ED-30, No. 7, pp. 764-769, July 1983.
- [3] B. E. Deal. "Thermal Oxidation Kinetics of Silicon in Pyrogenic H₂O and 5% HCl/H₂O Mixtures," *J. Electrochem. Soc.*, Vol. 125, No. 4, pp. 576-579, April 1978.
- [4] R. B. Fair and P. N. Pappas. "Diffusion of Ion-Implanted B in High Concentration P- and As-Doped Silicon," *J. Electrochem. Soc.*, Vol. 122, No. 9, pp. 1241-4, Sept. 1975.
- [5] N. E. B. Cowern and D. J. Godfrey. "A Model for Coupled Dopant Diffusion in Silicon," *Fundamental Research on the Numerical Modelling of Semiconductor Devices and Processes (Papers from NUMOS I, the First International Workshop on the Numerical Modelling of Semiconductors, 11th - 12th December 1986, Los Angeles, USA)*, edited by J. J. H. Miller, Dublin, Ireland: Boole Press, 1987.
- [6] F. Wittel and S. T. Dunham. "Diffusion of Phosphorus in Arsenic and Boron Doped Silicon," *Appl. Phys. Lett.*, Vol. 66, p. 1415, 1995.
- [7] R. E. Bank, W. M. Coughran Jr., W. Fichtner, E. H. Grosse, D. J. Rose, and R. K. Smith. "Transient Simulation of Silicon Devices and Circuits," *IEEE Trans. Elec. Dev.*, Vol. ED-32, No. 10, pp. 1992-2007, Oct. 1985.
- [8] M. E. Law, H. Park, and P. Novell. "Theory of Dopant Diffusion Assuming Nondilute Concentrations of Dopant-Defect Pairs," *Appl. Phys. Lett.*, Vol. 59, No. 26, pp. 3488-3489, Dec. 1991.
- [9] N. E. B. Cowern. "General Model for Intrinsic Dopant Diffusion in Silicon Under Nonequilibrium Point-Defect Conditions," *J. Appl. Phys.*, Vol. 64, No. 9, pp. 4484-4490, Nov. 1988.
- [10] F. A. Trumbore. "Solid Solubilities of Impurity Elements in Germanium and Silicon," *Bell Sys. Tech. J.*, pp. 205-233, Jan. 1960.
- [11] Y. OH and D. E. Ward, "A Calibrated Model for Trapping of Implanted Dopants at Material Interface During Thermal Annealing," 1998 IEDM *Tech. Dig.*, pp. 509-512, 1998.
- [12] C. S. Rafferty, G. H. Gilmer, J. Jaraiz, D. Eaglesham, and H.-J. Gossmann, "Simulation of Cluster Evaporation and Transient Enhanced Diffusion in Silicon," *Appl. Phys. Lett.*, Vol. 68, No. 17, pp. 2395-7, 22 April 1996.

- [13] J. M. Poate, D. J. Eaglesham, G. H. Gilmer, H.-J. Gossmann, M. Jaraiz, C. S. Rafferty, and P. A. Stolk, "Ion Implantation and Transient Enhanced Diffusion," 1995 IEDM *Tech. Dig.*, pp. 77-80, 1995.
- [14] S. M. Hu. "On Interstitial and Vacancy Concentrations in Presence of Injection," *J. Appl. Phys.*, Vol. 57, p. 1069, 1985.
- [15] P. B. Griffin and J. D. Plummer. "Process Physics Determining 2-D Impurity Profiles in VLSI Devices," *IEDM Technical Digest*, pp. 522-525, 1986.
- [16] P. B. Griffin. "Physics and Modeling of Two-Dimensional Diffusion in SUPREM-IV," Doctoral dissertation, Stanford University, Palo Alto, California, 1989.
- [17] B. E. Deal and A. S. Grove. "General Relationship for the Thermal Oxidation of Silicon," *J. Appl. Phys.*, Vol. 36, p. 3770, 1965.
- [18] H. Z. Massoud. "Thermal Oxidation of Silicon in Dry Oxygen—Growth Kinetics and Charge Characterization in the Thin Regime," Technical Report, Stanford Electronics Laboratories, Stanford University, Stanford, California, June 1983.
- [19] D. W. Hess and B. E. Deal. "Kinetics of the Thermal Oxidation of Silicon in O₂/HCl Mixtures," *J. Electrochem. Soc.*, Vol. 124, No. 5, pp. 735-739, May 1977.
- [20] R. R. Razouk, L. N. Lie, and B. E. Deal. "Kinetics of High Pressure Oxidation of Silicon in Pyrogenic Steam," *J. Electrochem. Soc.*, Vol. 128, No. 10, pp. 2214-2220, Oct. 1981.
- [21] L. N. Lie, R. R. Razouk, and B. E. Deal. "High Pressure Oxidation of Silicon in Dry Oxygen," *J. Electrochem. Soc.*, Vol. 129, No. 12, pp. 2828-2834, Dec. 1982.
- [22] N. Guillemot, G. Pananakakis, and P. Chenevier. "A New Analytical Model of the "Bird's Beak,"" *IEEE Trans. Elec. Dev.*, Vol. ED-34, No. 5, pp. 1033-1038, May 1987.
- [23] C. P. Ho and J. D. Plummer. "Si/SiO₂ Interface Oxidation Kinetics: A Physical Model for the Influence of High Substrate Doping Levels—I. Theory," *J. Electrochem. Soc.*, Vol. 126, No. 9, pp. 1516-1522, Sept. 1979.
- [24] C. P. Ho and J. D. Plummer. "Si/SiO₂ Interface Oxidation Kinetics: A Physical Model for the Influence of High Substrate Doping Levels—II. Comparison with Experiment and Discussion," *J. Electrochem. Soc.*, Vol. 126, No. 9, pp. 1523-1530, Sept. 1979.
- [25] C. P. Ho, J. D. Plummer, and J. D. Meindl. "Thermal Oxidation of Heavily Phosphorus-Doped Silicon," *J. Electrochem. Soc.*, Vol. 125, No. 4, pp. 665-671, April 1978.
- [26] D. Chin, S. Y. Oh, S. M. Hu, R. W. Dutton, and J. L. Moll. "Two-Dimensional Oxidation," *IEEE Trans. Elec. Dev.*, Vol. ED-30, No. 7, pp. 744-749, July 1983.

- [27] C. S. Rafferty. "Stress Effects in Silicon Oxidation—Simulation and Experiments," Doctoral dissertation, Stanford University, Stanford, California, 1989.
- [28] V. Senez, P. Ferreira, and B. Baccus. "Two-Dimensional Simulation of Local Oxidation of Silicon: Calibrated Viscoelastic Flow Analysis," *IEEE Trans. Electron Dev.*, Vol. 43, No. 5, pp. 720–731, May 1996.
- [29] S. L. Zhang, R. Buchta, and M. Ostling. "A Study of Silicide Formation from LPCVD-Tungsten Films: Film Texture and Growth Kinetics," *J. Mater. Res.*, Vol. 6, pp. 1886-1891, Sept. 1991.
- [30] R. B. Fair, "Physical Models of Boron Diffusion in Ultrathin Gate Oxides," *J. Electrochem. Soc.*, Vol. 144, No. 2, pp. 708-717, Sept. 1997.
- [31] A. M. Agarwal and S. T. Dunham, "Consistent Quantitative Model for the Spatial Extent of Point Defect Interactions in Silicon," *J. Appl. Phys.*, Vol. 78, No. 9, pp. 5313-5319, Nov. 1995.
- [32] S. Nedelec, D. Mathiot, E. Andre, A. Straboni, and M. Gauneau. "Influence of Nitrogen Content on B Diffusion in SiO_2 ," *Silicon Nitride and Silicon Dioxide Thin Insulating Films - The Electrochemical Society Proceedings Series*, pp.295-304, Pennington, NJ 1994.
- [33] T. Aoyama, K. Suzuki, H. Tashiro, Y. Toda, T. Yamazaki, K. Takasaki, and T. Ito. "Effect of Fluorine on Boron Diffusion in Thin Silicon Dioxides and Oxynitride," *J. Appl. Phys.*, Vol. 77, No. 1, pp. 417-419, Jan. 1995.
- [34] Locker and C. D. Capiro, "Reaction Kinetics of Tungsten Thin Films on Silicon (100) Surfaces," *J. Appl. Phys.*, Vol. 44, No. 10, pp. 4366-4369, Oct. 1973.
- [35] E. Ma, B. S. Lim, M. A. Nicolet, N. S. Alvi, and A. H. Hamdi. *J. Electron. Mater.*, Vol. 17, p. 207, 1988.
- [36] C. L. Chu, K. C. Saraswat, and S. S. Wong. "Measurement of Lateral Dopant Diffusion in Thin Silicide Layers," *IEEE Trans. Elec. Dev.*, Vol. 39, No. 10, pp. 2333-2340, Oct. 1992.
- [37] V. Probst, H. Schaber, A. Mitwalsky, and H. Kabza. "WSi₂ and CoSi₂ as Diffusion Sources for Shallow-Junction Formation in Silicon," *J. Appl. Phys.*, Vol. 70, No. 2, pp. 708-719, July 1991.
- [38] M. Y. Tsai, F. M. d'Heurle, C. S. Petersson, and R. W. Johnson. "Properties of Tungsten Silicide Film on Polycrystalline Silicon," *J. Appl. Phys.*, Vol. 52, No. 8, pp. 5350-5355, Aug. 1981.
- [39] G. Giroult, A. Nouailhat, and M. Gauneau. "Study of a WSi₂/Polycrystalline Silicon/Monocrystalline Silicon Structure for a Complementary Metal-Oxide-Semiconductor for a Compatible Self-Aligned Bipolar Transistor Emitter," *J. Appl. Phys.*, Vol. 67, No. 1, pp. 515-523, Jan. 1990.
- [40] C. M. Comrie and R. T. Newman, "Dominant diffusing species during cobalt silicide formation," *J. Appl. Phys.*, Vol. 79, No. 1, pp. 153 - 156, 1996.
- [41] T. Barge, "Analysis of the diffusion controlled growth of cobalt silicides in bulk and thin film couples," *J. Mater. Res.*, Vol. 10, No. 5, pp. 1134 - 1145, 1995.

- [42] S. B. Herner, K. S. Jones, H.-J. Gossmann *et al.*, "The influence of TiSi₂ and CoSi₂ growth on Si native point defects: The role of the diffusing species," *Appl. Phys. Lett.*, Vol. 68, No. 20, pp. 2870 - 2872, 1996.
- [43] S. B. Herner, H.-J. Gossmann and R. T. Tung, "Native point defects in Si with a CoSi₂ film: Single crystal versus polycrystalline," *Appl. Phys. Lett.*, Vol. 72, No. 18, pp. 2289 - 2291, 1998.
- [44] R. Stadler, W. Wolf, R. Podloucky *et al.*, "Ab initio calculations of the cohesive, elastic, and dynamical properties of CoSi₂ by pseudopotential and all-electron techniques," *Phys. Rev. B*, Vol. 54, No. 3, pp. 1729 - 1734, 1996.
- [45] T. L. Crandle, W. B. Grabowski, and M. R. Kump. "Empirically and Physically Based Approaches to Ion Implant Modeling," *Proceedings of the NASECODE VI Short Course on Software Tools for Process, Device and Circuit Modeling*, Dublin, Ireland, pp. 32-44, 1989.
- [46] A. F. Tasch, H. Shin, C. Park, J. Alvis, and S. Novak. "An Improved Approach to Accurately Model Shallow B and BF₂ Implants in Silicon," *J. Electrochem. Soc.*, Vol. 136, No. 3, March 1989.
- [47] S. Morris, V. Ghante, L. M. Lam, S.-H. Yang, M. Morris, and A. F. Tasch, "An Accurate and Computationally Efficient Model for Phosphorus Implants into (100) Single-Crystal Silicon," *Proc. of the XIth International Conference on Ion Implantation Technology*, Austin, TX, June 16-21, 1996.
- [48] M. Simard-Normandin and C. Slaby. "Empirical Modeling of Low Energy Boron Implants in Silicon," *J. Electrochem. Soc.*, Vol. 132, No. 9, pp. 2218-2223, Sept. 1985.
- [49] C. Park, K. Klein, A. Tasch, R. Simonton, and S. Novak. "Comprehensive Modeling of Boron Ion Implantation for the ULSI Era," Extended Abstract Volume, TECHCON'90 (San Jose, CA), pp. 443-446, 1990.
- [50] J. F. Gibbons. *Handbook on Semiconductors*, Vol. 3, Chapter 10, edited by T. S. Moss and S. P. Keller, Amsterdam: North-Holland, 1980.
- [51] W. K. Hofker. "Implantation of Boron in Silicon," *Philips Res. Reports*, Suppl. No. 8, pp. 1-121, 1975.
- [52] R. Tielert. "Two-Dimensional Numerical Simulation of Impurity Redistribution in VLSI Processes," *IEEE Trans. Elec. Dev.*, Vol. ED-27, No. 8, pp. 1479-1483, Aug. 1980.
- [53] J. Amaratunga, K. Sabine, and A. G. R. Evans. "The Modeling of Ion Implantation in a Three-Layer Structure Using the Method of Dose Matching," *IEEE Trans. Elec. Dev.*, Vol. ED-32, No. 9, pp. 1889-1890, Sept. 1985.
- [54] S. Furukawa, H. Matsumura, and H. Ishiwara. "Theoretical Considerations on Lateral Spread of Implanted Ions," *Jap. J. Appl. Phys.*, Vol. 11, No. 2, pp. 134-142, Feb. 1972.
- [55] G. Hobler and S. Selberherr. "Two-Dimensional Modeling of Ion Implantation Induced Point Defects," *IEEE Trans. Computer-Aided Design*, Vol. 7, No. 2, pp. 174-180, 1988.

- [56] B. J. Mulvaney, W. B. Richardson, and T. L. Crandle. "PEPPER—A Process Simulator for VLSI," *IEEE Trans. Computer-Aided Design*, Vol. 8, No. 4, pp. 336-349, 1989.
- [57] H. Goldstein. *Classical Mechanics*, Reading, Massachusetts: Addison-Wesley, 1950.
- [58] J. F. Ziegler, J. P. Biersack, and U. Littmark. *The Stopping and Ranges of Ions in Matter*, Vol. 1, New York: Pergamon Press, 1985.
- [59] M. T. Robinson and I. M. Torrens. "Computer Simulation of Atomic Displacement Cascades in Solids in the Binary-Collision Approximation," *Phys. Rev.*, Vol. B9, p. 5008, 1974.
- [60] J. P. Biersack and L. G. Haggmark. "A Monte Carlo Computer Program for the Transport of Energetic Ions in Amorphous Targets," *Nucl. Instr. and Meth.*, Vol. 174, p. 257, 1980.
- [61] G. Hobler and S. Selberherr. "Efficient Two-Dimensional Simulation of Ion Implantation," Proceedings of NASECODE V, Trinity College, Dublin, p. 225, 1987.
- [62] J. Lindhard, M. Scharff, and H. E. Schiott. *K. Dan Vidensk. Selsk. Mat. Fys. Medd.*, Vol. 33, No. 1, 1963; J. Lindhard and M. Scharff, *Phys. Rev.*, Vol. 124, p. 128, 1961.
- [63] D. V. Morgan. *Channeling*, New York: Wiley, 1973.
- [64] R. G. Wilson. "Ion Channeling in GaAs: Be, Mg, Zn, and Cd, and Calculations of Electronic Stopping Powers," *J. Appl. Phys.*, Vol. 53, p. 5641, 1982.
- [65] R. G. Wilson. "Random and Channeled Implantation Profiles and Range Parameters for P and Al in Crystalline and Amorphized Si," *J. Appl. Phys.*, Vol. 59, p. 2797-2805, Oct. 1986.
- [66] R. G. Wilson and V. R. Deline. "Ion Channeling in GaAs: Si, S, Se, and Te," *Appl. Phys. Lett.*, Vol. 37, pp. 793-796, 1980.
- [67] *Ion Beam Handbook for Material Analysis*, New York: Academic, 1977.
- [68] T. Saito, H. Yamakawa, S. Komiyama, H. J. Kang, and R. Shimuzu. "Dynamic Simulation of Ion Implantation with Damage Processes Included," *Nucl. Instr. and Meth.*, Vol. B21, p. 456, 1987.
- [69] T. L. Crandle, W. B. Richardson, and B. J. Mulvaney. "A Kinetic Model for Anomalous Diffusion During Post-Implant Annealing," *Intl. Elec. Dev. Mtng., Tech. Digest*, p. 636, 1988.
- [70] W. P. Maszara and G. A. Rozgonyi. "Kinetics of Damage Production in Silicon During Self-Implantation," *J. Appl. Phys.*, Vol. 60, p. 2310, 1986.
- [71] T. L. Crandle and B. J. Mulvaney. "An Ion Implantation Model Incorporating Damage Calculations in Crystalline Targets," *IEEE Elec. Dev. Lett.*, Vol. 11, No. 1, 1990.

- [72] L. Pelaz, G. H. Gilmer, M. Jaraiz, S. B. Herner, H.-J. Gossmann, D. J. Eaglesham, G. Hobler, C. S. Rafferty, and J. Barbolla, "Modeling of the Ion Mass Effect on Transient Enhanced Diffusion: Deviation from the '+1' Model," *Appl. Phys. Lett.*, Vol. 73, No. 10, pp. 1421–1423, 7 Sep 1998.
- [73] D. Chidambarrao and R. Young, private correspondence.
- [74] S. Chakravarthi and S. T. Dunham, "Influence of Extended Defect Models on Prediction of Boron Transient Enhanced Diffusion," in *Silicon Front End Technology—Materials Processing and Modeling*, N. Cowern, P. Griffin, D. Jacobsen, P. Packan, and R. Webb, Eds. (Mat. Res. Soc. Proc. Vol. 532, Pittsburgh, PA, 1998).
- [75] G. Hobler and V. Moroz, "Simple Formulae for the Effective Plus-Factor for Transient Enhanced Diffusion," to be published.
- [76] S. K. Jones and A. Gerodolle. "2D Process Simulation of Dopant Diffusion in Polysilicon," *NASECODE VII Transactions*, COMPEL, Vol. 10, No. 4, pp. 401-410, 1991.
- [77] A. Gerodolle and S. K. Jones. "Integration in the 2D Multi-Layer Simulator TITAN of an Advanced Model for Dopant Diffusion in Polysilicon," in *Simulation of Semiconductor Devices and Process*, Vol. 4, pp. 381-388, edited by W. Fichtner and D. Aemmer, Hartning-Gorre Verlag (Konstanz), 1991.
- [78] S. K. Jones, A. Gerodolle, C. Lombardi, and M. Schafer. "Complete Bipolar Simulation Using STORM," *IEDM-92, Tech. Digest*, p. 931, 1992.
- [79] A. G. O'Neill, C. Hill, J. King, and C. Please. "A New Model for The Diffusion of Arsenic in Polycrystalline Silicon," *J. Appl. Phys.*, Vol. 64, No. 1, p. 167, 1988.
- [80] M. M. Mandurah, K. C. Saraswat, R. W. Dutton, and T. I. Kamins. "Dopant Segregation in Polysilicon," *J. Appl. Phys.*, Vol. 51, pp. 5755, 1981.
- [81] L. Mei and R. W. Dutton. "A Process Simulation Model for Multilayer Structures Involving Polycrystalline Silicon," *IEEE Trans. Elec. Dev.*, Vol. ED-29, No. 11, p. 1726, 1982.
- [82] L. Mei, M. Rivier, Y. Kwark, and R. Dutton. "Grain Growth Mechanisms in Polysilicon," *J. Electrochem. Soc.*, Vol. 129, No. 8, p. 1791, 1982.
- [83] C. V. Thompson. "Secondary Grain Growth in Thin Films of Semiconductors: Theoretical Aspects," *J. Appl. Phys.*, Vol. 58, No. 2, p. 763, 1985.
- [84] D. Gupth, D. R. Campbell, and P. S. Ho. *Thin Films—Interdiffusion and Reactions*, New York: Wiley, p. 161, 1980.
- [85] C. Hill and S. K. Jones. "Modelling Dopant Diffusion in and from Polysilicon," *Mat. Res. Symp. Proc.*, No. 182, p. 129, 1990.
- [86] S. Ajuria and R. Reif. "Early Stage Evolution Kinetics of the Polysilicon/Single-Crystal Silicon Interfacial Oxide Upon Annealing," *J. Appl. Phys.*, Vol. 69, No. 2, p. 662, 1991.

- [87] J. D. Williams. "Epitaxial Alignment of Polycrystalline Silicon and its Implications for Analogue Bipolar Circuits," Doctoral dissertation, University of Southampton, Faculty of Engineering and Applied Science, 1992.
- [88] F. Benyaich, F. Priolo, E. Rimini, C. Spinella, and P. Ward. "Kinetic and Structural Study of the Epitaxial Realignment of Polycrystalline Si Films," *J. Appl. Phys.*, Vol. 71, No. 2, p. 638, 1992.
- [89] E. A. Irene. "Residual Stress in Silicon Nitride Films," *J. Electronic Mat.*, Vol. 5, No. 3, p. 287, 1976.
- [90] S. M. Hu. "Film-Edge-Induced Stress in Silicon Substrates," *Appl. Phys. Lett.*, Vol. 32, p. 5, 1978.
- [91] S. M. Sze. *Physics of Semiconductor Devices*, New York: Wiley-Interscience, pp. 28-34, 1969.
- [92] K. Yamaguchi. "Field-Dependent Mobility Model for Two-Dimensional Numerical Analysis of MOSFET's," *IEEE Trans. Electron Devices*, Vol. ED-206, pp. 1068-1074, July 1979.
- [93] K. Yamaguchi. "A Mobility Model for Carriers in the MOS Inversion Layer," *IEEE Trans. Electron Devices*, Vol. ED-30, pp. 658-663, June 1983.
- [94] N. D. Arora, J. R. Hauser, and D. J. Roulston. "Electron and Hole Mobilities in Silicon as a Function of Concentration and Temperature," *IEEE Trans. Electron Devices*, Vol. ED-29, pp. 292-295, Feb. 1982.
- [95] D. M. Caughey and R. E. Thomas. "Carrier Mobilities in Silicon Empirically Related to Doping and Field," *Proc. IEEE*, Vol. 55, pp. 2192-2193, 1967.
- [96] M. J. van Dort, P. H. Woerlee, and A. J. Walker. "A Simple Model for Quantization Effects in Heavily-Doped Silicon MOSFETs at Inversion Conditions," *Solid State Electronics*, Vol. 37, No. 3, pp. 411-414, 1994.
- [97] R. Y. S. Huang and R. W. Dutton. "Experimental Investigation and Modeling of the Role of Extended Defects During Thermal Oxidation," *J. Appl. Phys.*, Vol. 74, No. 9, pp. 5821-5827, Nov. 1993.
- [98] S. M. Hu. in *Defects in Semiconductors*, edited by J. Narayan and T. Y. Tan, Amsterdam: North-Holland, pp. 333-354, 1981.
- [99] H. Cerva and G. Hobler. "Comparison of Transmission Electron Microscope Cross Sections of Amorphous Regions in Ion Implanted Silicon with Point-Defect Density Calculations," *J. Electrochem. Soc.*, Vol. 139, No. 12, pp. 3631-8, 1992.
- [100] R. Rios, et al. "A Physical Compact MOSFET Model, Including Quantum Mechanical Effects for Statistical Circuit Design Applications," *IEDM '95 Tech. Digest*, pp. 937-940.
- [101] K. B. Parab, S. -H. Yang, S. J. Morris, S. Tian, M. Morris, B. Obradovich, A. F. Tasch, D. Kamenitsa, R. Simonton, and C. Magee. "Detailed Analysis and Computationally Efficient Modeling of Ultra-Shallow Profiles Obtained by Low Energy B, BF₂, and As Ion Implantation," *MRS Fall Meeting*, Boston, 1995.

Input Statement Descriptions

Introduction

The TSUPREM-4 user directs the program via input statements. These statements can appear in an input command file or can be entered directly from the user's terminal. They also appear in the *s4init* file.

This chapter describes the statements recognized by TSUPREM-4. The first section gives the general format of the input and defines the syntax used in the detailed documentation of the following sections. The following sections include:

- [3.1 Documentation and Control on page 3-7](#) discusses statements that control the execution of TSUPREM-4.
- [3.2 Device Structure Specification on page 3-49](#) discusses statements that specify the device structure used by TSUPREM-4.
- [3.3 Process Steps on page 3-89](#) discusses statements that simulate processing steps.
- [3.4 Output on page 3-127](#) discusses statements to print and plot results, and extract, structural, doping, and electrical characteristics.
- [3.5 Models and Coefficients on page 3-187](#) discusses statements that specify the models and coefficients used by TSUPREM-4.
- [3.6 Summary on page 3-309](#) presents a summary of the input statements recognized by TSUPREM-4.

Input Statements

The input to TSUPREM-4 consists of various statements. This section describes the format and syntax of these statements. Note that the input syntax recognized by TSUPREM-4 differs slightly from that used by other *Avant!* TCAD products.

Format

Each statement consists of a statement name followed by a list of parameters. Valid statement and parameter names are described in the following sections of this manual. Statements have the following general format:

- Statement and parameter names are recognized in either upper or lower case.
- Usually, each statement is terminated by the end of an input line. If a statement requires more than one input line, each line except the last must end with a plus (+) as the last nonblank character. A statement is automatically continued if the end of line occurs inside a quoted character string.
- A statement (other than **COMMENT**, **DEFINE**, or **UNDEFINE**) can also be terminated by a semicolon (;) appearing outside of a quoted character string; this allows more than one statement to be placed on a line.
- Blank lines are ignored.

Syntax

Parameter and statement names can be abbreviated by dropping characters from the end of the name. Ambiguous abbreviations are not permitted. The abbreviation for a parameter name must be long enough to distinguish among the parameters associated with a statement. The abbreviation for a statement name must be long enough to distinguish it from other statements. Parameter and statement names can also be extended by adding characters to the end of the unabbreviated name.

Most of the statements recognized by TSUPREM-4 obey a single, simple set of syntax rules. A few statements are treated specially, however. These are referred to as *special* statements in the discussion that follows.

Specifying Materials and Impurities

Many statements require the specification of materials. In all cases, materials can be specified by name using the **MATERIAL** and **/MATERIA** parameters. The material names recognized by these parameters are *silicon*, *polysili*, *oxide*, *oxynitri*, *nitride*, *photores*, *aluminum*, *titanium*, *TiSi2*, *tungsten*, *WSi2*, and the names of any user-defined materials. Note that material names are not case sensitive, and can be abbreviated or extended like parameters or statements names. Some statements also have the names of materials as parameters; only the materials available in older versions of TSUPREM-4 can be specified in this way.

Many statements require the specification of impurities or solution values. In all cases, impurities and solution values can be specified by name using the **IMPURITY** parameter. Names of impurities and solution values are not case sensitive. Note that the list of meaningful names depends on the semantics of the statement: some statements only take the names of impurities, while others work with any solution value. Some statements also have the names of impurities or

solution values as parameters; only the impurities and solution values available in older versions of TSUPREM-4 can be specified in this way.

Parameters

The parameters are separated from the statement name by one or more spaces. There are three types of parameters.

- Numerical
- Character
- Logical

Special statements may require other types of parameters, specified in a particular order. These are described in the detailed statement description sections.

Logical

Logical parameters are set true by specifying the parameter name only, and can be set false by preceding the name by one of the *not* characters, “^”, “!”, or “#”.

Numerical

Numerical parameters require that a value be specified. The value is separated from the parameter name by an equals character (=). Blanks on either side of the equals character are ignored. The value can be specified as an integer or fixed-point decimal number, or as a floating-point number using the character “E” or “e” to delimit the exponent (e.g., 101, 101.0, or 1.01E2).

The value of a numerical parameter can also be specified as a mathematical expression. The mathematical operators +, -, *, /, and ^ (for exponentiation) can be used, as well as a variety of mathematical functions. See the description of the **SELECT** statement on [page 3-127](#) for a list. If a mathematical expression includes spaces, the entire expression must be enclosed in parentheses. If an expression is given where an integer value is required, the value of the expression is rounded to the nearest integer.

Character

Character parameters require that a value be specified. The value is separated from the parameter name by an equals character (=). Character parameters can appear in one of two ways depending on the statement involved:

- A named character parameter is assigned a string value using an equals character in the same manner as a numerical parameter. If the string contains spaces it must be enclosed in quotes (").

- An unnamed character parameter is used when a statement consists of a single input line containing the statement name followed by an arbitrary character string. In this case there is no parameter name or equals character (=), and quotes are not needed.

Statement Description Format

The remainder of this chapter describes the input statements recognized by TSUPREM-4. The description of each statement consists of a formatted list of the parameters associated with the statement. This is followed by the parameter definition table.

Parameter Definition Table

The parameter definition table includes the following:

- Parameter name
- Parameter type
- A description of the parameter's function
- Synonyms (if any) which can be used instead of the standard parameter name
- The physical units (if any) for a numerical parameter
- The default value TSUPREM-4 uses in the absence of a user-specified value

Syntax of Parameter Lists

Several special characters are used in the formatted parameter list that appears at the beginning of each statement description:

- Angle brackets < >
- Square brackets []
- Vertical bar |
- Braces { }
- Parentheses ()



Note:

The special characters, < >, [], |, { }, and (), indicate parameter types, optional groups, alternate choices, and group hierarchy. They do not form part of the actual input to TSUPREM-4 (i.e., these special characters are not typed in). Only the information enclosed in the special characters is typed into command strings.

Parameter Types

< >

A lower case letter in angle brackets indicates the type of a parameter. Thus <n> represents the value of a numerical parameter and <c> represents the value of a

character parameter. Logical parameters are denoted by a parameter name without a value. For example,

WIDTH=<n>

indicates that the **WIDTH** parameter is assigned a numerical value. In a few special cases a descriptive word is used in angle brackets to denote a nonstandard parameter.

Parameter Groups

In the following, the term *group* refers to a parameter by itself or a set of parameters enclosed in a matched pair of square brackets, braces, or parentheses. For example,

{ (PARM1 [PARM2 [PARM3]] PARM4) PARM5 }

constitutes a valid group, composed of the subgroups **(PARM1 [PARM2 [PARM3]] PARM4)** and **PARM5**. The first subgroup can further be subdivided into the subgroups **PARM1**, **[PARM2 [PARM3]]** and **PARM4**, etc.

Optional Parameters []

Square brackets enclose groups that are optional. For example,

NEWCARD [PARM1] [PARM2 PARM3] [PARM4 [PARM5]]

indicates that in the **NEWCARD** statement, the parameter **PARM1** is optional. The group **[PARM2 PARM3]** is optional, but if **PARM2** is specified, then **PARM3** must also be specified. The group **[PARM4 [PARM5]]** is optional; **PARM5** can be specified only if **PARM4** is specified.

Choices { }, |

When one of a list of groups must be selected, the groups are enclosed in braces and separated by vertical bars. For example,

NEWCARD {PARM1 | PARM2 | (PARM3 PARM4)}

indicates that the **NEWCARD** statement requires that one and only one of the three groups **PARM1**, **PARM2**, or **(PARM3 PARM4)** be specified.

Group Hierarchy ()

Parentheses enclose groups that are to be considered as single items in higher level groupings. For example, in the above **NEWCARD** statement, the group **(PARM3 PARM4)** constitutes one of three possible choices and is therefore enclosed in parentheses.

3.1 Documentation and Control

The following statements control execution of TSUPREM-4:

| Statement Name | Description | Page |
|-----------------------------------|---|-------------|
| COMMENT | Documents the input file. | 3-8 |
| SOURCE | Reads input statements from a file. | 3-9 |
| RETURN | Exits from a sourced input file or from interactive mode. | 3-10 |
| INTERACTIVE | Enters interactive input mode. | 3-12 |
| PAUSE | Temporarily interrupts execution of the program. | 3-14 |
| STOP | Stops execution of the program. | 3-15 |
| FOREACH/END | Repeats a group of input statements. | 3-16 |
| LOOP/L.END | Begins an input statement loop. | 3-18 |
| L.MODIFY | Modifies processing of an input statement loop. | 3-22 |
| IF/ELSEIF/ ELSE/IF.END | Begins and terminates a sequence of one or more conditionally processed input statement blocks. | 3-23 |
| ASSIGN | Assigns values to an assigned name. | 3-25 |
| INTERMEDIATE | Species intermediate values | 3-32 |
| ECHO | Sends a string to the output file. | 3-38 |
| OPTION | Selects level of printed output and device for plotted output. | 3-39 |
| DEFINE | Defines macros (abbreviations). | 3-42 |
| UNDEFINE | Undefines macros. | 3-45 |
| CPULOG | Enables or disables reporting of execution times. | 3-46 |
| HELP | Prints a brief description of statements and parameters. | 3-47 |

COMMENT

The **COMMENT** statement is used to specify character strings for documenting the user input and the program output.

```

COMMENT
    [ <c> ]
or
$
    [ <c> ]

```

Description

The **COMMENT** statement is used to document the input file. **COMMENT** statements can appear at any point in the input file.

Examples

1. The following is a simple comment:

```
COMMENT    A SHORT COMMENT
```

2. The "\$" form is easier to type and stands out from other statements:

```
$ The "$" form is simpler
```

3. Comments can be continued using the "+" character:

```
$ THIS IS A LONGER COMMENT WHICH HAS BEEN +
  CONTINUED USING THE "+" CHARACTER.
```

4. It is simpler and less confusing to just put a "\$" before each line of a long comment:

```
$ THIS IS A LONG COMMENT WHICH SPANS TWO
$   LINES WITHOUT USING A CONTINUATION CHARACTER
```

Notes

The **COMMENT** statement has the following syntax considerations:

- The **\$** form of the **COMMENT** statement may produce unexpected results if the "\$" character is immediately followed by the name of a defined abbreviation. (See description of the **DEFINE** statement.) In that case, the "\$" character and the name are replaced by the expansion of the abbreviation.
- A **COMMENT**, like any other statement, can be continued by placing a "+" character at the end of the line.
- A **COMMENT**, unlike most other statements, is not terminated by a semicolon (;).

SOURCE

The **SOURCE** statement causes TSUPREM-4 to read input statements from a file.

```
SOURCE
```

```
  <filename>
```

Description

The **SOURCE** statement causes TSUPREM-4 to read input statements from the specified file. **SOURCE** statements can be nested.

Reusing Combinations of Statements

The **SOURCE** statement is useful for reading in previously defined sequences of input statements. Frequently used combinations of statements, such as implant/anneal or mask/expose/etch sequences, can be placed in files and read with the **SOURCE** statement.

Generating Templates

You can use the **SOURCE** statement with the **DEFINE** or **ASSIGN** statements to generate template files with variable input values. A template file can be constructed by replacing portions of the input, such as character strings, parameter names, and parameter values, with defined names and numerical expressions including defined names. An input file that uses the template file must first include **DEFINE** or **ASSIGN** statements that set the values of defined names appearing in the template file and then include a **SOURCE** statement that references the template file. The values of the defined names in the template file are replaced by the values specified in the original input file.

Examples

The following defines the variable *PTITLE* and then reads input statements from file *DOPLOT*:

```
DEFINE  PTITLE  "After Source/Drain Implant"
SOURCE  DOPLOT
```

The file *DOPLOT* might contain statements to plot the structure:

```
SELECT  Z=LOG10(BORON)  TITLE=@PTITLE
PLOT.2D  SCALE  Y.MAX=2
FOREACH X (15 TO 21)
  CONTOUR  VALUE=X
END
```

The value of the variable *PTITLE* is used by the called file as the title of the plot.

RETURN

The **RETURN** statement exits the current input mode and returns to the previous level of input. **BATCH** is a synonym for the **RETURN** statement.

RETURN

[<c>]

Description

The character string associated with the **RETURN** statement is ignored by the program and serves only to document user input.

The **RETURN** statement is used in batch input mode to exit an input command file before the end-of-file. Input statements after the **RETURN** statement are ignored.

Returning from Batch Mode

Three possibilities exist when returning from batch mode:

1. If TSUPREM-4 is started in batch input mode by specifying the input command file at the beginning of program execution, then execution of the program terminates after the **RETURN** statement.
2. If batch input mode is entered by executing the **SOURCE** statement from interactive input mode, then interactive input mode resumes.
3. If batch input mode is entered by executing a **SOURCE** statement from another input command file, then processing resumes with the statement following the **SOURCE** statement.

Exiting Interactive Input Mode

The **RETURN** (or **BATCH**) statement is also used to exit interactive input mode. There are two possibilities:

1. If interactive input mode is entered by executing the **INTERACTIVE** statement in an input command file, then processing resumes with the statement following the **INTERACTIVE** statement.
2. If TSUPREM-4 is started in interactive input mode at the beginning of program execution, then execution of the program terminates after the **RETURN** statement.

Typical end-of-file characters are *control-D (EOT)* and *control-Z (SUB)*. An end-of-file is equivalent to the **RETURN** statement. Thus, interactive input mode can be terminated either with a **RETURN** statement or an end-of-file character. Furthermore, a **RETURN** statement is not necessary at the end of an input command file.

Example

Suppose an input file contains the following lines:

```
$ LINE 1
$ LINE 2
RETURN
$ LINE 4
LINE 5 (BAD SYNTAX!)
```

When this file is read, lines 1 and 2 were executed, but lines 4 and 5 were ignored. Note that no syntax check is performed on lines that follow a **RETURN** statement.

INTERACTIVE

The **INTERACTIVE** statement starts the interactive input mode, allowing statements to be entered interactively from the terminal.

INTERACTIVE

[<c>]

Description

The character string associated with the **INTERACTIVE** statement is ignored by the program and serves only to document user input.

Interactive input mode can be initiated using either of the following methods:

1. An **INTERACTIVE** statement is executed from a batch input file. In this case, the program resumes processing statements from the batch input file when interactive input is terminated.
2. A blank file specification is given for the command input file at the beginning of program execution. In this case, the program automatically enters interactive input mode. All statements are entered interactively, and the program terminates when interactive input is terminated.

Interactive Input Mode

When interactive input mode is started, the program indicates this by printing a message on the terminal, printing the interactive input prompt (“TS4”), and awaiting input of statements. A statement can be continued on a subsequent line by ending the current input line with a plus (+). Continuation can be used repeatedly to generate input statements consisting of any number of input lines. The program indicates that continuation lines are expected by changing the interactive input prompt to “>” until the statement is complete. A continued statement can be completed by not ending the last line with a plus (+) or by entering a blank line.

Terminate interactive input mode either by entering a **RETURN** statement or by entering an end-of-file during interactive input from the terminal. Typical end-of-file characters are *control-D (EOT)* and *control-Z (SUB)*.

Example

Suppose the following input file is executed:

```
$ LINE 1
$ LINE 2
INTERACTIVE
$ LINE 4
$ LINE 5
```

Lines 1 and 2 are executed. When the **INTERACTIVE** statement is executed, the program prompts for input at your terminal. When you enter a **RETURN** statement or an end-of-file character, execution continues with lines 4 and 5.

PAUSE

The **PAUSE** statement causes the program to stop executing input statements until a response is received from your terminal.

PAUSE

[<c>]

Description

The character string associated with the **PAUSE** statement is ignored by the program and serves only to document user input. When a **PAUSE** statement is executed, the prompt

Type <RETURN> to continue, or a command to be executed:

is issued at your terminal. The program then prompts you to enter a line of input. If you enter a TSUPREM-4 input statement, the statement is executed and another prompt is issued. If the line is blank, the processing of input statements continues in the normal manner.

You can use the **PAUSE** statement to interrupt statement processing temporarily to view graphics output. The **INTERACTIVE** statement can be used for the same purpose.

Example

Consider the following input file:

```
PLOT.2D SCALE GRID
PAUSE
PLOT.2D SCALE FLOW VLENG=.1
```

The program pauses between the two plots. It continues when you enter a blank line.

STOP

The **STOP** statement terminates the execution of the program. **EXIT** and **QUIT** are synonyms for the **STOP** statement.

STOP

[<c>]

Description

Input statements following a **STOP** statement are not checked for syntax or executed. The character string associated with the **STOP** statement is ignored by the program and serves only to document the input. A **STOP** statement is not necessary to terminate program execution—an end-of-file condition on the input file has the same effect.

Example

Consider the input file

```
$ LINE 1
$ LINE 2
STOP
THIS IS LINE 4; IT WILL NOT BE READ
```

Lines 1 and 2 are read and executed. When the **STOP** statement is executed, the program terminates, so the last line is not read or executed.

FOREACH/END

The **FOREACH** statement causes a group of input statements to be processed repeatedly. The **END** statement marks the end of a **FOREACH** loop.

FOREACH

<name> <list>

Description

The **FOREACH** statement is used to specify loops in the input. The *<name>* takes on the values in the *<list>* consecutively until no values remain. The commands between the **FOREACH** statement and the matching **END** statement is executed once for each value in the *<list>*.

The *<list>* is a set of strings enclosed in parentheses and separated by commas or spaces. It can also take the form

(<start> TO <end> STEP <increment>)

where *<start>* is a numerical start value, *<end>* is the last value, and *<increment>* is the size of step to take between them. *<end>* must be greater than *<start>*, and *<increment>* must be greater than zero. The **STEP** and *<increment>* parameters can be omitted, in which case the *<increment>* defaults to one.

The *<name>* is set to the value in the *<list>* in a manner analogous to the **DEFINE** statement; the value is substituted for the *<name>* in the body of the loop accordingly.

Examples

1. In the following code fragment, the **ECHO** statement is executed four times.

```
FOREACH STRING (antimony, arsenic, boron,
phosphorus )
  ECHO STRING
END
```

The name *STRING* is set to the values “antimony,” “arsenic,” “boron,” and “phosphorus” consecutively. This produces the output:

```
antimony
arsenic
boron
phosphorus
```

2. The following increments *VAL* from 1.0 to 10.0 in steps of 0.5. The inner body of the loop is executed 19 times.

```
FOREACH VAL ( 1.0 TO 10.0 STEP 0.5 )  
    ECHO VAL  
END
```

Notes

1. If the value of <name> is changed with a **DEFINE** or **UNDEFINE** statement, the results are undefined.
2. Substitution of **DEFINED** names within a **FOREACH** loop does not occur in exactly the way one might expect. For details of the interactions between the **FOREACH** and **DEFINE** statements refer to [“DEFINE” on page 3-42](#).

LOOP / L . END

The **LOOP** statement begins an input statement loop and specifies the number of times necessary to process the statements within the loop. Optimization may be performed by a loop. The values of numerical and array parameters and assigned names may be varied on statements within loops. The **L . END** statement marks the end of a **LOOP**.

LOOP

```
[ STEPS=<c> ] [ INDEX=<c> ] [ START=<n> ] ]
[ OPTIMIZE [ DSSQ=<n> ] [ DNORM=<n> ] [ PLOT ] ]
```

| Parameter | Type | Definition |
|-----------------|-----------|--|
| STEPS | number | The maximum number of passes through the loop. The statements between the LOOP statement and its matching L . END statement are processed once during each pass through the loop. The loop terminates when the number of passes equals the value of the STEPS parameter. If OPTIMIZE is specified, the loop also terminates when the optimization is successful. The value of the STEPS parameter must be a positive integer. Units: none Default: 50 for OPTIMIZE |
| INDEX | character | The name of the variable to store the number of a looping count which starts with 1 by default. The value is substituted for the name whenever the name appears in the input file preceded by the "@" character. Default: none |
| START | number | The starting number of the index variable specified by the INDEX parameter. Default: 1 |
| OPTIMIZE | logical | Specifies that this loop performs optimization of values that are defined by ASSIGN statements specifying the LOWER and UPPER parameters. Default: false |
| DSSQ | number | The relative change in the sum of squares for convergence of optimization. This parameter can be used to change the criteria of convergence. Units: none Default: 1e-5 |
| DNORM | number | The change in norm of parameter vector for convergence of optimization. This parameter can be used to change the criteria of convergence. Units: none Default: 2e-3 |
| PLOT | logical | Specifies that a plot of optimization and sensitivity analysis results is to be drawn after optimization. The sensitivity graph is plotted only when a profile is specified as a target with the T . FILE parameter on an EXTRACT or ELECTRICAL statement. Default: false |

Description

The **LOOP** statement defines the beginning of a sequence of statements which are processed repeatedly. An **L.END** statement is used to indicate the end of the statement sequence. The statement sequence is processed the number of times specified by the **STEPS** parameter or, if **OPTIMIZE** is specified, until the optimization is completed.

Each **LOOP** statement must be paired with a matching **L.END** statement. The nesting of the loop levels is not restricted (as long as system memory is available). The repeated processing of a statement sequence behaves in the same manner as the case where the sequence of statements is explicitly repeated multiple times. However, if optimization is performed, the state of the simulation is saved internally before the optimization starts and restored at the start of each iteration.

Termination of Optimization Looping

The termination of looping for optimization is determined, in order, by the following:

1. When the RMS error for each target is less than the **TOLERANC**, specified in the **ELECTRICAL** or **EXTRACT** statement.

$$\text{RMSerror}(\text{Target}_i) < \text{TOLERANC}_i \quad \text{for each target } i$$

2. When the SSQ (Sum of Squares) value of the targets increases or the decremental ratio of SSQ value for targets is less than **DSSQ**, and the SSQ value of parameter variations is less than **DNORM**.

For j-th loop,

$$\text{SSQ}_{j-1} < \text{SSQ}_j \quad \text{or} \quad (\text{SSQ}_{j-1} - \text{SSQ}_j) / \text{SSQ}_j < \text{DSSQ} \quad \text{for targets,}$$

and

$$\text{SSQ}_j < \text{DNORM} \quad \text{for parameters}$$

3. When the looping number exceeds the maximum number, **STEPS**.

Parameter Sensitivity

The sensitivity of each parameter to each target is printed out after optimization finishes. The sensitivities show how much effect each parameter has on each target. The sensitivity is calculated with the normalized value of each parameter. The sensitivity is defined as

$$\text{Sensitivity}(\%) = 100 \times \frac{\Delta \text{Target}}{\Delta P_{\text{norm}}}$$

in which ΔT is the change of target with respect to the change of normalized parameter, ΔP_{norm} . (See ["ASSIGN" on page 3-25.](#))

Dependence and Variability

Dependence and variability show how each parameter is expected to depend on the other parameters, while sensitivity shows the dependence of each target on each parameter. Dependence is an estimate of how much the sensitivity to each parameter is decreased if the other parameters are adjusted for a minimum rather than being held fixed. A zero value of dependence implies that the parameter is independent of the other parameters so that the optimized value of the parameter is unique. In general, small values of dependence (<10) are desirable, while large values (>100) imply that an accurate parameter cannot be extracted from the given data. The variability is the possible change of parameter for the same increase in error if the other parameters are simultaneously adjusted for minimum error.

Example

The output below shows the target `tox` depends mainly on the parameter `temp`, while both of the parameters, `temp` and `dose` contribute to the target `xj`.

```
Sensitivities: 100*(change in target)/(change in normalized parameter)
```

| Parameter Name | Target Name | tox | xj |
|----------------|-------------|---------|--------|
| temp | | 115.205 | 16.325 |
| dose | | 4.688 | 13.444 |

If the target is given as multiple data points rather than a specified value, for example, as SIMS data, the sensitivity can be shown in the plot. The **PLOT** parameter in the **LOOP** statement specifies that the sensitivity graph is plotted if necessary (Figure 3-1). The output shows two kinds of sensitivity: RMS sensitivity and maximum sensitivity.

```
Sensitivities: 100*(change in target)/(change in normalized parameter)
```

| Parameter Name | Target Name | boron_sims | boron_sims |
|----------------|-------------|------------|------------|
| theta0 | * | 28.939 ^ | 46.885 |
| seg0 | * | 4.797 ^ | 16.798 |

```
(*) means a RMS sensitivity.
(^) means a maximum sensitivity.
```

In Figure 3-1, the x-axis named `variable` represents the variable of target data. In this example, the variable is a depth in microns. The negative sensitivity means that the simulation result decreases in comparison with the target as the parameter value increases. The figure implies that the diffusivity of boron increases as the parameter `theta0` increases, and that the parameter `seg0` is dominant at the surface.

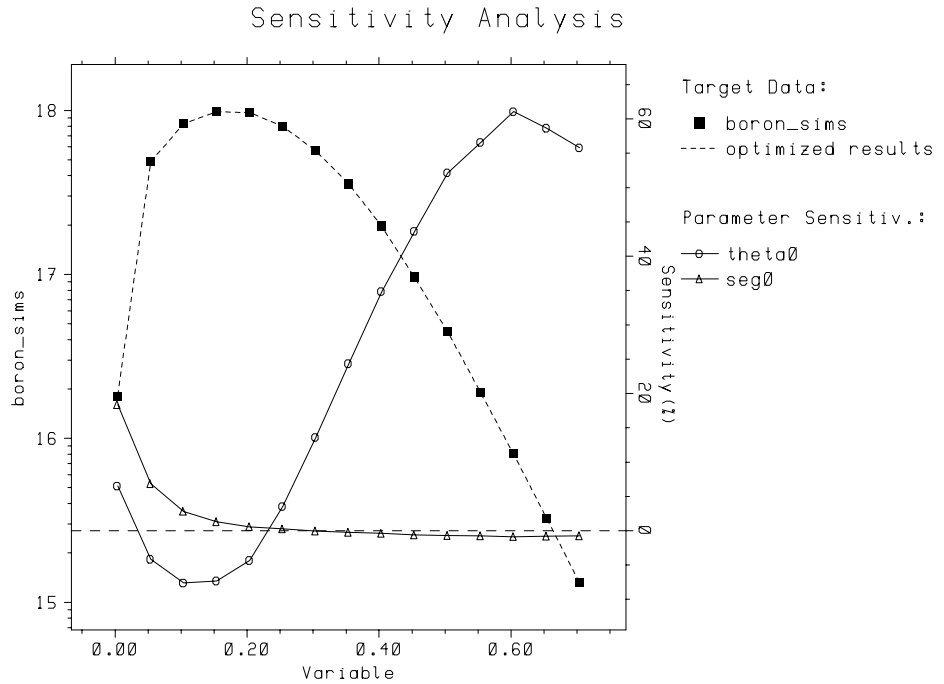


Figure 3-1 Example of sensitivity plot for target with multiple data

The **PLOT** parameter also specifies the plot of the optimization procedure, which shows how the parameters and the simulation results approach the optimized values and the desired values.

Advantages

The use of **LOOP** and **ASSIGN** statements instead of **FOREACH** provides some advantages as follows:

- Different assigned variables can be varied in unrelated ways for each pass through the loop.
- The number of passes through the loop is specified directly. (The number of passes through a **FOREACH** loop is sometimes less than expected due to round off error in computing the loop variable.)
- The **LOOP** statement allows optimization.
- The substitution of **ASSIGNED** variables in a **LOOP** is more logical and intuitive than the substitution of **DEFINED** names in a **FOREACH** loop.
- The **ASSIGN** statement evaluates numerical expressions before assigning them to variables.
- Assigned variables are only substituted when preceded by the “@” character. Thus, a statement or parameter name can never be inadvertently replaced by a variable specified by the **ASSIGN** statement.

L.MODIFY

The **L.MODIFY** statement modifies the processing of a currently active statement loop associated with a **LOOP** statement.

L.MODIFY

[**STEPS**=<n>] [{ **NEXT** | **BREAK** }]

| Parameter | Type | Definition |
|--------------|---------|--|
| STEPS | number | The number of times the statements between the LOOP statement and its matching L.END statement are processed for the current loop level. The value of this parameter must be a positive integer. Units: none Default: the current value for the current loop level |
| NEXT | logical | Specifies that the next statement processed is the L.END statement for the current loop level. The statement between the L.MODIFY statement and L.END statement for the current loop level are not processed during this pass through the loop. Default: false Synonym: CONTINUE |
| BREAK | logical | Specifies that the next statement processed is the statement following the L.END statement for the current loop level. No subsequent passes through the loop is performed. The statements between the L.MODIFY statement and the L.END statement for the current loop level are not processed during this pass through the loop. Default: false |

Description

The **L.MODIFY** statement can be used to modify the number of passes through a loop. Based on the results of previous statements, you may choose to modify the number of subsequent passes through the loop. An **L.MODIFY** statement specifying the **STEPS** statement can be used to increase or decrease the total number of passes through the loop.

IF/ELSEIF/ELSE/IF.END

The **IF** statement begins a sequence of one or more conditionally processed input statement blocks.

The **ELSEIF** statement terminates a conditionally processed input statement block associated with an **IF** or other previous **ELSEIF** statements and begins a new conditionally processed input statement block.

The **ELSE** statement specifies alternative action if the condition part of an **IF** and previous **ELSEIF** statements are false.

```

IF ( condition )
  [ ELSEIF ( condition ) ]
  [ ELSE ]
IF.END

```

Description

The **IF** statement defines the beginning of a sequence of conditionally processed blocks of statements. An **IF.END** statement is used to indicate the end of the sequence of statement blocks. The first statement block in the sequence begins with the **IF** statement, while subsequent statement blocks begin with **ELSEIF** statements and finally with the **ELSE** statement.

At most, one statement block in a sequence of blocks is processed. The statement block processed is the first in the sequence with a true value for the condition on the **IF** or **ELSEIF** statement that begins the block. Only the statement block that begins with the **ELSE** statement in a sequence is processed if the **IF** statement and all **ELSEIF** statements in a sequence have a false value for the condition.

Each **IF** statement must be paired with a matching **IF.END** statement, with possibly intervening **ELSEIF** or **ELSE** statements. Pairs of **IF** and **IF.END** statements may be nested as deeply as system memory allows.

The **ELSEIF** statement defines the beginning of one statement block within a sequence of conditionally processed blocks of statements begun with an **IF** statement. The statement block is terminated by the following **ELSEIF** statement, by the **ELSE** statement, or by the **IF.END** statement that terminates the sequence of statement blocks. The statement block is processed if the value of condition is non-zero (i.e., true) and no previous statement blocks in the sequence have been processed.

The **ELSE** statement defines the beginning of the last statement block within a sequence of conditionally processed blocks of statements begun by an **IF** statement and **ELSEIF** statements. One and only one **ELSE** statement can be used within an **IF-IF.END** block.

Conditional Operators

The relational operators are

> (greater than), *>=* (greater than or equal to), *<* (less than), *<=* (less than or equal to), *==* (equal to), *!=* (not equal to)

These operators all have the same precedence. Below them in precedence are the logical operators:

&& (and), *||* (or)

Expressions connected by *&&* or *||* are evaluated left to right.

Expression for Condition

The compared content can be a numerical expression or a character string. The following example shows the comparison between a number and a formula calculation,

```
ASSIGN NAME=X N.V=0.2 PROMPT="x? "
ASSIGN NAME=Y N.V=0.5 PROMPT="y? "
ASSIGN NAME=FUNC C.V=EXP PROMPT="Function? "
IF (@X*@FUNC(@Y)<10); ECHO x*@FUNC(y)<10; IF.END
```

The character string can be compared. In the expression of condition, the character string must be wrapped by the double quotation “ ”.

```
ASSIGN NAME=SHAPE C.V=none PROMPT=shape=
IF ("@SHAPE"=="triangle")
    ECHO "3 sides"
ELSEIF ("@SHAPE"=="none")
    ECHO "not specified"
ELSE
    ECHO "invalid shape"
IF.END
```



Note:

There should be at least one blank between an IF (or ELSEIF) statement and the open parenthesis “(” of expression for condition.

ASSIGN

The **ASSIGN** statement assigns values to an assigned name.

```

ASSIGN
{ ( NAME=<c> [PRINT] [DELETE] [PROMPT=<c>]
  [ { ( {N.EXPRES=<n> | N.VALUE=<c>}
    [ { DELTA=<n> | RATIO=<n> | (LOWER=<n> UPPER=<n> [LOG]) } ] )
    | C.VALUE=<c>
    | ( C.FILE=<c> [LINE=<n>] )
    | ( [C1=<c>] [C2=<c>] [C3=<c>] [C4=<c>] [C5=<c>]
      [C6=<c>] [C7=<c>] [C8=<c>] [C9=<c>] [C10=<c>] )
    } ] )
| ( ARRAY=<c>
  { ( IN.FILE=<c> DATA=<c> [ {TIF | ROW | COLUMN} ] )
    | IN.NVALU=<c> | IN.CVALU=<c> }
  [C.COUNT=<c> ] )
}
    
```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| NAME | character | The assigned name to which a value is being assigned or for which the current value is printed. Default: none |
| PRINT | logical | Specifies that the current value of the specified name is printed. Default: false |
| DELETE | logical | Specifies that the variable is deleted. Default: false |
| PROMPT | character | The character string used to prompt the user for interactive input of an alternative to the value specified by the N.VALUE , N.EXPRES , or C.VALUE parameter. If the character string read from the terminal input is blank, the value specified by the N.VALUE , N.EXPRES , or C.VALUE parameter is used. This parameter is only allowed with the N.VALUE , N.EXPRES , or C.VALUE parameters. Default: none |
| N.EXPRES | number | The numerical value assigned to the assigned name. If neither LOWER nor UPPER are specified, the DELTA or RATIO parameters can be specified to vary the value of the assigned name with each iteration of the enclosing LOOP statement. Units: none Default: none |

| Parameter | Type | Definition |
|------------------|-----------|---|
| N . VALUE | character | The numerical values assigned to the assigned name. If a single value is specified and neither LOWER nor UPPER are specified, the DELTA or RATIO parameters can be specified to vary the value of the assigned name with each iteration of the enclosing LOOP statement. If multiple values are specified, the value of the assigned name is varied by choosing successive values from the list of values specified with this parameter. After the last value in the list is taken, the sequence begins again with the first value in the list. Only a single value may be specified if LOWER and UPPER are specified. Default: none Synonyms: N . ARRAY, VALUE |
| DELTA | number | The constant difference by which the value of the assigned name is varied. This parameter is only allowed if a single value is specified with the N . VALUE parameter and neither LOWER nor UPPER is specified. This parameter works only within a LOOP / L . END block that does not specify optimization. Units: none Default: none |
| RATIO | number | The constant ratio by which the value of the assigned name is varied. The value of this parameter must be nonzero. This parameter is only allowed if a single value is specified with the N . VALUE parameter and neither LOWER nor UPPER is specified. This parameter works only within a LOOP / L . END block that does not specify optimization. Units: none Default: none |
| LOWER | number | The lower bound for the value of the assigned name during optimization. If the LOWER value is greater than the UPPER value, the LOWER value is automatically exchanged with the UPPER value. Units: none Default: none |
| UPPER | number | The upper bound for the value of the assigned name during optimization. If the UPPER value is less than the LOWER value, the UPPER value is automatically exchanged with the LOWER value. Units: none Default: none |
| LOG | logical | Specifies that the value assigned to the NAME is to be varied on a logarithmic scale during optimization. Default: false |
| C . VALUE | character | The character value assigned to the assigned name. Default: none |
| C . FILE | character | The name of the file including the character string to be read. Default: none |
| LINE | number | The line number of the character string to be read from the file C . FILE . Units: none Default: none |

| Parameter | Type | Definition |
|-----------|-----------|--|
| C1 | character | The first in a list of character values assigned to the assigned name. The value of the assigned name is varied by choosing successive values from the list of values specified with the parameters C1 through C10. If the looping step exceeds the maximum parameter, the assigned character value is circulated from C1 again. This parameter works only within a LOOP/L.END block that does not specify optimization. Default: none |
| C2 | character | The second in a list of character values assigned to the assigned name. The value of the assigned name is varied by choosing successive values from the list of values specified with the parameters C1 through C10. This parameter works only within a LOOP/L.END block that does not specify optimization. Default: none |
| C3 | character | The third in a list of character values assigned to the assigned name. The value of the assigned name is varied by choosing successive values from the list of values specified with the parameters C1 through C10. This parameter works only within a LOOP/L.END block that does not specify optimization. Default: none |
| C4 | character | The fourth in a list of character values assigned to the assigned name. The value of the assigned name is varied by choosing successive values from the list of values specified with the parameters C1 through C10. This parameter works only within a LOOP/L.END block that does not specify optimization. Default: none |
| C5 | character | The fifth in a list of character values assigned to the assigned name. The value of the assigned name is varied by choosing successive values from the list of values specified with the parameters C1 through C10. This parameter works only within a LOOP/L.END block that does not specify optimization. Default: none |
| C6 | character | The sixth in a list of character values assigned to the assigned name. The value of the assigned name is varied by choosing successive values from the list of values specified with the parameters C1 through C10. This parameter works only within a LOOP/L.END block that does not specify optimization. Default: none |
| C7 | character | The seventh in a list of character values assigned to the assigned name. The value of the assigned name is varied by choosing successive values from the list of values specified with the parameters C1 through C10. This parameter works only within a LOOP/L.END block that does not specify optimization. Default: none |
| C8 | character | The eighth in a list of character values assigned to the assigned name. The value of the assigned name is varied by choosing successive values from the list of values specified with the parameters C1 through C10. This parameter works only within a LOOP/L.END block that does not specify optimization. Default: none |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| C9 | character | The ninth in a list of character values assigned to the assigned name. The value of the assigned name is varied by choosing successive values from the list of values specified with the parameters C1 through C10 . This parameter works only within a LOOP/L.END block that does not specify optimization. Default: none |
| C10 | character | The tenth in a list of character values assigned to the assigned name. The value of the assigned name is varied by choosing successive values from the list of values specified with the parameters C1 through C10 . This parameter works only within a LOOP/L.END block that does not specify optimization. Default: none |
| ARRAY | character | The prefix name for an array to store the sequential data from the external data file. The index of an array starts with 1. Default: none |
| IN.FILE | character | The identifier for the file containing the data to be stored to the ARRAY parameter. The file format must be either a TIF or a columnwise format. Default: none |
| DATA | character | The name of the data in a TIF file or the column number of the data in a columnwise file to be stored to the ARRAY parameter. Default: none |
| TIF | logical | Specifies that the IN.FILE is a TIF file. Default: false |
| ROW | logical | Specifies that the IN.FILE is a rowwise file. Default: false |
| COLUMN | logical | Specifies that the IN.FILE is a columnwise file. Default: true |
| IN.NVALU | character | The string composed of numbers, for example, "1.2 3 4.5 6." Default: none |
| IN.CVALU | character | The string composed of character strings, for example, "a bc def g." Default: none |
| C.COUNT | character | The name of the variable to which the number of the stored data is assigned. Default: none |

Description

The **ASSIGN** statement associates a value with a name. The value is substituted for the name whenever the name appears in the input file preceded by the "@" character. The name must start with a letter and may contain only letters and numbers.

The **ASSIGN** statement performs a function similar to that of the **DEFINE** statement, but in a way that is more convenient for many applications. Some of its advantages are:

- An **ASSIGN**ed name can take on a different value for each iteration of a **LOOP** statement. Different names can be varied in unrelated ways during a single loop.
- The syntax for using **ASSIGN**ed names is simpler than for **DEFINED** names, leaving less chance for confusion or error.
- The **ASSIGN** statement works in the same way in TSUPREM-4 as it does in Medici and Davinci.

Varying During Statement Looping

The **ASSIGN**ed variable may be varied during statement looping by one of the following:

- A constant increment, by specifying the **DELTA** parameters:

```
LOOP STEPS=3
  ASSIGN NAME=X N.VAL=10 DELTA=10 PRINT
L.END
```

The variable X takes the values 10, 20, and 30.

- A constant ratio, by specifying the **RATIO** parameter:

```
LOOP STEPS=3
  ASSIGN NAME=X N.VAL=10 RATIO=10 PRINT
L.END
```

The variable X takes the values 10, 100, and 1000.

- A list of values from which successive values are taken during each pass through a statement loop:

```
LOOP STEPS=3
  ASSIGN NAME=X N.VAL="20, 10, 90" PRINT
L.END
```

The variable X takes the values 20, 10, and 90.

ASSIGN with Mathematical Expressions

The assigned value may be specified with another variable and/or a mathematical expression as well as a constant. For example,

```
ASSIGN NAME=X N.VAL=0.2
ASSIGN NAME=Y N.VAL=ERFC(@X) PRINT
LOOP STEPS=2
  ASSIGN NAME=Z N.VAL="@X+@Y, LOG10(@X*@Y)"
L.END
```



Note:

The N.VALUE (N.ARRAY) is handled as a character string; the separator of an N.VALUE is a blank or a comma character. Thus, if there is a blank character in the specified formula to N.VALUE, for example,

N.VALUE=(@X + @Y) *may not be accepted. Instead, use the N.EXPRES. Since the N.EXPRES is not a character string but a number, the blank character can be accepted as long as the formula is wrapped by parentheses. If the formula is not wrapped by parentheses, even N.EXPRES disallows a blank character.*

ASSIGN and Optimization

- The **ASSIGNED** variable specifies a parameter to be optimized when both **LOWER** and **UPPER** parameters are specified. The initial value, with which the optimization starts, is specified with the **N.VALUE** parameter. The optimization loop determines an appropriate value for each assigned name for each pass through the loop after the first. The range of allowed values for the assigned name is specified with the **LOWER** and **UPPER** parameters.
- The sensitivity shown after optimization is based on the value of normalized parameters in order to eliminate the unit dependency. The normalization is performed as follows:

$$Param_{,norm} = \frac{Param - LOWER}{UPPER - LOWER}$$

Thus, the reported sensitivity value depends on the values of **LOWER** and **UPPER**.

Expansion of ASSIGNED Variable

The combination of assigned variables is expanded from left to right. For example,

```
ASSIGN NAME=A C.V=T
ASSIGN NAME=B C.V=M
ASSIGN NAME=C C.V=A
ASSIGN NAME=D C.V=@A@B@C PRINT
```

The variable, D, is assigned with the character string, TMA.

```
ASSIGN NAME=EF C.V=BEST
ASSIGN NAME=GH C.V=F
$ 1)
ASSIGN NAME=IJ C.V=@E@GH PRINT

ASSIGN NAME=E C.V=GOL
$ 2)
ASSIGN NAME=F C.V=@E@GH PRINT
```

At 1), E is an unknown variable and, thus, is not expanded. @GH is expanded to the character, F. And then, @EF is expanded to the string BEST. However, at 2), E is not an unknown variable any longer so that @E is expanded to the string, GOL.

Finally, the variable, F, is assigned with the string, GOLF, to which GOL@GH has been expanded.

Reading the External Data File

The data in a TIF file or a rowwise or a columnwise file can be read through the **ASSIGN** statement. The data is sequentially read into the variable **ARRAY** to be appended with the index of which the starting number is 1. The parameter **DATA** specifies which data is to be read, the name of the data for a TIF file, the row number of data for a rowwise file, or the column number of data for a columnwise file. The variable defined by **C.COUNT** stores the number of data to be read.

```

ASSIGN ARRAY=VG IN.FILE=VGID.IVL TIF DATA=V(GATE)+
      C.COUNT=NDATA
ASSIGN ARRAY=ID IN.FILE=VGID.IVL TIF DATA=I(DRAIN)
LOOP STEPS=@NDATA

ASSIGN NAME=i N.V=1 DELTA=1
EXTRACT PREFIX="@VG@i @ID@i "

L.END

```

If the number of data is 12, the value of **NDATA** is 12, and each variable of **VG_n** and **ID_n**, in which *n* has the number from 1 to 12, stores the V(GATE) and I(DRAIN), respectively. The example shows how to read the .*ivl* (TIF format) file written by Medici. It prints out the data, V(GATE) vs. I(DRAIN), in a *vgid.ivl* file.



Note:

*If there is already a variable with the same name as the **ARRAY**, the previous value of the variable is overridden. In the above example, if the variable, for example, VG1, already exists before reading the data file, the current value of VG1 is lost and replaced with the data read.*

Also the character string can be read from an external file.

```
ASSIGN NAME=str C.FILE=text.dat LINE=10
```

The variable *str* stores the string of 10-th line in the file *text.dat*.

Reading the Array from a String

The array can be read from a data string composed of either numbers or characters.

```

ASSIGN ARRAY=vg IN.NVALU="0.05 1.0" C.COUNT=nvgs
ASSIGN ARRAY=type IN.CVALU="linear saturat"

```

The numbers of the variables *vg1* and *vg2* are 0.05 and 1.0, respectively. The strings of the variables *type1* and *type2* are "linear" and "saturat," respectively. The variable *nvgs* stores the number 2 of the count of the array.

INTERMEDIATE

The **INTERMEDIATE** statement specifies the intermediate quantities associated with equations. Aliases for the **INTERMEDIATE** statement are **MODEL** (see [Page 6-7](#)) or **ARRAY** (see [Page 6-9](#)).

INTERMEDIATE

```
NAME=<c> [EXPRESS=<c>] [SIZE=<c> [MININDEX=<c>][DEFAULT=<c>]]
[PRINT] [DELETE]
```

| Parameter | Type | Definition |
|-----------------|-----------|--|
| NAME | character | The intermediate name. Default: none |
| EXPRESS | character | The expression for the intermediate named by the NAME parameter. The EXPRESS parameter can specify a single expression or multiple expressions separated by commas (.). Multiple expressions are allowed only when the SIZE parameter is specified. Default: none Synonym: VALUE |
| SIZE | character | The array size. The SIZE specification implies that the intermediate is an array, the size of which is specified by the SIZE parameter. Default: none Synonym: LENGTH |
| MININDEX | character | The array minimum index number Default: 0 |
| DEFAULT | character | The default initial expression for array elements. Only one expression is allowed. The DEFAULT parameter may not be specified together with the EXPRESS parameter. Default: none |
| PRINT | logical | Specifies that the values of intermediates are to be printed. Default: false |
| DELETE | logical | Specifies that the intermediate is to be removed. Default: false |

Advantages of INTERMEDIATES

The advantages of using intermediates quantities are:

- Simplifying long and complicated equations.
- Avoiding the duplicated calculation of quantities used in several equations.
Once the intermediates are evaluated, the calculated values are shared by all associated equations until the intermediates need to be reset.
- Avoiding duplicated calculation of quantities depending *only* on temperature.

If the intermediates depend *only* on temperature, they are evaluated *only* when the temperature changes.

- Use intermediates in the same manner as the built-in solution values in the **SELECT** statement.



Note:

Intermediates are stored as symbolic expressions that are evaluated later (e.g. when solving the model equations).

Value Type

When the expression is described by the **EXPRESS** parameter, the value type is automatically determined to be one of the following:

- **CONSTANT**
The value is a constant number. It is recalculated whenever any other constant intermediate is redefined.

```
INTERMED NAME=C1 EXPRESS=1.0      ;$1
INTERMED NAME=C2 EXPRESS=C1      ;$2
INTERMED NAME=C1 EXPRESS=2.0     ;$3
```

In the third line, the *C2* value is changed from 1.0 to 2.0.

- **PARAMETER**
The value is dependent *only* on temperature. The value will be recalculated whenever the temperature changes.

```
INTERMED NAME=P1 EXPRESS=1E-5*exp(-2.0/kt) ;$1
INTERMED NAME=P2 EXPRESS=P1                ;$2
```

The values of the *P1* and *P2* intermediates are always the same. The value type of the intermediate *P2* is **PARAMETER**.

- **SOLUTION**
The value is a node solution or an edge solution. The *node solution* type specifies that the quantity varies on nodes. The *edge solution* type represents the quantity evaluated on the edges between nodes.

```
INTERMED NAME=S1 EXPRESS=BORON+PHOS ;$1
INTERMED NAME=S2 EXPRESS=MIN(S1,1E21);$2
```

The value type of the intermediate *S2* is **SOLUTION**.

In case the type of an intermediate is changed by reassignment, the value types of the others associated with the intermediate are also reviewed and changed, if needed.

```

INTERMED NAME=P1 EXPRESS=1E-5*exp(-2.0/kt) ;$1
INTERMED NAME=P2 EXPRESS=P1 ;$2
INTERMED NAME=P1 EXPRESS=2.0 ;$3

```

In the third line, the *P2* value type is changed to CONSTANT.

Array Values

The **SIZE** parameter specifies that the intermediate is an array as well as its array size. The values for the array elements are provided by the string. In the string, each element is separated by a comma (,) in the **EXPRESS** parameter. When the **DEFAULT** parameter is specified instead of the **EXPRESS** parameter, all elements are initialized to the **DEFAULT** parameter. The **DEFAULT** parameter allows only a single expression. The expression type can be CONSTANT, PARAMETER, and also SOLUTION. If the number of data items listed in the **EXPRESS** parameter is smaller than the **SIZE** parameter, the undefined array elements are assigned with N/A values. Conversely, if the number of data items listed are greater than the **SIZE** parameter, all the data items in the list beyond the **SIZE**th item are ignored. You can also define the starting value of array indices with the **MININDEX** parameter. The default value of the **MININDEX** parameter is 0. Each array element can have a different value type.

```

INTERMED NAME=a SIZE=3 MININDEX=-1 +
EXPRESS="1,BORON+PHOS,exp(-1.0/KT)" PRINT

```

The array size 3 and the minimum index -1 means that the first array element is *a[-1]* and the last one is *a[1]*. The array elements are assigned with the different value types CONSTANT, SOLUTION, and PARAMETER. The output shows:

```

Array Name : a
-----
Index      Type      Expression      Value
-----
[-1]      CONSTANT  1                1
[0]       SOLUTION  BORON+PHOS
[1]       PARAMETER exp(-1.0/KT)    2.01e-05
-----

```

Multi-dimensional arrays are defined in a similar way to single arrays. The array dimension is determined by the number of sizes listed in the **SIZE** parameter. Each number listed corresponds to the size of each dimension. Similarly, the minimum numbers of each index are specified with the string specified by the **MININDEX** parameter.

```

INTERMED NAME=b SIZE="2,3" MININDEX="0,-1" +
EXPRESS="1,2,3,SUM(i,1,10,i^2),KT,BORON" PRINT

```

There are two sizes listed in the **SIZE** parameter, 2 and 3. That means it is a two-dimensional array. The size of the first dimension is 2 and its index starts from 0.

The size of the second dimension is 3 and its minimum index is -1. Thus, the first array element is $b[0][-1]$ and the last one is $b[1][1]$. The output shows:

```
Array Name : b
```

| Index | Type | Expression | Value |
|---------|-----------|-----------------|--------|
| [0][-1] | CONSTANT | 1 | 1 |
| [0][0] | CONSTANT | 2 | 2 |
| [0][1] | CONSTANT | 3 | 3 |
| [1][-1] | CONSTANT | SUM(i,1,10,i^2) | 385 |
| [1][0] | PARAMETER | KT | 0.0925 |
| [1][1] | SOLUTION | BORON | |

Getting Array Values

Use the bracket symbol ([]) to get the value of an array element.

```
INTERMED NAME=a SIZE=2 EXPRESS="10,20" ;$1
INTERMED NAME=b EXPRESS=a[1] ;$2
INTERMED NAME=c SIZE=3 EXPRESS="a[0],b,30" ;$3
INTERMED NAME=c PRINT
```

The array element can be the other array element as well as an intermediate as shown in the third line. The above example prints out as:

```
Array Name : c
```

| Index | Type | Expression | Value |
|-------|----------|------------|-------|
| [0] | CONSTANT | a[0] | 10 |
| [1] | CONSTANT | b | 20 |
| [2] | CONSTANT | 30 | 30 |

Redefinition of Array Elements

Redefine array elements in the same manner as non-array intermediates.

```
INTERMED NAME=a SIZE="3,2" +
    EXPRESS="1,2,temp,4,arsenic,0"
INTERMED NAME=a[0][1] EXPRESS=boron
```

The $a[0][1]$ is changed to the solution variable *boron* from the constant 2.

Effect Due to Redefinition

Changing the value type of an array element affects the types of the other intermediates.

```
INTERMED NAME=a SIZE="3,2" +
    EXPR="1,2,temp,4,arsenic,0" ;$1
INTERMED NAME=b EXPR=sum(i,0,0,sum(j,0,1,a[i][j])) ;$2
INTERMED NAME=c EXPR=sum(i,0,1,sum(j,0,1,a[i][j])) ;$3
INTERMED NAME=d EXPR=sum(i,1,2,sum(j,0,1,a[i][j])) ;$4
INTERMED PRINT ;$5
INTERMED NAME=a[0][1] EXPR=boron ;$6
INTERMED PRINT ;$7
```

In the fifth line, the output shows:

| Intermed | Type | Expression | Value |
|----------|-----------|-------------------------------|----------|
| b | CONSTANT | sum(i,0,0,sum(j,0,1,a[i][j])) | 3 |
| c | PARAMETER | sum(i,0,1,sum(j,0,1,a[i][j])) | 1.08e+03 |
| d | SOLUTION | sum(i,1,2,sum(j,0,1,a[i][j])) | |

At line six, the value of the array element $a[0][1]$ is changed to a solution variable, *boron* from a constant of 1. It implies that the value type of $a[0][1]$ is changed to SOLUTION from CONSTANT. Thus, it affects the value types of the intermediates *b* and *c* of which the expressions include $a[0][1]$. In the seventh line, it prints out:

| Intermed | Type | Expression | Value |
|----------|----------|-------------------------------|-------|
| b | SOLUTION | sum(i,0,0,sum(j,0,1,a[i][j])) | |
| c | SOLUTION | sum(i,0,1,sum(j,0,1,a[i][j])) | |
| d | SOLUTION | sum(i,1,2,sum(j,0,1,a[i][j])) | |

Comparison to ASSIGN

The **ASSIGN** statement assigns numeric or string values. The values assigned by the **ASSIGN** statement are never changed unless they are respecified by **ASSIGN**. The value assigned by the **ASSIGN** statement is substituted for the name preceded by the '@' character. The differences between the **INTERMED** and **ASSIGN** statements are as follows:

| Differences | INTERMED | ASSIGN |
|---------------------------------|---|--|
| Numeric value type | INTERMED NAME=a EXPRESS=exp(2) | ASSIGN NAME=a N.VALUE=exp(2) |
| String value type | Not available | ASSIGN NAME=a C.VALUE="TS4" |
| Solution value type | INTERMED NAME=a EXPRESS=boron | Not available |
| Array definition | INTERMED NAME=b + EXPRESS="4,5" SIZE=2 | ASSIGN ARRAY=b + IN.NVALU="4,5" |
| Array dimension | Multi-dimension allowed | one dimension only |
| Default minimum index of arrays | 0 | 1 |
| Getting values | a b[1] | @a @b1 |

| Differences | INTERMED | ASSIGN |
|----------------------------------|--|--|
| <p>Evaluation of expressions</p> | <p>Expression are saved and evaluated only when necessary</p> <pre> INTERMED NAME=a EXPRESS=1 INTERMED NAME=b EXPRESS=a INTERMED NAME=a EXPRESS=2 INTERMED NAME=b PRINT ;\$ b=2 SELECT TEMP=800 INTERMED NAME=c EXPRESS=temp SELECT TEMP=900 INTERMED NAME=c PRINT;\$ c=1173 </pre> | <p>Expressions are evaluated immediately</p> <pre> ASSIGN NAME=a N.VALUE=1 ASSIGN NAME=b N.VALUE=@a ASSIGN NAME=a N.VALUE=2 ASSIGN NAME=b PRINT ;\$ b=1 SELECT TEMP=800 ASSIGN NAME=c N.VALUE=temp SELECT TEMP=900 ASSIGN NAME=c PRINT;\$ c=1073 </pre> |

ECHO

The **ECHO** statement prints its parameters, evaluating arithmetic expressions.

ECHO

<string>

Description

The **ECHO** statement prints the given <string> on the standard output. Defined abbreviations are substituted into the string. If the result is a valid arithmetic expression, it is evaluated. (See “**COMMENT**” on page 3-8 and “**SELECT**” on page 3-128.) Otherwise, the resulting string value is printed.

Examples

1. The statements

```
DEFINE W 2.0
ECHO The width is W - 0.5
ECHO W - 0.5
```

prints the two lines

```
The width is 2.0 - 0.5
1.5
```

to your terminal and to the standard output file. The first **ECHO** statement prints a string, because its argument is not a valid arithmetic expression. The second **ECHO** statement shows evaluation of a valid expression.

2. The statement

```
ECHO ( 15.0 - 12.0 * exp( 4.0 - 2.0 / 6.0 ) )
```

prints

```
-454.455
```

which is the value of the arithmetic expression.

OPTION

The **OPTION** statement sets terminal and plotting options.

OPTION

```
[DEVICE=<c>] [PLOT.OUT=<c>]
[ {QUIET | NORMAL | VERBOSE | DEBUG} ]
[INFORMAT] [DIAGNOST]
[ECHO] [EXECUTE] [V.COMPAT]
```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| DEVICE | character | The name of the graphics device to be used for plotting. Default: none |
| PLOT.OUT | character | The file to which graphics output is sent. If the name is preceded by a "+", the output is appended to an existing file; otherwise, an existing file is overwritten or a new file is created. Default: none Synonyms: FILE.SAV , PLOT.SAV |
| QUIET | logical | Print a minimum of information in the standard output and the output listing file. Default: the current value; initially false |
| NORMAL | logical | Print the "normal" amount of information in the standard output and the output listing file. Default: the current value; initially true |
| VERBOSE | logical | Print more than the normal amount of information in the standard output and the output listing file. Default: the current value; initially false |
| DEBUG | logical | Include all available debugging information in the standard output and the output listing file. Default: the current value; initially false |
| INFORMAT | logical | Send additional output to the informational output file. Default: the current value; initially false |
| DIAGNOST | logical | Send additional output to the diagnostic output file. Default: the current value; initially false |
| ECHO | logical | Echo each input statement to the standard output as it is processed. Default: the current value; initially true |
| EXECUTE | logical | Execute each input statement as it is read. If EXECUTE is false, input statements are checked for syntax but not executed. Default: the current value; initially true |
| V.COMPAT | number | Change certain models and algorithms to be compatible with the specified version of TSUPREM-4. Default: the current version of TSUPREM-4 |

Selecting a Graphics Device

The **DEVICE** parameter can be used to specify the type of graphics device to be used for plotting. The device types known to the program are described in the *s4pcap* file; a listing of the available devices is given in [Appendix B](#). If no **DEVICE** is specified, a default device is chosen as follows:

1. If the environment variable *DEFPDEV* is defined as the name of a valid plot device, its value is used as the plot device.
2. Otherwise, if the environment variable *TERM* is defined as the name of a valid plot device, its value is used as the plot device.
3. Otherwise, the **DEFAULT** device in the *s4pcap* file is used. Note that the *s4pcap* file can be modified to make the **DEFAULT** device refer to any available real plotting device (see [Appendix B](#)).

Redirecting Graphics Output

Graphics output normally goes to your terminal or to the file specified in the *s4pcap* file. In either case, the output can be redirected to the file specified by the **PLOT.OUT** parameter. This allows graphics output to be saved for later processing or display. When graphics output is redirected by **PLOT.OUT**, it does not appear on your terminal or in the file specified in *s4pcap*.

Printed Output

The **QUIET**, **NORMAL**, **VERBOSE**, and **DEBUG** parameters control the amount of printed output generated by the program. **NORMAL** mode is assumed when the program begins. Output from the **VERBOSE** and **DEBUG** modes is not needed during typical usage of TSUPREM-4 and may be difficult to interpret.

Informational and Diagnostic Output

The **INFORMAT** and **DIAGNOST** parameters specify whether informational and diagnostic output are written to the appropriate files. If output to one of these files is disabled before any information is written, the file is not created. The default initialization file *s4init* sets both these parameters false, disabling both files.

Echoing and Execution of Input Statements

The **ECHO** parameter controls whether input statements are listed on the standard output (usually your terminal) as they are processed. By default, this option is turned *on*. The **EXECUTE** parameter controls whether statements are executed as they are encountered. If **EXECUTE** is false, statements are checked for syntax but not processed. This is useful for doing a preliminary syntax check of an input file.

The **DEFINE**, **UNDEFINE**, **FOREACH**, **SOURCE**, **RETURN**, **OPTION**, and **STOP** statements are always executed, regardless of the setting of the **EXECUTE** option.

Version Compatibility

The **V.COMPAT** parameter is used to improve compatibility with older versions of TSUPREM-4. When an older version number is specified, certain models and algorithms from the older version are used in place of the current models and algorithms. Only some changes between versions are reversed by using **V.COMPAT**. For a description of changes in the current version that are affected by **V.COMPAT**, see [Appendix C](#).

Examples

1. The following statement causes the graphics output to be produced for a Tektronix 4100-series terminal and appended to the file *plotsave*:

```
OPTION DEVICE=4100 PLOT.OUT="+plotsave"
```

2. Additional information is printed on the standard output with the statement

```
OPTION VERBOSE
```

3. Enhanced compatibility with TSUPREM-4 version 6.4 can be obtained with

```
OPTION V.COMPAT=6.4
```

DEFINE

The **DEFINE** statement defines strings for substitution in subsequent input statements.

DEFINE

[<name> <body>]

Description

This statement defines the name <name> as an abbreviation for the string <body>. Any time <name> appears in an input line as a separate token (see below), <body> is substituted. This allows long, often-used sequences to be abbreviated. **DEFINE** without any parameters lists the current definitions. To undefine a <name>, use the **UNDEFINE** statement.

Format and Syntax

The **DEFINE** statement uses the following format and syntax. The defined <name> can only contain letters and digits. A name is recognized and expanded only if it is preceded and followed by one of the following separators: space, tab, newline, <, >, &, “;”, !, {, }, =, “”, \$, @, or (. To ensure that defined names are recognized in arithmetic expressions, the name should be preceded and followed by spaces, and the entire expression enclosed in parentheses. Expansion of abbreviations can be forced by preceding the defined name with the “@” character; in this case the name can also be enclosed in braces to separate it from surrounding characters. Older versions of the program used the “\$” character to force expansion; this usage is now obsolete, although it is still recognized by the program. Within a character string, you can force recognition of a defined name by enclosing the name in braces (“{” and “}”) and preceding the left brace with the “@” (or “\$”) character.

Substitution of abbreviations is inhibited by the “%” character. When “%” is encountered in an input line, expansion of abbreviations is inhibited for the rest of the line, except when forced by the “@” (or “\$”) character.

Examples

1. The following statements define and use the name *LIMITS*:

```
DEFINE LIMITS X.MIN=0.0 X.MAX=5.0 +
          Y.MIN=0.0 Y.MAX=20.0
PLOT.2D LIMITS
```

The parameter specifications for **X.MIN**, **X.MAX**, **Y.MIN**, and **Y.MAX** are substituted whenever the name *LIMITS* is encountered. Thus, the **PLOT.2D** statement is equivalent to

```
PLOT.2D X.MIN=0.0 X.MAX=5.0 Y.MIN=0.0 Y.MAX=20.0
```

- For the statements

```
DEFINE W 2.0
ECHO 1-W
ECHO 1-@W
```

the output is

```
1-W
-1
```

In the first **ECHO** statement, “-” is not recognized as a separator. In the second **ECHO** statement, the “@” forces expansion of the abbreviation *W*.

- The statements

```
DEFINE PROC C41
SAVEFILE OUT.FILE=MOS@{PROC}A.DAT
```

save the structure in file *MOSC41A.DAT*. The defined name *PROC* is inserted in the middle of a text string with the “@{ }” construct to force expansion.

Usage Notes

- Abbreviations are expanded whenever they are encountered, including in **DEFINE** and **UNDEFINE** statements. The expanded version of an abbreviation is not checked for further abbreviations, however. For example,

```
DEFINE a b
DEFINE c a
```

defines both *a* and *c* as “b,” but

```
DEFINE a b
%DEFINE c a
```

defines *a* as “b” and *c* as “a.” References to *c* produce “a,” which is *not* expanded to “b.”

- Similarly, to redefine an name, you must use the “%” character. For example,

```
DEFINE a b
DEFINE a c
```

defines *a* as “b” and *b* as “c.” To define *a* as “c,” use

```
DEFINE a b
%DEFINE a c
```

The substitution of “b” for *a* in the second line is prevented by the “%” character.

3. When undefining an abbreviation, be sure to use the “%” character on the **UNDEFINE** statement:

```
DEFINE a b
UNDEFINE a
```

does not work, because the **UNDEFINE** statement is expanded to

```
UNDEFINE b
```

and *b* is not defined. To undefine *a*, use

```
%UNDEFINE a
```

4. A line beginning with “\$” is not treated as a **COMMENT** statement if the “\$” is immediately followed by the name of a defined abbreviation. The expansion of the abbreviation is substituted. Thus, in the statements

```
DEFINE THIS 1.0
$THIS IS NOT A COMMENT
```

the value 1.0 is substituted for the name *THIS* in the second statement, giving

```
1.0 IS NOT A COMMENT
```

which results in a syntax error.

5. It is not possible to assign a null value to a name with the **DEFINE** statement.
6. Care should be used in choosing names to be defined. You normally want to avoid using names that are also valid statement or parameter names. For example, given

```
DEFINE TIME 20
DIFFUSE TIME=TIME
```

The second statement is expanded as

```
DIFFUSE 20=20
```

resulting in a syntax error.

7. For many applications it is easier to use the **ASSIGN** and **LOOP** statements instead of **DEFINE** and **FOREACH**. See [“ASSIGN” on page 3-25](#) and [“LOOP/L.END” on page 3-18](#).

UNDEFINE

The **UNDEFINE** statement undefines a previously defined abbreviation.

```
UNDEFINE
      <name>
```

Description

The **UNDEFINE** statement is used to turn *off* previously defined abbreviations. The character string *<name>* gives the name of the abbreviation to be deleted. The statement should normally be given as **%UNDEFINE**; the “%” is required to prevent substitution for abbreviations in the **UNDEFINE** statement itself.

The **UNDEFINE** statement is also useful for correcting mistakes made with the **DEFINE** statement. For more examples of the **UNDEFINE** statement, see [“DEFINE” on page 3-42](#).

Example

The statements

```
DEFINE  W  2.0
ECHO   1 - W
%UNDEFINE W
ECHO   1 - W
```

produce the output

```
-1
1 - W
```

Redefined Parameter Names

The **UNDEFINE** statement is most often needed when a parameter name has been redefined. For example, you may wish to do the following:

```
DEFINE  COMPRESS  VISCOUS
SOURCE  OXIDE1
%UNDEFINE COMPRESS
SOURCE  OXIDE2
```

This forces any **METHOD COMPRESS** statements in file *OXIDE1* to be interpreted as **METHOD VISCOUS** statements. By undefining the name *COMPRESS*, this substitution is not made when executing file *OXIDE2*.

CPULOG

The **CPULOG** statement controls the logging of CPU usage.

CPULOG

[**LOG**] [**OUT.FILE**=<c>]

| Parameter | Type | Definition |
|-----------------|-----------|--|
| LOG | logical | Enables logging of CPU usage when true; disables CPU logging when false. Default: true |
| OUT.FILE | character | The file to which the CPU log is written. Default: standard output Synonym: CPUFILE |

Description

The **CPULOG** statement instructs the program to record the amount of CPU time used in various internal operations. The information appears on the standard output and in the output listing file unless it is redirected with the **OUT.FILE** parameter. Most CPU-intensive operations report the time used.

Examples

1. The following statement enables reporting of CPU statistics to the standard output and the output listing file:

```
CPULOG LOG
```

2. The following statement enables CPU statistics reporting and stores the output in the file *timefile*.

```
CPU LOG OUT.FILE=timefile
```

Limitations

The accuracy of the times depends on the computer being used. A resolution of 1/60 seconds is typical. Some systems cannot report CPU time at all. Reported times are zero for those systems. Timing for a complete TSUPREM-4 simulation is best obtained using operating system commands, such as the **TIME** command in the UNIX C-shell.

HELP

The **HELP** statement prints a summary of statement names and parameters.

HELP

[<name>]

Description

The **HELP** statement prints a summary of statement names and parameters to the standard output. If no <name> is given, a summary of statement names is printed. If the name of a statement is given, a summary of parameters for that statement is printed. The parameter summary includes the name type, and default for each parameter.

Example

The statement

HELP DIFFUSION

prints a summary of the parameters on the **DIFFUSION** statement.

Notes

1. The default values printed by the **HELP** statement are not always helpful. This is especially true for statements that specify the values of model coefficients. These statements usually ignore parameters that are not specified rather than use the default values. The default value is likewise meaningless for parameters that must be specified on a statement.
2. Help is not always available for statements that use nonstandard syntax (i.e., special statements).

3.2 Device Structure Specification

The following statements specify the device structure used by TSUPREM-4:

| Statement Name | Description | Page |
|-------------------|--|----------------------|
| MESH | Sets grid spacing scale factor and defaults for automatic grid generation. | 3-50 |
| LINE | Specifies a grid line in a rectangular mesh. | 3-55 |
| ELIMINATE | Specifies grid lines to be removed from parts of the mesh. | 3-57 |
| BOUNDARY | Sets boundary conditions for a rectangular mesh. | 3-60 |
| REGION | Sets material types for a rectangular mesh. | 3-62 |
| INITIALIZE | Initializes a rectangular mesh or reads mesh and solution information from a file. | 3-64 |
| LOADFILE | Reads mesh and solution information from a file. | 3-68 |
| SAVEFILE | Writes mesh and solution information to a file. | 3-71 |
| STRUCTURE | Reflects, truncates, extends, or unrefines a structure. | 3-77 |
| MASK | Reads mask information from a file. | 3-82 |
| PROFILE | Reads a one-dimensional doping profile from a file. | 3-84 |
| ELECTRODE | Specifies the name and position of electrodes. | 3-87 |

MESH

The **MESH** statement specifies a grid spacing scale factor and default values for controlling automatic grid generation.

MESH

```
[ GRID . FAC = <n> ]
[ DX . MAX = <n> ] [ DX . MIN = <n> ] [ DX . RATIO = <n> ]
[ LY . SURF = <n> ] [ DY . SURF = <n> ] [ LY . ACTIV = <n> ] [ DY . ACTIV = <n> ]
[ LY . BOT = <n> ] [ DY . BOT = <n> ] [ DY . RATIO = <n> ]
[ FAST ]
```

| Parameter | Type | Definition |
|-------------------|--------|--|
| GRID . FAC | number | A factor by which all grid spacing specifications are multiplied. The scaling occurs when the spacings are used (not when they are specified.) Units: none Default: the current value |
| DX . MAX | number | The maximum grid spacing in the <i>x</i> direction (i.e., between vertical grid lines) in the default horizontal grid. This is the spacing to be used far from mask edges. This spacing is multiplied by GRID . FAC when it is used. Units: microns Default: the current value |
| DX . MIN | number | The minimum grid spacing in the <i>x</i> direction (i.e., between vertical grid lines) in the default horizontal grid. This is the spacing to be used at mask edges. This spacing is multiplied by GRID . FAC when it is used. Units: microns Default: the current value |
| DX . RATIO | number | The ratio by which grid spaces are increased from DX . MIN to DX . MAX in the default horizontal grid. Units: microns Default: the current value |
| LY . SURF | number | The depth of the surface region in the default vertical grid. Units: microns Default: the current value |
| DY . SURF | number | The grid spacing in the <i>y</i> direction (i.e., between horizontal grid lines) in the surface region of the default vertical grid. This spacing is used between <i>y</i> =0 and <i>y</i> = LY . SURF . This spacing is multiplied by GRID . FAC when it is used. Units: microns Default: the current value |
| LY . ACTIV | number | The depth of the bottom of the active region in the default vertical grid. Units: microns Default: the current value |

| Parameter | Type | Definition |
|-----------------|---------|---|
| DY.ACTIV | number | The grid spacing in the y direction (i.e., between horizontal grid lines) at the bottom of the active region in the default vertical grid. The grid spacing varies geometrically between DY.SURF at LY.SURF and DY.ACTIV at LY.ACTIV . This spacing is multiplied by GRID.FAC when it is used. Units: microns Default: the current value |
| LY.BOT | number | The depth of the bottom of the structure in the default vertical grid. Units: microns Default: the current value |
| DY.BOT | number | The grid spacing in the y direction (i.e., between horizontal grid lines) at the bottom of the structure in the default vertical grid. This spacing is multiplied by GRID.FAC when it is used. Units: microns Default: the current value |
| DY.RATIO | number | The ratio by which grid spaces are increased from DY.ACTIV to DY.BOT in the default vertical grid. Units: microns Default: the current value |
| FAST | logical | Specifies that one-dimensional simulation is used until an etch step creates a structure that varies in the x direction. The structure is automatically converted to two-dimensional status when required. Default: true |

Description

The **MESH** statement controls the automatic generation of simulation grids for TSUPREM-4. A grid is generated whenever an **INITIALIZE** statement without an **IN.FILE** parameter is processed. Automatic grid generation is used unless an appropriate set of **LINE** statements has been processed since the previous **INITIALIZE** statement was processed. Grids can be automatically generated in both the x and y directions, and automatic grid generation in one direction can be combined with manual grid specification on the other direction.

Grid Creation Methods

A simulation grid for TSUPREM-4 can be created in one of four ways:

1. It can be read from a saved structure file, using the **INITIALIZE** or **LOADFILE** statements.
2. It can be generated from user-specified **LINE**, **REGION**, **BOUNDARY**, and **ELIMINATE** statements.
3. It can be generated automatically from parameters supplied on the **MESH** statement and from information read from mask data files.

4. The last two methods can be combined, using one of the methods for generating the horizontal grid and the other method for generating the vertical grid.

If no **LINE** statements are supplied before an **INITIALIZE** statement, a grid is generated automatically. If **LINE X** statements are supplied, they are used (instead of automatic generation) for the horizontal grid; if **LINE Y** statements are supplied, they are used to generate the vertical grid.

Horizontal Grid Generation

The automatic generation of the horizontal grid depends on whether mask information has been read with the **MASK** statement.

- If no mask information has been read, the default horizontal grid consists of two (vertical) grid lines located at $y=0.0$ and $y=1.0$. This produces a quasi one-dimensional simulation.
- If mask information is available, it is used with the **DX.MIN**, **DX.MAX**, and **DX.RATIO** parameters to automatically generate a horizontal grid.

The horizontal grid has a spacing between (vertical) grid lines of approximately **DX.MIN** at mask edges, expanding to a spacing of **DX.MAX** at distances far from any mask edge. The **DX.RATIO** parameter specifies the rate at which the spacing increases away from mask edges. The minimum and maximum grid locations are set to the minimum and maximum x coordinates for which mask information is supplied in the mask data file(s).

For a complete description of the grid generation process, see [Chapter 2, “Automatic Grid Generation in the X Direction”](#) on page 2-5.

Vertical Grid Generation

The automatically generated vertical grid is controlled by the various **LY** and **DY** parameters.

- **LY.SURF**, **LY.ACTIV**, and **LY.BOT** specify the locations of the bottom of the surface region, active region, and structure, respectively.
- **DY.SURF**, **DY.ACTIV**, and **DY.BOT** specify the grid spacings at these locations.

Between the top of the structure and **LY.SURF**, the grid spacing **DY.SURF** is used. The spacing increases geometrically (i.e., with a constant ratio) from **DY.SURF** at **LY.SURF** to **DY.ACTIV** at **LY.ACTIV**. The spacing expands to **DY.BOT** below **LY.ACTIV**. The **DY.RATIO** parameter specifies the rate at which the spacing increases below **DY.ACTIV**.

In a automatically generated vertical grid, vertical grid lines deeper than **LY.ACTIV** are eliminated, to increase the speed of the simulation. It is assumed that only point defects diffuse below this depth and that the point defect profiles

are approximately constant in the horizontal direction, so that the low horizontal resolution does not cause significant loss of accuracy in the simulation.

For a complete description of the grid generation process see [Chapter 2, “Automatic Grid Generation in the Y Direction”](#) on page 2-6.

Scaling the Grid Spacing

The **GRID.FAC** parameter multiplies all grid spacing specifications, including those on the **MESH**, **LINE**, **DEPOSITION**, and **EPITAXY** statements. The default values of all grid spacings are set so that **GRID.FAC**=1.0 produces a moderately fine grid, useful for reasonably, accurate simulations. For faster simulations, **GRID.FAC** should be given a value greater than 1.0. For more accurate simulations, **GRID.FAC** should be reduced as needed.

The **GRID.FAC** parameter modifies the values of other parameters in the following ways:

- The **DX.MIN**, **DX.MAX**, **DY.SURF**, **DY.ACTIV**, and **DY.BOT** parameters on the **MESH** statement are multiplied by **GRID.FAC**.
- The **DY** and **ARC.SPAC** parameters on the **DEPOSITION** and **EPITAXY** statements are multiplied by **GRID.FAC**.
- The **SPACES** parameter on the **DEPOSITION** and **EPITAXY** statements is divided by **GRID.FAC**.
- The **DY.OXIDE** on the **METHOD** statement and the **DY.DEFAU** parameter on the **MATERIAL** statement are multiplied by **GRID.FAC** (the modification takes place when the parameters are used, not when they are specified).



Note:

*Moderation should be used in adjusting **GRID.FAC**—reducing the value from 1.0 to 0.1 increases the grid density by a factor of 10 in each direction, increasing the node count by a factor of 100 and the simulation time by a factor of more than 100.*

1D Mode

Normally, when a mesh is generated or read from a file, it is examined to determine whether there is any variation of the structure or solution values in the x direction. If there is no variation, then simulation proceeds in 1D mode. By setting the **FAST** parameter false, the use of 1D mode is disabled. If 1D mode is already in use, the structure is converted to 2D.

Examples

1. The following **MESH** statements could be used in the *s4init* file to set the default parameters for automatic grid generation:

```
MESH DX.MIN=0.1 DX.MAX=0.4 DX.RATIO=1.5  
MESH LY.SURF=0.1 DY.SURF=0.03 LY.ACTIV=4.0 +  
DY.ACTIV=0.3 LY.BOT=200 DY.BOT=100 +  
DY.RATIO=1.5
```

2. The following statement cuts all grid spacings to one half their specified value, giving twice the specified grid density in the horizontal and vertical directions:

```
MESH GRID.FAC=0.5
```

LINE

The **LINE** statement specifies a horizontal or vertical mesh line in a non-uniform rectangular grid.

LINE

```
{X | Y} LOCATION=<n> [SPACING=<n>] [TAG=<c>]
```

| Parameter | Type | Definition |
|-----------------|-----------|--|
| X | logical | If true, then LOCATION specifies the <i>x</i> coordinate of a vertical grid line. Default: false |
| Y | logical | If true, then LOCATION specifies the <i>y</i> coordinate of a horizontal grid line. Default: false |
| LOCATION | number | The <i>y</i> coordinate of a horizontal grid line or the <i>x</i> coordinate of a vertical grid line. Units: microns Default: none |
| SPACING | number | The local grid spacing. Units: microns Default: none |
| TAG | character | A name used to refer to this grid line on a BOUNDARY or REGION statement. Default: none |

Description

This statement specifies the position and spacing of one line in a rectangular mesh. A complete mesh specification consists of a group of **LINE** statements followed by **ELIMINATE** statements (optional), **REGION** and **BOUNDARY** statements (optional), and an **INITIALIZE** statement.

TSUPREM-4 uses an inverted Cartesian coordinate system, with *x* increasing from left to right and *y* increasing from the top surface of the structure into the substrate.

Placing Grid Lines

Grid lines are placed at the locations specified on **LINE** statements. Additional lines are added automatically so that the final grid spacing varies geometrically between the **SPACING** values at the user-specified lines. The spacing ratio between adjacent pairs of grid lines is guided by the value of the **RATIO** parameter on the **INITIALIZE** statement. The final grid spacing may be slightly larger than that specified on the **LINE** statement because of the need for an integral

number of spaces; it may be much smaller than specified to satisfy constraints imposed by neighboring grid lines and the **RATIO** parameter. A complete description of the grid setup process is given in [Chapter 2, “Explicit Specification of Grid Structure”](#) on page 2-3.

Example

The following statements include three user-specified vertical grid lines and two user-specified horizontal grid lines:

```

LINE X LOCATION=0 SPACING=1
LINE X LOCATION=1 SPACING=0.1
LINE X LOCATION=2 SPACING=1

LINE Y LOCATION=0 SPACING=0.02
LINE Y LOCATION=3 SPACING=0.5

```

After processing with **RATIO=1.5**, TSUPREM-4 produces a mesh with vertical grid lines at 0.0, 0.42, 0.69, 0.88, 1.0, 1.12, 1.31, 1.58, and 2.0. Around the center, the spacing is 0.12, approximately what was requested. At each edge, the spacing is 0.42, because that is as coarse as it could get without exceeding the allowed ratio between adjacent grid spaces. If the allowed ratio were 9, then you would get one space of 0.9 microns and one space of 0.1 micron on each side of the center line.

Additional Notes

Below are some additional notes on the **LINE** statement.

Structure Depth and Point Defect Models

When the **PD.TRANS** or **PD.FULL** point defect model is to be used, the structure should be deep to accommodate the deep diffusion of point defects. A depth of 50-200 microns is suggested. **ELIMINATE** statements can be used to eliminate all but two vertical grid lines deep in the structure. When **ELIMINATE** statements are used with a coarse vertical grid spacing, the computational overhead of using a deep structure can be made negligible.

Maximum Number of Nodes and Grid Lines

The program can handle up to 40,000 nodes; this includes one node per grid point, plus one node for each exposed point and another node for each point on a boundary between two materials. Thus, the grid must be smaller than 40,000 points. In addition, the initial grid must not have more than 1000 grid lines in either the *x* or *y* direction.

Default Regions and Boundaries

No tags are required if the default regions and boundary conditions are to be used, i.e., if the initial structure consists entirely of silicon with only the top surface exposed.

ELIMINATE

The **ELIMINATE** statement eliminates mesh nodes along lines in a grid structure over a specified rectangular region.

ELIMINATE

```
{ROWS | COLUMNS}
[X.MIN=<n>] [X.MAX=<n>] [Y.MIN=<n>] [Y.MAX=<n>]
```

| Parameter | Type | Definition |
|----------------|---------|---|
| ROWS | logical | Specifies that horizontal lines of nodes are eliminated. Default: false Synonym: X.DIREC |
| COLUMNS | logical | Specifies that vertical lines of nodes are eliminated. Default: false Synonym: Y.DIREC |
| X.MIN | number | The minimum horizontal location of the rectangular region over which nodes are eliminated. Units: microns Default: the minimum horizontal location of the device structure Synonym: X.LOW |
| X.MAX | number | The maximum horizontal location of the rectangular region over which nodes are eliminated. Units: microns Default: the maximum horizontal location of the device structure Synonym: X.HIGH |
| Y.MIN | number | The minimum vertical location of the rectangular region over which nodes are eliminated. Units: microns Default: the minimum vertical location of the device structure Synonym: Y.LOW |
| Y.MAX | number | The maximum vertical location of the rectangular region over which nodes are eliminated. Units: microns Default: the maximum vertical location of the device structure Synonym: Y.HIGH |

Description

The **ELIMINATE** statement is used to eliminate nodes in regions of the device structure where the grid would otherwise be more dense than necessary. The **ROWS** parameter specifies that every second row of nodes within the specified

region is to be removed. Similarly, the **COLUMNS** parameter specifies that every second column of nodes within the specified region is to be removed.

The **ELIMINATE** statement can only be used after the **LINE** statements and before the **INITIALIZE** statement.



Note:

*Elimination of grid lines takes place during initial structure generation. The **ELIMINATE** statement cannot be used to eliminate grid lines after the **INITIALIZE** statement has been processed.*

It is frequently a good idea to plot and examine the initial grid after the **LINE X**, **LINE Y**, **ELIMINATE**, and **INITIALIZE** statements to verify the desired grid structure before proceeding.

Reducing Grid Nodes

The **ELIMINATE** statement can be very useful for reducing the number of grid nodes (and hence execution time), especially for large structures. In particular, you might use **LINE X** and **LINE Y** statements to define a large structure that would exceed the limit of 40,000 nodes, then eliminate enough rows and columns to bring the final number of nodes under 40,000.

Overlapping Regions

If you get an error message stating that one of the **ELIMINATE** statements is causing a mesh generation error, it is due to overlapping or adjacent eliminate regions. For best results when performing multiple eliminations over a structure, either:

1. Do the multiple eliminations over identical regions of the structure, or
2. Avoid overlapping or adjacent eliminate regions; i.e., separate the regions by at least one grid line.

If you must have overlapping eliminate regions, best results are obtained if the **ELIMINATE** statements define successively smaller regions. That is, start with the largest region, then eliminate additional nodes in subsets of that region. If you need to eliminate over partially overlapping regions, best results are obtained if you minimize the amount of overlap.

Examples

1. The following statement eliminates every other row of nodes over the entire structure:

```
ELIMINATE ROWS
```

The same result could be obtained by changing the locations and grid spacings on the **LINE Y** statements.

2. The following statement eliminates every other column of nodes over the full width of the structure, but only for values of y greater than 4.5 microns:

```
ELIMINATE COLUMNS Y.MIN=4.5
```

3. The following statements perform multiple eliminations of rows and columns over the same region of the structure:

```
ELIMINATE ROWS X.MAX=3 Y.MAX=2.2
```

```
ELIMINATE ROWS X.MAX=3 Y.MAX=2.2
```

```
ELIMINATE COLUMNS X.MAX=3 Y.MAX=2.2
```

```
ELIMINATE COLUMNS X.MAX=3 Y.MAX=2.2
```

4. The following statements eliminate columns in two non-overlapping, non-adjacent regions of the structure:

```
ELIMINATE COLUMNS X.MIN=1 X.MAX=3
```

```
ELIMINATE COLUMNS X.MIN=7 X.MAX=9
```

Each region extends from the top to the bottom of the structure.

BOUNDARY

The **BOUNDARY** statement specifies a boundary condition along an edge of the structure.

BOUNDARY

```
{REFLECTI | EXPOSED}
XLO=<c> XHI=<c> YLO=<c> YHI=<c>
```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| REFLECTI | logical | The specified boundary is reflecting. Default: false |
| EXPOSED | logical | The specified boundary is exposed. Default: false |
| XLO | character | The tag corresponding to the left edge of the boundary. Default: none |
| XHI | character | The tag corresponding to the right edge of the boundary. Default: none |
| YLO | character | The tag corresponding to the top edge of the boundary. Default: none |
| YHI | character | The tag corresponding to the bottom edge of the boundary. Default: none |

Description

This statement specifies the boundary conditions that apply at the surfaces in a rectangular mesh. Two surface types are recognized: exposed and reflecting.

Exposed surfaces normally correspond to the top of the wafer. Deposition, oxidation, and out-diffusion occur at exposed surfaces. Impurity predeposition also happens at exposed surfaces, as do defect recombination and generation. The default boundary condition for the sides and the back of the structure is reflecting, while the default for the top surface is exposed. Thus the **BOUNDARY** statement is not required for most simulations. The tags specified with the **XLO**, **XHI**, **YLO**, and **YHI** parameters correspond to tags specified on the **LINE** statements used to define the mesh. **XLO**, **XHI**, **YLO**, and **YHI** must specify one entire edge of the structure. It is not permissible to specify part of an edge, a region of nonzero area, or a line in the interior of the structure.

Limitations

At present, the **BOUNDARY** statement is only useful in special situations, due to the following limitations in the various processing steps:

- **DEPOSITION** only works on the top and right side (when exposed) of the structure, and may fail if the bottom of the structure is exposed.
- **ETCH** with the **TRAPEZOI** and **OLD.DRY** options assumes that the exposed surface is etched from the top. Etching with the **TRAPEZOI** and **OLD.DRY** options may fail if the sides or bottom of the structure are exposed.
- Ion implantation always occurs on the top surface.
- **DIFFUSION** works correctly on all exposed surfaces, but the analytical and **VERTICAL** oxidation models assume that oxidation only occurs at the top surface.

The **BOUNDARY** statement is only used in setting up the initial structure. It cannot be used to change a boundary condition after the structure has been initialized.

Example

The following statement specifies that the top of the mesh is the exposed surface:

```
BOUNDARY EXPOSED XLO=left XHI=right YLO=surf YHI=surf
```

The tags **left**, **right**, and **surf** must have been previously specified on the **LINE** statements used to define the mesh. See the description of the **LINE** statement on [page 3-55](#).

REGION

The **REGION** statement specifies the material type of a mesh region.

REGION

```
{ MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
  | PHOTORES | ALUMINUM
}
```

XLO=<c> XHI=<c> YLO=<c> YHI=<c>

| Parameter | Type | Definition |
|-----------------|-----------|---|
| MATERIAL | character | The specified region is defined to be of the named material. Default: none |
| SILICON | logical | The specified region is defined to be silicon. Default: false |
| OXIDE | logical | The specified region is defined to be oxide. Default: false |
| OXYNITRI | logical | The specified region is defined to be oxynitride. Default: false |
| NITRIDE | logical | The specified region is defined to be nitride. Default: false |
| POLYSILI | logical | The specified region is defined to be polysilicon. Default: false |
| PHOTORES | logical | The specified region is defined to be photoresist. Default: false |
| ALUMINUM | logical | The specified region is defined to be aluminum. Default: false |
| XLO | character | The tag specifying the left edge of the region to be defined. Default: none |
| XHI | character | The tag specifying the right edge of the region to be defined. Default: none |
| YLO | character | The tag specifying the top edge of the region to be defined. Default: none |
| YHI | character | The tag specifying the bottom edge of the region to be defined. Default: none |

Description

This statement specifies the material type of rectangular subregions when generating a rectangular mesh. **REGION** statements appear after **LINE** statements but before the **INITIALIZE** statement. The default material is silicon; if no **REGION** statements are specified, the entire initial structure is assumed to be silicon.

The tags referenced on **REGION** statements must be defined on the preceding **LINE** statements.

Example

The following statement defines the region bounded by the x coordinates **left** and **right** and the y coordinates **surf** and **back** to be silicon. See the description of the **LINE** statement on [page 3-55](#).

```
REGION SILICON XLO=left XHI=right YLO=surf YHI=back
```

INITIALIZE

The **INITIALIZE** statement sets up the initial structure for a simulation.

INITIALIZE

```
{ ( IN.FILE=<c> { ( [SCALE=<n>] [FLIP.Y] ) | TIF } )
| ( [WIDTH=<n> [DX=<n>]] [ {<111> | <110> | <100> | ORIENTAT=<n>} ]
  [ {ROT.SUB=<n> | X.ORIENT=<n>} ] [RATIO=<n>] [LINE.DAT]
)
}
[ IMPURITY=<c> { I.CONC=<n> | I.RESIST=<n> } ]
[ MATERIAL=<c> ] [ANTIMONY=<n>] [ARSENIC=<n>] [BORON=<n>] [PHOSPHOR=<n>]
[ {CONCENTR | RESISTIV} ]
```

| Parameter | Type | Definition |
|--------------------|-----------|---|
| IN.FILE | character | Name of a saved structure file. If this parameter is omitted, a rectangular grid is generated, using previously specified LINE , ELIMINATE , REGION , and BOUNDARY statements. Default: none Synonym: INFILE |
| SCALE | number | The mesh read in from IN.FILE is scaled by this factor. Units: none Default: 1.0 |
| FLIP.Y | logical | Specifies that the input structure is to be reflected about $y=0$. Default: false |
| TIF | logical | Specifies that the input file is a TIF (Technology Interchange Format) file. Default: false |
| WIDTH | number | The width of the initial structure. (Only used if no LINE X statements are specified.) Units: microns Default: the width of the MASK information, if any, or 1.0 |
| DX | number | The grid spacing to use in the x direction. Units: microns Default: the current value of DX.MAX from the MESH statement |
| <111> | logical | Specifies that the crystalline orientation of the silicon substrate is <111> . Default: false |
| <110> | logical | Specifies that the crystalline orientation of the silicon substrate is <110> . Default: false |
| <100> | logical | Specifies that the crystalline orientation of the silicon substrate is <100> . Default: True, if no other orientation is specified. |

| Parameter | Type | Definition |
|------------------|-----------|--|
| ORIENTAT | number | The crystalline orientation of the silicon substrate. Only 100, 110, and 111 are recognized. Units: none Default: 100 Synonym: Y. ORIENT |
| ROT. SUB | number | The rotation of the substrate about the y axis. The reference orientation (ROT. SUB=0) is defined such that the x axis points in a <110> direction. Units: degrees Default: 0.0 |
| X. ORIENT | number | The crystalline orientation of the x axis. This must be a 3-digit integer value. Units: none Default: 110 |
| RATIO | number | The maximum ratio of adjacent grid spacings to be used in generating a grid. Units: none Default: 1.5 Synonym: INTERVAL |
| LINE. DAT | logical | Specifies that the location of each x and y grid line be listed on the standard output and in the output listing file. Default: false |
| IMPURITY | character | The name of the impurity with which the initial structure is doped. Default: none |
| I. CONC | number | The concentration of the specified IMPURITY in the initial structure. Units: atoms/cm ³ Default: none |
| I. RESIST | number | The resistivity of the initial structure. Units: ohm-cm Default: none |
| MATERIAL | character | Specifies the material of the initial structure. Default: SILICON |
| ANTIMONY | number | The uniform concentration or resistivity of antimony in the initial structure. Units: atoms/cm ³ or ohm-cm Default: 0.0 Synonym: SB |
| ARSENIC | number | The uniform concentration or resistivity of arsenic in the initial structure. Units: atoms/cm ³ or ohm-cm Default: 0.0 Synonym: AS |
| BORON | number | The uniform concentration or resistivity of boron in the initial structure. Units: atoms/cm ³ or ohm-cm Default: 0.0 Synonym: B |

| Parameter | Type | Definition |
|-----------|---------|---|
| PHOSPHOR | number | The uniform concentration or resistivity of phosphorus in the initial structure. Units: atoms/cm ³ or ohm-cm Default: 0.0 Synonym: P |
| CONCENTR | logical | Specifies that the impurity concentration in the initial structure is given. Default: true |
| RESISTIV | logical | Specifies that the impurity resistivity in the initial structure is given. Default: false |

Description

The **INITIALIZE** statement sets up the mesh from either a rectangular specification or from a previously saved structure file. This statement also initializes the background doping concentrations of the impurities specified.

Mesh Generation

If **IN.FILE** is not specified, a rectangular mesh is generated. If **LINE** statements have been specified for the x or y direction, they are used along with any **ELIMINATE**, **BOUNDARY**, and **REGION** statements and the value of the **RATIO** parameter to generate the mesh in that direction. If no **LINE** statements are specified for the y direction, a default y mesh is generated. If no **LINE** statements are specified for the x direction, and **WIDTH** is specified, a mesh of the requested width is generated, with spacing given by **DX**. If no **LINE X** statements are given and **WIDTH** is not specified, the width is taken from mask information read with a preceding **MASK** statement, if any. Otherwise, a one-micron wide structure with a grid spacing of one micron (i.e., one grid space wide) is generated. See [Chapter 2, “Grid Structure” on page 2-2](#) for a complete description of the mesh generation process. The locations of the generated grid lines are listed if **LINE.DAT** is true.

Previously Saved Structure Files

A mesh read from a file must be in either TSUPREM-4 format or TIF (Technology Interchange Format). Meshes read from TSUPREM-4 files can be scaled or flipped about $y=0$ by specifying the **SCALE** or **FLIP.Y** parameters, respectively.

When a structure is read from a file, the last processing temperature as well as the choice of silicon substrate orientation, oxidation model, and point defect model are automatically set through information stored in the file. You need not respecify these parameters after reading in a structure file at the start of a simulation.

Crystalline Orientation

The crystalline orientation of any silicon regions in a generated structure can be specified by $\langle 100 \rangle$, $\langle 110 \rangle$, or $\langle 111 \rangle$ parameters. The **ORIENTAT** parameter is also accepted, for compatibility with older versions of the program. The specified

orientation is used for all single-crystal silicon regions in the simulation, whether present in the initial structure or deposited later. By default, the x axis points in a $\langle 110 \rangle$ direction. This can be changed by specifying a rotation about the y axis (**ROT.SUB**) or the crystal orientation of the x axis (**X.ORIENT**).

The orientation parameters do not apply when reading a structure from a file; the orientation of the saved structure is used instead.

Specifying Initial Doping

The **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHOR** parameters can be used to specify the initial resistivity or impurity concentrations in the structure. Any combination of these parameters can be specified if impurity concentrations are given (**CONCENTR** true), but only one impurity can be specified if the resistivity is given (**RESISTIV** true). An impurity can also be specified by name with the **IMPURITY** parameter; **I.CONC** or **I.RESIST** are used to specify the concentration or resistivity, respectively, associated with the named impurity.

The resistivity is calculated from tables of mobility as a function of doping concentration. These tables are described in [Chapter 2, “Initial Impurity Concentration”](#) on page 2-12.

Although the source of the grid (read or generated) and the specification of doping are independent, the doping specification is normally used when a grid is generated but not when a grid is read from a file.

Examples

1. The following statement reads in a previously saved structure in file *oldstr*:

```
INITIALIZE IN.FILE=oldstr
```

2. The following statement generates a rectangular mesh and initializes the structure with a boron doping of $10^{15}/\text{cm}^3$:

```
INIT <111> X.ORIENT=211 BORON=1e15
```

The orientation of single-crystal silicon regions are $\langle 111 \rangle$, while the x axis points in a $\langle 211 \rangle$ direction.

3. The following statement generates a mesh and initializes the structure to contain arsenic with resistivity of 20 ohm-cm:

```
INIT IMPURITY=arsenic I.RESIST=20
```



Note:

*The conversion from a resistivity to a concentration is based on Masetti's mobility table, while the calculation of electrical characteristics in the **ELECTRICAL** statement uses the same mobility table as in Medici. Thus, the sheet resistances of the initial structure given by the **EXTRACT** statement do not correspond exactly to the resistivity specified on the **INITIALIZE** statement.*

LOADFILE

The **LOADFILE** statement reads mesh and solution information from a file.

LOADFILE

```
IN.FILE=<c>
{ ( [SCALE=<n>] [FLIP.Y] ) | TIF | DEPICT }
```

| Parameter | Type | Definition |
|----------------|-----------|---|
| IN.FILE | character | The identifier of a structure file to be read. Default: none Synonyms: INFILE |
| SCALE | number | A scaling factor to be applied to the mesh when reading TSUPREM-4 structure files. All coordinate values are multiplied by this factor as they are read. Units: none Default: 1.0 |
| FLIP.Y | logical | Specifies that the structure should be reflected about $y=0$ when reading TSUPREM-4 structure files. All y coordinates are multiplied by -1 as they are read. Default: false |
| TIF | logical | Specifies that the input file is a TIF (Technology Interchange Format) file. Default: false |
| DEPICT | logical | Specifies that the input file is a formatted file written by <i>Avant!</i> TCAD's Taurus-Lithography (formerly Depict). Default: false |

Description

The **LOADFILE** statement reads a mesh and solution from a file. Either TSUPREM-4 or TIF structure files (created with the **SAVEFILE** statement) or Taurus-Lithography structure files can be read. TSUPREM-4 structure files can be scaled or flipped about the x axis ($y=0$) during reading.

TSUPREM-4 Files

The silicon substrate orientation, last processing temperature, current oxidation model, and current point defect model are saved in TSUPREM-4 structure files, and are automatically restored when the file is read. There is no need to respecify these parameters after reading in a structure. (This does not apply to versions of TSUPREM-4 prior to version 9035; versions older than 9035 only saved the substrate orientation.)

Older Versions

Files saved with older versions of TSUPREM-4 can be loaded by newer versions of the program. Files saved with version 5.1 and newer of TSUPREM-4 cannot be read by older versions of the program.

In versions of TSUPREM-4 prior to version 5.1, the **STRUCTURE** statement is used to load structure files. Starting with version 5.1, the **LOADFILE** statement should be used instead. The **IN.FILE**, **SCALE**, **FLIP.Y**, and **DEPICT** parameters are still accepted on the **STRUCTURE** statement, however, for compatibility with older TSUPREM-4 input files.

User-Defined Materials and Impurities

Any user-defined materials and impurities referenced in a structure file should be defined before the file is loaded. If a file containing undefined materials or impurities is loaded, a warning is issued for each undefined material or impurity encountered. The material or impurity is defined, but its properties are not set. Further simulation using the material or impurity may fail unless the properties are set.

Taurus-Lithography Files

Files produced by Taurus-Lithography do not contain the complete mesh and solution information required by TSUPREM-4. Rather, they are used to update an existing structure with the results of a Taurus-Lithography simulation. Thus, the usual sequence for interfacing with Taurus-Lithography is:

1. Generate a structure in TSUPREM-4, either from an initial mesh definition or by reading a saved structure.
2. Save the structure in both TSUPREM-4 and Taurus-Lithography formats. This requires two **SAVEFILE** statements and two output files.
3. Use Taurus-Lithography to process the structure, starting with the saved Taurus-Lithography-format file. Save the results in a new Taurus-Lithography-format file.
4. Read the saved TSUPREM-4-format file into TSUPREM-4 (with an **INITIALIZE** or **LOADFILE** statement).
5. Read the new Taurus-Lithography-format file into TSUPREM-4 (with a **LOADFILE** statement).

Examples

1. The following statement reads in a previously saved structure from file *savestr*:

```
LOADFILE IN.FILE=savestr
```

2. The following statements save a structure to be processed further with Taurus-Lithography:

```
SAVEFILE OUT.FILE=STRTS4  
SAVEFILE OUT.FILE=STRTODEP DEPICT
```

Taurus-Lithography can read the file *STRTODEP*. The structure file *STRTS4* is needed when reading the results produced by Taurus-Lithography. If Taurus-Lithography stored its results in file *STRFRDEP*, they could be read into TSUPREM-4 with the statements

```
INITIALIZE IN.FILE=STRTS4  
LOADFILE IN.FILE=STRFRDEP DEPICT
```

3. The statements

```
LOADFILE IN.FILE=savestr
```

and

```
INITIALIZE IN.FILE=savestr
```

are equivalent, except that the program recognizes that an initial structure has been set up in the second case. An **INITIALIZE** statement must be given before any processing or output statement can be processed.

SAVEFILE

The **SAVEFILE** statement writes mesh and solution information to a file.

SAVEFILE

```

OUT.FILE=<c> [TEMPERAT=<n>]
{ ( [SCALE=<n>] [FLIP.Y] [ACTIVE] )
  | (TIF [TIF.VERS=<c>]) | DEPICT
  | ( MEDICI [POLY.ELE] [ELEC.BOT] ] )
  | ( MINIMOS5 X.MASK.S=<n>
    { HALF.DEV | ( FULL.DEV X.MASK.D=<n> [X.CHANNE=<n>] ) }
    [X.MIN=<n>] [X.MAX=<n>] [Y.MIN=<n>] [Y.MAX=<n>]
    [DX.MIN=<n>] [DY.MIN=<n>]
  )
  | ( WAVE [ACTIVE] [CHEMICAL] [DEFECT] [OXID] [MISC] )
}
    
```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| OUT.FILE | character | The identifier of the structure file to be written. Default: none Synonym: OUTFILE |
| TEMPERAT | number | The temperature used for evaluating active impurity concentrations. Units: °Celsius Default: the last processing temperature specified or 800°C |
| SCALE | number | A scaling factor to be applied to the mesh when writing TSUPREM-4 structure files. All coordinate values saved in the file are multiplied by this value. The SCALE parameter does not affect the structure used by subsequent simulation steps. Units: none Default: 1.0 |
| FLIP.Y | logical | Specifies that the structure should be reflected about y=0 when writing TSUPREM-4 structure files. All y coordinates are multiplied by -1 as they are written to the file. FLIP.Y does not affect the structure used by subsequent simulation steps. Default: false |
| TIF | logical | Specifies that the output file be saved as a TIF (Technology Interchange Format) file. Default: false |
| TIF.VERS | character | The version of TIF to be used for saving the file. The default is to use the latest version of TIF; a value of 0 produces files compatible with version 6.0 of TSUPREM-4. Units: none Default: 1.2.0 |

| Parameter | Type | Definition |
|-----------------|---------|--|
| DEPICT | logical | Specifies that the output file is a formatted file that can be read by <i>Avant!</i> TCAD's Taurus-Lithography programs. Default: false |
| MEDICI | logical | Specifies that the saved output file is a formatted file that can be read by the Medici device simulator. The output file can also be read by older versions of TMA PISCES-2B and by other versions of PISCES. Default: false Synonym: PISCES |
| POLY.ELE | logical | Specifies that polysilicon regions should be converted to electrodes in the Medici output file. Default: false |
| ELEC.BOT | logical | Specifies that an electrode should be placed along the backside of the structure in the Medici output file. Default: false |
| MINIMOS5 | logical | Specifies that the saved output file contains a two-dimensional doping profile that can be read by MINIMOS 5. Default: false |
| X.MASK.S | number | The <i>x</i> coordinate of the mask edge in the source area of the MINIMOS 5 simulation region. MINIMOS 5 interprets this coordinate as the left edge of the gate electrode. Units: microns Default: none |
| HALF.DEV | logical | Specifies that the MINIMOS 5 simulation region includes only the source area of the device. Either FULL.DEV or HALF.DEV must be specified if MINIMOS5 is specified. Default: false |
| FULL.DEV | logical | Specifies that the MINIMOS 5 simulation region includes both the source and drain areas of the device. Either FULL.DEV or HALF.DEV must be specified if MINIMOS5 is specified. Default: false |
| X.MASK.D | number | The <i>x</i> coordinate of the mask edge in the drain area of the MINIMOS 5 simulation region. MINIMOS 5 interprets this coordinate as the right edge of the gate electrode. X.MASK.D must be specified if FULL.DEV is specified; it must not be specified if HALF.DEV is specified. Units: microns Default: none |
| X.CHANNE | number | The <i>x</i> coordinate of the center of the channel of the MINIMOS 5 simulation region. Units: microns Default: $(X.MIN + X.MAX)/2$ if FULL.DEV is specified; not applicable if HALF.DEV is specified |
| X.MIN | number | The <i>x</i> coordinate of the left edge of the MINIMOS 5 simulation region. Units: microns Default: left edge of the TSUPREM-4 simulation region |

| Parameter | Type | Definition |
|-----------------|---------|---|
| X . MAX | number | The <i>x</i> coordinate of the right edge of the MINIMOS 5 simulation region. Units: microns Default: right edge of the TSUPREM-4 simulation region |
| Y . MIN | number | The <i>y</i> coordinate of the top edge of the MINIMOS 5 simulation region. MINIMOS 5 interprets this coordinate as the gate oxide/silicon interface. Units: microns Default: 0.0 |
| Y . MAX | number | The <i>y</i> coordinate of the bottom edge of the MINIMOS 5 simulation region. Units: microns Default: bottom edge of the TSUPREM-4 simulation region |
| DX . MIN | number | The minimum spacing in the <i>x</i> direction used to specify the doping profiles in the output file. Units: microns Default: $\min((X . MAX - X . MIN)/80, 0.01)$ |
| DY . MIN | number | The minimum spacing in the <i>y</i> direction used to specify the doping profiles in the output file. Units: microns Default: $\min((Y . MAX - Y . MIN)/80, 0.01)$ |
| WAVE | logical | Specifies that the output file is a formatted file in Wavefront Technologies' <i>wave</i> file format. These files can be read by Wavefront Technologies' Data Visualizer program. Default: false |
| ACTIVE | logical | Specifies that active impurity concentrations are to be saved in TSUPREM-4 and WAVE output files. Default: true |
| CHEMICAL | logical | Specifies that chemical impurity concentrations are to be included in the WAVE output file. Default: false |
| DEFECT | logical | Specifies that point defect concentrations are to be included in the WAVE output file. Default: false |
| OXID | logical | Specifies that oxidant concentrations, oxidation flow rates, and stresses (if available) are to be included in the WAVE output file. Default: false Synonym: STRESS |
| MISC | logical | Specifies that miscellaneous solution values are to be included in the WAVE output file. At present, this includes diffusivities of impurities and point defects. Default: false |

Description

The **SAVEFILE** statement writes mesh and solution information into a file, in one of several formats. If no format is specified, a TSUPREM-4 structure file is created. TSUPREM-4 structure files can be read with the **LOADFILE** or **INITIALIZE** statements. The mesh can be scaled or flipped about the x axis as the structure is written or when it is read. Scaling and flipping during writing only affect the saved structure, and do not affect the structure used by subsequent simulation steps.

TSUPREM-4 Files

The silicon substrate orientation, last processing temperature, current oxidation model, and current point defect model are saved in TSUPREM-4 structure files, and are automatically restored when the file is read. There is no need to respecify these parameters after reading in a structure. (This does not apply to versions of TSUPREM-4 prior to version 9035; versions older than 9035 only saved the substrate orientation.)

Older Versions

Files saved with older versions of TSUPREM-4 can be loaded by newer versions of the program. Files saved with version 5.1 and newer of TSUPREM-4 cannot be read by older versions of the program.

In versions of TSUPREM-4 prior to version 5.1, the **STRUCTURE** statement is used to load structure files. Starting with version 5.1, the **LOADFILE** statement should be used instead. The **IN.FILE**, **SCALE**, **FLIP.Y**, and **DEPICT** parameters are still accepted on the **STRUCTURE** statement, however, for compatibility with older TSUPREM-4 input files.

Effective in version 5.2 of TSUPREM-4, active impurity concentrations are saved by default in TSUPREM-4 structure files. Structure files without the active impurity concentrations can be produced by specifying **^ACTIVE** on the **SAVEFILE** statement. This is necessary if the structure files are to be read by older (prior to 5.2) versions of TSUPREM-4 or by other programs that cannot accept the active concentration information.

TIF Files

The **TIF** parameter specifies that the file should be saved as a TIF (Technology Interchange Format) file. The version of TIF can be specified with the **TIF.VERS** parameter; newer products use version 1.2.0, while older products (including version 6.0 of TSUPREM-4) use version 1.00 or version 0 (which are equivalent, as far as TSUPREM-4 is concerned).

Correct writing of a user-defined material or impurity to a TIF file requires that a **TIF.NAME** be specified when the material or impurity is defined; **MD.INDEX** must also be specified for materials. Before other programs can read the saved TIF

file, an entry corresponding to the TIF name must be added to the appropriate database (*mat.dbs* for materials, or *sol.dbs* for impurities).



Note:

*Versions 6.1 and later of TSUPREM-4 can read TIF files created by version 6.0, but version 6.0 cannot read TIF files created by versions 6.1 and later unless **TIF.VERS =0** is specified when the file is written.*

Medici Files

The **MEDICI** parameter creates an output file that can be read by the Medici device simulator. **MEDICI** structures can also be read by older versions of TMA PISCES-2B and by other versions of PISCES. The **POLY.ELE** and **ELEC.BOT** parameters are not needed when creating files for *Avant!* TCAD's device simulators, because these simulators allow the treatment of polysilicon and backside contacts to be specified in a **MESH** statement.

Correct writing of a user-defined material to a Medici file requires that **MD.INDEX** be specified when the material is defined (see "MATERIAL" on page 3-228).

Taurus-Lithography Files

The **DEPICT** parameter allows you to create files to be read by *Avant!* TCAD's Taurus-Lithography program.

Files produced by Taurus-Lithography do not contain the complete mesh and solution information required by TSUPREM-4. Rather, they are used to update an existing structure with the results of a Taurus-Lithography simulation. Thus, the usual sequence for interfacing with Taurus-Lithography is:

1. Generate a structure in TSUPREM-4, either from an initial mesh definition or by reading a saved structure.
2. Save the structure in both TSUPREM-4 and Taurus-Lithography formats. This requires two **SAVEFILE** statements and two output files.
3. Use Taurus-Lithography to process the structure, starting with the saved Taurus-Lithography-format file. Save the results in a new Taurus-Lithography-format file.
4. Read the saved TSUPREM-4-format file into TSUPREM-4 (with an **INITIALIZE** or **LOADFILE** statement).
5. Read the new Taurus-Lithography-format file into TSUPREM-4 (with a **LOADFILE** statement).

MINIMOS

The **MINIMOS5** parameter creates an output file that can be read by MINIMOS 5, a two-dimensional program for the simulation of MOSFETs and MESFETs. The output file contains a two-dimensional doping profile. Detailed instructions on interfacing TSUPREM-4 to MINIMOS 5 are given in [Appendix F](#).

Temperature

The final temperature of the last high-temperature processing step is needed to calculate active impurity concentrations. Normally, the value at the end of the last diffusion or epitaxy step or the value read when a structure is loaded is used, but if **TEMPERAT** is specified, its value is used instead. The value of **TEMPERAT** (if specified) is retained for use in subsequent input statements. The last temperature value is saved in TSUPREM-4 structure files and is restored when the structure is loaded with a **LOADFILE** or **INITIALIZE** statement.

Examples

1. The following statement saves a structure in the file *savestr*:

```
SAVEFILE OUT.FILE=savestr
```

2. The following statement saves the structure in file *PIOUTSTR*, in a format that can be read by Medici:

```
SAVEFILE OUT.FILE=PIOUTSTR MEDICI
```

3. The following statements save a structure to be processed further with Taurus-Lithography:

```
SAVEFILE OUT.FILE=STRTS4  
SAVEFILE OUT.FILE=STRTODEP DEPICT
```

Taurus-Lithography can read the file *STRTODEP*. The structure file *STRTS4* is needed when reading the results produced by Taurus-Lithography. If Taurus-Lithography stored its results in file *STRFRDEP*, they could be read into TSUPREM-4 with the statements

```
INITIALIZE IN.FILE=STRTS4  
LOADFILE IN.FILE=STRFRDEP DEPICT
```


STRUCTURE

The **STRUCTURE** statement reflects, truncates, or extends the current structure.

STRUCTURE

```
[ TRUNCATE { ( {RIGHT | LEFT} X=<n> ) | ( {BOTTOM | TOP} Y=<n> ) } ]
[ REFLECT [ {RIGHT | LEFT} ] ]
[ EXTEND [ {RIGHT | LEFT} ] WIDTH=<n>
  [ SPACES=<n> ] [ DX=<n> ] [ XDX=<n> ] [ Y.ELIM=<c> ] ]
[ UNREFINE [ REPEAT=<n> ] [ ROWS ] [ COLUMNS ]
  [ X.MIN=<n> ] [ X.MAX=<n> ] [ Y.MIN=<n> ] [ Y.MAX=<n> ] ]
[ TEMPERAT=<n> ]
```

| Parameter | Type | Definition |
|-----------------|---------|---|
| TRUNCATE | logical | Causes the current mesh to be truncated according to the values of the RIGHT , LEFT , BOTTOM , TOP , X , and Y parameters. The boundary condition is set to reflecting along the new surface created by the truncation. Default: false |
| RIGHT | logical | Specifies that truncation, reflection, or extension occurs at the right edge of the structure. Default: true unless LEFT , BOTTOM , or TOP is specified |
| LEFT | logical | Specifies that truncation, reflection, or extension occurs at the left edge of the structure. Default: false |
| X | number | The <i>x</i> coordinate where truncation occurs. If LEFT is specified, truncation occurs to the left of this location; if RIGHT is specified, truncation occurs to the right of this location. Only valid if TRUNCATE and RIGHT or LEFT are specified. Units: microns Default: none |
| BOTTOM | logical | Specifies that truncation occurs at the bottom edge of the structure. Default: false |
| TOP | logical | Specifies that truncation occurs at the top edge of the structure. Default: false |
| Y | number | The <i>y</i> coordinate where truncation occurs. If BOTTOM is specified, the portion of the structure below this location (higher values of <i>y</i>) is truncated; if TOP is specified, the portion of the structure above this location (lower values of <i>y</i>) is truncated. Only valid if TRUNCATE and BOTTOM or TOP are also specified. Units: microns Default: none |
| REFLECT | logical | Causes the current mesh to be reflected at its left or right edge, doubling the width of the structure. Default: false |

| Parameter | Type | Definition |
|-----------------|-----------|--|
| EXTEND | logical | Causes the current mesh to be extended from its left or right edge by the specified width. Default: false |
| WIDTH | number | The width by which the structure is extended. Units: microns Default: none |
| SPACES | number | The number of grid spaces to be added in the x direction when extending the structure. Units: none Default: 1 |
| DX | number | The nominal grid spacing in the x direction in the extension of the structure. Units: microns Default: the width of the extension/ SPACES |
| XDX | number | The absolute x location at which the grid spacing in the extension is equal to DX . Units: microns Default: the location of the new edge of the extended structure |
| Y.ELIM | character | A list of 10 or fewer y locations, separated by spaces or commas, below which vertical grid lines are eliminated in the extended portion of the structure. If spaces appear in the list, the entire list must be enclosed in quotation marks. Units: microns Default: no elimination is done |
| UNREFINE | logical | Causes one or more mesh unrefinement steps to be performed. Default: false |
| REPEAT | number | Specifies the maximum number of unrefinement steps to be performed. Units: none Default: 1 |
| ROWS | logical | Specifies that rows of points in the undoped (or background doped) substrate may be removed during unrefinement. Default: false |
| COLUMNS | logical | Specifies that columns of points in the undoped (or background doped) substrate may be removed during unrefinement. Default: false |
| X.MIN | number | The x coordinate of the left edge of the region in which unrefinement is allowed. Units: microns Default: the left edge of the structure |
| X.MAX | number | The x coordinate of the right edge of the region in which unrefinement is allowed. Units: microns Default: the right edge of the structure |

| Parameter | Type | Definition |
|-----------------|--------|--|
| Y . MIN | number | The y coordinate of the top of the region in which unrefinement is allowed. Units: microns Default: the top of the structure |
| Y . MAX | number | The y coordinate of the bottom of the region in which unrefinement is allowed. Units: microns Default: the bottom of the structure |
| TEMPERAT | number | Specifies the last processing temperature to be used for evaluating active impurity concentrations. Units: °Celsius Default: the last processing temperature specified, or 800°C |

Description

The **STRUCTURE** statement is used to reflect, truncate, extend, or unrefine the current structure. The structure can be truncated on the left or right or at the bottom or top; reflected about its left or right edge, doubling its width; or extended from its left or right edge to the specified width. Truncation occurs before reflection, and reflection occurs before extension. Unrefinement occurs after truncation and extension. The last processing temperature can also be set for use in calculating active impurity concentrations for the **SELECT** or **SAVEFILE** statements.

Reflecting a structure also reflects any mask information associated with the structure. The mask information is truncated or extended (if necessary) to the value specified by **X**, then reflected.

The **SPACES**, **DX**, and **XDX** parameters specify the grid when a structure is extended. These parameters are used in the same way as the **THICKNES**, **DY**, and **YDY** parameters are used in the **DEPOSITION** and **EPITAXY** statements. The use of these parameters is described in [Chapter 2, “Changes to the Mesh During Processing” on page 2-7](#).



Note:

*The **EXTEND** feature should be used with caution. Simulating a narrow structure and then extending it does not give the same results as simulation of the wider structure unless the topography and doping concentrations at the edge of the narrow structure are essentially one-dimensional.*

Order of Operations

The operations specified on the **STRUCTURE** statement are performed in the following order:

1. Set the last processing temperature, if specified with the **TEMPERAT** parameter.
2. Truncate the structure (if **TRUNCATE** is specified).
3. Reflect the structure (if **REFLECT** is specified).

4. Extend the structure (if **EXTEND** is specified).
5. Unrefine the structure (if **UNREFINE** is specified).

Usage

Reflecting the structure about an edge that is not absolutely vertical can result in very thin triangles that could cause numerical problems. A structure should only be reflected about a reflecting boundary, created during an initial structure specification or by the **TRUNCATE** parameter.

TRUNCATE TOP can be used to remove the entire exposed surface, giving a structure with reflecting boundary conditions on all four sides. The usefulness of such structures is limited because some process steps (e.g., **DEPOSITION** and **ETCH**) do not work on structures that have no exposed surface.

The **UNREFINE** parameter is used to remove unneeded mesh points from the structure. If **REPEAT** is more than 1, the unrefinement is repeated the specified number of times or until no further points can be removed. **ROWS** and **COLUMNS** allow removal of rows and columns of points in the undoped (or background doped) substrate. If both **ROWS** and **COLUMNS** are specified, rows and columns will be removed in alternating unrefinement steps. The **X.MIN**, **X.MAX**, **Y.MIN**, and **Y.MAX** parameters can be used to limit unrefinement to a portion of the structure.

TSUPREM-4 Version Compatibility

In versions of TSUPREM-4 prior to version 5.1, the **STRUCTURE** statement is used to save and load structure files. Starting with version 5.1, the **SAVEFILE** and **LOADFILE** statements should be used instead. Starting with version 2000.2, the parameters for saving TSUPREM-4, Taurus-Lithography, Medici, and MINIMOS 5 files and for loading TSUPREM-4 and Taurus-Lithography files are no longer accepted on the **STRUCTURE** statement.

Examples

1. The following statement truncates the structure to the right of $x=1.2$ microns, then mirrors the structure about the new right edge:

```
STRUCTURE TRUNCATE RIGHT X=1.2 REFLECT
```

2. The following statement extends the structure to the right by 0.5 microns. Two grid spaces are added, with elimination of every other vertical line below $y=1$ μm :

```
STRUCTURE EXTEND RIGHT WIDTH=0.5 +  
SPACES=2 Y.ELIM=1.0
```

3. The following statement causes up to four unrefinement steps to be performed:

```
STRUCTURE UNREFINE REPEAT=4 ROWS COLUMNS
```

The first and third steps allow rows of points to be removed in the substrate while the second and fourth steps allow columns to be removed.

MASK

The **MASK** statement reads masking information from a file.

MASK

```
[ IN . FILE=<c> [ SCALE=<n> ] [ GRID=<c> ] [ G . EXTENT=<c> ] ] [ PRINT ]
```

| Parameter | Type | Definition |
|-------------------|-----------|---|
| IN . FILE | character | Specifies a file containing mask information. Default: none |
| SCALE | number | Specifies a scale factor to be applied to all dimensions and locations in the mask file. Units: none Default: 1.0 |
| GRID | character | One or more names of mask levels (separated by spaces or commas) for which fine horizontal grid is to be used during automatic grid generation. Default: all mask levels used |
| G . EXTENT | character | The distance that fine grid spacing extends under the lines on a mask. If a single value is specified, it is used for all mask levels specified with the GRID parameter (or all mask levels if GRID is not specified). If multiple values (separated by spaces or commas) are specified, then each value corresponds to a mask level specified with the GRID parameter. Negative values specify that fine grid extends into the spaces between the mask lines. Units: microns Default: 0.0 |
| PRINT | logical | Prints mask information on the standard output and in the output listing file. The listing includes the names of all masks and the minimum and maximum <i>x</i> locations of each opaque segment of each mask. Default: false |

Description

The **MASK** statement reads a file describing the mask levels used in a process. This mask information is used by the **EXPOSE** statement for patterning photoresist, which can be used as a masking layer for subsequent processing. By using mask files, the process description can be separated from the layout information, making it easy to simulate multiple layouts with a single process or multiple processes with a given layout.

The **SCALE** parameter specifies a scale factor to be applied to all dimensions and locations in the mask file.

The **GRID** and **G.EXTENT** parameters control how mask information is used to set the horizontal grid spacing during automatic grid generation. The **GRID** parameter specifies which mask levels are used. Fine grid spacing is used at the edges of lines on the specified levels. The **G.EXTENT** parameter determines how far the fine grid extends under the lines of the mask. The **GRID** and **G.EXTENT** parameters are character strings that specify lists of values; be sure to enclose them in quotation marks if they contain spaces.

TSUPREM-4 input mask files are created by Taurus Layout. Taurus Layout lets you view a mask layout file and interactively select regions to be simulated.

Examples

1. The following statement reads mask information from the file *HVNCH.TL1* then prints the information that is read:

```
MASK IN.FILE=HVNCH.TL1 PRINT
```

2. The following statement specifies that only the *POLY* and *FIELD* masks should be considered when doing automatic grid generation. Fine grid should extend 0.5 microns under the edges of lines on the *POLY* mask and 0.3 microns under the edges of lines on the *FIELD* mask:

```
MASK IN.FILE=S4EX4M.TL1 GRID="Poly,Field"+  
G.EXTENT="0.5,0.3"
```

3. TSUPREM-4 input files can be structured to separate the masking information from the processing information:

```
$ MESH DESCRIPTION (DEPENDS ON MASKS)  
LINE X LOC=0.0 SPAC=0.01  
.  
.  
.  
LINE Y LOC=200.0 SPAC=50.0  
INITIALIZE BORON=1E14  
$ MASK INFORMATION  
MASK IN.FILE=MINNCH.TL1  
$ PROCESS INFORMATION (DOES NOT DEPEND ON MASKS)  
SOURCE CMOS07
```

To simulate a different portion of a layout with the same process, you only need to change the mesh description and the name of the **MASK** file. To simulate the same device with a modified process, you only need to change the name of the **SOURCE** file.

PROFILE

The **PROFILE** statement allows a one-dimensional impurity profile to be read from a file.

PROFILE

```
{ IMPURITY=<c> | ANTIMONY | ARSENIC | BORON | PHOSPHOR }
IN.FILE=<c> OFFSET=<n> [REPLACE]
```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| IMPURITY | character | The name of the impurity to be read from the input file. Default: none |
| ANTIMONY | logical | Specifies that the input file contains a profile of antimony concentration. Default: false Synonym: SB |
| ARSENIC | logical | Specifies that the input file contains a profile of arsenic concentration. Default: false Synonym: AS |
| BORON | logical | Specifies that the input file contains a profile of boron concentration. Default: false Synonym: B |
| PHOSPHOR | logical | Specifies that the input file contains a profile of phosphorus concentration. Default: false Synonym: P |
| IN.FILE | character | The name of the data file containing the one-dimensional profile information. Default: none Synonym: INFILE |
| OFFSET | number | The amount by which the profile is shifted in the y direction before being applied to the TSUPREM-4 structure. The concentration at y=0 in the data file is added at y= OFFSET in the TSUPREM-4 structure. Units: microns Default: 0 |
| REPLACE | logical | Specifies that the profile read from the file should replace the existing profile (rather than adding to it). Default: false |

Description

This statement allows a one-dimensional impurity profile (or other solution variable) to be read from a data file. The data file must be a text file containing two columns of numbers. The first column gives the y coordinate of a point, in microns, and the second contains the impurity concentration (or other solution

value), in appropriate units. Lines in the data file that do not contain exactly two recognizable numeric values are ignored.

If **REPLACE** is specified, the profile read from the file replaces any existing profile. Replacement occurs only at nodes whose y coordinates are within the range of y coordinates given in the file. Adaptive gridding does not apply to the **PROFILE** statement. You must insure that the grid is dense enough to resolve the profile.

OFFSET Parameter

The profile read from the data file is expanded in the x direction to the width of the structure and added to any existing impurities in the structure; negative data values can be used to reduce the concentration. The profile can be shifted in the y direction by specifying the **OFFSET** parameter. The profile is shifted by **OFFSET**, so that the concentration at $y=0$ in the data file appears at location $y=-\mathbf{OFFSET}$ in the TSUPREM-4 structure.

Interpolation

Values between the y coordinates specified in the input file are calculated using linear interpolation on the function

$$f(c) = \text{sign}(c) \log \left(\frac{|c|}{2k} + \sqrt{\left(\frac{c}{2k}\right)^2 + 1} \right) \quad \text{Equation 3-1}$$

where

- c is the concentration in the file
- k is a scaling factor.

The final value of c at each point is found by inverting $f(c)$ using the value of f obtained from the linear interpolation. This gives logarithmic interpolation for $|c| \gg k$ and linear interpolation for $|c| \ll k$. The value of k is taken to be 10^5 for impurity and point defect concentrations and 1.0 for other solution variables (e.g., *dloop* and *rloop*). At points outside the range of values in the input file, the impurity concentration remains unchanged.

IMPURITY Parameter

The **IMPURITY** parameter can be used to read in values for a user-specified impurity. This can be used to read in a profile for comparison without altering the desired doping:

```

IMPLANT BORON ...
SELECT Z=LOG10(BORON)
PLOT.1D ...
IMPURITY NEW IMPURITY=PBORON
PROFILE IMPURITY=PBORON ...
SELECT Z=LOG10(PBORON)
PLOT.1D ^AX ^CL ...

```

If *boron* had been specified on the **PROFILE** statement, the profile would have been added to the implanted boron, making it impossible to compare the implanted profile with the contents of the data file.

The **IMPURITY** parameter can be used to read in values of interstitial and vacancy profiles. Remember that the profiles read from the file are added to the profiles already in the structure, unless **REPLACE** is specified.

Example

The following statement reads boron concentration data from the file *bprof.dat*:

```

PROFILE BORON IN.FILE=bprof.dat OFFSET=-0.1

```

If the file contained the lines

```

0.0  1.0e17
0.1  1.4e17
0.2  1.8e17
0.4  2.0e17
0.6  1.8e17
0.8  1.4e17
1.0  1.0e17
2.0  1.0e16
5.0  1.0e14

```

then a boron concentration of 1.0e17 would be added at $y=-0.1$, a concentration of 1.4e17 would be added at $y=0$, a concentration of 1.8e17 at $y=0.1$, and so on.

ELECTRODE

The **ELECTRODE** statement is used to name the electrodes for device simulation.

```

ELECTRODE
  [NAME=<c>] [ { ( X=<n> [Y=<n>] ) | BOTTOM } ]
  [CLEAR [ALL]]
  [MERGE]
  [PRINT]
    
```

| Parameter | Type | Definition |
|---------------|-----------|---|
| NAME | character | The name of the electrode to define or delete. This parameter is required unless both CLEAR and ALL are specified. Default: none |
| X | number | The <i>x</i> coordinate of an (<i>x</i> , <i>y</i>) position in the region to be defined as an electrode. Units: microns Default: none |
| Y | number | The <i>y</i> coordinate of an (<i>x</i> , <i>y</i>) position in the region to be defined as an electrode. Units: microns Default: the <i>y</i> coordinate inside the conductor at the <i>x</i> position |
| BOTTOM | logical | Specifies that the electrode to be defined is at the bottom of structure. Default: false |
| CLEAR | logical | Specifies that the named electrode is to be deleted. Default: false |
| MERGE | logical | Specifies the polysilicon adjacent to the electrode to be merged as the electrode. Only applicable to the saved file in TIF format. Default: true |
| ALL | logical | Specifies that all defined electrodes are to be deleted. Default: false |
| PRINT | logical | Specifies that the summary of defined electrodes is printed. Default: true |

Description

The **ELECTRODE** statement allows you to specify names for electrodes. These electrode names are saved in TIF and **MEDICI** output files for use in Medici and other programs. The materials which can be named to an electrode are polysilicon and all materials which have been defined as conductors on a **MATERIAL** statement. Once an electrode is named, the information is kept unless it is undefined by **CLEAR** in an **ELECTRODE** statement.

Examples

1. The following statements show how to name the electrodes.

```

ELECTROD  X=0.1  NAME=Source
ELECTROD  X=1.2  NAME=Gate
ELECTROD  X=2.3  NAME=Drain
ELECTROD  BOTTOM NAME=Bulk
SAVEFILE  OUT.FILE=mos.mdc MEDICI POLY.ELE ELEC.BOT

```

2. The following statements show how to undefine the electrode.

```

ELECTROD  X=0.1 NAME=Wrong
...
ELECTROD  NAME=Wrong CLEAR

```

Additional ELECTRODE Notes

1. The polysilicon region named in **ELECTROD** is stored as an electrode in the saved file in TIF format. For Medici format, **POLY.ELE** must be specified in **SAVEFILE**.
2. The polysilicon region adjacent to the conductor named in **ELECTROD** is merged as the same electrode if **MERGE** parameter for TIF format, or **POLY.ELE** parameter (in **SAVEFILE**) for Medici format is specified.
3. If there are more than two conductors and/or polysilicon regions at the **X** position, for example, like EPROM structure, the **Y** must be specified individually.

```

ELECTROD  X=1.2 Y=-0.02 NAME=Erase
ELECTROD  X=1.2 Y=-0.35 NAME=Program

```

3.3 Process Steps

The following statements simulate processing steps:

| Statement Name | Description | Page |
|-----------------------|---|-----------------------|
| DEPOSITION | Deposits a material on the exposed surface. | 3-90 |
| EXPOSE | Exposes photoresist using mask. | 3-95 |
| DEVELOP | Removes exposed positive photoresist or unexposed negative photoresist. | 3-97 |
| ETCH | Etches an exposed material. | 3-98 |
| IMPLANT | Implants an impurity. | 3-103 |
| DIFFUSION | Performs a diffusion step, possibly with oxidation. | 3-115 |
| EPITAXY | Performs a silicon epitaxial deposition step. | 3-121 |
| STRESS | Calculates the stresses in the structure. | 3-124 |

DEPOSITION

The **DEPOSITION** statement is used to deposit a specified material on the exposed surface of the current structure.

DEPOSITION

```

{ MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE
  | POLYSILI | ALUMINUM
  | ( PHOTORES [ { POSITIVE | NEGATIVE } ] )
}
[ IMPURITY=<c> { I.CONC=<n> | I.RESIST=<n> } ]
[ANTIMONY=<n>] [ARSENIC=<n>] [BORON=<n>] [PHOSPHOR=<n>]
[ {CONCENTR | RESISTIV} ]
THICKNES=<n> [SPACES=<n>] [DY=<n>] [YDY=<n>] [ARC.SPAC=<n>]
[TEMPERAT=<n>] [GSZ.LIN]
TOPOGRAP=<c>

```

| Parameter | Type | Definition |
|-----------------|-----------|--|
| MATERIAL | character | The name of the material to be deposited. Default: none |
| SILICON | logical | Deposit silicon. Default: false |
| OXIDE | logical | Deposit oxide. Default: false |
| OXYNITRI | logical | Deposit oxynitride. Default: false |
| NITRIDE | logical | Deposit nitride. Default: false |
| POLYSILI | logical | Deposit polysilicon. Default: false |
| ALUMINUM | logical | Deposit aluminum. Default: false |
| PHOTORES | logical | Deposit photoresist. Default: false |
| POSITIVE | logical | Specifies that the deposited photoresist (and all other photoresist in the structure) is positive, i.e., that the DEVELOP statement removes exposed photoresist while leaving unexposed photoresist. Default: true, unless NEGATIVE is specified |
| NEGATIVE | logical | Specifies that the deposited photoresist (and all other photoresist in the structure) is negative, i.e., that the DEVELOP statement removes unexposed photoresist while leaving exposed photoresist. Default: false |

| Parameter | Type | Definition |
|-------------------|-----------|--|
| IMPURITY | character | The name of the impurity with which the deposited layer is doped. Default: none |
| I . CONC | number | The concentration of the specified IMPURITY in the deposited layer. Units: atoms/cm ³ Default: none |
| I . RESIST | number | The resistivity of the deposited layer. Units: ohm-cm Default: none |
| ANTIMONY | number | The uniform concentration or resistivity of antimony in the deposited layer. Units: atoms/cm ³ or ohm-cm Default: 0.0 Synonyms: SB |
| ARSENIC | number | The uniform concentration or resistivity of arsenic in the deposited layer. Units: atoms/cm ³ or ohm-cm Default: 0.0 Synonym: AS |
| BORON | number | The uniform concentration or resistivity of boron in the deposited layer. Units: atoms/cm ³ or ohm-cm Default: 0.0 Synonym: B |
| PHOSPHOR | number | The uniform concentration or resistivity of phosphorus in the deposited layer. Units: atoms/cm ³ or ohm-cm Default: 0.0 Synonym: P |
| CONCENTR | logical | Specifies that the impurity concentration in the deposited layer is given. Default: true |
| RESISTIV | logical | Specifies that the resistivity in the deposited layer is given. Default: false |
| THICKNES | number | The thickness of the deposited layer. Units: microns Default: none |
| SPACES | number | The number of vertical grid spacings in the layer. This value is divided by the value of GRID . FAC (see “MESH” on page 3-50). Units: none Default: 1/ GRID . FAC Synonym: DIVISION |
| DY | number | The nominal grid spacing to be used in the deposited material layer at the location specified by the YDY parameter. This value is multiplied by the value of GRID . FAC (see “MESH” on page 3-50). Units: microns Default: GRID . FAC*THICKNES/SPACES Synonym: DX |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| YDY | number | The location of the nominal grid spacing specified by DY relative to the top of the deposited layer. Units: microns Default: 0.0 Synonym: XDX |
| ARC.SPAC | number | The maximum spacing allowed along an arc on the new surface. This value is multiplied by the value of GRID.FAC (see “ MESH ” on page 3-50). Units: microns Default: 0.5* THICKNES * GRID.FAC |
| TEMPERAT | number | The deposition temperature; used to determine initial grain size when depositing polycrystalline materials. Units: degrees Default: 0.0 Kelvins |
| GSZ.LIN | logical | Specifies that the grain size increases linearly with depth from the bottom of the deposited layer. If false, the grain size is constant through the layer. Default: true |
| TOPOGRAP | character | The name of a file containing Taurus-Topography input commands that define the deposition to be performed. Default: none Synonym: TERRAIN |

Description

This statement provides a basic deposition capability. Material is deposited on the exposed surface of the structure, with the upper surface of the deposited layer becoming the new exposed surface. The **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHOR** parameters specify the initial impurity concentrations or resistivity in the deposited layer, depending on whether **CONCENTR** or **RESISTIV** is true. Doping can also be specified with the **IMPURITY** and **I.CONC** or **I.RESIST** parameters.

The deposited material conforms to the contours of the original surface. Outside corners on the original surface produce arcs on the new surface, which are approximated by straight line segments. The maximum segment length is set by the **ARC.SPAC** parameter. The **SPACES**, **DY**, and **YDY** parameters used to control the grid spacing in the deposited layer are scaled by the value of the **GRID.FAC** parameter on the **MESH** statement (see [Chapter 2, “Changes to the Mesh During Processing”](#) on page 2-7).



Note:

It is not possible to deposit a layer on the bottom of a structure, even if it is exposed. Attempting to do so may cause the program to fail.

Polycrystalline Materials

The **TEMPERAT** and **GSZ.LIN** parameters are used only when depositing a polycrystalline material. If no temperature is specified, or if the temperature is less than the value of **TEMP.BRE** for the material, the deposited layer is amorphous; recrystallization occurs at the start of the next high-temperature step.

Photoresist

Photoresist can be positive or negative, but all photoresist in a structure must be of the same type. If photoresist of one type is deposited on a structure containing photoresist of the other type, a warning is issued and the type of the old photoresist is changed to that of the newly deposited photoresist.

Deposition with Taurus-Topography

The **TOPOGRAP** parameter invokes Taurus-Topography with the specified command input file. The command input file contains Taurus-Topography commands describing one or more processing steps to be simulated by Taurus-Topography. It should not contain the **INITIALIZE** or **STOP** statements.

The values of variables set with the **ASSIGN**, **DEFINE**, and **EXTRACT** statements are substituted in the Taurus-Topography command input file. In addition to variables set explicitly by you, if the **THICKNES** parameter is set on the **DEPOSITION** statement then its value is assigned to the variable **THICK** prior to substitution. (If the variable **THICK** is assigned in this way, it will be unset after the **DEPOSITION** statement, even if it was set by you previously.) This allows parameter values (such as deposition thickness) to be passed to Taurus-Topography. The most recent mask file specified in the TSUPREM-4 input file is passed to Taurus-Topography for use in masked etch steps.

By default, Taurus-Topography is called by requesting that the command `topography` be executed by the operating system, but if the environment variable `S4TERRAIN` is set, its value is used instead. It may be necessary for you to define other environment variables (e.g., `TERR_LIB`) for Taurus-Topography to run correctly. For additional details refer to the *Taurus-Topography Reference Manual*.

When Taurus-Topography is invoked with the **DEPOSITION** statement, the full set of parameters is used for any deposited layer of the specified material. Thus you have full control over grid spacing, doping, and polycrystalline grain size. Note that the parameters specified on the **DEPOSITION** statement only apply to the specified material. They are ignored for other materials deposited by Taurus-Topography.

Examples

1. The following statement deposits 200 angstroms of silicon dioxide:

```
DEPOSIT OXIDE THICK=0.02
```

2. The following statement deposits a one-micron thick layer of photoresist, using four grid spaces in the layer:

```
DEPOSIT PHOTO THICK=1.0 SPACES=4 ARC.SPAC=0.1
```

The maximum segment length used to approximate arcs is 0.1 micron. By default, the newly deposited photoresist (and any photoresist already present in the structure) is assumed to be positive.

3. The following statement deposits 0.1 micron of poly:

```
DEPOSIT MAT=POLY THICK=0.1 TEMPERAT=650 GSZ.LIN
```

The initial grain size is calculated at 650 °C and increases linearly over the thickness of the layer, with minimum value at the bottom of the layer.

4. The following statement calls Taurus-Topography with the input command file *PolyDep.inp*:

```
DEPOSIT MAT=POLY THICK=0.1 TEMPERAT=650 GSZ.LIN +  
      TOPOGRAPHY=PolyDep.inp
```

References to “@THICK” in the Taurus-Topography input file are replaced with the value “0.1”. If the call to Taurus-Topography causes the deposition of a polysilicon layer then the specified temperature and grain size model are used for that layer.

Additional DEPOSITION Notes

1. The calculation of doping concentration from resistivity uses mobility tables for silicon and polysilicon. If the resistivity is specified when depositing some other material, a warning is issued. The mobility tables used for calculating the doping concentration are not the same as the tables used by the **ELECTRICAL** statement. The extracted sheet resistance for the deposited layer does not correspond exactly to the resistivity specified during deposition.
2. Deposition of one material on top of another can cause a third material to be added between them. This happens when titanium is deposited on silicon, for example—a layer of TiSi₂ is inserted. Insertion of extra layers is specified with the **REACTION** statement (see [page 3-254](#)).
3. Diffusion of impurities during a deposition step is not simulated, even though a temperature may be specified.

EXPOSE

The **EXPOSE** statement exposes photoresist using a mask.

EXPOSE

MASK=<c> [SHRINK=<n>] [OFFSET=<n>]

| Parameter | Type | Definition |
|---------------|-----------|--|
| MASK | character | Specifies the name of the mask to be used for the exposure step. The mask name must match the name of a mask previously read with a MASK statement. Default: none |
| SHRINK | number | The reduction in line width on each side of each line on the mask. The total reduction in line width is twice the specified value. Units: microns Default: 0.0 |
| OFFSET | number | The amount by which lines on the mask are shifted (in the positive <i>x</i> direction). Units: microns Default: 0.0 |

Description

The **EXPOSE** statement exposes photoresist, using masking information previously read with a **MASK** statement. Regions of photoresist that are directly below transparent regions of the specified mask (and not obscured by any other material) are marked as exposed. If positive photoresist is present, exposed regions are removed by a subsequent **DEVELOP** statement; if negative photoresist is present, the unexposed portion of the photoresist is removed.

The **SHRINK** and **OFFSET** parameters can be used to model the effects of deliberate or accidental adjustments to the mask-making or photolithographic processes. **SHRINK** specifies a reduction in the width of lines on the mask, while **OFFSET** specifies a shifting of the mask (relative to other masks). The program does not allow shrinks or offsets that are large enough to make mask lines or spaces disappear.



Note:

*The **OFFSET** parameter breaks the symmetry that is assumed at reflecting boundaries at the edge of a structure. Shifting a line towards a reflecting boundary also makes it narrower, while shifting a line away from a reflecting boundary makes it wider.*

Example

The following sequence illustrates a typical use of the **EXPOSE** statement:

```
MASK  IN.FILE=CMOS3.TL1
      .
      .
      .
DEPOSIT POLY  THICKNES=.2
DEPOSIT POSITIVE PHOTORES  THICKNES=1
EXPOSE  MASK=POLY SHRINK=0.05
DEVELOP
ETCH    POLY  TRAP  THICK=0.4  ANGLE=80
ETCH    PHOTORES  ALL
```

This sequence produces lines of polycrystalline silicon under the opaque regions of the mask named *POLY*. The width of each *POLY* line is reduced by 0.05 microns on each side (for a total reduction of 0.1 micron).

DEVELOP

The **DEVELOP** statement removes exposed positive photoresist and unexposed negative photoresist.

DEVELOP

[<c>]

Description

The **DEVELOP** statement is used to pattern photoresist by removing exposed positive resist and unexposed negative resist. The character string associated with the **DEVELOP** statement is ignored, and serves only to document the input.

Example

The following sequence illustrates a typical use of the **DEVELOP** statement:

```
MASK  IN.FILE=CMOS3.TL1
      .
      .
      .
DEPOSIT POLY  THICKNES=.2
DEPOSIT POSITIVE PHOTORES THICKNES=1
EXPOSE  MASK=POLY
DEVELOP
ETCH    POLY TRAP THICK=0.4 ANGLE=80
ETCH    PHOTORES ALL
```

This sequence produces lines of polycrystalline silicon under the opaque regions of the mask named *POLY*.

ETCH

The **ETCH** statement is used to remove portions of the current structure.

ETCH

```
[ { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE
  | POLYSILI | PHOTORES | ALUMINUM
}
]
{ ( TRAPEZOI [THICKNES=<n>] [ANGLE=<n>] [UNDERCUT=<n>] )
  ( {LEFT | RIGHT} [P1.X=<n>] [P1.Y=<n>] [P2.X=<n>] [P2.Y=<n>] )
  ( {START | CONTINUE | DONE} X=<n> Y=<n> )
  ISOTROPI
  ( OLD.DRY THICKNES=<n> )
  ALL
  TOPOGRAP=<c>
}
```

| Parameter | Type | Definition |
|-----------------|-----------|--|
| MATERIAL | character | The name of the material to be etched. Default: none |
| SILICON | logical | Etch silicon only. Default: false |
| OXIDE | logical | Etch oxide only. Default: false |
| OXYNITRI | logical | Etch oxynitride only. Default: false |
| NITRIDE | logical | Etch nitride only. Default: false |
| POLYSILI | logical | Etch polysilicon only. Default: false |
| PHOTORES | logical | Etch photoresist only. Default: false |
| ALUMINUM | logical | Etch aluminum only. Default: false |
| TRAPEZOI | logical | Use an etch model that removes material from a trapezoidal region when applied to a planar surface. This is a generalization of the DRY model in older versions of TSUPREM-4. Default: true Synonym: DRY |

| Parameter | Type | Definition |
|-----------------|---------|--|
| THICKNES | number | The thickness of the layer to be removed when the TRAPEZOI or OLD.DRY parameter is specified. Units: microns Default: $\text{UNDERCUT} * \tan(\text{ANGLE})$ for $\text{ANGLE} < 90$, infinite otherwise; no default if OLD.DRY is specified |
| ANGLE | number | The angle of the sidewalls produced when the TRAPEZOI parameter is specified. The angle is measured from the horizontal, so that vertical sidewalls have an angle of 90° . Units: degrees Default: $\arctan(\text{THICKNES}/\text{UNDERCUT})$ if THICKNES and undercut are both specified, 90 otherwise |
| UNDERCUT | number | The distance that the etch extends under masking layers when the TRAPEZOI parameter is specified. Units: microns Default: $\text{THICKNES}/\tan(\text{ANGLE})$ for $\text{ANGLE} < 90$, 0 otherwise |
| LEFT | logical | Etch material to the left of the specified position. Default: false |
| RIGHT | logical | Etch material to the right of the specified position. Default: false |
| P1.X | number | The x coordinate of the first point used when LEFT or RIGHT is specified. Units: microns Default: 0.0 |
| P1.Y | number | The y coordinate of the first point used when LEFT or RIGHT is specified. Units: microns Default: a point above the top of the structure |
| P2.X | number | The x coordinate of the second point used when LEFT or RIGHT is specified. Units: microns Default: the value of P1.X |
| P2.Y | number | The y coordinate of the second point used when LEFT or RIGHT is specified. Units: microns Default: a point below the bottom of the structure |
| START | logical | The point (X,Y) is the first point in a series defining the region to be etched. Default: false |
| CONTINUE | logical | The point (X,Y) is the next point in a series defining the region to be etched. Default: false |
| DONE | logical | The point (X,Y) is the last point in a series defining the region to be etched. Default: false Synonym: END |
| ISOTROPI | logical | Specifies removal of material that lies within THICKNES of an exposed surface. Default: false |

| Parameter | Type | Definition |
|-----------------|-----------|--|
| X | number | The <i>x</i> coordinate used with the START , CONTINUE , or DONE parameter. Units: microns Default: none |
| Y | number | The <i>y</i> coordinate used with the START , CONTINUE , or DONE parameter. Units: microns Default: none |
| OLD.DRY | logical | The exposed surface is etched vertically by an amount given by the THICKNES parameter. Default: false |
| ALL | logical | The specified material is etched away entirely. Default: false |
| TOPOGRAP | character | The name of a file containing Taurus-Topography input commands that define the etch to be performed. Default: none Synonym: TERRAIN |

Description

This statement is used to remove a portion of the current structure. you define the region to be removed and you can optionally specify a material to be removed; if no material is specified, all materials are considered to be etchable. Portions of the structure are removed provided that they are of an etchable material, lie within the defined etch region, and are exposed to the ambient. See [Chapter 2, “Etching” on page 2-107](#) for a more complete description. The surface under the etched portions of the structure is marked as exposed. The warning:

```
*** Warning: No material removed by ETCH statement.
```

is produced by an attempt to etch a material that is not exposed, or by specifying etch coordinates that do not include any etchable material.

Removing Regions

You can specify the region to be removed in one of the following ways:

1. If **TRAPEZOI** is specified, the etch region is found from a simple model of a primarily anisotropic (i.e., vertical, or directional) etch with a small isotropic component. This model can produce profiles with sloped sidewalls and undercutting of masking layers. See [Chapter 2, “Trapezoidal Etch Model” on page 2-108](#).
2. If **LEFT** or **RIGHT** is specified, the etch region includes all material to the left or right of the line between (**P1.X,P1.Y**) and (**P2.X,P2.Y**).
3. The **START**, **CONTINUE**, and **DONE** parameters are used with the **X** and **Y** parameters to define arbitrarily complex etch regions. The boundary of the region is determined by a series of **ETCH** statements, each specifying a point on the boundary. The first statement of the series should contain the **START**

parameter, the last should contain the **DONE** parameter, and statements in between should use the **CONTINUE** parameter. The last point is connected to the first point to produce a closed region defining the portion of the structure to be removed.

4. If **ISOTROPI** is specified, the etch region includes all material within the distance **THICKNES** of the exposed surface. This produces a simple isotropic etch, without rounding of outside corners.
5. If the **OLD.DRY** parameter is specified, the etch region includes all material within a vertical distance **THICKNES** of the exposed surface. This model (the **DRY** model in previous versions of TSUPREM-4) has been replaced by the **TRAPEZOI** model.

The **OLD.DRY** model in version 5.1 of TSUPREM-4 is the same as the **DRY** model in older versions of the program. In version 5.1, **DRY** is a synonym for **TRAPEZOI**. The **TRAPEZOI** model with default values of **ANGLE** and **UNDERCUT** is equivalent to the **OLD.DRY** model, except that surface layers of nonetchable material blocks etching of underlying material, even if the surface layer is thinner than **THICKNES**. Specifying **DRY** in version 5.1 is equivalent to specifying **DRY** in previous versions in cases of practical interest.

6. If the **ALL** parameter is specified, the etch region includes the entire structure.
 - If no region specification is given, **TRAPEZOI** is assumed.
 - If a material is specified, only that material is etched; otherwise, the entire region specified is subject to removal.

It is possible to cut the structure into two or more pieces with an **ETCH** statement. In this case, all pieces except the one with the largest area are removed. A warning is issued for each piece removed.



Note:

*The **ETCH** statement (except when used with the **TERRAIN** parameter) is not intended to simulate a physical etching process; its purpose is to provide a means to generate the required structures for simulation of diffusion and oxidation. Note in particular that the statement “**ETCH OXIDE TRAP**” does not implement a selective etch of oxide, but rather defines a region geometrically in which all exposed oxide is removed.*

Etching with Taurus-Topography

The **TOPOGRAP** parameter invokes Taurus-Topography with the specified command input file. The command input file contains Taurus-Topography commands describing one or more processing steps to be simulated by Taurus-Topography. It should not contain the **INITIALIZE** or **STOP** statements.

The values of variables set with the **ASSIGN**, **DEFINE**, and **EXTRACT** statements are substituted in the Taurus-Topography command input file. In addition to variables set explicitly by you, if the **THICKNES** parameter is set on the **ETCH** statement then its value is assigned to the variable **THICK** prior to substitution. (If

the variable **THICK** is assigned in this way, it will be unset after the **ETCH** statement, even if it was set by you previously.) This allows parameter values (such as etch thickness) to be passed to Taurus-Topography. The most recent mask file specified in the TSUPREM-4 input file is passed to Taurus-Topography for use in masked etch steps.

By default, Taurus-Topography is called by requesting that the command `topography` be executed by the operating system, but if the environment variable `S4TERRAIN` is set, its value is used instead. It may be necessary for you to define other environment variables (e.g., `TERR_LIB`) for Taurus-Topography to run correctly. For additional details refer to the *Taurus-Topography Reference Manual*.

Examples

1. The following statement etches the nitride to the left of 0.5 μ to a depth of 1 micron:

```
ETCH NITRIDE LEFT P1.X=0.5 P2.Y=-1.0
```

Note that **P1.Y** defaults to a location above the top of the structure and **P2.X** defaults to the value of **P1.X** (i.e., 0.5).

2. This statement etches the oxide in the square defined by (0,0), (1,0), (1,1), (0,1):

```
ETCH OXIDE START X=0.0 Y=0.0  
ETCH CONTINUE X=1.0 Y=0.0  
ETCH CONTINUE X=1.0 Y=1.0  
ETCH DONE X=0.0 Y=1.0
```

Material is removed only if there is an exposed oxide surface somewhere within the boundaries of the etch.

3. The following statement calls Taurus-Topography with the input command file *PolyDep.inp*:

```
ETCH THICK=0.1 TOPOGRAPHY=PolyDep.inp
```

References to “**@THICK**” in the Taurus-Topography input file are replaced with the value “0.1”.

IMPLANT

The **IMPLANT** statement is used to simulate the ion implantation of impurities into the structure.

IMPLANT

```

DOSE=<n> ENERGY=<n> [TILT=<n>] [ROTATION=<n>]
{IMPURITY=<c> | ANTIMONY | ARSENIC | BORON | BF2 | PHOSPHOR}
{ ( [ {GAUSSIAN | PEARSON} ] [RP.EFF] [IN.FILE=<c>]
  [IMPL.TAB=<c>] [MOMENTS] [BACKSCAT]
  )
  | ( TAUR.MOD=<c> [TAUR.EXE=<c>] [SURV.RAT=<n>] [AMOR.PAR=<n>]
    [LSS.PRE=<n>] [NLOC.PRE=<n>] [NLOC.EXP=<n>] [DAM.MOD=<c>])
  | ( MONTECAR [N.ION=<n>] [BEAMWIDT=<n>] [SEED=<n>]
    [CRYSTAL [TEMPERAT=<n>] [VIBRATIO [X.RMS=<n>] [E.LIMIT=<n>] ]
    [THRESHOL=<n>] [REC.FRAC=<n>] [CRIT.PRE=<n>]
    [CRIT.F=<n>] [CRIT.110=<n>]
    ]
    [ {PERIODIC | REFLECT | VACUUM} ]
  )
}

[POLY.GSZ=<n>] [INTERST=<c>]
[DAMAGE] [ D.PLUS=<n> | D.P1=<n> ] [D.SCALE=<n>]
[MAX.DAMA=<n>] [D.RECOMB]
[L.RADIUS=<n>] { (L.DENS=<n> [L.DMIN=<n>] [L.DMAX=<n>])
  | (L.THRESH=<n> [L.FRAC=<n>]) } ]
[PRINT]
    
```

| Parameter | Type | Definition |
|-----------|-----------|---|
| IMPURITY | character | The name of the impurity to be implanted. Default: none Synonym: IMP |
| DOSE | number | The dose of implanted ions. Units: atoms/cm ² Default: none |
| ENERGY | number | The acceleration energy of the ion implant beam. Units: keV Default: none |

| Parameter | Type | Definition |
|-----------------|---------|---|
| TILT | number | Specifies the tilt angle of the wafer, measured in a clockwise direction from the horizontal in the plane of the simulation. Positive tilt angles correspond to implanted ions arriving from the left side of vertical, and negative tilt angles correspond to implanted ions arriving from the right side of vertical (assuming the wafer rotation angle is less than 90° and greater than -90°). Tilting of the wafer with respect to the ion beam can contribute to shadowing, dose reduction and foreshortening of the implant distribution; see Chapter 2, “Analytic Ion Implant Models” on page 2-77 . Units: degrees Default: 0 |
| ROTATION | number | The angle by which the wafer has been rotated from the simulation plane, measured in a clockwise direction about an axis perpendicular to and facing into the wafer surface. Implanting with a value of 0.0 for this parameter corresponds to implantation in the plane of the simulation. Wafer rotation can contribute to shadowing of portions of the simulation structure (see Chapter 2, “Analytic Ion Implant Models” on page 2-77). Units: degrees Default: 0 |
| ANTIMONY | logical | Specifies that antimony is to be implanted. Default: false Synonym: SB |
| ARSENIC | logical | Specifies that arsenic is to be implanted. Default: false Synonym: AS |
| BORON | logical | Specifies that boron is to be implanted. Default: false Synonym: B |
| BF2 | logical | Specifies that boron in the form of BF2 is to be implanted. Default: false |
| PHOSPHOR | logical | Specifies that phosphorus is to be implanted. Default: false Synonym: P |
| GAUSSIAN | logical | Specifies that a simple Gaussian distribution is to be used to represent the implanted impurity profile. Default: false |
| PEARSON | logical | Specifies that a Pearson or dual-Pearson distribution is to be used to represent the implanted impurity profile. Default: true, unless GAUSSIAN is specified |
| RP.EFF | logical | Specifies that the effective range scaling method is to be used to calculate the implanted profile in multi-layer targets. If RP.EFF is specified to be false, the dose matching method is used to calculate the implanted profile in multi-layer targets. Default: true |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| IN.FILE | character | The identifier for the file containing implant range statistics. If specified, this file is used instead of the default implant moment data file <i>s4imp0</i> . Default: none Synonym: FILE |
| IMPL.TAB | character | The name used to choose the range statistics from the implant moment data file. Default: <i>antimony</i> , <i>tr.indium</i> , <i>dual.ars</i> , <i>chboron</i> , <i>dual.bf2</i> , and <i>dual.pho</i> for ANTIMONY , INDIUM , ARSENIC , BORON , BF2 , and PHOSPHOR implantations, respectively. Synonym: NAME |
| MOMENTS | logical | Use range statistics from previously specified MOMENT statements instead of from a moment data file. Default: false |
| BACKSCAT | logical | Enables modeling of backscattering of ions from the surface. Default: true |
| MONTECAR | logical | Specifies that a numerical Monte Carlo analysis is to be performed to simulate the implantation. Default: false |
| N.ION | number | The number of ion trajectories to calculate to generate the Monte Carlo implant distribution. Units: none Default: 1000 |
| BEAMWIDT | number | The divergence angle of the ion beam. This parameter is used to select a uniform distribution of angles about the normal values for TILT and ROTATION . The range of values by which TILT and ROTATION are varied is from $-1/2$ to $+1/2$ times the value specified for BEAMWIDT . Units: degrees Default: 0.0 |
| SEED | number | A positive integer less than 2^{31} used as an initial value for the random number generator. Changing the seed gives a different numerical simulation for a given implant. This can be used to estimate the statistical uncertainty inherent in the Monte Carlo simulation. Units: none Default: 101 |
| CRYSTAL | logical | Specifies that the crystal structure of the silicon target material is to be included in the Monte Carlo implant calculation. Default: true |
| TEMPERAT | number | The temperature of the target material during implantation. This parameter is used for calculating the amplitude of silicon lattice vibrations and the amount of damage self-annealing. This parameter is used only for the Monte Carlo implant calculation. Units: degrees Celsius Default: 26.84 |

| Parameter | Type | Definition |
|-----------|---------|---|
| VIBRATIO | logical | Specifies that lattice vibrations be included in the Monte Carlo implant calculation. By default a Debye calculation is performed to determine the amplitude of the displacement of silicon lattice atoms. The inclusion of this effect is an important determinant of dechanneling. Default: true |
| X.RMS | number | The RMS amplitude of lattice vibrations of silicon atoms during Monte Carlo implantation. The calculation of a default value for this parameter is described in Chapter 2, "Crystalline Implant Model" on page 2-94 . Units: microns Default: a value calculated in the program |
| E.LIMIT | number | An empirical parameter specifying the ion energy above which the effect of lattice vibrations is ignored during a Monte Carlo implantation. The calculation of a default value for this parameter is described in Chapter 2, "Crystalline Implant Model" on page 2-94 . Units: keV Default: 5 keV for boron; 40keV for other ions |
| THRESHOL | number | The energy threshold that must be imparted to a silicon lattice site to generate a secondary that is not self-annealed. The calculation of a default value for this parameter is described in Chapter 2, "Crystalline Implant Model" on page 2-94 . Units: eV Default: a value calculated in the program |
| REC.FRAC | number | The fraction of secondaries to be calculated by the Monte Carlo implant calculation. Units: none Default: a value calculated in the program |
| CRIT.PRE | number | The critical angle fraction used in determining whether an ion is channeled during a Monte Carlo implantation. Units: none Default: a value calculated by the program |
| CRIT.F | number | Specifies the power relationship between the critical channeling angle in the <100> direction and the ion energy during Monte Carlo implantation. Units: none Default: 0.25 |
| CRIT.110 | number | Specifies the fraction of the critical angle for the <100> direction that determines the critical angle for the <110> direction during a Monte Carlo implantation. Units: none Default: 1.2973 |
| PERIODIC | logical | Specifies that periodic boundary conditions be imposed at the left and right edges of the structure during Monte Carlo implantation. Default: true |
| REFLECT | logical | Specifies that reflecting boundary conditions be imposed at the left and right edges of the structure during Monte Carlo implantation. Default: false |

| Parameter | Type | Definition |
|-------------------|-----------|---|
| VACUUM | logical | Specifies that boundary conditions corresponding to vacuum material be imposed at the left and right edges of the structure during Monte Carlo implantation. Default: false |
| TAUR . MOD | character | Taurus implant models: one of Gauss, Pearson, DualPearson, or MonteCarlo. Default: none |
| TAUR . EXE | character | Name of Taurus executable. Default: taurus |
| SURV . RAT | number | Taurus MC parameter: point defect survival rate. Units: none. Default: Taurus default value. |
| AMOR . PAR | number | Taurus MC parameter: amorphization rate. |
| LSS . PRE | number | Taurus MC parameter: adjustable factor for LSS electron stopping power model. Units: none. Default: Taurus default value. |
| NLOC . PRE | number | Taurus MC parameter: prefactor for nonlocal component of electron stopping power model. Units: none. Default: Taurus default value. |
| NLOC . EXP | number | Taurus MC parameter: exponent for nonlocal component of electron stopping power model. Units: none. Default: Taurus default value. |
| DAM . MOD | character | Taurus MC damage model: one of recoil or norecoil. |
| POLY . GSZ | number | Specifies the polysilicon grain size in units of angstroms. Units: angstroms Default: none |
| INTERST | string | For implants into SiC only. Specifies which interstitial (either silicon or carbon) is to be plotted when selecting interstitial on the SELECT statement. Default: silicon |
| DAMAGE | logical | Specifies that point defects generated by ion implantation are to be retained and used as an initial condition for subsequent process steps. Default: true when PD . TRANS or PD . FULL is selected |
| D . PLUS | number | The scale factor applied to the implanted profile to obtain the <i>plus one</i> component of the damage profile. Units: none Default: effective +n model used |
| D . P1 | number | The +1 component of the effective +n model for implant damage. Units: none Default: 1.0 |

| Parameter | Type | Definition |
|-----------------|---------|--|
| D.SCALE | number | The scale factor applied to the Frenkel pair component of the damage profile. Units: none Default: 1.0 |
| MAX.DAMA | number | The number of point defects corresponding to amorphization of silicon. The final point defect concentrations is limited to this value. Units: #/cm ³ Default: 5e22 |
| D.RECOMB | logical | Compute the recombination of interstitials and vacancies analytically at the end of the IMPLANT step. Default: true |
| L.RADIUS | number | The radius of dislocation loops (if any) resulting from the implant. (This parameter is available only with the Extended Defects AAM.) Units: cm Default: none |
| L.DENS | number | The density of dislocation loops (if any) resulting from the implant. (This parameter is available only with the Extended Defects AAM.) Units: #/cm ³ Default: none |
| L.DMIN | number | The minimum damage level used to define the region where dislocation loops are produced. (This parameter is available only with the Extended Defects AAM.) Units: #/cm ³ Default: 1e20 |
| L.DMAX | number | The maximum damage level used to define the region where dislocation loops are produced. (This parameter is available only with the Extended Defects AAM.) Units: #/cm ³ Default: 1.15e22 |
| L.THRESH | number | The interstitial concentration above which dislocation loops are produced. (This parameter is available only with the Extended Defects AAM.) Units: #/cm ³ Default: none |
| L.FRAC | number | The fraction of interstitials in excess of L.THRESH that are incorporated into dislocation loops. (This parameter is available only with the Extended Defects AAM.) Units: none Default: 1.0 |
| PRINT | logical | Prints a summary of the current implantation on the standard output and in the output listing file. Default: false |

Description

The **IMPLANT** statement is used to simulate the implantation of impurities into the structure. The implanted impurity distribution can be calculated either numer-

ically or analytically. For more detailed descriptions of the ion implantation models see, [Chapter 2, “Ion Implantation”](#) on page 2-77.

Gaussian and Pearson Distributions

The analytic implant model uses either Gaussian or Pearson distributions. Both use an implant moment data file (*s4imp0*) or moments supplied on **MOMENT** statements if the **MOMENTS** parameter is specified. The shapes of the functions used by the analytic calculations are defined by their first four moments:

- Range
- Standard Deviation
- Skewness (also called gamma)
- Kurtosis (also called beta)

The Gaussian implant uses the first two moments, while the Pearson distribution uses the first four. A dual-Pearson model uses two sets of four moments for the dual-Pearson function. In the lateral direction, a Gaussian distribution is used in all cases, with a characteristic length specified in the implant moment data file or on a **MOMENT** statement.

Table of Range Statistics

The **IMPL.TAB** parameter can be used to specify the table of range statistics from the implant moment data file. The following tables are currently available:

| | |
|-------------------|--|
| <i>antimony</i> | default data for antimony (energies: 5–1000 keV) |
| <i>arsenic</i> | data for arsenic (energies: 5–11000 keV) |
| <i>dual.ars</i> | default dual-Pearson data for arsenic (energies: 10–1000 keV) |
| <i>tr.arsenic</i> | dual-Pearson data for arsenic in <100> silicon with full energy, dose, tilt and rotation dependence, from University of Texas (energy: 0.5–180 keV; dose: 10^{13} – 8×10^{15} atoms/cm ² ; tilt: 0–10°; rotation: 0–45°) |
| <i>bf2</i> | data for boron from BF ₂ source (energies: 5–120 keV) |
| <i>dual.bf2</i> | default dual-Pearson data for BF ₂ (energies: 10–200 keV) |
| <i>ut.bf2</i> | dual-Pearson data for boron from a BF ₂ source (energies: 15–120 keV) |
| <i>tr.bf2</i> | dual-Pearson data for boron from a BF ₂ source in <100> silicon with full energy, dose, tilt and rotation dependence, from University of Texas (energy: 0.5–65keV; dose: 10^{13} – 8×10^{15} atoms/cm ² ; tilt: 0–10°; rotation: 0–45°) |
| <i>boron</i> | original boron data with extended ranges fitted to results of amorphous Monte Carlo calculations (energies: 5-4000 keV) |

| | |
|----------------------|---|
| <i>leboron</i> | data for low-energy boron with channeling in silicon (energies: 10–30 keV) |
| <i>chboron</i> | default data for boron with channeling in silicon (energies: 5–2000 keV) |
| <i>ut.boron</i> | dual-Pearson data for boron (energies: 15–100 keV) |
| <i>tr.boron</i> | dual-Pearson data for boron in <100> silicon with full energy, dose, tilt and rotation dependence, from University of Texas (energy: 0.5–80 keV; dose: 10^{13} – 8×10^{15} atoms/cm ² ; tilt: 0–10°; rotation: 0–45°) |
| <i>scr.boron</i> | Dual-Pearson data for boron in <100> silicon with full energy, dose, tilt, rotation, and <i>screening oxide thickness</i> dependence. Data from The University of Texas at Austin (energy: 15 keV–80 keV; dose: 10^{13} – 8×10^{15} atoms/cm ² ; tilt: 0°–10°; rotation: 0°–45°; oxide thickness: 15–400 Angstroms). |
| <i>phosphorus</i> | original phosphorus data with extended energy ranges fitted to results of amorphous Monte Carlo calculations (energies: 5–7000 keV) |
| <i>dual.pho</i> | default dual-Pearson data for phosphorus with channeling in silicon (energies: 10–200 keV) |
| <i>tr.phosphorus</i> | dual-Pearson data for phosphorus into bare <100> silicon with full energy, dose, tilt, and rotation dependence (energy: 15–180 keV; dose: 10^{13} – 8×10^{15} atoms/cm ² ; tilt: 0°–10°; rotation: 0°–45°) |
| <i>tr.indium</i> | default dual-Pearson data for indium in <100> silicon with full energy, dose, tilt, and rotation dependence (energy: 1–100 keV; dose: 10^{13} – 8×10^{15} atoms/cm ² ; tilt: 0°–10°; rotation: 0°–45°). The 200 keV parameters are based on tilt=0 implants, while 300 keV are based on tilt=7 and rotation=30 implants. |

The energy ranges shown are for implantation into silicon; the ranges may be different for other materials. The dual-Pearson model is used when one of the dual-Pearson distribution tables is specified with the **IMPL . TAB** parameter.

The default table for each impurity (except BF₂) can be changed with the **IMPL . TAB** parameter on the **IMPURITY** statement (see [page 3-239](#)).

Monte Carlo Implant Model

An alternative to the analytic implant calculation is provided by the Monte Carlo based calculation. This model is physically based and allows more general implant conditions and characteristics to be modeled. The Monte Carlo calculation allows simulation of varying rotation angles and temperature dependence of damage self-annealing, and of reflection of ions from the surface.

When **BF2** is specified with **MONTECAR**, the BF_2 implant is approximated by a boron implant with an energy of 0.2215 times the specified energy.

Point Defect Generation

Generation of point defects during the implantation is specified by the **DAMAGE** parameter. **DAMAGE** defaults to true if the **PD . TRANS** or **PD . FULL** point defect model is in effect and defaults to false if **PD . FERMI** is in effect. If **DAMAGE** is specified true and **PD . FERMI** is being used, the **PD . TRANS** point defect model is used instead. The damage model is controlled by the **D . PLUS**, **D . P1**, **D . SCALE**, **MAX . DAMA**, and **D . RECOMB** parameters. Details of the damage model are given in [Chapter 2, "Implant Damage Model"](#) on page 2-98.

Note that the **DAMAGE** parameter does not control the calculation of crystalline damage in the Monte Carlo model. It controls the retention or omission of point defect damage information for process steps subsequent to either analytic or Monte Carlo implantations. Calculation of crystalline damage is controlled using the **CRYSTAL** parameter. To reduce the effects of damage in the Monte Carlo model, use the **MAX . DAMA** parameter set to a small value, increase the temperature so that damage self-annealing becomes more dominant, or modify the final damage calculation using **D . SCALE**.

Extended Defects

If the Extended Defects AAM is enabled, the creation of dislocation loops can be modeled by specifying **L . DENS** and **L . RADIUS** (see [Chapter 2, "Dislocation Loop Model"](#) on page 2-126).

Channeling Effects

The default implant tables for arsenic, BF_2 , and phosphorus include significant channeling effects. When implanting one of these impurities through a screening oxide, it may be preferable to use one of the tables that does not include channeling, i.e., *arsenic*, *bf2*, or *phosphorus*. Some users may wish to change the default implant table for these impurities (with the **IMPL . TAB** parameter on the **IMPURITY** statement).

Boundary Conditions

Ion implantation does not obey the reflecting boundary conditions that are usually used at the left and right edges of the structure. Instead, the analytic implantation extends the structure at a reflecting boundary out to infinity while the Monte Carlo model uses the boundary condition specified by your choice of the parameters **VACUUM**, **PERIODIC**, or **REFLECT**. Thus, there is a loss of accuracy in the implanted profile unless the lateral spread of the implant distribution is small compared to the distance between the edge of the structure and the nearest mask edge. In some cases you may need to reflect the structure before implantation and truncate it afterwards to ensure the accuracy of the implanted profile.

For Monte Carlo implants, three boundary conditions are available.

1. The default, **PERIODIC**, specifies that ions leaving one side of the structure re-enter on the other side, with the same velocity.

This condition gives accurate answers for one-dimensional structures and for structures where the sequence and thickness of layers are the same at the left and right edges.

2. **REFLECT** specifies that ions hitting the edge of the structure are reflected back into the structure.

This condition is accurate for **TILT=0** implants and for pairs of implants with opposite tilts.

3. **VACUUM** specifies that ions leaving the structure through the sides are lost.

This boundary condition is a poor approximation for most structures.

There are many situations (particularly single tilted implants into two-dimensional structures) where none of the available boundary conditions is perfectly accurate. As with the analytical implant model, you lose some accuracy unless the lateral spread of the implant distribution is small compared to the distance between the edge of the structure and the nearest mask edge. Again, you may need to reflect the structure before implantation and truncate it afterwards to ensure the accuracy of the implanted profile.

TSUPREM-4 Version Considerations

The **IMPLANT** statement also accepts the parameters **DX . MIN** and **DY . MAX**. Prior to version 5.2 of TSUPREM-4, these parameters were used to specify the spacings for a rectangular grid used internally in performing the implantation calculations. These parameters are still accepted for compatibility with older versions of the program, but their values are ignored.

When **BACKSCAT** is true, TSUPREM-4 uses the same model for backscattering of ions as in TMA SUPREM-3. When **BACKSCAT** is false, the backscattering model is disabled, giving the same results as in versions of TSUPREM-4 prior to version 5.2. Note that the effect of backscattering is usually very small.

Examples

1. The following statement specifies a 100 keV implant of phosphorus with a dose of 10^{14} atoms/cm²:

```
IMPLANT PHOSPH DOSE=1E14 ENERGY=100 IN.FILE=altmom
```

The Pearson model (the default) is used for the distribution function using moment data from the file *altmom*.

2. The following statement specifies a 50 keV implant of boron from a BF₂ source with a dose of 10^{13} atoms/cm²:

```
IMPLANT BF2 DOSE=1E13 ENERGY=50 TILT=15 ROTATION=45
```

The default range statistics when the **BF2** parameter is specified are taken from the *dual.bf2* table in the standard moments file. The wafer is tilted at an angle of 15° (clockwise from the horizontal) in the plane of the simulation and rotated by an angle of 45° (clockwise from the simulation plane, facing into the wafer) about the surface normal of the wafer. This means that the typical shadowing effect of the 15° tilt is moderated by the effect of the rotation out of the plane of the simulation. The tilting of the wafer with respect to the ion beam also leads to some degree of dose reduction and foreshortening of the resulting implant distribution in the simulation plane (see [Chapter 2, “Analytic Ion Implant Models”](#) on page 2-77).

- The following statement performs a boron implant of 2×10^{13} atoms/cm² at an energy of 500 keV with ion beam tilt and rotation of 7 and 30°, respectively, using the energy, dose, tilt and rotation dependent data for boron in <100> silicon.

```
IMPLANT BORON DOSE=2E13 ENERGY=500 +
IMPL.TAB=tr.boron TILT=7 ROTATION=30 PRINT
```

The **PRINT** parameter causes a summary of the implantation to be printed on the standard output and in the output listing file. The dual-Pearson model is used with the *tr.boron* moment tables.

- The following statements show how the **MOMENT** statement can be used with the **IMPLANT** statement to specify the range statistics for an implant:

```
MOMENT SILICON RANGE=0.21 SIGMA=0.073 +
GAMMA=-0.1 KURT=12

IMPLANT BORON DOSE=1E13 +
ENERGY=50 TILT=15 MOMENTS
```

The **MOMENT** statement specifies the range statistics for implantation into silicon. Moments for implantation into other materials are read from the default moment data file.

- In the following statement, the Monte Carlo method is used to implant arsenic with a dose of 10^{14} atoms/cm², an energy of 50 keV, and a tilt angle of 45° counterclockwise from vertical.

```
IMPLANT ARSENIC DOSE=1E14 ENERGY=50 MONTECAR +
N.ION=10000 TILT=45
```

The number of ions is increased to 10,000 (from the default of 1000) to improve the accuracy in the tail of the implant profile. This implant uses the default boundary condition (periodic). Thus, ions that leave the simulation regime on one side of the structure re-enter at the other side.

- The following statements perform symmetric plus and minus 7 degree tilted implants:

```
IMPLANT ARSENIC DOSE=5E13 ENERGY=50 MONTECAR +
N.ION=10000 TILT=7 REFLECT
IMPLANT ARSENIC DOSE=5E13 ENERGY=50 MONTECAR +
N.ION=10000 TILT=-7 REFLECT
```

Specifying the reflecting boundary condition causes ions that exit from either side of the simulation regime to re-enter at the same point with the horizontal velocity reversed.

7. The following statement is similar to the previous example except that ions that exit from either side re-enter on the opposite side with their velocity unchanged:

```
IMPLANT ARSENIC DOSE=1E14 ENERGY=50 MONTECAR +  
N.ION=10000 TILT=7 PERIODIC
```

This corresponds to reproducing the simulation regime on each side, as in simulating one cell of a periodic structure.

8. The following statement simulates ion implantation using analytic models and generates damage for use as an initial condition for subsequent high-temperature processing steps:

```
IMPLANT ARSENIC DOSE=1E14 ENERGY=50 DAMAGE +  
D.PLUS=0.2
```

The **PD.TRANS** model is enabled if it is not already in use. The *plus one* component of damage is reduced to 0.2 times the concentration of implanted arsenic.

DIFFUSION

The **DIFFUSION** statement is used to model high temperature diffusion in both oxidizing and nonoxidizing ambients. **DIFFUSE** is accepted as a synonym for the **DIFFUSION** statement.

DIFFUSION

```

TIME=<n> [CONTINUE]
TEMPERAT=<n> [ {T.RATE=<n> | T.FINAL=<n>} ]
[ { DRYO2 | WETO2 | STEAM | N2O | INERT
  | AMB.1 | AMB.2 | AMB.3 | AMB.4 | AMB.5
  | ( [F.O2=<n>] [F.H2O=<n>] [F.N2O=<n>]
    [F.H2=<n>] [F.N2=<n>] [F.HCL=<n>] )
  }
]
[IMPURITY=<c> I.CONC=<n>]
[ANTIMONY=<n>] [ARSENIC=<n>] [BORON=<n>] [PHOSPHOR=<n>]
[PRESSURE=<n>] [ {P.RATE=<n> | P.FINAL=<n>} ] [HCL=<n>]
[D.RECOMB=<n>] [MOVIE=<c>] [DUMP=<n>]
    
```

| Parameter | Type | Definition |
|-----------------|---------|--|
| TIME | number | The duration of the diffusion step. Units: minutes Default: none |
| CONTINUE | logical | Indicates that this step is a continuation of a previous diffusion step. No native oxide deposition occurs and the time step is not reset. No processing steps should be specified between the preceding DIFFUSION statement and the DIFFUSION CONTINUE statement. The starting temperature of the step should be the same as the final temperature of the preceding step, and the ambient must also be the same. Default: false |
| TEMPERAT | number | The ambient temperature at the beginning of the step. Units: degrees Celsius Default: none |
| T.RATE | number | The time rate of change of the ambient temperature. Units: degrees Celsius/minute Default: 0.0 |
| T.FINAL | number | The ambient temperature at the end of the step. Units: degrees Celsius Default: TEMPERAT |
| DRYO2 | logical | Specifies that the ambient gas is dry oxygen. Default: false |
| WETO2 | logical | Specifies that the ambient gas is wet oxygen. Default: false |

| Parameter | Type | Definition |
|----------------|---------|---|
| STEAM | logical | Specifies that the ambient gas is steam. Default: false |
| N2O | logical | Specifies that the ambient gas is N ₂ O. Default: false |
| INERT | logical | Specifies that the ambient gas is inert. Default: true Synonyms: NEUTRAL, NITROGEN, ARGON |
| AMB . 1 | logical | Specifies that the ambient gas is ambient number one. Ambient number one is defined by the user with the AMBIENT statement. Default: false |
| AMB . 2 | logical | Specifies that the ambient gas is ambient number two. Ambient number two is defined by the user with the AMBIENT statement. Default: false |
| AMB . 3 | logical | Specifies that the ambient gas is ambient number three. Ambient number three is defined by the user with the AMBIENT statement. Default: false |
| AMB . 4 | logical | Specifies that the ambient gas is ambient number four. Ambient number four is defined by the user with the AMBIENT statement. Default: false |
| AMB . 5 | logical | Specifies that the ambient gas is ambient number five. Ambient number five is defined by the user with the AMBIENT statement. Default: false |
| F . O2 | number | The flow of O ₂ associated with the ambient gas. If H ₂ is also present, the O ₂ and H ₂ is assumed to react completely to form H ₂ O. The flows of O ₂ and H ₂ are reduced and the flow of H ₂ O is increased. Units: none Default: 0.0 |
| F . H2O | number | The flow of H ₂ O associated with the ambient gas. If O ₂ and H ₂ are also present, the O ₂ and H ₂ are assumed to react completely to form H ₂ O. The flows of O ₂ and H ₂ are reduced and the flow of H ₂ O is increased. Units: none Default: 0.0 |
| F . N2O | number | The flow of N ₂ O associated with the ambient gas. Units: none Default: 0.0 |
| F . H2 | number | The flow of H ₂ associated with the ambient gas. If O ₂ is also present, the O ₂ and H ₂ are assumed to react completely to form H ₂ O. The flows of O ₂ and H ₂ are reduced and the flow of H ₂ O is increased. Units: none Default: 0.0 |
| F . N2 | number | The flow of N ₂ (and other inert components) associated with the ambient gas. Units: none Default: 0.0 |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| F.HCL | number | The flow of chlorine associated with the ambient gas. Units: none Default: 0.0 |
| IMPURITY | character | The name of an impurity present in the ambient gas at the surface of the structure. Default: none |
| I.CONC | number | The concentration of IMPURITY in the ambient gas at the surface of the wafer. Units: atoms/cm ³ Default: none |
| ANTIMONY | number | The concentration of antimony in the ambient gas at the surface of the structure. Units: atoms/cm ³ Default: 0.0 Synonym: SB |
| ARSENIC | number | The concentration of arsenic in the ambient gas at the surface of the structure. Units: atoms/cm ³ Default: 0.0 Synonym: AS |
| BORON | number | The concentration of boron in the ambient gas at the surface of the structure. Units: atoms/cm ³ Default: 0.0 Synonym: B |
| PHOSPHOR | number | The concentration of phosphorus in the ambient gas at the surface of the structure. Units: atoms/cm ³ Default: 0.0 Synonym: P |
| PRESSURE | number | The total pressure of the ambient gas at the start of the step. Units: atmospheres Default: the pressure specified in the corresponding AMBIENT statement, or 1.0 if flows are specified |
| P.RATE | number | The time rate of change of the ambient gas pressure. Units: atmospheres/minute Default: 0.0 |
| P.FINAL | number | The ambient gas pressure at the end of the step. Units: atmospheres Default: PRESSURE |
| HCL | number | The percentage of chlorine present in the ambient gas. Units: percent Default: value calculated from F.HCL or specified on AMBIENT statement |
| D.RECOMB | number | The fraction of Frenkel pair implant damage remaining after initial recombination. Units: none Default: 0.0 |

| Parameter | Type | Definition |
|--------------|-----------|---|
| MOVIE | character | A string of TSUPREM-4 commands to be executed at the beginning of each time step. Multiple input statements can be given, separated by semicolons (;). Default: no commands executed |
| DUMP | number | Write a solution file after every DUMP time step. The files are readable with the LOADFILE and INITIALIZE statements. The names are of the form <i>s<time></i> , where <i><time></i> is the time in seconds from the start of the diffusion step. Units: none Default: no intermediate solutions saved |

Description

This statement specifies a diffusion step, with or without oxidation. Any impurities present in the wafer are diffused. If the wafer is exposed to a gas, predeposition and/or oxidation can be performed. If an oxidizing ambient is specified and the **VISCOUS** oxidation model is in effect, reflow of surface layers occurs.

The duration of the step must be specified with the **TIME** parameter. The ambient temperature must be specified with the **TEMPERAT** keyword (unless **CONTINUE** is specified). For linear ramping of the temperature, specify either the ramp rate (with **T .RATE**) or the temperature at the end of the step (with **T .FINAL**).

Ambient Gas

The ambient gas used during the diffusion step can be specified in one of two ways:

1. Specify one of the **DRYO2**, **WETO2**, **STEAM**, **INERT**, or **AMB .1** through **AMB .5** parameters. These select an ambient that has been predefined with the **AMBIENT** statement. The **DRYO2**, **WETO2**, **STEAM**, and **INERT** ambients are defined by the standard initialization file; the **AMB .1** through **AMB .5** ambients must be defined by the user before they are used. The predefined ambients include a default pressure and HCl percentage, which can be overridden with the **PRESSURE** and **HCL** parameters, respectively, on the **DIFFUSION** statement.
2. Define the ambient by specifying the flows of oxidizing (O₂ and H₂O) and nonoxidizing (H₂, N₂, and HCl) species. The flows can be specified as flow rates, fractions, or percentages, but the units of all the flows in a single **DIFFUSION** statement are assumed to be the same.

Any O₂ and H₂ in the gas are assumed to react (two units of H₂ for each unit of O₂) to form H₂O. Thus, the effective flow contains zero units of O₂ or H₂ (or both). If the effective gas contains nonzero amounts of both O₂ and H₂O, the oxidation rate is based on the partial pressure of H₂O.

Ambient Gas Parameters

The amount of chlorine in the ambient can be specified either by the flow of HCl (**F .HCL** parameter) or by the percentage of HCl (**HCL** parameter), but not both. If

the specified ambient contains O₂ or H₂O, oxidation takes place at interfaces between silicon dioxide and silicon or polysilicon. A native oxide (with thickness given by the **INITIAL** parameter on the **AMBIENT** statement) is deposited on any exposed silicon or polysilicon surfaces before the start of the diffusion step. **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHOR** and the combination of **IMPURITY** and **I . CONC** specify the concentration of impurities at the wafer surface for predeposition. The total pressure for an oxidizing ambient is given by **PRES-SURE**. To ramp the pressure, specify either the ramp rate (**P . RATE**) or the pressure at the end of the step (**P . FINAL**).

The parameters for oxidation are set by the **AMBIENT** statement. Diffusivities and segregation parameters are set on the various impurity statements (i.e., the **IMPURITY**, **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHORUS** statements). The oxidation and point defect models and the numerical methods to be used are specified on the **METHOD** statement. The default values for these parameters are normally set by the *s4init* file, which is read each time TSUPREM-4 is executed.

See [Chapter 2](#) for complete descriptions of the models used for diffusion and oxidation and [Appendix A](#) for a list of default model coefficients.

Oxidation Limitations

Oxidation of polysilicon is simulated only when the **COMPRESS**, **VISCOELA**, or **VISCOUS** model has been specified (with the **METHOD** statement).

The oxidation algorithms provide limited support for the case where silicon (or polysilicon), oxide, and a third material meet at a point. The results are reasonably accurate when only one of the materials in contact with oxide is oxidizing; results are less accurate if both materials in contact with oxide are oxidizing at a significant rate.

Impurities present in the ambient during an oxidation step are incorporated into the growing oxide. Note, however, that the program does not currently contain models for the changes in physical properties of heavily-doped glasses.

Reflow

Reflow of surface layers occur whenever oxidation with the **VISCOUS** model is specified. The amount of reflow is proportional to the ratio of the surface tension (specified by the **SURF . TEN** parameter on the **MATERIAL** statement) to the viscosity for each material. Reflow can occur in any material having a nonzero value of **SURF . TEN**. Only exposed layers flow due to surface tension, but underlying layers can deform due to stresses produced by reflow of the exposed layers.

Reflow in an inert ambient can be approximated by specifying an oxidizing ambient with a negligible partial pressure of oxidant, e.g., by setting **PRESSURE**=1e-6 or by a combination such as **F . N2**=1.0 and **F . O2**=1e-6. Note that a native oxide is added whenever the partial pressure of the oxidizing species is nonzero; it may be necessary to do a shallow etch to remove this oxide following a reflow step.

Examples

1. The following statement specifies a 1000-degree, 30-minute boron predeposition:

```
DIFFUSION TIME=30 TEMP=1000 BORON=1E20
```

2. The following statement calls for a 30-minute diffusion in an inert ambient:

```
DIFFUSION TIME=30 TEMP=800 T.FINAL=1000 INERT
```

The temperature is ramped from 800°C to 1000°C during the step.

3. The following statement calls for a 60-minute dry oxidation at 900°C with an ambient containing 2 percent HCl:

```
DIFFUSION TIME=60 TEMP=900 DRYO2 HCL=2
```

4. The following statement performs a 30-minute, 1000°C diffusion:

```
DIFFUSION TIME=30 TEMP=1000 +  
MOVIE="SELECT Z=log10(Boron)  
PLOT.1D X.V=1.0"
```

The boron concentration is plotted before each time step.

EPITAXY

The **EPITAXY** statement is used to grow silicon material on the exposed silicon surface of the current structure.

EPITAXY

```

TIME=<n> TEMPERAT=<n> [ {T.RATE=<n> | T.FINAL=<n>} ]
[ IMPURITY=<c> {I.CONC=<n> | I.RESIST=<n>} ]
[ ANTIMONY=<n> ] [ ARSENIC=<n> ] [ BORON=<n> ] [ PHOSPHOR=<n> ]
[ {CONCENTR | RESISTIV} ]
THICKNES=<n> [ SPACES=<n> ] [ DY=<n> ] [ YDY=<n> ] [ ARC.SPAC=<n> ]
    
```

| Parameter | Type | Definition |
|-----------------|-----------|--|
| TIME | number | The duration of the epitaxy step. Units: minutes Default: none |
| TEMPERAT | number | The ambient temperature at the beginning of the step. Units: degrees Celsius Default: none |
| T.RATE | number | The time rate of change of the ambient temperature. Units: degrees Celsius/minute Default: 0.0 |
| T.FINAL | number | The ambient temperature at the end of the step. Units: degrees Celsius Default: TEMPERAT |
| IMPURITY | character | The name of an impurity present in the ambient gas at the surface of the structure. Default: none |
| I.CONC | number | The concentration of IMPURITY in the ambient gas at the surface of the wafer. Units: atoms/cm ³ Default: none |
| I.RESIST | number | The resistivity produced by the presence of IMPURITY in the ambient gas at the surface of the wafer. Units: ohm-cm Default: none |
| ANTIMONY | number | The uniform concentration or resistivity of antimony in the deposited layer. Units: atoms/cm ³ or ohm-cm Default: zero concentration Synonym: SB |
| ARSENIC | number | The uniform concentration or resistivity of arsenic in the deposited layer. Units: atoms/cm ³ or ohm-cm Default: zero concentration Synonym: AS |

| Parameter | Type | Definition |
|-----------------|---------|--|
| BORON | number | The uniform concentration or resistivity of boron in the deposited layer. Units: atoms/cm ³ Default: zero concentration Synonym: B |
| PHOSPHOR | number | The uniform concentration or resistivity of phosphorus in the deposited layer. Units: atoms/cm ³ or ohm-cm Default: zero concentration Synonym: P |
| CONCENTR | logical | Specifies that the impurity concentration in the deposited layer is given. Default: true |
| RESISTIV | logical | Specifies that the resistivity of the deposited layer is given. Default: false |
| THICKNES | number | The thickness of the deposited layer. Units: microns Default: none |
| SPACES | number | The number of vertical grid spacings in the layer. This value is divided by the value of GRID.FAC (see “MESH” on page 3-50). Units: none Default: 1/ GRID.FAC Synonym: DIVISION |
| DY | number | The nominal grid spacing to be used in the deposited material layer at the location specified by the YDY parameter. This value is multiplied by the value of GRID.FAC (see “MESH” on page 3-50). Units: microns Default: GRID.FAC*THICKNES/SPACES Synonym: DX |
| YDY | number | The location of the nominal grid spacing specified by DY relative to the top of the deposited layer. Units: microns) Default: 0.0 Synonym: DX |
| ARC.SPAC | number | The maximum spacing allowed along an arc on the new surface. This value is multiplied by the value of GRID.FAC (see “MESH” on page 3-50). Units: microns Default: 0.5* THICKNES*GRID.FAC |

Description

This statement provides a basic epitaxy capability. Silicon is deposited on the exposed surface of the structure, and its upper surface becomes the new exposed surface. A uniform concentration of each dopant can be specified, either directly or by the resistivity of the grown material. Impurities diffuse according to the models in [Chapter 2, “Diffusion” on page 2-13](#).

The deposited material conforms to the contours of the original surface. Outside corners on the original surface produce arcs on the new surface, which are approximated by straight line segments. The maximum segment length is set by the **ARC . SPAC** parameter.

The **SPACES**, **DY**, and **YDY** parameters used to control the grid spacing in the deposited layer are scaled by the value of the **GRID . FAC** parameter on the **MESH** statement (see [Chapter 2, “Changes to the Mesh During Processing” on page 2-7](#)).

The **EPITAXY** works by alternately depositing layers and diffusing impurities in the resulting structure. The number of deposit/diffuse steps is equal to the number of grid spaces in the deposited layer, i.e., it is controlled by the **SPACES**, **DY**, and **YDY** parameters.

Example

The following statement deposits 1 micron of silicon while simultaneously diffusing at 1100°C:

```
EPITAXY THICK=1.0 TIME=180 TEMPERAT=1100 +
      ANTIMONY=1E19 SPACES=10
```

The deposition and diffusion processes are subdivided into 10 steps corresponding to the value specified by the **SPACES** parameter.



Note:

*The mobility tables used to calculate the doping from the resistive are not the same as the tables used by the **ELECTRICAL** statement. The extracted sheet resistance for the epitaxial layer does not correspond exactly to the resistivity specified during epitaxy.*

STRESS

The **STRESS** statement calculates the stresses caused by thermal mismatch between materials or due to intrinsic stress in deposited films.

STRESS

```
[ TEMP1=<n> TEMP2=<n> ]
[ NEL=<n> ]
```

| Parameter | Type | Definition |
|--------------|--------|--|
| TEMP1 | number | The initial temperature for calculating thermal mismatch stresses. Units: degrees Celsius Default: none |
| TEMP2 | number | The final temperature for calculating thermal mismatch stresses. Units: degrees Celsius Default: none |
| NEL | number | The number of nodes per triangle to use. Valid values are 6 and 7, with 6 being faster. Units: none Default: 6 |

Description

This statement calculates stresses due to thin film intrinsic stress and thermal mismatch. For thermal mismatch stresses, **TEMP1** and **TEMP2** specify the initial and final temperatures, respectively. If **TEMP1** and **TEMP2** are not specified or are equal, no thermal stresses are calculated.

The magnitude of the intrinsic stress in a thin film is specified by the **INTRIN.S** parameter on the **MATERIAL** statement. By default, the intrinsic stresses are all set to zero.

Printing and Plotting of Stresses and Displacements

The calculated stresses can be accessed for printing or plotting by referencing the variables S_{xx} , S_{yy} , and S_{xy} on the **SELECT** statement; the calculated x and y displacements can be referenced as $x.vel$ and $y.vel$. The stresses and displacements can also be displayed with the **STRESS** and **FLOW** parameters on the **PLOT.2D** statement.



Note:

*Stresses and displacements calculated by the **STRESS** statement replace any values of stress or velocity calculated by a preceding oxidation step.*

For more information, see [Chapter 2, “Modeling Stress with the STRESS Statement”](#) on page 2-75.

Reflecting Boundary Limitations

The stress calculation does not allow expansion or shrinkage perpendicular to a reflecting boundary. Thus, the results are not correct when there are reflecting boundaries on both the left and right edges of a structure. For calculating stress, you should either use a structure with an exposed boundary on the right side (created with the **BOUNDARY** statement or by etching away the right edge of the structure), or use coefficients of thermal expansion relative to the value for silicon (assuming a silicon substrate).

Example

The following statements calculate the stresses in the substrate and film arising from a nitride layer that has an intrinsic stress of 1.4×10^{10} dynes/cm², when deposited uniformly:

```
MATERIAL NITRIDE INTRIN.S=1.4E10  
STRESS
```


3.4 Output

The following statements print and plot results:

| Statement Name | Description | Page |
|-----------------------|---|-----------------------|
| SELECT | Evaluates the quantity to be printed or plotted; specifies titles and axis labels. | 3-128 |
| PRINT . 1D | Prints values of a quantity along a line through the structure. | 3-132 |
| PLOT . 1D | Plots a quantity along a line through the structure. | 3-136 |
| PLOT . 2D | Plots axes and boundaries for two-dimensional structure plots; plots grid and/or velocity and stress vectors. | 3-144 |
| CONTOUR | Plots contours in two dimensions. | 3-148 |
| COLOR | Fills areas of a two-dimensional plot. | 3-150 |
| PLOT . 3D | Plots a 3D projection plot of the selected quantity. | 3-152 |
| LABEL | Adds labels to a plot. | 3-155 |
| EXTRACT | Extracts information about a structure and prints it or writes it to a file. | 3-160 |
| ELECTRICAL | Extracts electrical information and prints it or writes it to a file. | 3-174 |
| VIEWPORT | Specifies a subset of the plotting surface to plot on. | 3-184 |

SELECT

The **SELECT** statement evaluates a quantity to be printed or plotted. It is also used to specify plot titles and axis labels.

SELECT

[Z=<c>] [TEMPERAT=<n>] [LABEL=<c>] [TITLE=<c>]

| Parameter | Type | Definition |
|-----------------|-----------|---|
| Z | character | A mathematical expression defining the quantity to be printed or plotted. If the expression contains spaces, it must be enclosed in parentheses. Default: "0" |
| TEMPERAT | number | The temperature at which the solution is to be evaluated. Units: degrees Celsius Default: last specified temperature or 800 |
| LABEL | character | The label to be used on the y axis of a one-dimensional plot, or the z axis of a three-dimensional plot. Default: the expression given by Z |
| TITLE | character | The title to be used on plots. Default: the name and version number of the program |

Description

The **SELECT** statement evaluates the quantity to be displayed by the **CONTOUR**, **PLOT . 1D**, **PRINT . 1D**, **PLOT . 2D**, and **PLOT . 3D** statements or to be extracted by the **EXTRACT** statement. No solution data can be printed or plotted until this statement is specified. The values calculated by a **SELECT** statement are used until another **SELECT** statement is specified. If the solution changes, a new **SELECT** statement is given in order for the new values to be printed or plotted.

Solution Values

The **Z** parameter specifies a mathematical expression for the quantity to be plotted. The following solution values can be used in the expression, provided that the solution is available:

| | |
|-------------------|---|
| <i>antimony</i> | antimony concentration (atoms/cm ³) |
| <i>arsenic</i> | arsenic concentration (atoms/cm ³) |
| <i>boron</i> | boron concentration (atoms/cm ³) |
| <i>phosphorus</i> | phosphorus concentration (atoms/cm ³) |

| | |
|----------------------|---|
| <i>doping</i> | net active concentration (atoms/cm ³) |
| <i>oxygen</i> | oxidant concentration (atoms/cm ³) |
| <i>silicon</i> | silicon concentration (in silicide) (atoms/cm ³) |
| <i>interstitial</i> | interstitial concentration (#/cm ³) |
| <i>vacancy</i> | vacancy concentration (#/cm ³) |
| <i>damage</i> | damage concentration (#/cm ³) |
| <i>ci.star</i> | equilibrium interstitial concentration (#/cm ³) |
| <i>cv.star</i> | equilibrium vacancy concentration (#/cm ³) |
| <i>trap</i> | concentration of filled interstitial traps (#/cm ³) |
| <i>cl_interst</i> | concentration of clustered interstitials (#/cm ³) |
| <i>dloop</i> | density of dislocation loops (#/cm ³) |
| <i>rloop</i> | radius of dislocation loops (cm) |
| <i>lgrain</i> | average polycrystalline grain size (um) |
| <i>electron</i> | concentration of electrons (#/cm ³) |
| <i>x.v</i> | x velocity (cm/sec) |
| <i>y.v</i> | y velocity (cm/sec) |
| <i>Sxx, Sxy, Syy</i> | components of the stress tensor (dynes/cm ²) |
| <i>x</i> | x coordinate (microns) |
| <i>y</i> | y coordinate (microns) |

- The names of user-specified impurities can also be used, giving the concentration of the impurity in atoms/cm³.
- The chemical concentration is reported unless the active function is used, for example, *active(phosphorus)*.
- The net concentration is defined as the sum of the donor concentrations minus the sum of the acceptor concentrations.
- The electron concentration is calculated using the assumptions of local charge neutrality and complete ionization of impurities.

Mathematical Operations and Functions

The symbols “+”, “-”, “*”, “/”, and “^” are used for the mathematical operations of addition, subtraction, multiplication, division, and exponentiation, respectively. In addition, the following functions are available:

| | |
|--|--|
| <i>active</i> | electrically active part of impurity concentration |
| <i>gb</i> | concentration of impurity in polycrystalline grain boundaries |
| <i>abs</i> | absolute value |
| <i>diffusivity</i> | diffusivity (in cm ² /sec) of an impurity or point defect species |
| <i>erf</i> | error function |
| <i>erfc</i> | complementary error function |
| <i>exp</i> | exponential |
| <i>log</i> | natural logarithm of the absolute value |
| <i>log10</i> | base-10 logarithm of the absolute value |
| <i>slog10</i> | base-10 logarithm of the absolute value times the sign of the value |
| <i>sqrt</i> | square root |
| <i>sin, cos, tan</i> | trigonometric functions (arguments in radians) |
| <i>asin, acos, atan</i> | inverse trigonometric functions (results in radians) |
| <i>sinh, asinh, cosh, acosh, tanh, atanh</i> | hyperbolic and inverse hyperbolic functions |

The *log*, *log10*, and *slog10* functions return the value 0.0 if their argument is zero; the *log*, *log10*, *slog10*, and *sqrt* functions take absolute value of their arguments. The following constant is available:

Kb Boltzmann's constant (eV/°C)



Note:

The active and net concentrations depend on the temperature. If TEMPERAT is not specified, the last processing temperature is used. If the last process step ended with a ramp to a low temperature, you may need to specify a higher value of TEMPERAT in order to obtain realistic levels of dopant activation.

Examples

1. The logarithm (base 10) of the arsenic concentration is evaluated with

```
SELECT Z=log10(Arsenic)
```

2. The phosphorus concentration minus a constant profile of 5×10^{14} is evaluated with

```
SELECT Z=(Phosphorus - 5.0e14)
```

3. The difference between the phosphorus concentration and an analytic profile is evaluated with

```
SELECT Z=(phos - 1.0e18 * exp ( y * y ) )
```

4. The excess vacancy-interstitial product is evaluated with

```
SELECT Z=(inter * vacan - ci.star * cv.star)
```

5. The diffusivity (in cm^2/sec) of boron at each point in the structure is evaluated with

```
SELECT Z=diffusivity(Boron)
```

Note that when the **PD.TRANS** or **PD.FULL** model for point defects is used, the diffusivity can be different at each point in the structure and may vary with time.

6. The following statements print junction depths:

```
SELECT Z=doping  
PRINT LAYERS X.V=0
```

The **PRINT LAYERS** statement assumes that a new layer begins whenever the selected value (net doping in this case) changes sign.

7. The following statements print the thicknesses of material layers:

```
SELECT Z=1.0  
PRINT LAYERS X.V=0
```

The **SELECT** statement specifies a constant value of one, so the **PRINT** statement only uses material boundaries to define layers. Further, when the value "1.0" is integrated over each layer, the result is just the layer thickness.

8. The following statement specifies the title to be used on the next plot:

```
SELECT TITLE="Final N-Channel Structure"
```

Because no **Z** value is specified, **Z=0** is assumed, and any attempt to print or plot solution data uses the value zero.

PRINT.1D

The **PRINT.1D** statement prints the value of the selected expression along a line through the structure. It can also print layer thickness and integrated doping information.

PRINT.1D

```

{ X.VALUE=<n> | Y.VALUE=<n>
  | ( { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE
      | POLYSILI | PHOTORES | ALUMINUM
      }
      { /MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE
      | /POLYSIL | /PHOTORE | /ALUMINU | /AMBIENT | /REFLECT
      }
    )
}
[OUT.FILE=<c> [APPEND] ]
[SPOT=<n>] [LAYERS]
[X.MIN=<n>] [X.MAX=<n>]

```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| X.VALUE | number | The x coordinate of a vertical section along which values are to be printed. Units: microns Default: 0.0 |
| Y.VALUE | number | The y coordinate of a horizontal section along which values are to be printed. Units: microns Default: none |
| MATERIAL | character | Print values in the named material, at the interface with the other specified material. Default: none |
| SILICON | logical | Print values in silicon, at the interface with the other specified material. Default: false |
| OXIDE | logical | Print values in oxide, at the interface with the other specified material. Default: false |
| OXYNITRI | logical | Print values in oxynitride, at the interface with the other specified material. Default: false |
| NITRIDE | logical | Print values in nitride, at the interface with the other specified material. Default: false |
| POLYSILI | logical | Print values in polysilicon, at the interface with the other specified material. Default: false |
| PHOTORES | logical | Print values in photoresist, at the interface with the other specified material. Default: false |

| Parameter | Type | Definition |
|-------------------|-----------|--|
| ALUMINUM | logical | Print values in aluminum, at the interface with the other specified material. Default: false |
| /MATERIA | character | Print values in the other specified material, at the interface with the named material. Default: none |
| /SILICON | logical | Print values in the other specified material, at the interface with silicon. Default: false |
| /OXIDE | logical | Print values in the other specified material, at the interface with oxide. Default: false |
| /OXYNITR | logical | Print values in the other specified material, at the interface with oxynitride. Default: false |
| /NITRIDE | logical | Print values in the other specified material, at the interface with nitride. Default: false |
| /POLYSIL | logical | Print values in the other specified material, at the interface with polysilicon. Default: false |
| /PHOTORE | logical | Print values in the other specified material, at the interface with photoresist. Default: false |
| /ALUMINU | logical | Print values in the other specified material, at the interface with aluminum. Default: false |
| /AMBIENT | logical | Print values in the other specified material, at the interface with the exposed surface (if any). Default: false Synonym: /EXPOSED, /GAS |
| /REFLECT | logical | Print values in the other specified material, at the interface with the reflecting boundary (if any). Default: false |
| OUT . FILE | character | The name of the output file in which the printed profile is saved. Default: data not written to file |
| APPEND | logical | Specifies that saved profiles are to be appended to the file specified by OUT . FILE . If APPEND is false, any existing data in the specified file are discarded before writing the new data. Default: false |
| SPOT | number | Print the coordinate along the cross-section at which the selected quantity equals the specified value. Units: units of the selected quantity Default: none |
| LAYERS | logical | Report the integral of the selected quantity over each layer of the device structure. Default: false |

| Parameter | Type | Definition |
|----------------|--------|--|
| X . MIN | number | The minimum position along the cross-section to be printed. Units: microns Default: none |
| X . MAX | number | The maximum position along the cross-section to be printed. Units: microns Default: none |

Description

The **PRINT . 1D** statement prints the values of the selected quantity along a cross-section through the device. Cross-sections are defined as vertical or horizontal (by the **X . VALUE** and **Y . VALUE** parameters, respectively), along the interface between two materials, or along a boundary of the device structure. The quantity to be printed must be specified on a **SELECT** statement preceding the **PRINT . 1D** statement. The **SPOT** parameter finds all points along the specified path at which the selected quantity equals the specified value.

Layers

If **LAYERS** is specified, the integral of the selected quantity over each layer is printed. The integration is along the path defined by the **X . VALUE**, **Y . VALUE**, or interface specification. Layers are delimited by those points along the path where the material type changes or the sign of the selected quantity changes. If a cross-section passes out a structure into the ambient and then re-enters the structure, the ambient "layer" may be omitted from the **LAYERS** output.

Interface Values

The values along the interface between two materials depend on the order in which the materials are specified. Thus **SILICON /OXIDE** and **OXIDE /SILICON** print values at the same interface, but the first prints the values in the silicon, while the second prints the values in the oxide.

Values along an interface are sorted by their x coordinates. The values printed may not be in order if there are vertical or re-entrant interfaces, or if the structure contains more than one interface between the specified materials.

Saving Profiles in a File

If **OUT . FILE** is specified, profiles printed by **PRINT . 1D** are saved in a file. The file contains the same information that is sent to the standard output except that a '/' is substituted for the first blank space in the heading line. Thus the file consists of a heading line (with a '/' in the first column) followed by a series of data lines, each containing three values: the x or y location, the solution value (specified by the **Z** parameter on a preceding **SELECT** statement), and the material type. Results produced by the **SPOT** and **LAYERS** parameters are not saved in the file. If **APPEND** is specified, the profile information is appended following any previously existing data in the output file; if **APPEND** is false (the default), existing data in the output file is discarded before the new information is saved. Files produced

by the **OUT.FILE** parameter on the **PRINT.1D** statement can be read with the **PROFILE** statement or used as targets for optimizing profiles.

Examples

1. The following statements print the boron concentration at $x=1.0$ micron between the top of the mesh and $y=3.0$ microns:

```
SELECT    Z=Boron
PRINT.1D  X.VAL=1.0 X.MAX=3.0 OUT.FILE=boron.dat
```

The profile is saved in the file *boron.dat*.

2. The following statements print the x and y coordinates of the interface between silicon and oxide:

```
SELECT    Z=y
PRINT.1D  SILICON /OXIDE
```

3. The following statements prints junction depths:

```
SELECT    Z=doping
PRINT LAYERS X.V=0
```

The **PRINT** statement assumes that a new layer begins whenever the selected value (net doping in this case) changes sign.

4. The following statements print the thicknesses of material layers:

```
SELECT    Z=1.0
PRINT LAYERS X.V=0
```

The **SELECT** statement specifies a constant value of one, so the **PRINT** statement only uses material boundaries to define layers. Further, when the value "1.0" is integrated over each layer, the result is just the layer thickness.

PLOT.1D

The **PLOT.1D** statement plots the value of the selected expression along a line through the structure.

PLOT.1D

```

{ { [ { X.VALUE=<n> | Y.VALUE=<n> } ]
  | ( { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE
      | POLYSILI | PHOTORES | ALUMINUM
      }
    { /MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE
      | /POLYSIL | /PHOTORE | /ALUMINU | /AMBIENT | /REFLECT
      }
    )
  }
| { IN.FILE=<c>
    { (TIF X.AXIS=<c> Y.AXIS=<c>)
      | ( { (COLUMN [X.COLUMN=<n>] [Y.COLUMN=<n>])
          | (ROW [X.ROW=<n>] [Y.ROW=<n>]) }
        [X.LABEL=<c>] [Y.LABEL=<c>] )
      }
      [X.SHIFT=<n>] [Y.SHIFT=<n>]
      [X.SCALE=<n>] [Y.SCALE=<n>]
      [Y.LOG] [X.LOG]
    }
  | ELECTRIC
  }
[BOUNDARY] [CLEAR] [AXES]
[SYMBOL=<n>] [CURVE] [LINE.TYP=<n>] [COLOR=<n>]
[LEFT=<n>] [RIGHT=<n>] [BOTTOM=<n>] [TOP=<n>]
[X.OFFSET=<n>] [X.LENGTH=<n>] [X.SIZE=<n>]
[Y.OFFSET=<n>] [Y.LENGTH=<n>] [Y.SIZE=<n>]
[T.SIZE=<n>]

```

| Parameter | Type | Definition |
|-----------------|-----------|--|
| X.VALUE | number | A vertical cross-section is to be plotted at this value of <i>x</i> . Units: microns Default: 0.0 |
| Y.VALUE | number | A horizontal cross-section is to be plotted at this value of <i>y</i> . Units: microns Default: none |
| MATERIAL | character | Plot a cross-section through the named material, at the interface with the other specified material. Default: none |

| Parameter | Type | Definition |
|-----------------|-----------|--|
| SILICON | logical | Plot a cross-section through silicon, at the interface with the other specified material. Default: false |
| OXIDE | logical | Plot a cross-section through oxide, at the interface with the other specified material. Default: false |
| OXYNITRI | logical | Plot a cross-section through oxynitride, at the interface with the other specified material. Default: false |
| NITRIDE | logical | Plot a cross-section through nitride, at the interface with the other specified material. Default: false |
| POLYSILI | logical | Plot a cross-section through polysilicon, at the interface with the other specified material. Default: false |
| PHOTORES | logical | Plot a cross-section through photoresist, at the interface with the other specified material. Default: false |
| ALUMINUM | logical | Plot a cross-section through aluminum, at the interface with the other specified material. Default: false |
| /MATERIA | character | Plot a cross-section through the other specified material, at the interface with the named material. Default: none |
| /SILICON | logical | Plot a cross-section through the other specified material, at the interface with silicon. Default: false |
| /OXIDE | logical | Plot a cross-section through the other specified material, at the interface with oxide. Default: false |
| /OXYNITR | logical | Plot a cross-section through the other specified material, at the interface with oxynitride. Default: false |
| /NITRIDE | logical | Plot a cross-section through the other specified material, at the interface with nitride. Default: false |
| /POLYSIL | logical | Plot a cross-section through the other specified material, at the interface with polysilicon. Default: false |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| /PHOTORE | logical | Plot a cross-section through the other specified material, at the interface with photoresist. Default: false |
| /ALUMINU | logical | Plot a cross-section through the other specified material, at the interface with aluminum. Default: false |
| /AMBIENT | logical | Plot a cross-section through the other specified material, at the interface with the exposed surface. Default: false Synonyms: /GAS , /EXPOSED |
| /REFLECT | logical | Plot a cross-section through the other specified material, at the interface with the reflecting boundary (if any). Default: false |
| IN.FILE | character | The identifier for the file containing the data to plot. This file may contain experimental data or data produced by the EXTRACT or ELECTRICAL statements. Default: none |
| TIF | logical | Specifies that the format of IN.FILE is TIF (.ivl file from Medici). Default: false |
| X.AXIS | character | The quantity used for the horizontal axis when plotting data stored in a TIF file. The label is automatically assigned with the string composite of the X.AXIS and the unit associated with X.AXIS in a TIF file. Default: none |
| Y.AXIS | character | The quantity used for the vertical axis when plotting data stored in a TIF file. The label is automatically assigned with the string composite of the Y.AXIS and the unit associated with Y.AXIS in a TIF file. Default: none |
| COLUMN | logical | Specifies that the format of IN.FILE is column-wise. Default: true |
| X.COLUMN | number | The index of the column in the file specified by the IN.FILE parameter that contains the horizontal coordinates of the plot. Units: none Default: 1 |
| Y.COLUMN | number | The index of the column in the file specified by the IN.FILE parameter that contains the vertical coordinates of the plot. Units: none Default: 2 |
| ROW | logical | Specifies that the format of IN.FILE is row-wise. Default: false |
| X.ROW | number | The index of the row(line) in the file specified by the IN.FILE parameter that contains the horizontal coordinates of the plot. Units: none Default: 1 |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| Y.ROW | number | The index of the row(line) in the file specified by the IN.FILE parameter that contains the vertical coordinates of the plot. Units: none Default: 2 |
| X.LABEL | character | The label of the horizontal axis for X.COLUMN data in the IN.FILE file. Default: "Distance (microns)" |
| Y.LABEL | character | The label of the vertical axis for Y.COLUMN data in the IN.FILE file. Default: if the SELECT statement is defined, LABEL in SELECT , otherwise, "Concentration (#/cm ³)" |
| X.SHIFT | number | The offset by which X.COLUMN data are shifted when reading IN.FILE file. Units: the same as for the X.COLUMN data Default: 0.0 |
| Y.SHIFT | number | The offset by which Y.COLUMN data are shifted when reading IN.FILE file. Units: the same as for the Y.COLUMN data Default: 0.0 |
| X.SCALE | number | The scaling factor by which X.COLUMN data are multiplied when reading IN.FILE file. Units: none Default: 1.0 |
| Y.SCALE | number | The scaling factor by which Y.COLUMN data are multiplied when reading IN.FILE file. Units: none Default: 1.0 |
| Y.LOG | logical | Specifies that the vertical axis for Y.COLUMN data in the IN.FILE file is logarithmic. Default: the current value dependent on Z quantity in the SELECT statement Synonym: LOG |
| X.LOG | logical | Specifies that the horizontal axis for X.COLUMN data in the IN.FILE file is logarithmic. Default: the current value in the previous specification. Otherwise, false. |
| ELECTRIC | logical | Specifies plotting of results from a preceding ELECTRICAL statement. Default: false |
| BOUNDARY | logical | If true, material boundaries that are crossed are indicated by dashed vertical lines on the plot. Default: true |
| CLEAR | logical | If true, the graphics screen is cleared before the graph is drawn. Default: true |

| Parameter | Type | Definition |
|-----------------|---------|--|
| AXES | logical | Specifies that axes should be drawn, using scaling information from this statement and/or the current structure. If AXES is false, no axes are drawn and scaling information from the previous plotting statement is used (i.e., LEFT , RIGHT , BOTTOM , and TOP are ignored). If AXES is false and no previous plotting statement has been given, an error is reported. Default: true Synonym: AXIS |
| SYMBOL | number | The type of centered symbol to be drawn at each point where the cross-section intersects a mesh line. This value must be in the range 1 to 15. Values of this parameter are associated with the following symbols: <ol style="list-style-type: none"> 1 Square 2 Circle 3 Triangle 4 Plus 5 Upper case X 6 Diamond 7 Up-arrow 8 Roofed upper case X 9 Upper case Z 10 Upper case Y 11 Curved square 12 Asterisk 13 Hourglass 14 Bar 15 Star Units: none Default: no symbols drawn |
| CURVE | logical | Specifies that a line is to be drawn through the data points. Default: true |
| LINE.TYP | number | The dashed line type used for the plotted data. (The axes are always drawn with line type 1.) Units: none Default: 1 |
| COLOR | number | The color of line used for the plotted data. (The axes are always drawn with color 1.) Units: none Default: 1 |
| LEFT | number | The minimum value to be plotted on the x axis. Units: microns Default: minimum x or y coordinate of the structure Synonym: X.MIN |

| Parameter | Type | Definition |
|-------------------|--------|--|
| RIGHT | number | The maximum value to be plotted on the x axis. Units: microns Default: maximum x or y coordinate of the structure Synonym: X . MAX |
| BOTTOM | number | The minimum value of the selected expression to be plotted, in units of the plot variable. Units: units of the selected expression Default: minimum value of the selected expression Synonym: Y . MIN |
| TOP | number | The maximum value of the selected expression to be plotted, in units of the plot variable. Units: units of the selected expression Default: maximum value of the selected expression Synonym: Y . MAX |
| X . OFFSET | number | The distance by which the left end of the horizontal axis is offset from the left edge of the graphics viewport. Units: cm Default: 2.0 |
| X . LENGTH | number | The length of the horizontal axis. Units: cm Default: viewport width - X . OFFSET - 1.25 |
| X . SIZE | number | The height of the characters used to label the horizontal axis. Units: cm Default: 0.25 |
| Y . OFFSET | number | The distance by which the bottom end of the vertical axis is offset from the bottom edge of the graphics viewport. Units: cm Default: 2.0 |
| Y . LENGTH | number | The length of the vertical axis. Units: cm Default: viewport height - Y . OFFSET - 1.25 |
| Y . SIZE | number | The height of the characters used to label the vertical axis. Units: cm Default: 0.25 |
| T . SIZE | number | The height of the characters in the character string used as the plot title. Units: cm Default: 0.4 |

Description

The **PLOT . 1D** statement plots cross-sections vertically or horizontally through the device, or along an interface between two materials, or along a boundary of the device. The statement has options to provide for initialization of the graphics

device and plotting of axes. The statement can optionally draw vertical lines whenever a material boundary is crossed. The vertical axis corresponds to the variable selected with the **SELECT** statement.

Limits can be specified so that only a portion of the entire device is shown, or more than one variable can be conveniently plotted. By default the limits of the x axis extend to the edges of the structure, and the y axis is scaled according to the minimum and maximum values of the selected value over the entire structure.

The quantity to be plotted must be defined by a preceding **SELECT** statement. The type of graphics device must be set, either with an **OPTION** statement or through use of a suitable default. (See “**OPTION**” on page 3-39 and Appendix B.)

If two materials are specified (e.g., **OXIDE / SILICON**), a cross-section is plotted in the first material (e.g., oxide) at the interface with the second material (e.g., silicon). Note that **OXIDE / SILICON** produces different results from **SILICON / OXIDE**. For interface plots, the points along the interface are sorted by their x coordinates; specifying interfaces containing vertical segments or reentrant angles may not produce useful plots.

Line Type and Color

The **LINE.TYP** parameter specifies the dashed line type for plotting the data. Line type 1 is solid, while types 2 through 7 are dashed lines with increasing dash sizes. Types 8 through 10 produce more complicated patterns of dashes.

The **COLOR** parameter specifies the color for plotting the data. Color 1 is the default, and produces a line that contrasts with the background color (e.g., black on white or white on black). The **COLOR** parameter has no effect on monochrome displays.

The colors produced by the **COLOR** parameter depend on the type of display being used. Where possible, the colors 2 through 7 have been set up to produce the colors red, green, blue, cyan (light blue), magenta (light purple), and yellow, in that order. Colors 8 and above produce a repeating series of 12 colors in rainbow order, from red to violet.

IN.FILE Parameter

The format of the file specified by the **IN.FILE** parameter can be either columnwise or TIF. In the case of a columnwise format, the file may contain the following two types of lines:

1. Lines that are blank or contain a slash (/) as the first nonblank character are ignored and can be used to document the file.
2. Other lines define the data at one point in the distribution. These lines must contain the following values:
 - a. Value number **X.COLUMN** is the horizontal coordinate of the point.

b. Value number **Y.COLUMN** is the vertical coordinate of the point.

If this line contains fewer than *N* numerical values in free-field format, where *N* is the maximum of indices (**X.COLUMN** and **Y.COLUMN**) for the values listed above, the line is ignored.

Since the *.ivl* log file of Medici is a TIF format file, the results of device simulation can be plotted if **TIF**, **X.AXIS** and **Y.AXIS** are specified. This capability makes it possible to easily compare the electrical calculations of TSUPREM-4 and Medici.

The transformation of data read by the **IN.FILE** parameter is as follows:

Data of X axis = X.SCALE x Data of X.COLUMN or X.AXIS + X.SHIFT
Data of Y axis = Y.SCALE x Data of Y.COLUMN or Y.AXIS + Y.SHIFT

Examples

1. The following statement clears the screen, draws a set of axes, and plots a (vertical) cross-section at $x=1.0$ micron:

```
PLOT.1D X.V=1.0 SYMB=1 ^CURVE
```

Symbol 1 (a small square) is drawn at each data point; the line through the data points is suppressed.

2. The following statement plots a cross-section at $x=2.0$ microns on the previous set of axes, without clearing the screen:

```
PLOT.1D X.V=2.0 ^AXES ^CLEAR LINE.TYP=2 COLOR=3
```

A line consisting of short dashes is used, and appears in color 3 on color displays.

PLOT . 2D

The **PLOT . 2D** statement specifies a two-dimensional plot of the device structure.

PLOT . 2D

```
[X.MIN=<n>] [X.MAX=<n>] [Y.MIN=<n>] [Y.MAX=<n>] [SCALE]
[CLEAR] [AXES] [BOUNDARY] [L.BOUND=<n>] [C.BOUND=<n>]
[GRID] [L.GRID=<n>] [C.GRID=<n>]
[ [STRESS] [FLOW] VLENG=<n> [VMAX=<n>]
  [L.COMPRES=<n>] [C.COMPRES=<n>] [L.TENSIO=<n>] [C.TENSIO=<n>]
]
[DIAMONDS]
[X.OFFSET=<n>] [X.LENGTH=<n>] [X.SIZE=<n>]
[Y.OFFSET=<n>] [Y.LENGTH=<n>] [Y.SIZE=<n>]
[T.SIZE=<n>]
```

| Parameter | Type | Definition |
|----------------|---------|---|
| X . MIN | number | The x coordinate of the left edge of the region to be plotted. Units: microns Default: left edge of the device Synonym: LEFT |
| X . MAX | number | The x coordinate of the right edge of the region to be plotted. Units: microns Default: right edge of the device Synonym: RIGHT |
| Y . MIN | number | The y coordinate of the top edge of the region to be plotted. Units: microns Default: $Y_{top} - (Y . MAX - Y_{top})/10$, where Y_{top} is the y coordinate of the top of the device Synonym: TOP |
| Y . MAX | number | The y coordinate of the bottom edge of the region to be plotted. Units: microns Default: bottom edge of the device Synonym: BOTTOM |
| SCALE | logical | If true, the x and y axes are scaled to preserve the true aspect ratio of the device. If false, the axes are scaled to fill the available plotting area. Default: false |
| CLEAR | logical | If true, the graphics screen is cleared before the graph is drawn. Default: true |
| AXES | logical | Specifies that axes should be drawn, using scaling information from this statement and/or the current structure. If AXES is false, no axes are drawn and scaling information from the previous plotting statement is used (i.e., X . MIN , X . MAX , Y . MIN , and Y . MAX are ignored). If AXES is false and no previous plotting statement has been given, an error is reported. Default: true Synonym: AXIS |

| Parameter | Type | Definition |
|-------------------|---------|--|
| BOUNDARY | logical | Plot the device boundary and material interfaces. Default: true |
| L . BOUND | number | The dashed line type used for plotting the device boundary and material interfaces. Units: none Default: 1 Synonyms: LINE . BOU , LINE . TYP |
| C . BOUND | number | The line color used for plotting the device boundary and material interfaces. Units: none Default: 1 |
| GRID | logical | Plot the grid used for the numerical solution. Default: false |
| L . GRID | number | The dashed line type used for plotting the grid. Units: none Default: 1 Synonym: LINE . GRI |
| C . GRID | number | The line color used for plotting the grid. Units: none Default: 1 |
| STRESS | logical | Plot the principal stresses in the structure. Vectors are drawn along the two principal axes of the stress tensor at each mesh point. Default: false |
| FLOW | logical | Plot vectors indicating the velocity (due to oxidation) or displacement (calculated by the STRESS statement) at each mesh point. Default: false |
| VLENG | number | The length of the vector drawn for the maximum value of stress or velocity. Units: microns Default: none |
| VMAX | number | The maximum value of stress or velocity to be plotted. Units: dynes/cm ² for stress, cm/sec for flow Default: no limit on maximum value |
| L . COMPRE | number | The dashed line type used for plotting compressive stress vectors and flow vectors. Units: none Default: 1 Synonyms: L . FLOW , L . VELOCI , LINE . COM |
| C . COMPRE | number | The line color used for plotting compressive stress vectors and flow vectors. Units: none Default: 1 Synonyms: C . FLOW , C . VELOCI |

| Parameter | Type | Definition |
|-----------------|---------|---|
| L.TENSIO | number | The dashed line type used for plotting tensile stress vectors. Units: none Default: 1 Synonym: LINE.TEN |
| C.TENSIO | number | The line color used for plotting tensile stress vectors. Units: none Default: 1 |
| DIAMONDS | logical | Plot a small symbol at each mesh point location. Default: false |
| X.OFFSET | number | The distance by which the left end of the horizontal axis is offset from the left edge of the graphics viewport. Units: cm Default: 2.0 |
| X.LENGTH | number | The length of the horizontal axis. Units: cm Default: viewport width - X.OFFSET - 1.25 |
| X.SIZE | number | The height of the characters used to label the horizontal axis. Units: cm Default: 0.25 |
| Y.OFFSET | number | The distance by which the bottom end of the vertical axis is offset from the bottom edge of the graphics viewport. Units: cm Default: 2.0 |
| Y.LENGTH | number | The length of the vertical axis. Units: cm Default: viewport height - Y.OFFSET - 1.25 |
| Y.SIZE | number | The height of the characters used to label the vertical axis. Units: cm Default: 0.25 |
| T.SIZE | number | The height of the characters in the character string used as the plot title. Units: cm Default: 0.4 |

Description

The **PLOT.2D** statement specifies a two-dimensional plot. It is useful in itself for plotting structure outlines and interface locations, grid, stress, and velocities. It is also used to plot axes for the **CONTOUR** and **COLOR** statements.

The type of graphics device must be set, either with an **OPTION** statement or use of a suitable default. See "[OPTION](#)" on page 3-39 and in [Appendix B](#).

The x and y limits are in microns, and refer to the coordinates of the structure being simulated. The x coordinate increases from left to right on the plot, while y increases from top to bottom. If **Y .MIN** is greater than **Y .MAX**, the plot is flipped. Thus, the value at the top of the plot (**Y .MIN**) is greater than the value at the bottom of the plot (**Y .MAX**). Similarly, **X .MIN** can be greater than **X .MAX**.

For **STRESS** and **FLOW** plots, the maximum vector length must be given by **VLENG**. **VLENG** is in the same units as the device structure, namely microns. If **VMAX** is specified, values of stress or velocity larger than the specified value are not plotted.

Line Type and Color

The **L .BOUND**, **L .GRID**, **L .COMPRES**, and **L .TENSIO** parameters specify dashed line types to be used for plotting various quantities. Type 1 produces a solid line while types 2 through 10 produce various dashed line styles.

The **C .BOUND**, **C .GRID**, **C .COMPRES**, and **C .TENSIO** parameters specify the colors to be used in plotting the various quantities. Color 1 (the default) gives a line color that contrasts with the background (e.g., black on white or green on black). Colors 2 through 7 give red, green, blue, cyan (light blue), magenta (light purple), and yellow on most displays, while colors 8 and above produce a repeating sequence of 12 colors in rainbow order from red to violet. Note that not all display types give these colors; the color parameters have no effect on monochrome displays.

Examples

1. The following statement plots the axes, triangular grid, and structure boundaries (in that order):

```
PLOT.2D SCALE GRID C.GRID=2
```

The grid is plotted with color 2 (usually red on color devices). The plot is scaled the same in the x and y directions in order to display the correct aspect ratio of the structure.

2. The following statement draws the material interfaces and axis between 2.0 and 5.0 microns, without clearing the screen first:

```
PLOT.2D X.MIN=2 X.MAX=5 SCALE ^CLE
```

The plot is scaled to show the true aspect ratio of the structure.

3. The following statement draws the structure boundaries and material interfaces using line type 2, and shows the grid points as diamonds:

```
PLOT.2D LINE.TYP=2 DIAMONDS ^AXES ^CLEAR
```

The plotting area is not cleared, and axes from the previous plot are used. The aspect of the structure can be distorted because **SCALE** is not specified.

CONTOUR

The **CONTOUR** statement plots a contour of the selected variable on a two-dimensional plot.

CONTOUR

VALUE=<n> [**LINE.TYP**=<n>] [**COLOR**=<n>] [**SYMBOL**=<n>]

| Parameter | Type | Definition |
|-----------------|--------|--|
| VALUE | number | The value of the selected variable at which to plot a contour. Units: units of the selected variable Default: none |
| LINE.TYP | number | The dashed line type to be used for the contour. Units: none Default: 1 |
| COLOR | number | The line color to be used for the contour. Units: none Default: 1 |
| SYMBOL | number | The type of centered symbol to be drawn at each mesh line intersection on the contour. This value must be in the range 1 to 15. Values of this parameter are associated with the following symbols: <ol style="list-style-type: none"> 1 Square 2 Circle 3 Triangle 4 Plus 5 Upper case X 6 Diamond 7 Up-arrow 8 Roofed upper case X 9 Upper case Z 10 Upper case Y 11 Curved square 12 Asterisk 13 Hourglass 14 Bar 15 Star Units: none Default: no symbols drawn |

Description

The **CONTOUR** statement draws a contour of the selected variable at the value specified. The value must be specified in units of the selected variable.

This statement assumes that a **PLOT . 2D** statement has been specified previously and the screen has been set up for plotting a two-dimensional picture. The variable to be plotted must have been specified on a preceding **SELECT** statement.



Note:

*The results of this statement are undefined unless valid **PLOT . 2D** and **SELECT** statements have been executed beforehand.*

Line Type and Color

Line type 1 gives a solid line, while types 2 through 10 give dashed lines of various sorts. Color 1 produces contours that contrast with the background (e.g., black on white or green on black). On most color devices, colors 2 through 7 give red, green, blue, cyan (light blue), magenta (light purple), and yellow, while colors 8 and above give a repeating series of 12 colors in rainbow order (from red to violet). On monochrome devices, the **COLOR** parameter has no effect.

Example

The following statements plot a series of contours with line type 2, where the boron concentration is equal to 10^{15} , 10^{16} , 10^{17} , 10^{18} , and 10^{19} :

```
SELECT  Z=log10(Boron)
FOREACH X ( 15 TO 19 STEP 1 )
  CONTOUR VALUE=X LINE.TYP=2
END
```

Note the use of the **CONTOUR** statement inside a **FOREACH** loop, to plot a series of contours.

Additional CONTOUR Notes

- Values of contours of $\log_{10}(\text{concentration})$ should be specified as exponents, e.g., 16 and not 1e16.
- Symbols are placed where the contour crosses a boundary between mesh elements. Thus, the density of symbols reflects the density of the mesh along a contour.

COLOR

The **COLOR** statement fills an area of a two-dimensional plot.

COLOR

```
[COLOR=<n>] [MIN.VALU=<n>] [MAX.VALU=<n>]
[ { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
  | ALUMINUM | PHOTORES
  }
]
```

| Parameter | Type | Definition |
|-------------------|-----------|---|
| COLOR | number | The “color” to be used for the area fill. This can correspond to an actual color, a shade of gray, or to a fill pattern, depending on the plot device. Units: none Default: 1 |
| MIN . VALU | number | The minimum value of the selected variable to be filled. Units: units of the selected variable Default: the minimum value of the selected variable |
| MAX . VALU | number | The maximum value of the selected variable to be filled. Units: units of the selected variable Default: the maximum value of the selected variable |
| MATERIAL | character | Only regions of the named material are filled. Default: none |
| SILICON | logical | Only regions of the device composed of silicon are filled. Default: false |
| OXIDE | logical | Only regions of the device composed of oxide are filled. Default: false |
| OXYNITRI | logical | Only regions of the device composed of oxynitride are filled. Default: false |
| NITRIDE | logical | Only regions of the device composed of nitride are filled. Default: false |
| POLYSILI | logical | Only regions of the device composed of polysilicon are filled. Default: false |
| ALUMINUM | logical | Only regions of the device composed of aluminum are filled. Default: false |
| PHOTORES | logical | Only regions of the device composed of photoresist are filled. Default: false |

Description

The **COLOR** statement performs area fills on isoconcentration bands in the selected variable. If either the **MIN . VALU** or **MAX . VALU** keyword is specified, regions of the device having values of the plot variable between the two values are filled. If a material type is specified, only regions of the structure composed of the specified material are filled. If neither **MIN . VALU** or **MAX . VALU** is specified, all regions of the specified material are filled. If no values or materials are specified, a warning is printed and the statement is ignored.

The quantity referred to by **MIN . VALU** and **MAX . VALU** must be specified on a **SELECT** statement preceding the **COLOR** statement. The axes for the plot must be set up by a preceding **PLOT . 2D** statement.

Plot Device Selection

The effect of the **COLOR** statement depends on the type of plot device selected. On color devices, the specified areas are colored. In most cases, colors 2 through 7 are red, green, blue, cyan (light blue), magenta (light purple), and yellow, while colors 8 and above produce a repeating series of 12 colors in rainbow order (red to violet). On some monochrome devices, different values of **COLOR** produce varying gray-scale or halftone values. On devices without hardware area fill capabilities (defined in the *s4pcap* file), a cross-hatch pattern is used.

On some devices an area fill may overwrite previously plotted information, such as grid or material boundaries. In this case the grid or boundaries can be re-plotted after doing the area fill, to ensure that they are visible.

Examples

1. The following statement fills all oxide regions with color 4:

```
COLOR OXIDE COLOR=4
```

2. The following statements fill those portions of the structure having a boron concentration between 10^{15} and 10^{16} with color 3:

```
SELECT Z=log10(Boron)  
COLOR MIN.V=15 MAX.V=16 COLOR=3
```

PLOT . 3D

Create a three-dimensional projection plot of the solution.

PLOT . 3D

```
[ THETA=<n> ] [ PHI=<n> ]
[ CLEAR ]
[ X.MIN=<n> ] [ X.MAX=<n> ] [ Y.MIN=<n> ] [ Y.MAX=<n> ]
[ Z.MIN=<n> ] [ Z.MAX=<n> ] [ LINE.TYP=<n> ] [ COLOR=<n> ]
[ NUM.CNTR=<n> ]
[ BOUNDARY ] [ L.BOUND=<n> ] [ C.BOUND=<n> ]
```

| Parameter | Type | Definition |
|-----------|---------|--|
| THETA | number | The angle above the x - y plane from which the device is viewed. If an angle of 0 is specified, the viewpoint is in the x - y plane. Values between -90 and 90 are permitted. Units: degrees Default: 45 Synonym: ELEVATIO |
| PHI | number | The angle of counter-clockwise rotation of the device in the x - y plane. Only values within 30° of 45 , 135 , 225 , and 315° should be used; values outside this range are not supported. Never use values of 0 , 90 , 180 , or 270° . Units: degrees Default: 45 Synonym: AZIMUTH |
| CLEAR | logical | Specifies that the graphics display area is to be cleared before beginning the plot. Default: true |
| X . MIN | number | The minimum value to be plotted along the x axis. Units: microns Default: the minimum x coordinate of the current structure |
| X . MAX | number | The maximum value to be plotted along the x axis. Units: microns Default: the maximum x coordinate of the current structure |
| Y . MIN | number | The minimum value to be plotted along the y axis. Units: microns Default: the minimum y coordinate of the current structure |
| Y . MAX | number | The maximum value to be plotted along the y axis. Units: microns Default: the maximum y coordinate of the current structure |
| Z . MIN | number | The minimum value on the z axis. Units: units of the plot variable Default: the minimum value of the plot variable |

| Parameter | Type | Definition |
|-------------------|---------|--|
| Z . MAX | number | The maximum value on the z axis. Units: units of the plot variable Default: the maximum value of the plot variable |
| LINE . TYP | number | The dashed line type to be used for plotting the data. Units: none Default: 1 |
| COLOR | number | The line color to be used for plotting the data. Units: none Default: 1 |
| NUM . CNTR | number | The number of grid lines drawn to represent the surface. One axis has this number of lines, the other is drawn with a number calculated from it dependent on the value of PHI . Units: none Default: 20 |
| BOUNDARY | logical | Draw the device boundaries on the $z=Z . MIN$ plane of the plot. Default: true |
| L . BOUND | number | The dashed line type used for plotting the boundary. Units: none Default: 1 Synonym: LINE . BOU |
| C . BOUND | number | The line color used for plotting the boundary. Units: none Default: 1 |

Description

The **PLOT . 3D** statement plots a “bird’s eye view” of a three-dimensional wire diagram of the selected data. The routine interpolates a series of cross-section lines and plots them with the given viewpoint parameters. Axes can be drawn and labeled.

The variable to be plotted must be specified with a **SELECT** statement prior to the **PLOT . 3D** statement. The graphics device to be used must be set with an **OPTION** statement, unless an appropriate default device is available.



Note:

*The algorithms used work only for certain values of **PHI**; values near 0, 90, 180, or 270° should be avoided.*

Line Type and Color

The **LINE.TYP** and **L.BOUND** parameters specify the style of dashed line to be used for plotting the data and boundaries, respectively. Type 1 produces a solid line, while types 2 through 10 produce various dashed styles.

The **COLOR** and **C.BOUND** parameters specify the color of lines to be used for data and boundaries. Color 1 produces lines that contrast with the background (e.g., black on white or green on black). On most color displays, colors 2 through 7 give red, green, blue, cyan (light blue), magenta (light purple), and yellow, while colors 8 and above give a repeating sequence of 12 colors in rainbow order (red through violet).

Examples

1. The following statement plots the device as viewed from straight above:

```
PLOT.3D NUM.C=30 THETA=90
```

Thirty contour lines are selected.

2. The following statement plots the bird's eye view plot from 60° above the horizon and 30° off the x axis:

```
PLOT.3D PHI=30 THETA=60 COLOR=4
```

Color 4 (usually blue) is used.

Additional PLOT . 3D Notes

- Nonrectangular areas are padded with **Z.MIN** values to make the final surface rectangular.
- **Z.MIN** and **Z.MAX** specify the limits of the z axis, but values being plotted are not clipped at these values.

LABEL

The **LABEL** statement is used to add a label to a plot.

LABEL

```
{ ( X=<n> Y=<n> [CM] ) | ( [X.CLICK=<c>] [Y.CLICK=<c>] ) }
[SIZE=<n>] [COLOR=<n>]
[ LABEL=<c> [ {LEFT | CENTER | RIGHT} ] ]
[LINE.TYP=<n>] [C.LINE=<n>] [LENGTH=<n>]
[ { ( [SYMBOL=<n>] [C.SYMBOL=<n>] )
  | ( [RECTANGL] [C.RECTAN=<n>] [W.RECTAN=<n>]
    [H.RECTAN=<n>] )
}
]
```

| Parameter | Type | Definition |
|----------------|-----------|--|
| X | number | The horizontal location corresponding to the left end, center, or right end of the character string (depending on whether LEFT , CENTER , or RIGHT is specified). If the CM parameter is specified, then this parameter specifies a location in centimeters relative to the left edge of the graphics viewport. Otherwise, this parameter specifies the location in axis units along the horizontal axis. Units: cm or horizontal axis units Default: none |
| Y | number | The vertical location corresponding to the bottom of the character string. If the CM parameter is specified, then this parameter specifies a location in centimeters relative to the bottom edge of the graphics viewport. Otherwise, this parameter specifies the location in axis units along the vertical axis. Units: cm or vertical axis units Default: none |
| CM | logical | Specifies that the X and Y parameters are locations in centimeters relative to the lower left edge of the graphics viewport. Default: false |
| X.CLICK | character | The variable name to store the x-coordinate of the position at which a mouse is clicked. Units: horizontal axis units Default: none |
| Y.CLICK | character | The variable name to store the y-coordinate of the position at which a mouse is clicked. Units: cm or vertical axis units Default: none |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| SIZE | number | The height of the characters in the character string, and the default size to be used for rectangles and centered symbols. Units: cm Default: 0.25 Synonym: C.SIZE |
| COLOR | number | The color of the label text, and the default color for rectangles, centered symbols, and line segments. Units: none Default: 1 |
| LABEL | character | The character string to be used to label the plot. Default: none |
| LEFT | logical | Specifies that the character string is to start at the position given by X and Y . Default: true if neither CENTER or RIGHT is true |
| CENTER | logical | Specifies that the character string is to be centered horizontally about the position given by X and Y . Default: false |
| RIGHT | logical | Specifies that the character string is to end at the position given by X and Y . Default: false |
| LINE.TYP | number | The dashed type of a line segment to be plotted before the label. If LABEL is not specified, the line segment is centered at the point given by X and Y . Units: none Default: 1 |
| C.LINE | number | The color of the line segment to be plotted before the label. Units: none Default: COLOR |
| LENGTH | number | The length of the line segment to be plotted before the label. Units: cm Default: 4* SIZE |

| Parameter | Type | Definition |
|-----------------|---------|--|
| SYMBOL | number | <p>The type of centered symbol to be drawn before the label. This value must be in the range 1 to 15. Values of this parameter are associated with the following symbols:</p> <ul style="list-style-type: none"> 1 Square 2 Circle 3 Triangle 4 Plus 5 Upper case X 6 Diamond 7 Up-arrow 8 Roofed upper case X 9 Upper case Z 10 Upper case Y 11 Curved square 12 Asterisk 13 Hourglass 14 Bar 15 Star <p>If LABEL is specified, the symbol is placed to the left of the label with one character space between the symbol and the label text. If LABEL is not specified, the rectangle is centered at the point given by X and Y.</p> <p>Units: none Default: 1</p> |
| C.SYMBOL | number | <p>The color of the symbol (if any).</p> <p>Units: none Default: COLOR</p> |
| RECTANGL | logical | <p>Specifies that a filled rectangle be plotted with the label. If LABEL is specified, the rectangle is placed to the left of the label with one character space between the rectangle and the label text. If LABEL is not specified, the rectangle is centered at the point given by X and Y.</p> <p>Default: false</p> |
| C.RECTAN | number | <p>The color of the filled rectangle.</p> <p>Units: none Default: COLOR</p> |
| W.RECTAN | number | <p>The width of the filled rectangle.</p> <p>Units: cm Default: SIZE</p> |
| H.RECTAN | number | <p>The height of the filled rectangle.</p> <p>Units: cm Default: SIZE</p> |

Description

The **LABEL** statement is used to add text, symbols, and/or filled rectangles to a plot. This statement is meaningless unless a **PLOT . 1D**, **PLOT . 2D**, or **PLOT . 3D** statement has been previously specified.

Label Placement

The rules for placing these annotations are as follows:

1. Labels are always placed at the location specified by **X** and **Y**. The label is left justified, centered, or right justified at this location, depending on whether **LEFT**, **CENTER**, or **RIGHT** is specified.
2. The placement of line segments depends on whether a **LABEL** is specified:
 - a. If a **LABEL** is specified, the line segment is placed one character width to the left of the label.
 - b. If no **LABEL** is specified, the line segment is centered at the location specified by **X** and **Y**.
3. The placement of symbols and filled rectangles depends on whether a line segment or label is specified:
 - a. If a line segment is specified, the symbol or filled rectangle is centered on the line segment.
 - b. If no line segment is specified but a **LABEL** is specified, the symbol or filled rectangle is placed one character width to the left of the label.
 - c. If neither a line segment nor a **LABEL** is specified, the symbol or filled rectangle is centered at the location specified by **X** and **Y**.
4. The coordinates **X** and **Y** should be in the units of the plot axes, e.g., microns or $1/\text{cm}^3$, unless **CM** is specified.



Note:

*Some graphics devices have a cursor whose position can be read by the program. On such devices, a **LABEL** statement without **X** and **Y** coordinates attempts to read the cursor position and plot the label at that point. On terminals from which the cursor position cannot be read, a **LABEL** statement without **X** and **Y** coordinates may produce unpredictable results.*

Line, Symbol, and Rectangle

If **LINE . TYP**, **C . LINE**, or **LENGTH** is specified, a line segment is drawn. **LINE . TYP** specifies the dashed line type of the line segment. Type 1 produces a solid line, while types 2 through 10 produce various styles of dashed lines. **C . LINE** specifies the color of the line and **LENGTH** gives the length. If either

SYMBOL or **C.SYMBOL** is specified, a symbol is drawn. If **RECTANGL**, **C.RECTAN**, **W.RECTAN**, or **H.RECTAN** is specified, a filled rectangle is drawn.

The **SIZE** parameter specifies the character size to be used for the label and the default width and height for filled rectangles.

Color

The **COLOR** parameter specifies the color to be used for the label and the default color for any line segment, symbol, or filled rectangle. Color 1 contrasts with the background (e.g., black on white or white on black). On most color devices, colors 2 through 7 produce red, green, blue, cyan (light blue), magenta (light purple), and yellow, while colors 8 and above give a repeating sequence of 12 colors in rainbow order (red through violet). The **COLOR** parameter has no effect on monochrome devices.

Examples

1. The following statements put two labels on the plot starting at $x=3$ microns and $y=1.4$ and 1.6 microns, with a short line of the specified type before each one:

```
LABEL X=3.0 Y=1.4 LABEL="Arsenic" LINE=3
LABEL X=3.0 Y=1.6 LABEL="Phosphorus" LINE=4
```

2. The following statement plots a label preceded by a filled rectangle:

```
LABEL RIGHT X=12.5 Y=9.0 CM LABEL="Oxide" C.RECT=2
```

The label ends at a point 12.5 cm from the x axis and 9.0 cm from the y axis. It is preceded by a rectangle filled with color 2.

3. The following statement stores the coordinate of the position at which a mouse is clicked.

```
LABEL LABEL=x X.CLICK=px Y.CLICK=py
```

The variables, px , py store the coordinate of the position at which a mouse is clicked.



Note:

*The unit of the stored value in **X.CLICK** is the same as the x -coordinate unit. However, in the case of **Y.CLICK**, the unit is cm for the distance. In **PLOT.2D** graph, for example, the variable of **Y.CLICK** stores the y -coordinate value in cm unit, while the variable of **X.CLICK** value stores the x -coordinate value in μm unit.*

EXTRACT

The **EXTRACT** statement extracts information about the structure, prints the results, and/or writes them to a file.

EXTRACT

```
[ { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI |
  PHOTORES | ALUMINUM } ]
[P1.X=<n>] [P1.Y=<n>] [P2.X=<n>] [P2.Y=<n>]
[ { /MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE |
  /POLYSIL | /PHOTORE | /ALUMINU | /AMBIENT } [CLOCKWIS] ]
[X=<n>] [Y=<n>] [ { DISTANCE=<n> | MINIMUM | MAXIMUM | VALUE=<n> } ]
{ ( [X.EXTRAC] [Y.EXTRAC] [D.EXTRAC] [VAL.EXTR] )
  | ( [INT.EXTR] [AREA.EXT] [AVG.EXTR] ) }
[PREFIX=<c>] [SEPARAT=<c>] [SUFFIX=<c>]
[WRITE] [PRINT]
[ NAME=<c> [ASSIGN]
  [ { TARGET=<n>
    | ( T.FILE=<c> [V.COLUMN=<n>] [V.LOWER=<n>] [V.UPPER=<n>]
      [T.COLUMN=<n>] [T.LOWER=<n>] [T.UPPER=<n>]
      [V.TRANSF=<c>] [T.TRANSF=<c>]
      [Z.VALUE=<c>] [SENSITIV])
    }
  [TOLERANC=<n>] [WEIGHT=<n>]
  [MIN.REL=<n>] [MIN.ABS=<n>]
]
]
[ OUT.FILE=<c> [APPEND] ] [CLOSE]
```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| MATERIAL | character | Extract parameters in the named material. Default: none |
| SILICON | logical | Extract parameters in silicon. Default: false |
| OXIDE | logical | Extract parameters in oxide. Default: false |
| OXYNITRI | logical | Extract parameters in oxynitride. Default: false |
| NITRIDE | logical | Extract parameters in nitride. Default: false |
| POLYSILI | logical | Extract parameters in polysilicon. Default: false Synonym: POLY |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| PHOTORES | logical | Extract parameters in photoresist. Default: false |
| ALUMINUM | logical | Extract parameters in aluminum. Default: false |
| P1.X | number | The <i>x</i> coordinate of the starting point of a line through the structure. Units: microns Default: the left edge of the structure if an interface is specified, or 0.0, otherwise |
| P1.Y | number | The <i>y</i> coordinate of the starting point of a line through the structure. Units: microns Default: 0.0 if an interface is specified, or the top of the structure otherwise |
| P2.X | number | The <i>x</i> coordinate of the end point of a line through the structure. Units: microns Default: the right edge of the structure, or P1.X otherwise |
| P2.Y | number | The <i>y</i> coordinate of the end point of a line through the structure. Units: microns Default: 0.0 if an interface is specified, or the bottom of the structure, otherwise |
| /MATERIA | character | Extract parameters along the interface with the named material. Default: none |
| /SILICON | logical | Extract parameters along the interface with silicon. Default: false |
| /OXIDE | logical | Extract parameters along the interface with oxide. Default: false |
| /OXYNITR | logical | Extract parameters along the interface with oxynitride. Default: false |
| /NITRIDE | logical | Extract parameters along the interface with nitride. Default: false |
| /POLYSIL | logical | Extract parameters along the interface with polysilicon. Default: false |
| /PHOTORE | logical | Extract parameters along the interface with photoresist. Default: false |
| /ALUMINU | logical | Extract parameters along the interface with aluminum. Default: false |
| /AMBIENT | logical | Extract parameters along exposed surfaces. Default: false Synonyms: /EXPOSED , /GAS |
| CLOCKWIS | logical | If the specified material interface forms a closed loop, traverse it in the clockwise direction. Default: false |

| Parameter | Type | Definition |
|-------------------|-----------|--|
| X | number | Define a line or a point on a line by the value of its x coordinate. Units: microns Default: none |
| Y | number | Define a line or a point on a line by the value of its y coordinate. Units: microns Default: 0.0 |
| DISTANCE | number | Define a point on a line by its distance from the start of the line. Units: microns Default: 1.0 |
| MINIMUM | logical | Extract at the first point on a line where the value of the selected variable is at its minimum. Default: false |
| MAXIMUM | logical | Extract at the first point on a line where the value of the selected variable is at its maximum. Default: false |
| VALUE | number | Extract at the first point on a line where the selected variable has the specified value. Units: the units of the selected variable Default: none |
| X . EXTRAC | logical | Extract the x coordinate of the specified point. Default: false |
| Y . EXTRAC | logical | Extract the y coordinate of the specified point. Default: false |
| D . EXTRAC | logical | Extract the distance of the specified point from the start of the specified line. Default: false |
| VAL . EXTR | logical | Extract the value of the selected variable at the specified point. Default: false |
| INT . EXTR | logical | Extract the integral of the selected variable along the specified line. Default: false Synonym: INTEGRAL |
| AREA . EXT | logical | Extract the length of the specified line. Default: false Synonym: THICKNES |
| AVG . EXTR | logical | Extract the average value of the selected variable along the specified line. Default: false |
| PREFIX | character | A character string to be printed before the extracted value(s). Note that if a space is desired before the first value, it must be explicitly specified as part of the prefix. Default: none |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| SEPARAT | character | A character string to be printed between extracted values if more than value is extracted. Default: " " (a single space) |
| SUFFIX | character | A character string to be printed after the extracted value(s). Note that if a space is desired after the last value, it must be explicitly specified as part of the suffix. Default: none |
| WRITE | logical | Specifies that the extracted result is to be written to the extract output file, if one is open. Default: true |
| PRINT | logical | Specifies that the extracted result is to be printed on the standard output (including the output listing file and the user's terminal). Default: true |
| NAME | character | Specifies that the extracted result is to be assigned to the specified name, as if it had been defined with a DEFINE statement or an ASSIGN statement when either TARGET or T.FILE is specified. Note that if the specified name has been defined with a DEFINE statement previously, you need to precede the EXTRACT statement with a "%" (percent character) to prevent substitution of its previous value. The NAME parameter allows extracted results to be used in subsequent extractions or simulations. Default: none |
| ASSIGN | logical | Specifies that the extracted quantity to be assigned to the variable specified by NAME is the same style as the ASSIGN statement. Default: false |
| TARGET | number | The desired value of the target being defined for an optimization loop. The optimization attempts to match the extracted value with the value of this parameter. Units: determined by the extracted values Default: none |
| T.FILE | character | The file name containing the desired values of the target being defined for an optimization loop. The desired values are in the column specified by the T.COLUMN parameter. The optimization attempts to match the extracted values with the desired values. Default: none |
| V.COLUMN | number | The index of the column in the file specified by the T.FILE parameter, which contains the variable at which the extraction is performed. Units: none Default: 1 |
| V.LOWER | number | The lower limit of the variable to be read from T.FILE . The target data at variable values less than the value of V.LOWER parameter are excluded during an optimization loop. Note that the V.LOWER specifies the value prior to the transformation by V.TRANSF . Units: The same as the unit of variable specified by V.COLUMN Default: -1e+30 |

| Parameter | Type | Definition |
|-------------------|-----------|---|
| V . UPPER | number | The upper limit of the variable to be read from T . FILE . The target data at variable values greater than the value of V . UPPER parameter are excluded during an optimization loop. Note that the V . UPPER specifies the value prior to the transformation by V . TRANSF . Units: the same as the unit of variable specified by V . COLUMN Default: 1e+30 |
| T . COLUMN | number | The index of the column in the file specified by the T . FILE parameter, which contains the desired values of the target being defined for an optimization loop. Units: none Default: 2 |
| T . LOWER | number | The lower limit of the desired value of a target to be read from T . FILE . Target values less than the value of T . LOWER parameter are excluded during an optimization loop. Note that the T . LOWER specifies the value prior to the transformation by T . TRANSF . Units: the same as the unit of variable specified by T . COLUMN Default: -1e+30 |
| T . UPPER | number | The upper limit of the desired value of a target to be read from T . FILE . Target values greater than the value of T . UPPER parameter are excluded during an optimization loop. Note that the T . UPPER specifies the value prior to the transformation by T . TRANSF . Units: the same as the unit of variable specified by T . COLUMN Default: 1e+30 |
| V . TRANSF | character | The function for transformation of the variable data read from T . FILE . The specified character string represents the function of a variable V , with which the variable data specified by V . COLUMN parameter are transformed. Default: none |
| T . TRANSF | character | The function for transformation of the desired data of a target read from T . FILE . The specified character string represents the function of a variable T , with which the target data specified by T . COLUMN parameter are transformed. Default: none |
| Z . VALUE | character | The function which generates the data instead of simulation. The specified character string represents a function of the variable V , which corresponds to the variable specified by V . COLUMN parameter. Units: the same as the unit of target data specified by T . COLUMN Default: none |
| TOLERANC | number | The RMS (root-mean-square) error for convergence criterion. An optimization loop terminates when the RMS errors of all of specified targets are smaller than TOLERANC . Units: % Default: 0.0 |
| SENSITIV | logical | Specifies the plot of sensitivity analysis of the target defined by this EXTRACT statement. This parameter works only if the PLOT parameter in the LOOP statement is true. Default: true |

| Parameter | Type | Definition |
|-----------------|-----------|--|
| WEIGHT | number | The weighting factor applied to the target being defined for an optimization loop. The weights are used to control the importance of individual targets in calculations of the error during optimization. Units: none Default: 1.0 |
| MIN.REL | number | The minimum target ratio for which relative error is used to calculate the error during optimization. This value is compared with the ratio of the absolute target value to the maximum absolute target value defined by this EXTRACT statement. Units: none Default: 1e-10 |
| MIN.ABS | number | The minimum target value for which relative error is used to calculate the error during optimization. This value is compared with the absolute target value. Units: determined by the extracted values Default: 1e-10 |
| OUT.FILE | character | The name of an extract output file to be opened. The file is opened before extracting any values, so extracted results appear in the file. Default: none |
| APPEND | logical | Specifies that extracted values are to be appended to the file specified by the OUT.FILE parameter. If APPEND is false, any existing data in the specified file is discarded when the file is opened. Default: false |
| CLOSE | logical | Specifies that the extract output file is to be closed. The file is closed after any values extracted by this EXTRACT statement are written. Default: false |

Description

The **EXTRACT** statement is used to extract values along a line through a structure or at a point. It has the following properties and uses:

- When a line is specified, layer thicknesses and integrals and averages of solution values can be extracted.
- Lines through the structure can be vertical, horizontal, or oblique; material interfaces can also be treated as lines for purposes of extraction.
- When a point is specified, the solution value, the coordinates of the point, and the distance from the start of a line can be extracted.
- Points can be specified by their coordinates or by the distance from the start of a line. You can also request extraction at the point at which the solution value is a minimum or maximum or takes on a specified value.
- Extraction can be limited to regions of specified materials, or can encompass the entire structure.
- Extracted results can be printed on the standard output or written to a separate output file. Text surrounding and between extracted values can be specified by you.

- Parameters are available for opening and closing extract output files and for appending to existing files.
- The **EXTRACT** statement can be used to define targets for optimization.

Solution Variables

Any solution variable to be used in the extraction must have been previously specified by the **Z** parameter on the **SELECT** statement (see [page 3-128](#)). Note that some extractions (e.g., layer thicknesses) do not require a solution variable. In the description that follows, the quantity specified on the **SELECT** statement is referred to as the *selected value* or *selected variable*.

Extraction Procedure

Extraction proceeds as follows:

1. If **OUT.FILE** is specified, any previously opened output file is closed and the new file is opened. If the file already exists, its contents are discarded unless **APPEND** is specified.
2. The line along which the extraction is to occur is determined as follows:
 - a. If a material interface is specified (i.e., exactly one of **MATERIAL**, **SILICON**, **OXIDE**, **OXYNITRI**, **NITRIDE**, **POLYSILI**, **PHOTORES**, and **ALUMINUM** is specified along with exactly one of **/MATERIA**, **/SILICON**, **/OXIDE**, **/OXYNITR**, **/NITRIDE**, **/POLYSIL**, **/PHOTORE**, **/ALUMINU**, and **/AMBIENT**), it is used as the line.

The extraction can be limited to a portion of an interface by specifying a starting point with **P1.X** and **P1.Y** and an ending point with **P2.X** and **P2.Y**, in which case the start of the line is taken to be the point on the interface closest to the specified starting point, and the end of the line is taken to be the point on the interface closest to the specified ending point.

By default, the starting and ending points are taken to be at the left and right edges of the structure, respectively, at $y = 0$. If more than one path exists along the interface between the starting and ending points, a choice is made based on the value of the **CLOCKWIS** parameter.

If two or more interfaces exist between the specified materials, only one is used; the interface to be used can be selected using **P1.X**, **P1.Y**, **P2.X**, and **P2.Y**.

- b. If at least one of **P1.X**, **P1.Y**, **P2.X**, and **P2.Y** (but no material interface) is specified, the points (**P1.X**, **P1.Y**) and (**P2.X**, **P2.Y**) are used as the starting and ending points of a straight line.
- c. If neither of the above is specified, the value of **Y** is used to define a horizontal line through the structure. If **Y** is not specified, the value of **X** is used to define a vertical line through the structure. If neither **Y** nor **X** is specified, then no line is defined and no values are extracted.

If one or more materials are specified, the starting point for extraction is taken to be the point where the line first enters a region of one of the specified materials, and the ending point is taken to be the point where the line first leaves one of the specified materials. If no materials are specified, the starting and ending points are taken to be the points where the line first enters and leaves the structure. Distance is always measured from the first point on the line that lies in one of the specified materials (or in any portion of the structure, if no materials were specified).

3. The point at which extraction is to occur is determined as follows:
 - a. If both **X** and **Y** are specified, their values are used as the coordinates of the point.
 - b. If **X** or **Y** is specified and a line is defined by its endpoints or as an interface, extraction occurs at the point on the line where the x or y coordinate takes the specified value.
 - c. If **DISTANCE** is specified, extraction occurs at the specified distance from the start of the line.
 - d. If **VALUE** is specified, extraction occurs at the first point along the line where the selected variable has the specified value.
 - e. If **MINIMUM** or **MAXIMUM** is specified, extraction occurs at the first point along the line where the selected variable takes on its minimum or maximum value. Only values along the line are considered when computing the minimum and maximum values.

If none of the above is specified, then no point is defined and extraction occurs along the line specified by step 2 above.

4. If a point is specified in step 3, the extracted result consists of the **PREFIX** followed (with no added spaces) by the values of the x location, y location, distance from the start of the line, and the selected value (if specified by **X.EXTRAC**, **Y.EXTRAC**, **D.EXTRAC**, and **VAL.EXTR**, respectively) separated by **SEPARAT**, followed (again with no added spaces) by the **SUFFIX**. If no point is specified in step 3, the extracted result consists of the **PREFIX** followed (with no added spaces) by the integral of the selected variable, the length of the line, and the average of the selected variable along the line (if specified by **INT.EXTR**, **AREA.EXTR**, and **AVG.EXTR**, respectively) separated by **SEPARAT**, followed (again with no added spaces) by the **SUFFIX**.
5. If **PRINT** is true, the extracted result is printed to the user's terminal and to the output listing file; if **WRITE** is true and an extract output file is open, the extracted result is written to the extract output file.
6. If **NAME** is specified, the extracted result is assigned as its value. The assigned variable follows the macro expansion rule as default. If the **ASSIGN** parameter is specified, the variable is assigned in the same way as it is assigned by the **ASSIGN** statement.
7. If **CLOSE** is specified and an extract output file is open, the file is closed.

8. If **T.FILE** is specified, extraction along a line horizontal or vertical must be specified by step 2 above, unless **Z.VALUE** is defined. The value of variable axis corresponds to the data of the column specified by **V.COLUMN**.

Some of the capabilities of the **EXTRACT** statement are illustrated by the following examples.

Targets for Optimization

The **EXTRACT** statement can be used to define targets for optimization by specifying the **NAME** parameter and either the **TARGET** or **T.FILE** parameter. Within an optimization loop, the **TARGET** and **T.FILE** parameters define the desired target values which the optimization attempts to achieve by varying the values of assigned names. The optimization attempts to simultaneously achieve the desired values of all targets defined within an optimization loop.

File Formats

The file specified by the **T.FILE** parameter is written in the same format as used for the file specified by the **IN.FILE** parameter in **PLOT.1D** statement:

1. Lines that are blank or contain a slash (/) as the first nonblank character are ignored and can be used to document the file.
2. Other lines define the data at one point in the distribution. These lines must contain the following values:
 - a. Value number **V.COLUMN** is the variable data of the point.
 - b. Value number **T.COLUMN** is the desired data of the point.

If this line contains less than N numerical values in free-field format, where N is the maximum of indices (**V.COLUMN** and **T.COLUMN**) for the values listed above, the line is ignored.

Error Calculation

Targets that are defined for an optimization loop require the calculation of the error between the extracted and desired target values. A single RMS error is obtained by combining these errors for all targets defined within the optimization loop. The RMS error is used to control the search for an optimal solution and to determine when to terminate the optimization process. The error for a target value is calculated either as relative error or as absolute error, depending on the desired target value V_{des} and the minimum significant target value V_{min} . V_{des} is either a single value specified by the **TARGET** parameter or a set of values obtained from the input data file specified by the **T.FILE** parameter. V_{min} is given by

$$V_{min} = \max(\mathbf{MIN.REL}|V_{des}|, \mathbf{MIN.ABS}) \quad \text{Equation 3-2}$$

A relative error calculation is used if V_{des} is greater than V_{min} . The relative error is given by

$$\text{error} = V_{weight} \frac{V_{ext} - V_{des}}{V_{des}} \quad \text{Equation 3-3}$$

where V_{ext} is the target value extracted by the **EXTRACT** or **ELECTRICAL** statements and V_{weight} is the product of the target weight defined by the **WEIGHT** parameter.

An absolute error calculation is used if V_{des} is less than V_{min} . The absolute error is given by

$$error = V_{weight} \frac{V_{ext} - V_{des}}{V_{min}} \quad \text{Equation 3-4}$$

Relative error calculations are performed for single target values. Both relative and absolute error calculations can be performed for a set of target values obtained from the input data file specified by the **T.FILE** parameter. In this case, absolute error calculations are used for target values that are smaller by a factor of **MIN.REL** than the maximum absolute target value in the set. The use of absolute error calculation for these target values prevents large relative errors associated with insignificant target values from controlling the optimization process.

Examples

1. Open a file named *data.ext* to receive extracted results:

```
EXTRACT OUT.FILE=data.ext
```

2. Extract the value of net doping at $(x,y) = (1.5,0.25)$ and write it to the extract file:

```
SELECT Z=DOPING  
EXTRACT X=1.5 Y=0.25 VAL.EXTR
```

3. Extract the depth of the first junction in silicon at $x=1.5$:

```
SELECT Z=DOPING  
EXTRACT SILICON X=1.5 VALUE=0.0 D.EXTRAC
```

4. Extract the width and integrated doping of the base of a bipolar transistor, assuming that the center of the emitter is at $x=1.5$:

```
SELECT Z=DOPING  
%EXTRACT SILICON X=1.5 VALUE=0.0 Y.EXTRAC NAME=J1  
%EXTRACT SILICON P1.X=1.5 P1.Y=@{J1}+0.001) +  
VALUE=0.0 Y.EXTRACT NAME=J2  
EXTRACT P1.X=1.5 P1.Y=@{J1} P2.Y=@{J2} +  
INT.EXT AREA.EXT
```

The first **EXTRACT** statement extracts the y location of the emitter-base junction by finding the first point along the line at $x=1.5$ where the net doping is zero. The second **EXTRACT** statement extracts the y location of the base-collector junction using a similar technique, but starting at a point 0.001 microns below the first junction. The third **EXTRACT** statement calculates the length and the integral of the doping along the line between the two junctions. The “%” characters are used

on the first two **EXTRACT** statements to prevent substitution of any previous definitions of the names *J1* and *J2*. Note that the results of the first two extractions are printed and/or written to the extract file; this can be prevented by specifying **^PRINT** or **^WRITE**.

5. Extract the thickness of the topmost oxide layer at $x=0$:

```
EXTRACT OXIDE X=0.0 THICKNES
```

Note that **THICKNES** is a synonym for the **AREA.EXT** parameter.

6. Extract the thickness of the gate oxide of a polysilicon-gate transistor at $x=0$:

```
%EXTRACT POLY X=0.0 DISTANCE=0.001 Y.EXTRACT +  
NAME=YPOLY  
EXTRACT OXIDE P1.X=0.0 P1.Y=@{YPOLY} THICKNES
```

The first **EXTRACT** statement finds the y location of a point 0.001 microns into the poly gate. The second **EXTRACT** statement uses this as a starting point to search for the gate oxide.

7. Find the gate length of a poly-gate MOSFET assuming that the gate is located near $y=0$:

```
EXTRACT POLY /OXIDE ^CLOCKWIS AREA.EXT
```

This statement finds the length of the polysilicon/oxide interface, starting at the point closest to the left edge of the structure at $y=0$ and ending at the point closest to the right edge of the structure at $y=0$. The interface is traversed in the counterclockwise direction.

8. Create a file containing doping as a function of y location in silicon at $x=0$:

```
SELECT Z=DOPING  
EXTRACT OUT.FILE=profile +  
PREFIX="/Net doping vs. Y"  
FOREACH DEPTH (0 TO 4.0 STEP 0.02)  
EXTRACT SILICON X=0.0 DISTANCE=@{DEPTH} +  
Y.EXT VAL.EXT  
END  
EXTRACT CLOSE
```

The first **EXTRACT** statement opens the output file and writes a line of text to it. The **FOREACH** loop steps through the structure from a depth of zero to four microns, in 0.02 micron increments. The second **EXTRACT** statement extracts the y location and doping values. The third **EXTRACT** statement closes the file.

9. Extract the arsenic surface concentration at $x=0.25$:

```
SELECT Z=ARSENIC  
EXTRACT SILICON X=0.25 DISTANCE=0.0 VAL.EXT
```

Optimization Examples

1. Extract the process conditions for the given oxide thickness and junction depth:

```

INIT B=1E15
LOOP OPTIMIZE

$ Specify the parameters to be optimized
ASSIGN NAME=TEMP N.V=1000 LOWER=900 +
UPPER=1200
ASSIGN NAME=DOSE N.V=1E14 LOWER=1E12 +
UPPER=1E15 LOG

$ Process
IMPLANT PHOS DOSE=@DOSE ENERGY=80
DIFFUSE TEMP=@TEMP TIME=20 DRYO2

$ Specify the targets - TOX and XJ
EXTRACT NAME=TOX X=0 THICKNESS +
OXIDE TARGET=0.06
SELECT Z=DOPING
EXTRACT NAME=XJ X=0 VALUE=0 D.EXTRAC +
TARGET=0.9

L.END
ASSIGN NAME=TOX PRINT
ASSIGN NAME=XJ PRINT

```

The value of implant dose is varied logarithmically by the **LOG** parameter during optimization because the difference between its lower and upper bound values is too large. Note that the extracted result is to be assigned to the **NAME** parameter, as if it had been defined with a **ASSIGN** statement, when either **TARGET** or **T.FILE** is specified.

2. Extract the SPICE JCAP area capacitance parameters:

```

INIT      P=1E16
IMPLANT  BORON DOSE=1E14 ENERGY=50
DIFFUSE  TEMP=1000 TIME=30 STEAM
ELECTRI  JCAP JUNCTION=1 V="0 5 0.5" +
          OUT.F=jcap.dat NAME=CJ0 V.SELECT=0.0

LOOP OPTIMIZE

$ Specify the SPICE JCAP parameter
  to be optimized
ASSIGN NAME=CJA N.V=@CJ0 LOWER=@CJ0/2 +
      UPPER=@CJ0*2
ASSIGN NAME=VJA N.V=0.7 LOWER=0.1 +
      UPPER=1.0
ASSIGN NAME=MJA N.V=0.33 LOWER=0.1
      UPPER=1.0

$ Define the JCAP model in SPICE
EXTRACT NAME=JCAP T.FILE=jcap.dat +
      Z.VALUE=@CJA/(1+V/@VJA)^@MJA

L.END

```

The **Z.VALUE** describes the modeling of a junction capacitance in SPICE. Note that a character “V” is used as a reserved keyword in the definition of the **Z.VALUE** parameter. Since the CJA is the value of a unit areal junction capacitance at zero bias, its initial value is taken at zero value of the **V.SELECT** parameter in the **ELECTRICAL** statement.

3. Extract the model coefficients concerned with oxidation-enhanced-diffusion and segregation by fitting a SIMS profile:

```

$ Extract the coefficients, THETA.0 and SEG.0
INIT      P=1E15
DEPOSIT   OXIDE THICKNES=0.03
IMPLANT   BORON DOSE=5E13 ENERGY=40

LOOP      OPTIMIZE PLOT

          ASSIGN NAME=THETA0 N.V=0.01    LOWER=0.0 +
          UPPER=0.1
          ASSIGN NAME=SEG0    N.V=1.126E3 LOWER=1.0 +
          UPPER=1E5 LOG

          INTERS SILICON /OXIDE THETA.0=@THETA0
          BORON   SILICON /OXIDE SEG.0=@SEG0
          DIFFUS  TEMP=950 TIME=30 DRYO2

          EXTRAC NAME=YSURF SILICON DISTANCE=0 +
          X=0 Y.EXTRAC ASSIGN
          SELECT Z=LOG10(BORON)
          EXTRAC NAME=boron_sims SILICON X=0 VAL.EXT +
          T.FILE=boron.sims T.LOWER=1E15 +
          V.TRANSF=V+@YSURF T.TRANSF=LOG10(T) +
          TOLER=0.1 SENS

L.END

```

This example assumes that the file `boron.sims` includes the SIMS data on a linear scale measured from a silicon sample. The **T.LOWER** parameter is used to eliminate the noise in the SIMS measurement at low concentrations. **V.TRANSF** is used to match the vertical coordinate at the surface of silicon since the segregation coefficient has the most sensitivity for the data near interface. The fitting for impurity profile is based on the logarithmic scale data because the distribution of impurity profile has a large range, so **Z** is specified as the log of the simulation result and **T.TRANSF** is used to take the log of the SIMS data. Note that the character “V” is used as a reserved keyword in the definition of the **V.TRANSF** parameter and that the character “T” is used in the definition of the **T.TRANSF** parameter.

ELECTRICAL

The **ELECTRICAL** statement is used to extract electrical characteristics.

ELECTRICAL

```
[X=<n>]
[ { ( SRP
  [ANGLE=<n>] [PITCH=<n>] [ {POINT=<n> | DEPTH=<n>} ]
  [Y.SURFAC=<n>]
)
| ( {V=<c> | (VSTART=<n> VSTOP=<n> VSTEP=<n>)}
  { ( RESISTAN [EXT.REG=<n>] [BIAS.REG=<n>] )
  | ( JCAP [JUNCTION=<n>] )
  | ( { ( MOSCAP [HIGH] [LOW] [DEEP] )
      | ( THRESHOL [VB=<n>] )
      }
    {NMOS | PMOS} [QM]
    [QSS=<n>] [GATE.WF=<n>] [GATE.ELE]
    [BULK.REG=<n>] )
  }
  [BULK.LAY=<n>] [PRINT] [DISTRIB]
)
}
]
[TEMPERAT=<n>]
[OUT.FILE=<c>]
[NAME=<c> [V.SELECT=<n>]
 { TARGET=<n> [SENSITIV]
 | T.FILE=<c> [V.COLUMN=<n>] [V.LOWER=<n>] [V.UPPER=<n>]
   [T.COLUMN=<n>] [T.LOWER=<n>] [T.UPPER=<n>]
   [V.TRANSF=<c>] [T.TRANSF=<c>] )
 [Z.VALUE]
}
[TOLERANC=<n>] [WEIGHT=<n>] [MIN.REL=<n>] [MIN.ABS=<n>]
]
```

| Parameter | Type | Definition |
|------------|---------|--|
| X | number | The <i>x</i> coordinate of a vertical section along which electrical characteristics are extracted. Units: microns Default: none |
| SRP | logical | Specifies that a simulated spreading resistance profile is to be extracted. Default: false |

| Parameter | Type | Definition |
|-------------------|-----------|---|
| ANGLE | number | The beveling angle for SRP extraction. Units: degrees Default: 2.0 |
| PITCH | number | The probing pitch for SRP extraction. Units: microns Default: 5.0 Synonyms: SPACE |
| POINT | number | The number of probing points for SRP extraction. Default: 50 |
| DEPTH | number | The beveling depth for SRP extraction. Units: microns Default: none |
| Y . SURFAC | number | The starting value of y for the SRP profile. Units: microns Default: 0.0 |
| V | character | This parameter is interpreted as a series of numeric values, separated by spaces or commas. The series must be composed of three values which is ordered by starting voltage, final voltage and incremental voltage. Units: volts Default: none |
| VSTART | number | The starting voltage. Units: volts Default: none |
| VSTOP | number | The final voltage. Units: volts Default: none |
| VSTEP | number | The incremental voltage. Units: volts Default: none |
| RESISTAN | logical | Specifies that the sheet resistance is to be extracted. Default: true |
| EXT . REG | number | The bottom-up number of the doping region in which the sheet resistance is to be extracted. Default: none Synonym: PLOT . REG |
| BIAS . REG | number | The bottom-up number of the doping region to which the given bias is applied when extracting the sheet resistance. Default: none |
| JCAP | logical | Specifies that junction capacitance is to be extracted. Default: false |

| Parameter | Type | Definition |
|-----------------|---------|--|
| JUNCTION | number | The bottom-up number of the junction at which the capacitance is to be extracted. Default: 1 |
| MOSCAP | logical | Specifies that MOS capacitance is to be extracted. Default: false |
| HIGH | logical | Select the slow DC and fast AC input signal to extract MOS capacitance. Default: true |
| LOW | logical | Select the slow DC and slow AC input signal to extract MOS capacitance. Default: false |
| DEEP | logical | Select the fast DC and fast AC input signal to extract MOS capacitance. Default: false |
| THRESHOL | logical | Specifies that the MOS threshold voltage is to be extracted. The extracted threshold voltage is the x-intercept value extrapolated from the gate bias at which the slope, G_m , is maximum. Default: false |
| VB | number | The back bias applied to the bulk of the MOS transistor. Units: volts Default: 0.0 |
| NMOS | logical | Specifies N-channel MOSFET. Default: false |
| PMOS | logical | Specifies P-channel MOSFET. Default: false |
| QM | logical | Specifies that the quantum effect is to be considered. Default: false |
| QSS | number | The surface fixed-state density at the interface between silicon and gate oxide. Units: $\#/cm^2$ Default: $1e10$ Synonym: QF |
| GATE.WF | number | The work function of the gate material. Units: volts Default: The current value for this material Synonym: WORKFUNC |
| GATE.ELE | logical | Specifies that the region defined as polysilicon gate is treated as electrodes. Default: false Synonym: CONDUCTO |
| BULK.REG | number | The bottom-up index of the doping region of the bulk of MOS transistor within BULK.LAY . Default: automatically decided |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| BULK.LAY | number | The bottom-up index of the material layer in which the extraction is to be performed. Default: the number of the first semiconductor layer |
| PRINT | logical | Specifies that the electron and hole charges, conductances, and resistances be printed for each bias. Default: true |
| DISTRIB | logical | Specifies that the spatial distributions of potential, electron concentration, and hole concentration be printed for each bias. Default: false |
| TEMPERAT | number | The device temperature used during the solutions of Poisson's equation. Units: degrees Celsius Default: 25.0 |
| OUT.FILE | character | The identifier for the file in which the electrical information is saved. Default: none |
| NAME | character | Specifies that the extracted result is to be assigned the specified name, as if it had been defined with an ASSIGN statement. The NAME parameter allows extracted results to be used in subsequent extractions or simulations. Default: none |
| V.SELECT | number | The variable value at which the assigned value to the NAME parameter is to be extracted. The V.SELECT parameter can not be used together with THRESHOL parameter. Units: determined by the extracted values Default: none |
| TARGET | number | The desired value of a target to be used for optimization. The optimization attempts to match the extracted value with the value of this parameter. Units: determined by the extracted values Default: none |
| SENSITIV | logical | Specifies the plot of sensitivity analysis of the target defined by this ELECTRICAL statement. This parameter works only if the PLOT parameter in the LOOP statement is true. Default: true |
| T.FILE | character | The file name containing the desired values of the target being defined for an optimization loop. The desired values are in the column specified by the T.COLUMN parameter. The optimization attempts to match the extracted values with the desired values. Default: none |
| V.COLUMN | number | The index of the column in the file specified by the T.FILE parameter, which contains the variable at which the extraction is performed. Units: none Default: 1 |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| V.LOWER | number | The lower limit of the variable to be read from T.FILE . The target data at variable values less than the value of V.LOWER parameter are excluded during an optimization loop. Note that the V.LOWER specifies the value prior to the transformation by V.TRANSF . Units: the same as the unit of variable specified by V.COLUMN Default: -1e+30 |
| V.UPPER | number | The upper limit of the variable to be read from T.FILE . The target data at variable values greater than the value of V.UPPER parameter are excluded during an optimization loop. Note that the V.UPPER specifies the value prior to the transformation by V.TRANSF . Units: the same as the unit of variable specified by V.COLUMN Default: 1e+30 |
| T.COLUMN | number | The index of the column in the file specified by the T.FILE parameter, which contains the desired values of the target being defined for an optimization loop. Units: none Default: 2 |
| T.LOWER | number | The lower limit of the desired value of a target to be read from T.FILE . Target values less than the value of T.LOWER parameter are excluded during an optimization loop. Note that the T.LOWER specifies the value prior to the transformation by T.TRANSF . Units: the same as the unit of variable specified by T.COLUMN Default: -1e+30 |
| T.UPPER | number | The upper limit of the desired value of a target to be read from T.FILE . Target values greater than the value of T.UPPER parameter are excluded during an optimization loop. Note that the T.UPPER specifies the value prior to the transformation by T.TRANSF . Units: the same as the unit of variable specified by T.COLUMN Default: 1e+30 |
| V.TRANSF | character | The function for transformation of the variable data read from T.FILE . The specified character string represents the function of a variable V , with which the variable data specified by V.COLUMN parameter are transformed. Default: none |
| T.TRANSF | character | The function for transformation of the desired data of a target read from T.FILE . The specified character string represents the function of a variable T , with which the target data specified by T.COLUMN parameter are transformed. Default: none |
| Z.VALUE | character | The function which generates the data instead of simulation. The specified character string represents a function of the variable V , which corresponds to the variable specified by V.COLUMN parameter. Units: the same as the unit of target data specified by T.COLUMN Default: none |

| Parameter | Type | Definition |
|-----------------|--------|--|
| TOLERANC | number | The RMS (root-mean-square) error for convergence criterion. An optimization loop terminates when the RMS errors of all of the specified targets are smaller than TOLERANC . Units: % Default: 0.0 |
| WEIGHT | number | The weighting factor applied to the target being defined for an optimization loop. The weights are used to control the importance of individual targets in calculations of the error during optimization. Units: none Default: 1.0 |
| MIN.REL | number | The minimum target ratio for which relative error is used to calculate the error during optimization. This value is compared with the ratio of the absolute target value to the maximum absolute target value defined by this ELECTRICAL statement. The use of this parameter is described at the end of this section. Units: none Default: 1e-10 |
| MIN.ABS | number | The minimum target value for which relative error is used to calculate the error during optimization. This value is compared with the absolute target value. The use of this parameter is described at the end of this section. Units: determined by the extracted values Default: 1e-10 |

Description

The **ELECTRICAL** statement solves the one-dimensional Poisson's equation along a vertical section through the structure. The coordinate of the vertical section is specified by **X**. Locations of features within the structure are specified by material layer number and by doping region numbers within layers. Layers and regions are numbered from bottom up, starting with layer or region number 1.

Files and Plotting

The extracted values can be plotted by specifying the **ELECTRIC** parameter on the next **PLOT.1D** statement. They can also be saved in the file specified by **OUT.FILE**. The file is in a text format with two columns; the first represents the independent variable (voltage or depth) and the second contains the extracted information. Extracted values saved in a file can be plotted using the **IN.FILE** parameter on the **PLOT.1D** statement.

Examples

1. The following statement shows the summary including the information of structure and the sheet resistance for each diffusion region.

```
ELECTRIC X=1.0
```

2. The following statements extract the threshold voltage and plot the gate voltage vs. the sheet conductance in channel.

```
ELECTRIC X=0.0 THRESHOLD NMOS V="0 2 0.05"
PLOT.1D ELECTRIC
```

3. The following statement extracts the threshold voltage of NMOS with a highly ($\geq 10^{20}/\text{cm}^3$) phosphorus-doped polysilicon gate.

```
ELECTRIC X=0.0 THRESHOLD NMOS V="0 2 0.05" +
GATE.WF=4.35 GATE.ELE
```

4. The following statement extracts the high-frequency MOS capacitance for each bias and saves it in file *vgvscap*, in a text format.

```
ELECTRIC X=0.0 MOSCAP NMOS V="-5 5 0.5" +
OUT.FILE=vgvscap
```

5. The following statement extracts the junction capacitance for each reverse bias.

```
ELECTRIC X=1.0 JCAP JUNCTION=2 V="0 5 0.1"
```

6. The following statements extract the sheet resistance for each bias and plot the voltage vs. sheet resistance. This gives the information about the increase of resistance due to the expansion of depletion region.

```
ELECTRIC X=2.5 RESIST V="0 5 0.5" PLOT.REG=2 +
BIAS.REG=2
PLOT.1D ELECTRICAL
```

7. The following statements show how to save the output file to be used in the Taurus WorkBench environment.

```
ELECTRIC NAME=VTH THRESHOLD NMOS V="0 3 0.1"
EXTRACT OUT.FILE=TWB.out
SELECT Z=@VTH
EXTRACT NAME=VT X=0 Y=0 VAL.EXT +
PREFIX="Vth_ext " SUFFIX=" Volts"
EXTRACT CLOSE
```


Optimization Examples

1. Extraction of process conditions obtains the desired threshold voltage.

```

INITIAL      P=1E15
DEPOSIT      OXIDE THICKNES=0.03

LOOP         OPTIMIZE
  ASSIGN     NAME=DOSE N.V=1E13 LOWER=1E11 +
            UPPER=1E14 LOG

            IMPLANT BORON DOSE=@DOSE ENERGY=40
            ETCH     OXIDE ALL
            DIFFUSE  TEMP=1000 TIME=30 DRYO2
            DIFFUSE  TEMP=950  TIME=20 INERT
            DEPOSIT  POLYSILI THICKNES=0.2

            ELECTRI  NAME=VTH THRESHOLD NMOS +
            V="0 3 0.1" GATE.WF=4.35 +
            GATE.ELE TARGET=0.7

L.END
    
```

2. Extraction of process conditions obtains the desired sheet resistance.

```

INITIAL      P=1E15
LOOP         OPTIMIZE
  ASSIGN     NAME=DOSE N.V=1E13 LOWER=1E11 +
            UPPER=1E15 LOG

            IMPLANT BORON DOSE=@DOSE ENERGY=30
            DIFFUSE  TEMP=1000 TIME=50 DRYO2

            ELECTRI  NAME=RS RESIST EXT.REG=2 TARGET=200

L.END
    
```

Quantum Effect in CV Plot

The quantum effect becomes important when the gate oxide is thinner and the substrate doping concentration is higher. The following example shows the quantum effect in the case of 68Å for the gate oxide thickness and $1.37 \times 10^{17}/\text{cm}^3$ for the substrate doping concentration. The graph shown in [Figure 3-2](#) compares three types of capacitance extraction:

1. Nonquantum effect and a polysilicon gate as conductor
2. Nonquantum effect and a polysilicon gate as semiconductor
3. Quantum effect and a polysilicon gate as semiconductor

The measured data are referenced from Rafael Rios, et al. [\[100\]](#).

The depletion in a polysilicon gate occurs in the very thin layer near the interface. Although the polysilicon for a gate electrode is heavily doped, the thin layer near the interface can be depleted as the voltage bias to the polysilicon gate increases. The formation of a thin depletion layer usually occurs when the gate voltage is higher than its threshold voltage. Therefore, the depletion effect of polysilicon can

be ignored for the subthreshold current and the threshold voltage characteristics. However, as the gate voltage increases enough to deplete the interface region in the polysilicon, the effect no longer can be neglected. If the parameter **GATE.ELE** is not specified, the polysilicon gate is considered to be a semiconductor. Accurate simulation requires regrid for making the grid dense near the interface because the depletion layer is very thin. TSUPREM-4 regrids automatically if the parameter **E.REGRID** is specified as true (default) in the **METHOD** statement.

```
# without poly-depletion effect
ELECTRIC X=0 MOSCAP NMOS V="-4 4 0.1" LOW GATE.ELE

# with poly-depletion effect
ELECTRIC X=0 MOSCAP NMOS V="-4 4 0.1" LOW
```

The introduction of the quantum effect produces a much better result. The parameter **QM** specifies that the quantum effect is to be considered. Similar to polysilicon depletion, quantization occurs near the interface so that automatic regridding is performed. The parameter **QM.YCRIT** (default: 20Å) in the **MATERIAL** statement determines how far the quantization goes.

```
# QM effect and gate material as semiconductor
ELECTRIC X=0 MOSCAP NMOS V="-4 4 0.1" LOW QM
```

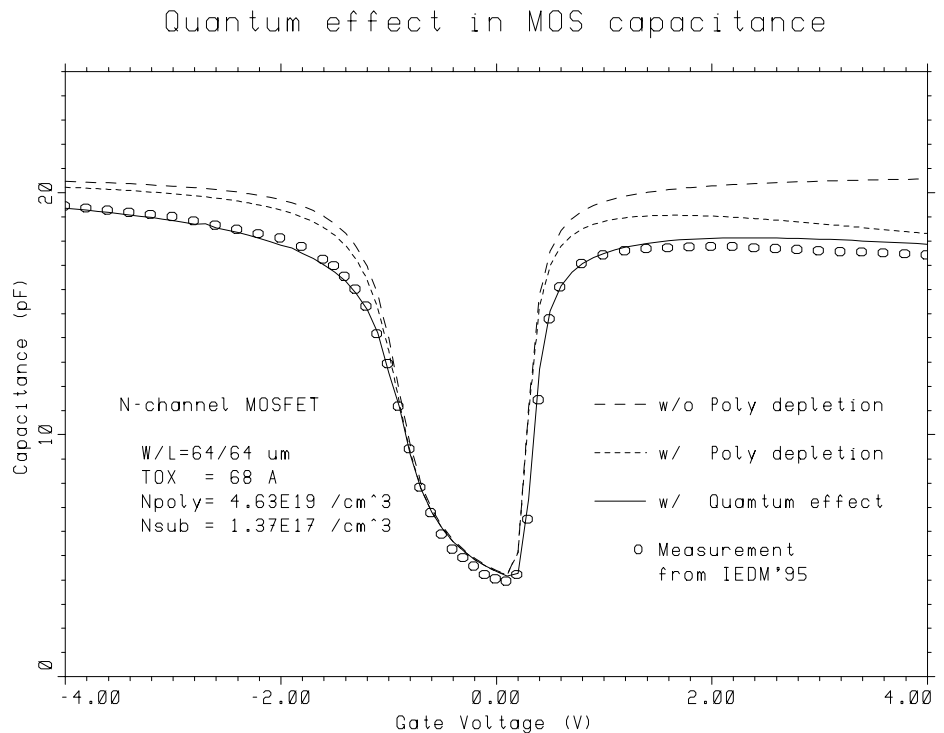


Figure 3-2 Quantum effect in MOS capacitance

Additional ELECTRICAL Notes

1. In the MOS capacitance of depletion MOSFET, only the **LOW** capacitance can be extracted.
2. The characteristics of gate material can be specified by the **MATERIAL** statement. For instance, the above example 3 can be replaced as follows:

```
MATERIAL POLYSILI CONDUCTOR WORKFUNC=4.35  
ELECTRIC X=0.0 THRESHOLD NMOS V="0 2 0.05" +  
QSS=1e10
```

3. The default table used in the **ELECTRICAL** statement is the same used in Medici, while the table used for conversion from resistivity to concentration in the **INITIALIZE** statement comes from Masetti's work [2]. The mobility table used in the **ELECTRICAL** statement can be redefined by using the **MOBILITY** statement.

VIEWPORT

The **VIEWPORT** statement limits plotting to a subset of the available drawing surface. It can be used to scale plots.

VIEWPORT

[**X.MIN**=<n>] [**X.MAX**=<n>] [**Y.MIN**=<n>] [**Y.MAX**=<n>]

| Parameter | Type | Definition |
|--------------|--------|---|
| X.MIN | number | A value between 0 and 1 specifying the left edge of the plotting region to use, as a fraction of the total width. Units: none Default: 0 |
| X.MAX | number | A value between 0 and 1 specifying the right edge of the plotting region to use, as a fraction of the total width. Units: none Default: 1 |
| Y.MIN | number | A value between 0 and 1 specifying the bottom edge of the plotting region to use, as a fraction of the total height. Units: none Default: 0 |
| Y.MAX | number | A value between 0 and 1 specifying the top edge of the plotting region to use, as a fraction of the total height. Units: none Default: 0 |

Description

This statement specifies a subset of the available plotting area on which to plot. It works with all plotting calls. The viewport remains in the current state until it is reset with a subsequent **VIEWPORT** statement. A **VIEWPORT** statement with no parameters resets the viewport to the full extent of the plotting area.

The **VIEWPORT** statement does not take effect until the next plotting statement that specifies that axes be drawn.

The **CLEAR** options of the various plotting statements clear the whole screen, not just the current **VIEWPORT** area.

Scaling Plot Size

The **VIEWPORT** statement can be used to scale plots because the default axis lengths for **PLOT . 1D**, **PLOT . 2D**, and **PLOT . 3D** depend on the viewport size. Scaling the plot size with **VIEWPORT** does not scale the size of titles, axis labels, or other objects that have absolute sizes (i.e., sizes given in centimeters).

Examples

The following statement causes the next plot to occupy 80% of the available plotting width, centered within that width:

```
VIEWPORT X.MIN=.1 X.MAX=.9
```


3.5 Models and Coefficients

The following statements specify the models and coefficients used by TSUPREM-4:

| Statement Name | Description | Page |
|---------------------|---|-----------------------|
| METHOD | Specifies models and numerical methods to be used. | 3-188 |
| EQUATION | Defines user-specified equations. Only available with the USEIT Option. | 3-204 |
| AMBIENT | Specifies parameters for the oxidation models. | 3-207 |
| MOMENT | Specifies moment parameters for ion implantation. | 3-224 |
| MATERIAL | Defines materials and specifies their properties. | 3-228 |
| IMPURITY | Defines impurities and specifies their properties. | 3-239 |
| REACTION | Defines the reactions that occur between impurities and materials at material interfaces. | 3-254 |
| MOBILITY | Defines or modifies the parameters describing carrier mobility. | 3-259 |
| INTERSTITIAL | Specifies coefficients for interstitials. | 3-265 |
| VACANCY | Specifies coefficients for vacancies. | 3-276 |
| ANTIMONY | Specifies coefficients for antimony. | 3-284 |
| ARSENIC | Specifies coefficients for arsenic. | 3-290 |
| BORON | Specifies coefficients for boron. | 3-296 |
| PHOSPHORUS | Specifies coefficients for phosphorus. | 3-302 |

METHOD

The **METHOD** statement selects models for oxidation and diffusion, and specifies numerical methods.

METHOD

```
[ {ERFC | ERF1 | ERF2 | ERF3 | VERTICAL
  | COMPRESS | VISCOELA | VISCOUS} ] [ST.HISTO]
[DY.OXIDE=<n>] [DY.EXACT] [DY.LOCAL] [GRID.OXI=<n>] [SKIP.SIL]
[ {PD.FERMI | PD.TRANS | PD.FULL} ]
[NSTREAMS=<n>] [PAIR.GRA] [PAIR.SAT] [PAIR.REC] [PD.PFLUX]
  [PD.PTIME] [PD.PREC] [PD.NREC]
[IMP.ADAP] [DIF.ADAP] [OX.ADAPT] [ERR.FAC=<n>] [UNREFINE=<n>]
[ {ACT.EQUI | ACT.TRAN | ACT.FULL} ]
[INIT.TIM=<n>] [ {TRBDF | MILNE | HYBRID | FORMULA=<c>} ]
[ {CG | GAUSS} ] [BACK=<n>] [BLK.ITLI=<n>]
[MIN.FILL] [MIN.FREQ=<n>] [MF.METH=<n>] [MF.DIST=<n>]
( [IMPURITY=<c>] [VACANCY] [INTERSTI] [ANTIMONY] [ARSENIC] [BORON]
  [PHOSPHOR] [OXIDANT] [TRAP]
  [ {LU | SOR | SIP | ICCG} ]
  [ {FULL | PART | NONE} ] [SYMMETRY]
  [ {TIME.STE | ERROR | NEWTON} ]
  [REL.ERR=<n>] [ABS.ERR=<n>]
  ( [MATERIAL=<c>] [SILICON] [POLYSILI] [OXIDE] [OXYNITRI]
    [NITRIDE] [ALUMINUM] [PHOTORES]
    [REL.ADAP=<n>] [ABS.ADAP=<n>] [MIN.SPAC=<n>] [MAX.SPAC=<n>]
  )
)
)
[OX.REL=<n>] [CONTIN.M=<n>] [VE.SMOOT=<n>]
[E.ITMIN=<n>] [E.ITMAX=<n>] [E.RELERR=<n>] [E.RVCAP=<n>] [E.AVCAP=<n>]
  [E.USEAVC]
[E.REGRID] [E.TSURF=<n>] [E.DSURF=<n>] [E.RSURF=<n>]
[ {MOB.TABL | MOB.AROR | MOB.CAUG} ]
[ ITRAP [IT.CPL] [IT.ACT] {IT.ZERO | IT.THERM | IT.STEAD}
[MODEL=<c>] [ENABLE] ]
```

| Parameter | Type | Definition |
|-----------------|---------|---|
| ERFC | logical | Use the simplest analytical model for oxidation. Default: the current value; initially false |
| ERF1 | logical | Use Guillemot's "shape 1" analytical model for oxidation. Default: the current value; initially false |
| ERF2 | logical | Use Guillemot's "shape 2" analytical model for oxidation. Default: the current value; initially false |
| ERF3 | logical | Use Guillemot's "shape 1" or "shape 2" analytical model, as appropriate, for oxidation. Default: the current value; initially false |
| VERTICAL | logical | Use the vertical growth numerical model for oxidation. Default: the current value; initially true |

| Parameter | Type | Definition |
|-----------------|---------|--|
| COMPRESS | logical | Use the compressible viscous flow numerical model for oxidation. Default: the current value; initially false |
| VISCOELA | logical | Use the viscoelastic numerical model for oxidation. Default: the current value; initially false |
| VISCOUS | logical | Use the incompressible viscous flow numerical model for oxidation. Default: the current value; initially false |
| ST.HISTO | logical | Calculate the stresses in the structure caused by material growth, thermal mismatch, intrinsic strain, and surface tension during all high-temperature steps. Only effective when the VISCOELA model is also active. Default: the current value; initially false |
| DY.OXIDE | number | The grid spacing to be used in growing oxides. This value is scaled by the value of the GRID.FAC parameter on the MESH statement (see page 3-50). Units: microns Default: the current value; initially 0.1 |
| DY.EXACT | logical | Specifies that the simulation time step should be reduced if needed to achieve precise grid spacing in growing layers. Default: the current value; initially true |
| DY.LOCAL | logical | Specifies that the grid spacing in a growing layer should be inversely proportional to the growth rate. Default: the current value; initially true |
| GRID.OXI | number | The ratio of grid spacing in a growing oxide to the grid spacing in the consumed silicon. Used only if DY.OXIDE is zero. A value of zero produces no grid in the oxide. Units: none Default: 0 for ERFC , ERF1 , and ERF2 ; 2.2 for others |
| SKIP.SIL | logical | If true, silicon regions are treated as fixed, rigid structures during oxidation, with no stress calculation in the silicon. If false, silicon is treated as a viscous material, with stress calculations. Default: the current value; initially true |
| PD.FERMI | logical | Selects a model in which the point defect concentrations depend only on the Fermi level. Equivalent to setting NSTREAMS=1 , ^PAIR.GRA , ^PAIR.SAT , ^PAIR.REC , ^PD.PFLUX , ^PD.PTIME , and ^PD.PREC . Does not model oxidation-enhanced diffusion. Recommended only where speed is more important than accuracy. This is the default selection at the start of a simulation. Equivalent to the FERMI model in older versions of TSUPREM-4. Default: the current value; initially true Synonym: FERMI |
| PD.TRANS | logical | Selects the simplest model that includes a full two-dimensional solution for the point defect concentrations. Equivalent to setting NSTREAMS=3 , ^PAIR.GRA , ^PAIR.SAT , ^PAIR.REC , ^PD.PFLUX , ^PD.PTIME , and ^PD.PREC . Models oxidation-enhanced diffusion. Recommended for routine simulations. Equivalent to the TWO.DIM model in older versions of TSUPREM-4. Default: the current value; initially false Synonym: TWO.DIM |

| Parameter | Type | Definition |
|-----------------|---------|--|
| PD.FULL | logical | Selects the most complete diffusion model available. Equivalent to setting NSTREAMS=3 , PAIR.GRA , PAIR.SAT , PAIR.REC , PD.PFLUX , PD.PTIME , and PD.PREC . Recommended for simulations where the maximum available accuracy is needed, or where it is known that high-concentration effects are important. Recommended when using the implant damage model. Default: the current value; initially false Synonym: FULL.CPL |
| NSTREAMS | number | Specifies the level of point defect diffusion modeling. The level is specified as the number of diffusion equations required to simulate diffusion with a single dopant species. A value of 1 indicates that no equations are to be solved for the point defects (i.e., the PD.FERMI model, while a value of 3 indicates that two equations are to be used, 1 for interstitials and one for vacancies (the PD.TRANS or PD.FULL models). Values other than 1 and 3 are not meaningful. Units: none Default: the current value; initially 1 |
| PAIR.GRA | logical | Specifies that the pair concentration terms be included in the gradient term (i.e., use Equations 2-30 and 2-31 rather than Equation 2-70 in Chapter 2). Default: the current value; initially false |
| PAIR.SAT | logical | Specifies that pair saturation effects (the a_m and a_n terms in Equations 2-53 and 2-54) be included in the equations for dopant diffusion. Default: the current value; initially false |
| PAIR.REC | logical | Specifies that dopant-assisted recombination effects (the K_{mv} and K_{ni} terms in Equations 2-53 and 2-54) be included in the equations for dopant diffusion. Default: the current value; initially false |
| PD.PFLUX | logical | Specifies that the dopant-defect pair fluxes (the J_m and J_n terms) be included in the equations for point defect diffusion (Equations 2-110 , 2-111 , and 2-112). Also causes the pair concentration terms to be included in the gradient term (i.e., use Equations 2-30 and 2-31 rather than Equation 2-70 in Chapter 2 .) Default: the current value; initially false |
| PD.PTIME | logical | Specifies that the time derivative of the dopant-defect pair concentrations (the dM/dt and dN/dt terms) be included in the equations for point defect diffusion (Equations 2-110 , 2-111 , and 2-112). Default: the current value; initially false |
| PD.PREC | logical | Specifies that dopant-assisted recombination effects (K_{mv} and K_{ni} terms) be included in the equation for interstitial-vacancy recombination (Equation 2-109). Default: the current value; initially false |
| PD.NREC | logical | Specifies that the recombination reduction due to trapping nitrogen be taken into account in the equation for interstitial-vacancy recombination (Equation 2-131). Default: the current value; initially false |
| IMP.ADAP | logical | Enables adaptive grid modification during ion implantation. Default: the current value; initially true |
| DIF.ADAP | logical | Enables adaptive grid modification during diffusion. Default: the current value; initially true |
| OX.ADAPT | logical | Enables adaptive grid modification based on oxidant concentration. Default: the current value; initially false |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| ERR.FAC | number | A factor that multiplies the relative errors REL.ADAP for adaptive gridding (both refinement and unrefinement). Larger values allow larger errors and produce coarser grids; smaller values reduce the error and produce finer grids. Units: none Default: the current value; initially 1.0 |
| UNREFINE | number | A factor that multiplies the relative errors REL.ADAP for mesh unrefinement. Larger values allow more unrefinement while smaller values allow less. A value of 0.0 disables mesh unrefinement. Values greater than 1.0 may cause excessive addition and removal of grid points and should be used with caution. Units: none Default: the current value; initially 0.25 |
| ACT.EQUI | logical | Use the equilibrium model for activation of dopants. Default: the current value; initially true |
| ACT.TRAN | logical | Use the transient model for activation of dopants. Available only with the Extended Defects AAM. Default: the current value; initially false |
| ACT.FULL | logical | Use the dopant-defect clustering model for activation of dopants. Default: the current value; initially false |
| INIT.TIM | number | The size of the initial time step. Units: minutes Default: the current value; initially 0.002 |
| TRBDF | logical | Use trapezoidal/backward-difference time integration with a trapezoidal step for local truncation error control. Default: the current value; initially true |
| MILNE | logical | OBSOLETE. Use Milne's time integration and local truncation error control method. This may not work in current or future versions of the program and should not be used. Default: the current value; initially false |
| HYBRID | logical | OBSOLETE. Use trapezoidal/backward-difference time integration with a divided-difference estimate of local truncation error. This may not work in current or future versions of the program and should not be used. Default: the current value; initially false |
| FORMULA | character | OBSOLETE. An equation giving the time step as a function of the time t , in seconds. Does not include local truncation error control. This may not work in current or future versions of the program and should not be used. Units: seconds Default: the current value; initially none |
| CG | logical | Use a conjugate residual method to solve the system of blocks. Default: the current value; initially true |
| GAUSS | logical | OBSOLETE. Use Gauss-Seidel iteration to solve the system of blocks. This may not work in current or future versions of the program and should not be used. Default: the current value; the initially false |

| Parameter | Type | Definition |
|-------------------|-----------|--|
| BACK | number | The number of back vectors to be used in the CG outer iteration. The maximum value is 5. More back vectors should give faster convergence but require more memory. Units: none Default: the current value; initially 18 |
| BLK . ITLI | number | The maximum number of block iterations allowed. The block iteration terminates after this many iterations whether convergence is obtained or not. Units: none Default: the current value; initially 20 |
| MIN . FILL | logical | Use a minimum fill reordering for the matrices to increase solution speed. Default: the current value; initially true |
| MIN . FREQ | number | Controls how much the solution matrix may increase before a new minimum fill reordering is done. A value of zero forces a minimum fill reordering whenever the matrix structure changes. Units: none Default: the current value; initially 1.1 |
| MF . METH | number | Selects the algorithm to use for minimum fill reordering. A value of 0 selects the algorithm used prior to version 6.6, while a value of 1 selects a simpler, faster algorithm. Units: none Default: the current value; initially 1 |
| MF . DIST | number | The maximum distance between nodes for which fill terms are included when FULL fill is specified. Smaller values decrease the time and memory required for matrix decomposition but may increase the number of iterations required for solution; very large values (larger than the size of the structure) retain all fill terms, as in versions prior to 6.6. Units: microns Default: the current value; initially 0.5 |
| IMPURITY | character | A list of one or more impurities (separated by spaces or commas) to which the remaining parameters apply. (The list must be enclosed in quotes if it contains spaces.) Default: none Synonym: IMP |
| VACANCY | logical | The remaining parameters apply to the solution for vacancies. Default: false Synonym: VACANCIE |
| INTERSTI | logical | The remaining parameters apply to the solution for interstitials. Default: false |
| ANTIMONY | logical | The remaining parameters apply to the solution for antimony. Default: false Synonym: SB |
| ARSENIC | logical | The remaining parameters apply to the solution for arsenic. Default: false Synonym: AS |

| Parameter | Type | Definition |
|-------------------|---------|--|
| BORON | logical | The remaining parameters apply to the solution for boron. Default: false Synonym: B |
| PHOSPHOR | logical | The remaining parameters apply to the solution for phosphorus. Default: false Synonym: P |
| OXIDANT | logical | The remaining parameters apply to the solution for oxidant. Default: false Synonym: OXYGEN |
| TRAP | logical | The remaining parameters apply to the solution for interstitial traps. Default: false |
| LU | logical | Use LU decomposition for solving the block equations for the selected solution variables. Default: the current value; initially true for all blocks |
| SOR | logical | OBSOLETE. Use Gauss-Seidel iteration with over-relaxation for solving the block equations for the selected solution variables. This may not work in current or future versions of the program and should not be used. Default: the current value; initially false for all blocks |
| SIP | logical | OBSOLETE. Use Stone's implicit method for solving the block equations for the selected solution variables. This may not work in current or future versions of the program and should not be used. Default: the current value; initially false for all blocks |
| ICCG | logical | OBSOLETE. Use a conjugate residual iteration for solving the block equations for the selected solution variables. This may not work in current or future versions of the program and should not be used. Default: the current value; initially false for all blocks |
| FULL | logical | Use all terms produced in factorization of the block matrices for the selected solution variables. Default: the current value; initially true for VACANCY and INTERSTI blocks, false for others |
| PART | logical | Use only nearest neighbor fill terms produced in factorization of the block matrices for the selected solution variables. Default: the current value; initially true for mobile species except VACANCY and INTERSTI , and false for all others |
| NONE | logical | Do not use fill terms produced during factorization of the block matrices for the selected solution variables. Default: the current value; initially true for immobile species, false for others |
| SYMMETRY | logical | Treat the block matrix equations for the selected solution variables as if they were symmetric. Default: the current value; initially false for all blocks Synonym: SYMMETRI |
| TIME . STE | logical | The matrices for the selected solution variables are refactored at each time step. Default: the current value; initially true for all blocks |

| Parameter | Type | Definition |
|------------------|-----------|--|
| ERROR | logical | The matrices for the selected solution variables are refactored whenever the error in the block is decreasing. Default: the current value; initially false for all blocks |
| NEWTON | logical | The matrices for the selected solution variables are refactored at each Newton step. Default: the current value; initially false for all blocks |
| REL . ERR | number | The relative error tolerance for solution of the impurity blocks for the selected solution variables. Units: none Default: the current value; initially 0.01 for all blocks |
| ABS . ERR | number | The absolute error tolerance for solution of the impurity blocks for the selected solution variables. Units: The units of the solution variable. Default: the current value; initially 1e5 for VACANCY and INTERSTI blocks, 1e9 for others |
| MATERIAL | character | A list of one or more materials (separated by spaces or commas) in which the specified values of REL . ADAP , ABS . ADAP , and MIN . SPAC apply. (The list must be enclosed in quotes if it contains spaces.) Default: none |
| SILICON | logical | The specified values of REL . ADAP , ABS . ADAP , and MIN . SPAC apply in single-crystal silicon. Default: false |
| POLYSILI | logical | The specified values of REL . ADAP , ABS . ADAP , and MIN . SPAC apply in polysilicon. Default: false |
| OXIDE | logical | The specified values of REL . ADAP , ABS . ADAP , and MIN . SPAC apply in oxide. Default: false |
| OXYNITRI | logical | The specified values of REL . ADAP , ABS . ADAP , and MIN . SPAC apply in oxynitride. Default: false |
| NITRIDE | logical | The specified values of REL . ADAP , ABS . ADAP , and MIN . SPAC apply in nitride. Default: false |
| ALUMINUM | logical | The specified values of REL . ADAP , ABS . ADAP , and MIN . SPAC apply in aluminum. Default: false |
| PHOTORES | logical | The specified values of REL . ADAP , ABS . ADAP , and MIN . SPAC apply in photoresist. Default: false |

| Parameter | Type | Definition |
|-------------------|--------|---|
| REL . ADAP | number | The relative error targets for adaptive gridding for the specified solutions and materials. A value of zero disables adaptive gridding for the specified solutions and materials. Units: none Default: the current value; see Appendix A for initial values |
| ABS . ADAP | number | The absolute error targets for adaptive gridding for the specified solutions and materials. Units: The units of the solution variable. Default: the current value; see Appendix A for initial values |
| MIN . SPAC | number | The minimum grid spacing produced by adaptive gridding for the specified solutions and materials. Units: microns Default: the current value; see Appendix A for initial values |
| MAX . SPAC | number | The maximum grid spacing produced by mesh unrefinement for the specified materials. Note that MAX . SPAC does not depend on the specification of impurities. Units: microns Default: the current value; initially 0.4 for silicon and 0.0 for other materials |
| OX . REL | number | The maximum relative error allowed when solving the stress-dependent oxidation equations. Values between 1e-4 and 1e-6 are recommended. Units: none Default: the current value; initially 1e-6 |
| CONTIN . M | number | Specifies which continuation method should be used for solving the stress-dependent oxidation equations. Values from 2 to 7 are recognized, but only 2 or 7 should be used. Units: none Default: the current value; initially 2 |
| VE . SMOOT | number | The amount of smoothing to be applied to stress values when the VISCOELA model is used. A value of 0.0 specifies a minimum of smoothing, while 1.0 specifies maximum smoothing. Units: none Default: the current value; initially 0.04 |
| E . ITMIN | number | The minimum number of iterations required for each solution of Poisson's equation in order to extract the electrical information by the ELECTRICAL statement. Units: none Default: 0 |
| E . ITMAX | number | The maximum number of iterations allowed for each solution of Poisson's equation in order to extract the electrical information by the ELECTRICAL statement. Units: none Default: 50 |

| Parameter | Type | Definition |
|-----------------|---------|---|
| E.RELERR | number | The allowed relative error used to test for convergence during the iterative numerical solution of Poisson's equation. This value determines the maximum relative change between successive approximations to the solution during iteration. This value is used to extract the electrical information by the ELECTRICAL statement. Units: none Default: 0.0001 |
| E.RVCAP | number | The ratio of AC disturbance to DC incremental voltage to extract the capacitance by the ELECTRICAL statement. Units: none Default: 0.2 |
| E.AVCAP | number | Disturbance voltage to extract the capacitance by the ELECTRICAL statement. Units: voltages Default: 0.025 |
| E.USEAVC | number | Specifies that E.AVCAP is to be used for capacitance extraction in the ELECTRICAL statement. If false, E.RVCAP is used. Default: false |
| E.REGRID | logical | Specifies that the automatic regriding is to be performed before solving Poisson's equation. Default: true |
| E.TSURF | number | The thickness of surface region to be regrided by specifying E.REGRID . Units: um Default: 0.01 |
| E.DSURF | number | The first grid space after regriding by specifying E.REGRID . Units: um Default: 0.0002 |
| E.RSURF | number | The ratio between sequent grid spaces after regriding by specifying E.REGRID . Units: none Default: 1.2 |
| MOB.TABL | logical | Specifies that the mobility tables are used to determine electron and hole mobilities. Default: true |
| MOB.AROR | logical | Specifies that the analytic mobility model based on the work of Arora, et al., is used to determine electron and hole mobilities. Default: false |
| MOB.CAUG | logical | Specifies that the analytic mobility model based on the work of Caughey, et al., is used to determine electron and hole mobilities. Default: false |
| ITRAP | logical | Use the interface trap model for segregation flux. Default: false |
| IT.CPL | logical | Specifies that each impurity occupies trap sites exclusive from each other. Default: false |

| Parameter | Type | Definition |
|-----------------|-----------|--|
| IT.ACT | logical | Specifies that the active concentration is used when the material adjacent to the interface is either silicon or polysilicon. Default: true |
| IT.ZERO | logical | Specifies that the initial value of occupied trap density is zero. Default: true |
| IT.THERM | logical | Specifies that the initial value of occupied trap density is calculated with the coefficients on the IMPURITY statement, i.e., $Q.INI.0 \exp(-Q.INI.E/kT)$. Default: false |
| IT.STEAD | logical | Specifies that the initial value of occupied trap density is calculated to satisfy the steady state in which neither accumulation nor depletion of trapped dopants at the interface occurs. Default: false |
| MODEL | character | The name of user-specified model which is defined in the IMPURITY or REACTION statements. Default: none |
| ENABLE | logical | Specifies that the user-specified model named by the MODEL parameter is turned on. Default: true |

Description

The **METHOD** statement selects the models to be used for local oxidation and point defect kinetics, and specifies numerical algorithms to be used for solving the simulation equations. Most users need be concerned only with the parameters for selecting the local oxidation model and the point defect models. Appropriate defaults for all values are given in the *s4init* file.

Oxidation Models

The oxidation model is selected by specifying one of the **ERFC**, **ERF1**, **ERF2**, **ERFG**, **VERTICAL**, **COMPRESS**, **VISCOELA**, or **VISCOUS** parameters. Parameters for the oxidation models are given on the **AMBIENT** statement. See [“AMBIENT” on page 3-207](#) for an overview of the models and their parameters, and see [Chapter 2, “Oxidation” on page 2-46](#), for a complete description.

Grid Spacing in Growing Oxide

The **DY.OXIDE**, **DY.EXACT**, **DY.LOCAL**, and **GRID.OXI** parameters control the addition of grid to growing oxide layers as described in [Chapter 2, “Addition of Nodes in a Growing Layer” on page 2-8](#). **DY.OXIDE** is the preferred parameter for controlling the grid spacing in growing oxides; the **GRID.OXI** parameter is considered obsolete and is provided only for compatibility with old input files. Note that the default value for **GRID.OXI** is set every time a **METHOD** statement is processed; for **GRID.OXI** to have an effect, it must be specified on the last **METHOD** statement before an oxidation step.

Rigid vs. Viscous Substrate

The **SKIP.SIL** parameter determines whether silicon regions are treated as a rigid substrate (**SKIP.SIL** true) or a viscous or viscoelastic material (**SKIP.SIL** false) during oxidation with the **VISCOUS** or **VISCOELA** model. Simulation of oxidation is much faster when **SKIP.SIL** is true, but stresses in the silicon are only calculated when **SKIP.SIL** is false.

To obtain the stresses in the silicon at the end of an oxidation step using the **VISCOUS** model, **SKIP.SIL** can be set to false for a very short oxidation step, and then reset to true:

```
$ Assume SKIP.SIL is true to start with
DIFFUSE TIME=60 TEMPERAT=1000 WETO2
```

```
$ Very short step with skip.sil false
$ for stress calculation
METHOD ^SKIP.SIL
DIFFUSE TIME=1E-6 TEMPERAT=1000 WETO2
```

```
$ Reset skip.sil to true for next time
METHOD SKIP.SIL
```

When the **VISCOELA** model is used, **SKIP.SIL** must be set to false for the entire simulation if stresses in the substrate are needed.

Another case in which **SKIP.SIL** should be set false is when simulating mesa structures, where the lifting of the edge of a silicon mesa may occur. Note that the **SKIP.SIL** parameter only affects oxidation with the **VISCOUS** and **VISCOELA** models; silicon is always treated as a rigid substrate when the other oxidation models are used.

Point Defect Modeling

The level of point defect modeling is set by the **NSTREAMS** parameter, and the details of the model are selected by the **PAIR.GRA**, **PAIR.SAT**, **PAIR.REC**, **PD.PFLUX**, **PD.PTIME**, and **PD.PREC** parameters. The **PD.FERMI**, **PD.TRANS**, and **PD.FULL** parameters provide convenient ways of setting the most useful combinations of **NSTREAMS**, **PAIR.GRA**, **PAIR.SAT**, **PAIR.REC**, **PD.PFLUX**, **PD.PTIME**, and **PD.PREC**.

PD.FERMI Model

The simplest (and fastest) model is **PD.FERMI**, in which the point defect concentrations depend only on the Fermi level in the silicon. (The actual point defect concentrations are not calculated, the effects of the Fermi level being included implicitly in the models of impurity diffusion.) The **PD.FERMI** model does not model oxidation-enhanced diffusion, high concentration, or implant damage effects. This model should be used when speed of the simulation is more important than accuracy, or when it is known that the features of the more complicated models are not needed.

PD . TRANS Model

The **PD . TRANS** parameter causes a full, transient simulation of the two-dimensional point defect distributions to be performed. The model includes the generation of point defects at interfaces, the diffusion of point defects into the substrate, and recombination at interfaces and in bulk silicon. The **PD . TRANS** model simulates oxidation-enhanced diffusion, but does not model high concentration effects (e.g., phosphorus kink and tail). It is less accurate than the **PD . FULL** model when simulating implant damage effects. The parameters for the point defect models are specified on the **INTERSTITIAL** and **VACANCY** statements. This model is recommended for routine simulations.

PD . FULL Model

The **PD . FULL** model is the most accurate diffusion model available, but requires the most computer time. This model includes all the effects of the **PD . TRANS** model plus the effects of dopant diffusion on the point defect concentration. It also includes pair saturation and dopant-assisted recombination effects. The **PD . FULL** model simulates oxidation-enhanced diffusion and high concentration (e.g., phosphorus kink and tail), and implant damage effects (when used with the **DAMAGE** parameter on the **IMPLANT** statement). This model is recommended only when the maximum available accuracy is needed, or when it is known that high-concentration or implant damage effects are important.

Customizing the Point Defect Models

The **PD . FERMI**, **PD . TRANS**, and **PD . FULL** parameters are processed before the other diffusion model specifications, so you can modify these models by simultaneous specification of the **PAIR . GRA**, **PAIR . SAT**, **PAIR . REC**, **PD . PFLUX**, **PD . PTIME**, and **PD . PREC** parameters. For example,

```
METHOD PD . TRANS PD . PFLUX
```

selects the basic 3-stream diffusion model, but includes the dopant-defect pair flux terms in the equations for interstitials and vacancies. This combination is useful because it provides a reasonable approximation to high-concentration effects without the computational overhead of the complete **PD . FULL** model. Similarly, the statement

```
METHOD PD . FULL ^PAIR . REC
```

removes the complicated (but generally insignificant) dopant-assisted recombination factors from the equations for dopant diffusion.

Enable/Disable User-Specified Models

The **MODEL** and **ENABLE** parameters provides the way to enable or disable the equations specified by users in the **IMPURITY** or **REACTION** statements. For example, the below statement designates that all equations named to myModel in the **IMPURITY** or **REACTION** statements must be solved.

```
METHOD MODEL=myModel ENABLE
```

On the other hand, the below statement disables the equations.

```
METHOD MODEL=myModel ^ENABLE
```

Adaptive Gridding

The **IMP .ADAP**, **DIF .ADAP**, **OX .ADAPT**, **ERR .FAC**, and **UNREFINE** parameters provide high-level control over adaptive gridding. **IMP .ADAP** enables or disables adaptive mesh refinement during ion implantation, which ensures that the grid is fine enough to resolve the implanted profile.



Note:

Adaptive grid only works with the analytical implant models; it is not available when the Monte Carlo implant model is specified.

DIF .ADAP enables or disables adaptive gridding during diffusion. Adaptive mesh refinement during diffusion ensures that accuracy is not degraded when profiles diffuse into regions where the grid is too coarse. If the value of **UNREFINE** is greater than zero, mesh unrefinement will be performed to remove mesh points that are no longer needed to ensure an accurate solution.



Note:

You may want to disable adaptive gridding during diffusion immediately following a Monte Carlo implant. Adaptive gridding can be turned back on after any statistical noise from the Monte Carlo implant has been smoothed by the diffusion process.

OX .ADAPT controls adaptive gridding in oxide based on oxidant concentration. Its use can significantly improve the accuracy of oxide shapes, especially when specifying a coarse starting grid or large value of **DY .OXIDE**. On the other hand, there are some situations in which **OX .ADAPT** can produce an unnecessarily large number of grid points. For example, when oxidizing through a poly layer it may be advisable to disable **OX .ADAPT** between the time when the oxidation first breaks through the poly and the time when the poly layer has been consumed across its entire width. This avoids adding unnecessary extra nodes in the pad oxide under the poly.

ERR .FAC controls the level of adaptive gridding by scaling the relative error targets specified by **REL .ADAP**. Larger values of **ERR .FAC** allow larger relative errors and produce coarser grids; smaller values of **ERR .FAC** reduce the relative errors, producing finer grids. **UNREFINE** controls the level of mesh unrefinement by scaling the relative error used for refinement. A value of 1.0 gives the same error level for refinement and unrefinement, while a value of 0.0 disables unrefinement.

Fine Control

Fine control over adaptive gridding is provided with the **REL . ADAP**, **ABS . ADAP**, **MIN . SPAC**, and **MAX . SPAC** parameters. **REL . ADAP** and **ABS . ADAP** specify the relative and absolute error targets for the specified solution variables and materials. **MIN . SPAC** specifies the minimum grid spacing produced by adaptive refinement and **MAX . SPAC** specifies the maximum grid spacing produced by unrefinement. The value of **MIN . SPAC** should be small enough to resolve implanted and diffused impurity profiles, but if it is made too small, simulation times may become very large, especially during oxidation steps. **MAX . SPAC** can be set to zero to disable mesh unrefinement in selected materials.

For further information on the use of adaptive gridding see [Chapter 3, “Adaptive Gridding”](#) on page 3-200

Initial Time Step

The **INIT . TIM** parameter specifies the initial time step to be used for diffusion steps. The default value is usually adequate, but some speedup can be obtained by specifying a larger value when appropriate. Smaller values may be advisable following an ion implantation with damage, when growing an initial oxide, or when the structure contains sharply peaked impurity distributions.

Internal Solution Methods

The remaining parameters specify the internal solution methods to be used.



CAUTION

These values should not be changed except by the experienced user. Some of the remaining parameters may be changed or eliminated in future releases of the program, in which case input files that use these parameters need to be modified.

Time Integration

The **TRBDF** method is used for time integration. The **MILNE**, **HYBRID**, and **FORMULA** parameters are provided for compatibility with old input files, but only the **TRBDF** method is supported.

A **TRBDF** integration step consists of a trapezoidal step followed by a backward difference step. A second trapezoidal solution is used to estimate the local truncation error and to determine the size of the next time step.



Note:

The parameters governing the matrix solution algorithms should be changed only if there is a demonstrated need for doing so. The use of inappropriate combinations of values for these parameters may prevent the solution algorithms from converging.

System Solutions

The equations for each impurity (and for oxidant) form a block. Each block is solved independently, then the system of blocks is solved by an outer iteration method. The **CG** parameter selects a conjugate residual method for the outer iteration method. The **BACK** parameter specifies the maximum number of back vectors to be used in the **CG** method. The number of outer iterations is limited to **BLK . ITLI**; the iteration is terminated at this point whether convergence has been reached or not. The **GAUSS** method is no longer supported and may not work in this or future versions of the program.

Minimum-Fill Reordering

A minimum-fill reordering is performed to reduce the solution time for the impurity blocks if the **MIN . FILL** parameter is true. The **MIN . FREQ** parameter sets a limit on how much the solution matrix may expand before the minimum-fill reordering is repeated. A value of zero forces a reordering whenever the grid structure changes; a value greater than one reduces the time spent reordering equations but increases the time required to solve them. Using **MIN . FILL** is strongly advised.

The algorithm for minimum-fill reordering is selected by the **MF . METH** parameter. **MF . METH=0** specifies a complex algorithm designed to give optimum results when **FULL** factorization with a very large value of **MF . DIST** is used. **MF . METH=1** specifies a simpler algorithm that is much faster and appears to work at least as well as the older algorithm for most cases encountered in TSUPREM-4. **MF . DIST** specifies the distance between nodes in the structure beyond which fill terms can be ignored when **FULL** fill is used.

Block Solution

The remaining parameters apply to solving the equations for the specified impurity.

Solution Method

The **LU** solution method has been found to be most effective for all impurities, and is much faster than the others for point defects. The **SOR**, **SIP**, and **ICCG** methods are no longer supported and may not work in this or future versions of the program.

Matrix Structure

The **FULL**, **PART**, and **NONE** parameters specify the degree to which fill-in terms are to be included in the matrix solution. **PART** and **NONE** produce smaller matrices that are faster to decompose, but which may require more iterations. **PART** is useful for impurities. Using **FULL** for point defects is generally fastest, but may generate a large number of floating-point underflow conditions. This can cause a large reduction in speed on computers that use software trapping to process sub-normal floating point numbers. On such machines it is better to use **PART** for point defects. An appropriate value for **MF . DIST** can also alleviate this problem. **NONE** can be used for interstitial traps, and other species that do not diffuse.

SYMMETRY forces the matrix to be treated as symmetric, whether it is or not. The program may fail if this parameter is set true when it should be false.

Matrix Refactoring

The **NEWTON**, **TIME.STE**, and **ERROR** parameters determine how often the matrix is refactored. Using **NEWTON** gives the fastest convergence, but spends the most time refactoring the matrix; using **TIME.STE** causes the matrix to be refactored only at the start of each time step. With **ERROR**, the matrix is refactored as needed. The **TIME.STE** choice has been found to give the fastest solution for most problems.

Error Tolerances

The relative and absolute error tolerances for solution of the impurity blocks are given by **REL.ERR** and **ABS.ERR**, respectively. The relative error tolerance for stress-dependent oxidation solutions with the **VISCOUS** model is given by **OX.REL**, which should be between 1e-4 and 1e-6. **CONTIN.M** selects a continuation method for solving the stress-dependent oxidation equations (**VISCOUS** model only). Values from 2 to 7 are recognized, but only methods 2 and 7 should be used. Method 2 is the method used in versions of TSUPREM-4 prior to version 5.1; method 7 is a new method. Method 7 is more likely to converge than method 2, but may give worse results when it does not converge.

Examples

1. The following statement specifies that the initial time step should be 0.1 minutes:

```
METHOD INIT.TIM=0.1 PD.FERMI VERTICAL
```

The **PD.FERMI** model is used for the point defects and the **VERTICAL** model for the oxide growth.

2. The following statement indicates that the arsenic blocks should be solved to a relative error of 0.001 (0.1%) or an absolute error of $10^{12}/\text{cm}^3$, whichever is greater:

```
METHOD ARSENIC REL.ERR=0.001 ABS.ERR=1.0e12
```

EQUATION

The **EQUATION** statement is used to specify user-specified equations. The **EQUATION** statement is available only with the USEIT (User-Specified Equation Interface for TSUPREM-4) Option.

EQUATION

```
VARIABLE=<c> MATERIAL=<c> [MODEL=<c>] [TIF.NAME=<c>]
[ADDTOEXP=<c>] [INITIAL=<c>] [UPDATE=<c>]
[/MATERIA=<c> INTERFAC=<c> [BOUNDARY] [TRANSPOR]]
```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| VARIABLE | character | The solution variable name to be solved. Default: none |
| MATERIAL | character | The name of the material in which the equations are to be solved. Default: none Synonyms: MAT, MAT1 |
| MODEL | character | The identifier of the specified equations. The MODEL name can be used in the METHOD statement in order to enable or disable the application of the equations. Default: none Synonym: NAME |
| TIF.NAME | character | The name by which the solution variable VARIABLE is known in the TIF materials database. Note: The value of this parameter is case-sensitive. Default: the name of the solution variable |
| ADDTOEXP | character | The expression to be added to the right-hand-side (RHS) of the equation for the solution variable VARIABLE . Default: none Synonym: ADD.RHS |
| INITIAL | character | The expression for the initial values of the solution variable VARIABLE . Default: none |
| UPDATE | character | The expression which describes the update of the solution variable during the solving of the equation. Default: none |
| /MATERIA | character | The name of the material adjacent to MATERIAL for interface specification. Default: none Synonym: /MAT2 |
| INTERFAC | character | The expression for the flux from /MATERIA to MATERIAL Default: none |
| BOUNDARY | logical | Specifies that the ADDTOEXP expression describes the interface equation. Default: false |

| Parameter | Type | Definition |
|-----------|---------|--|
| TRANSPOR | logical | Specifies that the INTERFAC flux is taken into account on the nodes in /MATERIA as well. Default: true |

Description

The **EQUATION** statement specifies model equations that have the general form:

$$\frac{\partial X}{\partial t} = Previous_RHS + ADDTOEXP + INTERFAC \text{ at specified boundaries}$$

where X is the quantity to be solved.

TSUPREM-4 assembles the matrices to solve equations by assuming that the time derivative term of X is placed at the left-hand-side (LHS) of the equation and all the other terms at the right-hand-side (RHS). To describe this equation using **USEIT**, any new terms that appear on the RHS of the equation are specified by the **ADDTOEXP** (alias **ADD.RHS**) parameter and the **INTERFAC** parameter on the **EQUATION** statement. The **INTERFAC** parameter is used to define the flux at interfaces. The term *Previous_RHS* in the above expression represents terms that are already present on the RHS of the equation being modified. The initial condition of the variable X is specified with the **INITIAL** parameter in the **EQUATION** statement. The **INTERMED** statement specifies the intermediate quantities associated with equations.

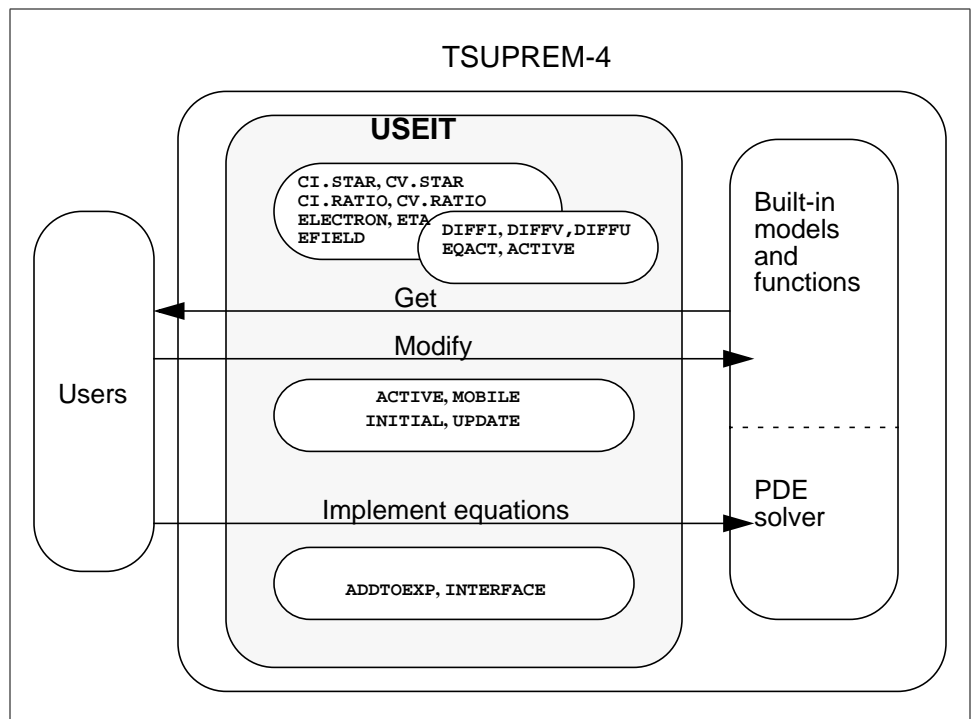


Figure 3-3 USEIT working diagram

Since USEIT is designed as a part of TSUPREM-4, you can easily use the physical quantities such as **CI . STAR(=I^{*})** and **ETA(n/n_i)** already calculated in the built-in routines. Similarly, you have the choice of using the built-in functions for calculating active and mobile concentrations or defining your own functions with the **ACTIVE** and **MOBILE** parameters in the **IMPURITY** statement.

When a new solution variable is first specified in the **EQUATION** statement, it is automatically defined. A message is displayed warning that the **VARIABLE** does not currently exist and is being defined. Alternatively, the solution variable can be defined in the **IMPURITY** statement with the **NEW** parameter. The **METHOD** statement specifies the numerical methods used to solve the equations. The factorization method of block matrices, the absolute, and the relative error tolerances for solutions should be specified.

Additional Information

For additional information see the following sections:

- See [Chapter 3, “INTERMEDIATE” on page 3-32](#) for the definition of **INTERMEDIATES** used in the expressions of equations.
- See [Chapter 3, “METHOD” on page 3-188](#) for the description of numerical methods to solve equations.
- See [Chapter 3, “IMPURITY” on page 3-239](#) for the descriptions of calculation methods for active and mobile concentrations.
- See [Chapter 6, "User-Specified Equation Interface-USEIT"](#) for additional details.

AMBIENT

The **AMBIENT** statement is used to specify oxidation coefficients. **OXIDE** is a valid synonym for the **AMBIENT** statement.

AMBIENT

```
[ { DRYO2 | WETO2 | STEAM | N2O | INERT | AMB.1 | AMB.2 | AMB.3 | AMB.4
  | AMB.5 }
  [F.O2=<n>] [F.H2O=<n>] [F.N2O=<n>] [F.H2=<n>] [F.N2=<n>]
  [F.HCL=<n>][PRESSURE=<n>] [HCL=<n>]
]
[ {O2 | H2O | N2O }
  [ {<111> | <110> | <100> | ORIENTAT=<n> | POLYSILI}
    [THINOX.0=<n>] [THINOX.E=<n>] [THINOX.L=<n>]
    [L.LIN.0=<n>] [L.LIN.E=<n>] [H.LIN.0=<n>] [H.LIN.E=<n>]
    [N.0=<n>] [N.E=<n>] [N.THIN.0=<n>] [N.THIN.E=<n>]
  ]
  [L.PAR.0=<n>] [L.PAR.E=<n>] [H.PAR.0=<n>] [H.PAR.E=<n>]
  [LIN.BREA=<n>] [PAR.BREA=<n>] [LIN.PDEP=<n>] [PAR.PDEP=<n>]
  [GAMMA.0=<n>] [GAMMA.E=<n>]
  [ { LIN.PCT | PAR.PCT
    | ( {LIN.CLDE | PAR.CLDE} COLUMN=<n> )
  }
  ]
  TABLE=<c>
]
[ { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
  | AMBIENT }
  [D.0=<n>] [D.E=<n>] [VC=<c>]
  [HENRY.CO=<n>] [THETA=<n>]
  [ { /MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR
    | /NITRIDE | /POLYSIL | /AMBIENT
  }
  ]
  [SEG.0=<n>] [SEG.E=<n>] [TRANS.0=<n>] [TRANS.E=<n>]
  [ALPHA=<n>]
  [N.G.0=<n>] [N.G.E=<n>] [N.GPOW.0=<n>] [N.GPOW.E=<n>]
  [N.V.0=<n>] [N.V.E=<n>]
]
]
]
[STRESS.D] [VR=<c>] [VT=<c>] [VD=<c>] [VDLIM=<n>]
[INITIAL=<n>] [SPREAD=<n>] [MASK.EDG=<n>]
[ERF.Q=<n>] [ERF.DELT=<n>] [ERF.LBB=<c>] [ERF.H=<c>]
[NIT.THIC=<n>]
[CLEAR]
[TEMPERAT=<c>]
[CM.SEC]
```

| Parameter | Type | Definition |
|-----------|---------|---|
| DRYO2 | logical | Specifies that the pressure, chlorine percentage, and flows of oxidizing and non-oxidizing species are associated with the dry oxygen ambient. Default: false |
| WETO2 | logical | Specifies that the pressure, chlorine percentage, and flows of oxidizing and non-oxidizing species are associated with the wet oxygen ambient. Default: false |
| STEAM | logical | Specifies that the pressure, chlorine percentage, and flows of oxidizing and non-oxidizing species are associated with the steam ambient. Default: false |
| N2O | logical | Specifies that the pressure, chlorine percentage, and flows of oxidizing and non-oxidizing species are associated with the N ₂ O oxygen ambient. Default: false |
| INERT | logical | Specifies that the pressure, chlorine percentage, and flows of oxidizing and non-oxidizing species are associated with the inert ambient. Default: false Synonyms: NEUTRAL, NITROGEN |
| AMB . 1 | logical | Specifies that the pressure, chlorine percentage, and flows of oxidizing and non-oxidizing species are associated with ambient number 1. Ambient number 1 is defined by the user. Default: false |
| AMB . 2 | logical | Specifies that the pressure, chlorine percentage, and flows of oxidizing and non-oxidizing species are associated with ambient number 2. Ambient number 2 is defined by the user. Default: false |
| AMB . 3 | logical | Specifies that the pressure, chlorine percentage, and flows of oxidizing and non-oxidizing species are associated with ambient number 3. Ambient number 3 is defined by the user. Default: false |
| AMB . 4 | logical | Specifies that the pressure, chlorine percentage, and flows of oxidizing and non-oxidizing species are associated with ambient number 4. Ambient number 4 is defined by the user. Default: false |
| AMB . 5 | logical | Specifies that the pressure, chlorine percentage, and flows of oxidizing and non-oxidizing species are associated with ambient number 5. Ambient number 5 is defined by the user. Default: false |
| F . O2 | number | The flow of O ₂ associated with the specified ambient. If H ₂ is also present, the O ₂ and H ₂ are assumed to react completely to form H ₂ O. The flows of O ₂ and H ₂ are reduced and the flow of H ₂ O is increased. Units: none Default: 0.0 |

| Parameter | Type | Definition |
|--------------------|---------|---|
| F.H2O | number | The flow of H ₂ O associated with the specified ambient. If O ₂ and H ₂ are also present, the O ₂ and H ₂ are assumed to react completely to form H ₂ O. The flows of O ₂ and H ₂ are reduced and the flow of H ₂ O is increased. Units: none Default: 0.0 |
| F.N2O | number | The flow of N ₂ O associated with the specified ambient. If H ₂ O or O ₂ are also present, the flow of N ₂ O gas is assumed to be N ₂ . Units: none Default: 0.0 |
| F.H2 | number | The flow of H ₂ associated with the specified ambient. If O ₂ is also present, the O ₂ and H ₂ are assumed to react completely to form H ₂ O. The flows of O ₂ and H ₂ were reduced and the flow of H ₂ O is increased. Units: none Default: 0.0 |
| F.N2 | number | The flow of N ₂ (or other inert gasses) associated with the specified ambient. Units: none Default: 0.0 |
| F.HCL | number | The flow of chlorine associated with the specified ambient. Units: none Default: 0.0 |
| PRESSURE | number | The default value of total gas pressure for the specified ambient. Units: atmospheres Default: the current value for this ambient; initially 1.0 |
| HCL | number | The default percentage of chlorine present for the specified ambient. Units: percent Default: calculated from F.HCL |
| O2 | logical | Specifies that the oxidation coefficients are associated with the O ₂ oxidizing species. Default: false |
| H2O | logical | Specifies that the oxidation coefficients are associated with the H ₂ O oxidizing species. Default: false |
| <111> | logical | Specifies that linear and thin oxide growth rate coefficients apply to <111> orientation silicon. Default: false |
| <110> | logical | Specifies that linear and thin oxide growth rate coefficients apply to <110> orientation silicon. Default: false |
| <100> | logical | Specifies that linear and thin oxide growth rate coefficients apply to <100> orientation silicon. Default: false |

| Parameter | Type | Definition |
|-----------------|---------|--|
| ORIENTAT | number | Specifies that linear and thin oxide growth rate coefficients apply to silicon of the specified orientation. Allowed values are 111, 110, and 100. Units: none Default: 100 |
| POLYSILI | logical | The specified coefficients apply to polysilicon, or the interface between polysilicon and some other material. Default: false |
| THINOX.O | number | The pre-exponential constant in the expression for the thin oxide growth rate parameter. Units: microns/min or cm/sec Default: current value for this orientation and oxidant |
| THINOX.E | number | The activation energy in the expression for the thin oxide growth rate parameter. Units: electron volts Default: current value for this orientation and oxidant |
| THINOX.L | number | The characteristic length in the expression for the thin oxide growth rate parameter. Units: microns Default: current value for this orientation and oxidant |
| L.LIN.O | number | The pre-exponential constant in the expression for the linear oxidation rate for temperatures below the temperature breakpoint set by LIN.BREA . Units: microns/min or cm/sec Default: current value for this orientation and oxidant Synonym: LIN.L.O |
| L.LIN.E | number | The activation energy in the expression for the linear oxidation rate for temperatures below the temperature breakpoint set by LIN.BREA . Units: electron volts Default: current value for this orientation and oxidant Synonym: LIN.L.E |
| H.LIN.O | number | The pre-exponential constant in the expression for the linear oxidation rate for temperatures above the temperature breakpoint set by LIN.BREA . Units: microns/min or cm/sec Default: current value for this orientation and oxidant Synonym: LIN.H.O |
| H.LIN.E | number | The activation energy in the expression for the linear oxidation rate for temperatures above the temperature breakpoint set by LIN.BREA . Units: electron volts Default: current value for this orientation and oxidant Synonym: LIN.H.E |
| N.O | number | The pre-exponential constant in the expression for the maximum trap density in N ₂ O linear oxidation rate. Units: cm ⁻² Default: current value for this orientation |

| Parameter | Type | Definition |
|-----------------|--------|---|
| N.E | number | The activation energy in the expression for the maximum trap density in N ₂ O linear oxidation rate Units: electron volts Default: current value for this orientation |
| N.THIN.O | number | The pre-exponential constant in the expression for the maximum trap density in N ₂ O thin oxidation rate. Units: cm ⁻² Default: current value for this orientation |
| N.THIN.E | number | The activation energy in the expression for the maximum trap density in N ₂ O thin oxidation rate Units: electron volts Default: current value for this orientation |
| L.PAR.O | number | The pre-exponential constant in the expression for the parabolic oxidation rate for temperatures below the temperature breakpoint set by PAR.BREA . Units: microns ² /min or cm ² /sec Default: current value for this oxidant Synonym: PAR.L.O |
| L.PAR.E | number | The activation energy in the expression for the parabolic oxidation rate for temperatures below the temperature breakpoint set by PAR.BREA . Units: electron volts Default: current value for this oxidant Synonym: PAR.L.E |
| H.PAR.O | number | The pre-exponential constant in the expression for the parabolic oxidation rate for temperatures above the temperature breakpoint set by PAR.BREA . Units: microns ² /min or cm ² /sec Default: current value for this oxidant Synonym: PAR.H.O |
| H.PAR.E | number | The activation energy in the expression for the parabolic oxidation rate for temperatures above the temperature breakpoint set by PAR.BREA . Units: electron volts Default: current value for this oxidant Synonym: PAR.H.E |
| LIN.BREA | number | The temperature breakpoint at which the temperature dependence of the linear oxidation rate changes. Units: degrees Celsius Default: current value for this oxidant Synonym: L.BREAK |
| PAR.BREA | number | The temperature breakpoint at which the temperature dependence of the parabolic oxidation rate changes. Units: degrees Celsius Default: current value for this oxidant Synonym: P.BREAK |
| LIN.PDEP | number | The exponent of the pressure in the expression for the linear oxidation rate. Units: none Default: current value for this oxidant Synonym: L.PDEP |

| Parameter | Type | Definition |
|-----------------|---------|--|
| PAR.PDEP | number | The exponent of the pressure in the expression for the parabolic oxidation rate. Units: none Default: current value for this oxidant Synonym: P.PDEP |
| GAMMA.O | number | The pre-exponential constant in the expression for the impurity concentration dependence of the linear oxidation rate. Units: none Default: current value for this oxidant |
| GAMMA.E | number | The activation energy in the expression for the impurity concentration dependence of the linear oxidation rate. Units: electron volts Default: current value for this oxidant |
| LIN.PCT | logical | Specifies that the TABLE parameter defines chlorine percentages associated with the rows in the table of coefficients modifying the linear oxidation rate in the presence of chlorine. Default: false |
| PAR.PCT | logical | Specifies that the TABLE parameter defines chlorine percentages associated with the rows in the table of coefficients modifying the parabolic oxidation rate in the presence of chlorine. Default: false |
| LIN.CLDE | logical | Specifies that the TABLE parameter defines entries in a column of the table of coefficients modifying the linear oxidation rate in the presence of chlorine. The column number is specified with the COLUMN parameter and is associated with the temperature specified by the TEMPERAT parameter. Default: false |
| PAR.CLDE | logical | Specifies that the TABLE parameter defines entries in a column of the table of coefficients modifying the parabolic oxidation rate in the presence of chlorine. The column number is specified with the COLUMN parameter and is associated with the temperature specified by the TEMPERAT parameter. Default: false |
| COLUMN | number | The column number in the tables of coefficients modifying the linear or parabolic oxidation rates. The coefficients defined by the TABLE parameter are stored in this column of a table. The column number is associated with the temperature specified by the TEMPERAT parameter. There can be at most 8 columns in a table and each column corresponds to one temperature. Units: none Default: none |

| Parameter | Type | Definition |
|-----------------|-----------|--|
| TABLE | character | <p>This parameter is interpreted as a series of numeric values, separated by spaces or commas. If the LIN . PCT or PAR . PCT parameter is specified, the TABLE parameter defines the chlorine percentages associated with the rows in the table of coefficients modifying the linear or parabolic oxidation rates, respectively. If the LIN . CLDE or PAR . CLDE parameter is specified, the TABLE parameter defines the entries in a column of the table of coefficients modifying the linear or parabolic oxidation rates, respectively. There can be at most eight rows in a table and each row corresponds to one chlorine percentage. At most 8 values can be defined with this parameter.</p> <p>Units: percent or none Default: none</p> |
| MATERIAL | character | <p>The specified coefficients apply to the named material or to the interface between the named material and some other material.</p> <p>Default: none</p> |
| SILICON | logical | <p>The specified coefficients apply to silicon, or the interface between silicon and some other material.</p> <p>Default: false</p> |
| OXIDE | logical | <p>The specified coefficients apply to oxide, or the interface between oxide and some other material.</p> <p>Default: true if no other first material is specified</p> |
| OXYNITRI | logical | <p>The specified coefficients apply to oxynitride, or the interface between oxynitride and some other material.</p> <p>Default: false</p> |
| NITRIDE | logical | <p>The specified coefficients apply to nitride, or the interface between nitride and some other material.</p> <p>Default: false</p> |
| POLYSILI | logical | <p>The specified coefficients apply to polysilicon, or the interface between polysilicon and some other material.</p> <p>Default: false</p> |
| AMBIENT | logical | <p>The specified coefficients apply to gas, or the interface between gas and some other material.</p> <p>Default: false Synonym: GAS</p> |
| D . 0 | number | <p>The pre-exponential constant in the expression for the diffusivity of oxidant in the specified material.</p> <p>Units: microns²/min or cm²/sec Default: current value for this oxidant and material Synonym: DIFF . 0</p> |
| D . E | number | <p>The activation energy in the expression for the diffusion of oxidant in the specified material.</p> <p>Units: electron volts Default: current value for this oxidant and material Synonym: DIFF . E</p> |

| Parameter | Type | Definition |
|-----------|-----------|---|
| VC | character | A table of activation volumes as a function of temperature for the dependence of material viscosity on shear stress for the specified material and oxidizing species (O ₂ or H ₂ O). Entries in the table correspond to temperatures given by the TEMPERAT parameter (see text). Units: Å ³ Default: the current value; initially 300 |
| HENRY.CO | number | The solubility of oxidant in the specified material at one atmosphere. Units: atoms/cm ³ /atm Default: current value for this oxidant and material |
| THETA | number | The number of oxide molecules per cubic centimeter of oxide. Units: atoms/cm ³ Default: current value |
| /MATERIA | character | The specified coefficients apply to the interface between the other specified material and this named material. Default: none |
| /SILICON | logical | The specified coefficients apply to the interface between the specified material and silicon. Default: True if no other second material is specified. |
| /OXIDE | logical | The specified coefficients apply to the interface between the specified material and oxide. Default: false |
| /OXYNITR | logical | The specified coefficients apply to the interface between the specified material and oxynitride. Default: false |
| /NITRIDE | logical | The specified coefficients apply to the interface between the specified material and nitride. Default: false |
| /POLYSIL | logical | The specified coefficients apply to the interface between the specified material and polysilicon. Default: false |
| /AMBIENT | logical | The specified coefficients apply to the interface between the specified material and gas. Default: false Synonym: /GAS |
| SEG.O | number | The pre-exponential constant in the expression for segregation of oxidant between the two specified materials. Units: none Default: current value for this oxidant and these materials |
| SEG.E | number | The activation energy in the expression for segregation of oxidant between the two specified materials. Units: electron volts Default: current value for this oxidant and these materials |

| Parameter | Type | Definition |
|---------------------|---------|--|
| TRANS . 0 | number | The pre-exponential constant in the expression for transport of oxidant between the two specified materials. Units: microns/min or cm/sec Default: the current value for this oxidant and these materials Synonym: TRN . 0 |
| TRANS . E | number | The activation energy in the expression for transport of oxidant between the two specified materials. Units: electron volts Default: the current value for this oxidant and these materials Synonym: TRN . E |
| ALPHA | number | The volume expansion ratio between the two specified materials. The defaults are 0.44 for silicon/oxide and polysilicon/oxide, and 1.0 for all other combinations. Units: none Default: the current value for these materials |
| N . G . 0 | number | The pre-exponential constant in the expression for nitrogen generation rate due to N ₂ O oxidation. Units: cm ⁻³ Default: the current value for these materials |
| N . G . E | number | The activation energy in the expression for nitrogen generation rate due to N ₂ O oxidation. Units: electron volts Default: the current value for these materials |
| N . GPOW . 0 | number | The pre-exponential constant in the expression for the exponent in the normalized boundary velocity. Units: none Default: the current value for these materials |
| N . GPOW . E | number | The activation energy in the expression for the exponent in the normalized boundary velocity of N ₂ O oxidation. Units: electron volts Default: the current value for these materials |
| N . V . 0 | number | The pre-exponential constant in the expression for the normalization velocity of N ₂ O oxidation. Units: cm/sec Default: the current value for these materials |
| N . V . E | number | The activation energy in the expression for the normalization velocity of N ₂ O oxidation. Units: none Default: the current value for these materials |
| STRESS . D | logical | Specifies that the stress-dependent models for oxide viscosity, oxidant diffusivity, and surface reaction rate are to be used. Default: the current value |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| VR | character | A table of activation volumes as a function of temperature for the dependence of oxidation rate at the Si/SiO ₂ interface on normal stress for the specified oxidizing species (O2 or H2O). Entries in the table correspond to temperatures given by the TEMPERAT parameter (see text). Units: Å ³ Default: the current value; initially 15 |
| VT | character | A table of activation volumes as a function of temperature for the dependence of oxidation rate at the Si/SiO ₂ interface on tangential stress for the specified oxidizing species (O2 or H2O). Entries in the table correspond to temperatures given by the TEMPERAT parameter (see text). Units: Å ³ Default: the current value; initially 0.0 |
| VD | character | A table of activation volumes as a function of temperature for the dependence of oxidant diffusivity in oxide on pressure for the specified oxidizing species (O2 or H2O). Entries in the table correspond to temperatures given by the TEMPERAT parameter (see text). Units: Å ³ Default: the current value; initially 75 |
| VDLIM | number | The maximum increase in oxidant diffusivity produced by the VD parameter. Units: none Default: the current value; initially 1.2 |
| INITIAL | number | The thickness of the existing oxide at the start of oxidation. Exposed silicon surfaces are covered with this thickness of native oxide before oxidation begins. Units: microns Default: the current value; initially 0.002 |
| SPREAD | number | The ratio of width to height for the bird's beak, used in the ERFC model of local oxide shape. Units: none Default: the current value; initially 1.0 |
| MASK.EDG | number | The assumed position of the mask, used by the analytical models for local oxidation. Oxide grows to the right of the mask edge. Units: microns Default: the current value; initially -200 |
| ERF.Q | number | The <i>q</i> parameter for the ERFG (Guillemot) model. Units: microns Default: the current value; initially 0.05 |
| ERF.DELT | number | The <i>delta</i> parameter for the ERFG (Guillemot) model. Units: microns Default: the current value; initially 0.04 |
| ERF.LBB | character | The length of the bird's beak for the ERFG (Guillemot) model. This is an arithmetic expression involving the variables <i>Fox</i> (the field oxide thickness, in microns), <i>eoX</i> (the pad oxide thickness, in microns), <i>Tox</i> (the oxidation temperature, in degrees Kelvin), and <i>en</i> (the nitride thickness, in microns). Units: microns Default: the current value; initially $(8.25e-3*(1580.3-Tox)*(Fox^{0.67})*(eoX^{0.3})*exp(-((en-0.08)^2)/0.06)).$ |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| ERF.H | character | The ratio of the nitride lifting to the field oxide thickness for the ERFG (Guillemot) model. This is an arithmetic expression involving the variables F_{ox} (the field oxide thickness, in microns), e_{ox} (the pad oxide thickness, in microns), T_{ox} (the oxidation temperature, in degrees Kelvin), and en (the nitride thickness, in microns). Units: none Default: the current value; initially $(402*(0.445-1.75*en)*exp(-T_{ox}/200))$ |
| NIT.THIC | number | The nitride thickness, en used in the equations for ERF.LBB and ERF.H . Units: microns Default: none |
| TEMPERAT | character | The temperature associated with the column in the chlorine tables given by the COLUMN parameter, or a list of temperatures corresponding to the values of the VC , VD , VR , and/or VT parameters. Units: degrees Celsius Default: none |
| CLEAR | logical | Clear table(s) specified by the VC , VD , VR , and/or VT parameters before adding new values (see text). Default: none |
| CM.SEC | logical | If true, parameters involving time are specified in centimeters and seconds; if false, parameters involving time are in microns and minutes. Default: false |

Description

All parameters relating to oxidation are specified on this statement. The necessary parameters are set by **AMBIENT** statements in the *s4init* file, but can be changed by you.

Oxidation Models

The following models are available:

1. An error-function fit to bird's beak shapes (the **ERFC** model)
2. A parameterized error-function model from the literature (the Guillemot or **ERFG** model)
3. A model in which oxidant diffuses and the oxide grows vertically at a rate determined by the local oxidant concentration (the **VERTICAL** model)
4. A compressible viscous flow model (the **COMPRESS** model)
5. A viscoelastic flow model (the **VISCOELA** model)
6. An incompressible viscous flow model (the **VISCOUS** model)

A summary of the features and characteristics of these models follows; full descriptions are given in [Chapter 2, "Oxidation" on page 2-46](#).

**Note:**

*Oxidation of polycrystalline silicon is modeled by the **COMPRESS**, **VISCOELA**, and **VISCOUS** models only.*

ERFC Model

The **ERFC** model is the fastest of the oxidation models. It can be used for uniform oxidation of bare silicon, provided that modeling of the concentration dependence of the oxidation rate is not needed. It can be used for nonuniform oxidation of planar surfaces provided that fitting data for the lateral spread of the bird's beak is available.

The **ERFC** model is controlled by the **SPREAD**, **MASK.EDG**, and **INITIAL** parameters. The growth rate vs. time is computed assuming an initial oxide thickness **INITIAL** at the start of each diffusion step. This model should not be used with a structure having an unmasked initial oxide thickness other than **INITIAL**.

ERFG Model

The **ERFG** model is by Guillemot, et al., *IEEE Transactions on Electron Devices*, ED-34, May 1987. The bird's beak shape and nitride lifting are functions of process conditions. The **ERFG** model is controlled by the **ERF.Q**, **ERF.DELT**, **ERF.LBB**, **ERF.H**, **NIT.THIC**, and **INITIAL** parameters. The above comments regarding **INITIAL** apply the **ERFG** model as well. The **ERF1** and **ERF2** models use the two shapes derived by Guillemot, et al; the **ERFG** model chooses between them based on process conditions.

**VERTICAL
Model**

The **VERTICAL** model has no fitting parameters, but is only accurate when the growth is approximately vertical (within about 30° of vertical). The **VERTICAL** model does not simulate oxidation of polysilicon. The **VERTICAL** model can be used for oxidation of uniform substrates with arbitrary initial oxide thicknesses, and for approximating nonrecessed LOCOS processes. Concentration dependence of the oxidation rate is included in the **VERTICAL** model.

**COMPRESS
Model**

The **COMPRESS** model simulates the viscous flow of the oxide in two dimensions. It uses simple (three nodes per triangle) elements for speed, but must allow some compressibility as a consequence. It is more accurate than the **VERTICAL** model, but requires more computer time. It uses Young's modulus (**YOUNG.M**) and Poisson's ratio (**POISS.R**), specified for each material with the **MATERIAL** statement. The **COMPRESS** model is recommended for general use on arbitrary structures. It includes the concentration dependence of oxidation rate and models the oxidation of polysilicon.

**VISCOELA
Model**

The **VISCOELA** model simulates viscoelastic flow in two dimensions. It uses simple (three nodes per triangle) elements for speed, but simulates elastic deformation as well as viscous flow. When used with stress dependent parameters (i.e., **STRESS.D** true), it can produce very accurate results with reasonable simulation times. It is slower than the **COMPRESS** model, but 10-100 times faster than the **VISCOUS** model with stress dependence. It uses the **YOUNG.M**, **POISS.R**,

VISC.0, **VISC.E**, and **VISC.X** parameters for mechanical properties of materials plus the **VC**, **VR**, **VD**, and **VDLIM** parameters for describing stress dependence.

VISCOUS Model

The **VISCOUS** model simulates incompressible viscous flow of the oxide using more complicated (seven nodes per triangle) elements. It calculates stresses and is the only model that models reflow. The **VISCOUS** model is slower than the **COMPRESS**, and **VISCOELA** models and may require large amounts of memory; it may be impossible to simulate large structures with this model on some computers, due to memory limitations. It uses the viscosity parameters (**VISC.0**, **VISC.E**, and **VISC.X**) specified for each material with the **MATERIAL** statement.

The **VISCOUS** model is needed only when stress calculations are required, when the stress-dependent oxidation parameters are used, or when **SKIP.SIL** must be set false in order to simulate structures with floating silicon mesas.

Stress Dependence

The parameter **STRESS.D** determines whether the stress dependence of oxidant diffusivity, surface reaction rate, and oxide viscosity are included when oxidizing with the **VISCOELA** or **VISCOUS** models. When **STRESS.D** is true, these stress dependencies are included; when **STRESS.D** is false, they are not.

The magnitude of the various stress effects are specified by the **VC**, **VR**, **VT**, **VD**, and **VDLIM** parameters.

- **VC** is the activation volume for the dependence of oxide viscosity on shear stress. **VC** can be specified for arbitrary materials using the **MATERIAL** statement.
- **VR** and **VT** are the activation volumes for the dependence of the surface reaction rate on normal and tangential stresses, respectively.
- **VD** is the activation volume for the dependence on pressure of the diffusivity of the oxidizing species in the oxide.
- **VDLIM** is the maximum increase in oxidant diffusivity produced by **VD**.

The parameters **VC**, **VD**, **VR**, **VT**, and **TEMPERAT** are used to specify the activation volumes as functions of temperature. A separate table is maintained for each oxidizing species, and for each material in the case of **VC**. Table entries are added or changed by specifying lists of values (with **VC**, **VD**, **VR**, or **VT**) and temperatures (with **TEMPERAT**). The portion of the table spanned by the specified temperatures is replaced by the specified values; the number of values must be the same as the number of temperatures, and the temperatures must be given in order, lowest to highest. The **CLEAR** parameter is used to clear a table before setting any values.

For example, the statement

```
AMBIENT O2 CLEAR VD="40 50 60" TEMP="800 900 1050"
```

removes any old values from the table of **VD** vs. temperature for O2 and adds three new values. The statement

```
AMBIENT O2 VD="55 75" TEMP="900 1100"
```

would then replace the values at 900° C and 1050° C with new values at 900° C and 1100° C.

If no oxidizing species is specified, the values apply to ambients containing either O₂ or H₂O. The material should be specified when setting **VC**; if no material is specified, **OXIDE** is assumed. If **V.COMPAT** (on the **OPTION** statement) is less than 6.6, the specified values of the activation volumes apply to oxide in all ambients, including inert ambients.

An Arrhenius interpolation is used between values in the table. For temperatures outside the range of the table the nearest value is used.

Coefficients

The diffusion and segregation coefficients can be used to model oxidant diffusion in arbitrary layers, but the diffusion coefficient in oxide is derived from the parabolic rate constant. The transport coefficient between the ambient and oxide is interpreted as the gas-phase mass-transport coefficient for the specified oxidizing species.

Chlorine

The effects of chlorine in the ambient gas on the oxidation rate of silicon are specified by tables of coefficients that modify the linear and parabolic oxidation rates. There are two tables for each oxidizing species, one each for the linear and parabolic oxidation rates. The tables are two-dimensional with at most 8 rows corresponding to chlorine percentages and at most 8 columns corresponding to ambient temperatures. Linear interpolation is used to obtain values for temperatures or percentages between the values in the table. For temperatures or percentages outside of the range of values present in the table, the values in the first or last rows or columns, as appropriate, are used.

Examples

For example, consider the following table of chlorine coefficients with six rows of chlorine percentages and five columns of temperatures:

| row | % | 1 800 | 2 900 | 3 1000 | 4 1100 | 5 1200 | column temperature |
|-----|----|----------|----------|-----------|-----------|-----------|-----------------------|
| 1 | 0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | |
| 2 | 1 | 1.1 | 1.2 | 1.3 | 1.4 | 1.5 | |
| 3 | 3 | 1.6 | 1.7 | 1.8 | 1.9 | 2.0 | |
| 4 | 5 | 2.1 | 2.2 | 2.3 | 2.4 | 2.5 | |
| 5 | 7 | 2.6 | 2.7 | 2.8 | 2.9 | 3.0 | |
| 6 | 10 | 3.1 | 3.2 | 3.3 | 3.4 | 3.5 | |

If this table represented the modification coefficients for the linear oxidation rates for the O2 oxidizing species, it could have been defined with the following series of input statements:

```

AMBIENT O2 LIN.PCT TABLE="0, 1, 3, 5, 7, 10"
AMBIENT O2 LIN.CLDE COLUMN=1 TEMPERAT=800 +
                TABLE="1.0, 1.1, 1.6, 2.1, 2.6, 3.1"
AMBIENT O2 LIN.CLDE COLUMN=2 TEMPERAT=900
                TABLE="1.0, 1.2, 1.7, 2.2, 2.7, 3.2"
AMBIENT O2 LIN.CLDE COLUMN=3 TEMPERAT=1000 +
                TABLE="1.0, 1.3, 1.8, 2.3, 2.8, 3.3"
AMBIENT O2 LIN.CLDE COLUMN=4 TEMPERAT=1100 +
                TABLE="1.0, 1.4, 1.9, 2.4, 2.9, 3.4"
AMBIENT O2 LIN.CLDE COLUMN=5 TEMPERAT=1200 +
                TABLE="1.0, 1.5, 2.0, 2.5, 3.0, 3.5"
    
```

The following values are obtained from this table for the indicated percentages and temperatures:

| percent | temperature | table value | row | column |
|---------|-------------|-------------|-----|--------|
| 1.0 | 1000 | 1.3 | 2 | 3 |
| 1.0 | 1050 | 1.35 | 2 | 3,4 |
| 2.0 | 1000 | 1.55 | 2,3 | 3 |
| 12.0 | 1000 | 3.3 | 6 | 3 |
| 1.0 | 700 | 1.1 | 2 | 1 |
| 1.0 | 1250 | 1.5 | 2 | 5 |
| 2.0 | 1250 | 1.75 | 2,3 | 5 |

Parameter Dependencies

Parameters which have special dependencies are listed below. If insufficient information is given with a parameter (e.g., **L.LIN.0** without an orientation and an oxidant), the parameter is ignored (without warning).

Orientation

The following parameters are dependent on the specified orientation: **L.LIN.0**, **L.LIN.E**, **H.LIN.0**, **H.LIN.E**, **THINOX.0**, **THINOX.E**, and **THINOX.L**. Note that **POLYSILI** can be used in place of the orientation to specify coefficients for oxidation of polysilicon.

Oxidizing Species

The following parameters are dependent on the oxidizing species (O₂ or H₂O): **L.LIN.0**, **L.LIN.E**, **H.LIN.0**, **H.LIN.E**, **LIN.BREA**, **LIN.PDEP**, **L.PAR.0**, **L.PAR.E**, **H.PAR.0**, **H.PAR.E**, **PAR.BREA**, **PAR.PDEP**, **GAMMA.0**, **GAMMA.E**, **LIN.PCT**, **PAR.PCT**, **LIN.CLDE**, **PAR.CLDE**, **COLUMN**, **TEMPERAT**, **TABLE**, **THINOX.0**, **THINOX.E**, **THINOX.L**, **D.0**, **D.E**, **VC**, **HENRY.CO**, **SEG.0**, **SEG.E**, **TRANS.0**, and **TRANS.E**.

VD, **VR**, and **VT** also depend on the oxidizing species but apply to both O₂ and H₂O if neither is specified.

Specified Material

The following parameters are dependent only on the first material specified: **D . 0**, **D . E**, **VC**, **HENRY . CO**, and **THETA**.

The following parameters are dependent on both materials specified: **SEG . 0**, **SEG . E**, **TRANS . 0**, **TRANS . E**, and **ALPHA**.

Specified Units

Parameters whose units include time are specified in units of microns and minutes, unless **CM . SEC** is true, in which case units of centimeters and seconds are assumed.

Examples

1. The statement

```
AMBIENT AMB.1 F.O2=0.90 F.N2=0.08 F.HCL=0.02 +
PRESSURE=2.0
```

defines ambient **AMB . 1** to consist of 90% oxygen and 2% chlorine at a pressure of two atmospheres.

2. The statement

```
AMBIENT H2O MAT=NITRIDE +
VC="130 170" TEMP="900 1000"
```

replaces any values for temperatures between 900°C and 1000°C in the table for **VC** of nitride in ambients containing H₂O.

3. The initialization file *s4init* contains the definitive set of examples of use for the **AMBIENT** statement.

Additional AMBIENT Notes

1. Oxidant in materials other than oxide is allowed to diffuse and segregate, but its concentration is then ignored (no oxynitridation, for instance). The diffusion coefficients in oxide and transport coefficients between oxide and silicon are derived from the Deal-Grove coefficients, so these parameters are ignored if read from input statements.
2. The analytic models use the thickness of the oxide to compute the growth rate, and the **ERFG** model also uses the nitride thickness. These values are *not* inferred from the structure. Instead, the value of **NIT . THIC** is used for the nitride thickness, and the oxide thickness is calculated by adding the oxide grown in a given high-temperature step to the specified **INITIAL** oxide thickness. Thus if the structure has other than **INITIAL** microns of oxide on it at the start of a diffusion step, the thickness must be specified with the **INITIAL** parameter. If there is no oxide on an exposed silicon surface, a layer of oxide of thickness **INITIAL** is deposited. If the **INITIAL** parameter doesn't correspond to the actual oxide thickness, the growth rate is incor-

rect. The **INITIAL** parameter need not be set when an oxidation is continued with the **CONTINUE** parameter on the **DIFFUSION** statement.

3. The analytic models do not recognize masking layers in the structure. The location of the presumed mask edge must be specified by the **MASK.EDG** parameter.
4. The material viscosities have been calibrated for the **VISCOELA** model with stress dependence enabled. For use without stress dependence (with either the **VISCOELA** or **VISCOUS** model), it may be necessary to modify the viscosity values.

MOMENT

The **MOMENT** statement sets distribution moments for use in ion implantation.

MOMENT

```
[ CLEAR ]
[ { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
  | PHOTORES } ]
[ RANGE=<n> SIGMA=<n> [ GAMMA=<n> ] [ KURTOSIS=<n> ] [ LSIGMA=<n> ]
  [ LSLOPE=<n> ]
  [ D.FRAC=<n> D.RANGE=<n> D.SIGMA=<n> [ D.GAMMA=<n> ] [ D.KURTOS=<n> ]
  [ D.LSIGMA=<n> ] [ D.LSLOPE=<n> ]
  [ TILT=<n> ] ]
]
```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| CLEAR | logical | Clears all moments specified on previous MOMENT statements. Default: false |
| MATERIAL | character | The name of the material to which the given range statistics apply. Default: none |
| SILICON | logical | Specifies that the given range statistics apply to implantation into silicon. Default: true, unless another material is specified |
| OXIDE | logical | Specifies that the given range statistics apply to implantation into oxide. Default: false |
| OXYNITRI | logical | Specifies that the given range statistics apply to implantation into oxynitride. Default: false |
| NITRIDE | logical | Specifies that the given range statistics apply to implantation into nitride. Default: false |
| POLYSILI | logical | Specifies that the given range statistics apply to implantation into polysilicon. Default: false |
| PHOTORES | logical | Specifies that the given range statistics apply to implantation into photoresist. Default: false |
| RANGE | number | The first moment (projected range) of the first Pearson or Gaussian distribution. units: microns Default: none |
| SIGMA | number | The second moment (standard deviation) of the first Pearson or Gaussian distribution. Units: microns Default: none |
| GAMMA | number | The third moment ratio (skewness) of the first Pearson distribution. Units: none Default: 0.0 |

| Parameter | Type | Definition |
|-----------------|--------|--|
| KURTOSIS | number | The fourth moment ratio (kurtosis) of the first Pearson distribution. Units: calculated from first three moments Default: calculated from first three moments |
| LSIGMA | number | The lateral (perpendicular to the ion beam) standard deviation for the first Pearson or Gaussian distribution. Units: microns Default: SIGMA |
| LSLOPE | number | The slope of the lateral standard deviation for the first Pearson distribution of the dual-Pearson model. Units: microns Default: 0.0 |
| D.FRAC | number | The fraction of the implant dose that uses the first set of Pearson distribution moments of the dual-Pearson model. Units: none Default: 1.0 |
| D.RANGE | number | The first moment (projected range) of the second Pearson distribution of the dual-Pearson model. Units: microns Default: none |
| D.SIGMA | number | The second moment (standard deviation) of the second Pearson distribution of the dual-Pearson model. Units: microns Default: none |
| D.GAMMA | number | The third moment ratio (skewness) of the second Pearson distribution of the dual-Pearson model. Units: none Default: 0.0 |
| D.KURTOS | number | The fourth moment ratio (kurtosis) of the second Pearson distribution of the dual-Pearson model. Units: none Default: calculated from first three moments |
| D.LSIGMA | number | The lateral (perpendicular to the ion beam) standard deviation for the second Pearson distribution of the dual-Pearson model. Units: microns Default: D.SIGMA |
| D.LSLOPE | number | The slope of the lateral standard deviation for the second Pearson distribution of the dual-Pearson model. Units: microns Default: 0.0 |
| TILT | number | The angle between the direction of the ion beam and direction along which the specified moments are measured. Units: degrees Default: 0.0 |

Description

The **MOMENT** statement specifies moments to be used by the ion implantation model. The values specified by **MOMENT** statements are used instead of values in the implant data file *s4imp0* on **IMPLANT** statements that specify the **MOMENTS** parameter.

One full set of moments for the Gaussian, single Pearson, or dual Pearson implant model can be specified for each material. A **MOMENT** statement for a particular material replaces any previously specified moments for that material. The **CLEAR** parameter clears any user-specified moment data for all materials.

When the **MOMENTS** parameter is specified on an **IMPLANT** statement, values specified on **MOMENT** statements are used for materials for which **MOMENT** statements were given. Moments for other materials are obtained from the moments data file, as usual.

Optional and Required Model Parameters

The models or the **MOMENT** parameter have mandatory and optional parameters. They are listed below.

- For the Gaussian implant model, **RANGE** and **SIGMA** must be specified and **LSIGMA** is optional.
- For the (single) Pearson model, **RANGE**, and **SIGMA** must be specified and **GAMMA**, **LSIGMA**, and **KURTOSIS** are optional.
- For the dual Pearson model, the **D.FRAC**, **D.RANGE**, and **D.SIGMA** parameters must be specified in addition to the parameters of the single Pearson model; **D.GAMMA**, **D.LSIGMA**, and **D.KURTOS** are optional.
- The **TILT** parameter is optional.

Using the MOMENT Statement

The **MOMENT** statement makes it easy to experiment with different implant distribution parameters, or to specify alternate parameters for critical implant steps. If a particular set of moments is needed frequently, it is easier to create an alternate moment data file, however.

When the **MOMENTS** parameter is specified on an **IMPLANT** statement, moments from preceding **MOMENT** statements are used without regard to the implant energy or species being implanted. It is your responsibility to ensure that the moments are appropriate for the implant being simulated.

The **TILT** parameter specifies the angle between the direction of the ion beam during implant and the direction along which the specified moments were (or would be) measured. In a typical experiment, ions are implanted into a tilted wafer and the resulting dopant distribution is measured perpendicular to the surface. If the moments of the measured distribution are specified on the **MOMENT** statement,

then the tilt of the wafer during the implant should be specified by the **TILT** parameter.

Examples

1. In the following series of statements

```
MOMENT SILICON RANGE=0.195 SIGMA=0.072 +  
      GAMMA=-0.65 KURTOSIS=22.1  
      IMPLANT BORON DOSE=1e15 ENERGY=70 MOMENTS
```

The implantation of boron into silicon uses the (single) Pearson distribution specified on the **MOMENT** statement. In materials other than silicon, distribution parameters from the implant data file are used.

2. In the statement

```
IMPLANT ARSENIC DOSE=2e15 ENERGY=40
```

parameters for the arsenic distribution comes from the implant data file. Parameters specified on **MOMENT** statements are not used, because **MOMENTS** is not specified on the **IMPLANT** statement.

Additional Note

The implanted dose in the silicon is reduced by the dose that is stopped in screening layers or backscattered from the surface, and thus depends weakly on the implant moments in the screening layer. Because the calculated damage is proportional to the dose in the silicon, the amount of damage is also weakly dependent on the moments in the screening layer.

MATERIAL

The **MATERIAL** statement sets the physical properties of materials.

MATERIAL

```
{ ( MATERIAL=<c> [NEW]) | SILICON | OXIDE | OXYNITRI | NITRIDE
  | POLYSILI | PHOTORES | ALUMINUM | AMBIENT
}
[TIF.NAME=<c>] [MD.INDEX=<n>] [IMPL.TAB=<c>]
[DY.DEFAU=<n>] [E.FIELD] [ION.PAIR] [IP.OMEGA=<n>]
[NI.0=<n>] [NI.E=<n>] [NI.F=<n>] [EPSILON=<n>]
[DENSITY=<n>] [AT.NUM=<n>] [AT.WT=<n>] [MOL.WT=<n>]
[VISC.0=<n>] [VISC.E=<n>] [VISC.X=<n>] [VC=<c>] [TEMPERAT=<c>]
[YOUNG.M=<n>] [POISS.R=<n>] [LCTE=<c>] [INTRIN.S=<n>] [SURF.TEN=<n>]
[ { ( SEMICOND [AFFINITY=<n>] [BANDGAP=<n>] [EGALPH=<n>] [EGBETA=<n>]
      [N.CONDUC=<n>] [N.VALENC=<n>] [NC.F=<n>] [NV.F=<n>]
      [G.DONOR=<n>] [E.DONOR=<n>]
      [G.ACCEP=<n>] [E.ACCEP=<n>]
      [BOLTZMAN] [IONIZATI]
      [QM.BETA=<n>] [QM.YCRIT=<n>])
  | ( CONDUCTO [WORKFUNC=<n>] ) }
]
[MAX.DAMA=<n>] [DAM.GRAD=<n>] [D.PCONST=<n>] [D.PME=<n>] [D.PDLF=<n>]
[D.PDLME=<n>] [D.PDRE=<n>] [D.PNORM=<n>] [D.PDE=<n>] [D.PHDF=<n>]
[D.PME=<n>] [D.PLF=<n>] [D.PLME=<n>]
[POLYCRYS] [GRASZ.0=<n>] [GRASZ.E=<n>] [TEMP.BRE=<n>] [MIN.GRAI=<n>]
[FRAC.TA=<n>] [G.DENS=<n>] [F11=<n>] [F22=<n>] [ALPHA=<n>]
[GEOM=<n>] [GAMMA.0=<n>] [GAMMA.E=<n>] [DSIX.0=<n>] [DSIX.E=<n>]
[DSIM.0=<n>] [DSIM.E=<n>] [DSIMM.0=<n>] [DSIMM.E=<n>] [DSIP.0=<n>]
[DSIP.E=<n>] [GBE.0=<n>] [GBE.H=<n>] [GBE.E=<n>] [NSEG=<n>]
[TBU.0=<n>] [TBU.E=<n>] [TOXIDE=<n>] [EAVEL.0=<n>] [EAVEL.E=<n>]
[DLGX.0=<n>] [DLGX.E=<n>] [POLY.FAC=<n>]
```

| Parameter | Type | Definition |
|-----------------|-----------|--|
| MATERIAL | character | The following parameters apply to the named material. Default: none Synonym: NAME |
| NEW | logical | Used to define a new material. The name specified by the MATERIAL parameter must not have been previously used as a material name. Default: false |
| SILICON | logical | The following parameters apply to silicon. Default: false |
| OXIDE | logical | The following parameters apply to oxide. Default: false |
| OXYNITRI | logical | The following parameters apply to oxynitride. Default: false |

| Parameter | Type | Definition |
|-------------------|-----------|--|
| NITRIDE | logical | The following parameters apply to nitride. Default: false |
| POLYSILI | logical | The following parameters apply to polysilicon. Default: false Synonym: POLY |
| PHOTORES | logical | The following parameters apply to photoresist. Default: false |
| ALUMINUM | logical | The following parameters apply to aluminum. Default: false |
| AMBIENT | logical | The following parameters apply to ambient. Default: false Synonym: GAS |
| TIF . NAME | character | The name by which this material is known in the TIF materials database. Note: This value of this parameter is case-sensitive. Default: the name of the material |
| MD . INDEX | number | The index to be used for this material when saving the structure in Medici file format. These correspond to the <i><matx></i> indices listed in Appendix E . Units: none Default: 0 (material ignored by Medici) |
| IMPL . TAB | character | The name of the material in the implant moment file to be associated with this material. Default: the current value for this material; initially the name of the material |
| DY . DEFAU | number | The grid spacing to be used in growing layers of the material. This value is scaled by the value of the GRID . FAC parameter on the METHOD statement. A value of zero specifies that no grid is to be added in growing layers. Units: microns Default: the current value; initially 0.0 Synonym: DX . DEFAU |
| E . FIELD | logical | Specifies that the electric field terms are to be included in the impurity diffusion equations for this material. (Meaningful only for silicon and polysilicon.) Default: the current value for this material; initially true |
| ION . PAIR | logical | Enables the model for pairing between donor and acceptor ions in this material. Default: the current value for this material; initially true for silicon and polysilicon |
| IP . OMEGA | number | The multiple of the intrinsic carrier concentration at which ion pairing becomes significant in this material. Units: none Default: the current value for this material; initially 6.0 for silicon and polysilicon |
| NI . 0 | number | The prefactor in the expression for the intrinsic carrier concentration. Units: carriers/cm ³ *(degrees Kelvin) ^{NI . F} Default: the current value for this material; initially 3.87e16 |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| NI.E | number | The activation energy in the expression for the intrinsic carrier concentration. Units: electron volts Default: the current value for this material; initially 0.605 |
| NI.F | number | The temperature exponent in the expression for the intrinsic carrier concentration. Units: none Default: the current value for this material; initially 1.5 Synonym: NI.POW |
| EPSILON | number | The relative permittivity of the material. Units: none Default: the current value for this material; see Appendix A for initial values |
| DENSITY | number | The density of the material. Units: gm/cm ³ Default: the current value; see Appendix A for initial values. |
| AT.NUM | number | The average atomic number of the material. This is the sum of the atomic numbers of the atoms in a molecule divided by the number of atoms. Units: none Default: the current value; see Appendix A for initial values. |
| AT.WT | number | The average atomic weight of the atoms of the material. This is the molecular weight of the material divided by the number of atoms per molecule. Units: atomic mass units Default: the current value; see Appendix A for initial values. |
| MOL.WT | number | The molecular weight of the material. Units: atomic mass units Default: the current value; see Appendix A for initial values. |
| VISC.0 | number | The exponential prefactor for the viscosity. Units: gm/(cm*sec) Default: the current value for this material |
| VISC.E | number | The activation energy for the viscosity. Units: electron volts Default: the current value for this material |
| VISC.X | number | The incompressibility factor. A value of 0.5 corresponds to an infinitely incompressible material. The value must be strictly less than 0.5. Units: none Default: the current value for this material |
| VC | character | A table of activation volumes as a function of temperature for the dependence of material viscosity on shear stress for the specified material in inert-ambient diffusions. Entries in the table correspond to temperatures given by the TEMPERAT parameter (see text). Units: Å ³ Default: the current value for this material |
| TEMPERAT | character | A list of temperatures corresponding to the values of the VC parameter. Units: degrees Celsius Default: none |

| Parameter | Type | Definition |
|-----------------|-----------|--|
| YOUNG.M | number | Young's modulus for the material. Units: dynes/cm ² Default: the current value for this material |
| POISS.R | number | Poisson's ratio for the material. Units: none Default: the current value for this material |
| LCTE | character | An expression giving the linear coefficient of thermal expansion as a function of absolute temperature, called T in the expression. It is given as a fraction, not as a percentage. Units: none Default: the current value for this material |
| INTRIN.S | number | The initial uniform stress state of a material such as a thin film of nitride deposited on the substrate. Units: dynes/cm ² Default: the current value for this material |
| SURF.TEN | number | The surface tension for this material. Affects reflow during oxidation with the VISCOUS model. Units: dynes/cm Default: the current value for this material |
| SEMICOND | logical | Specifies that the material is a semiconductor. Default: false |
| AFFINITY | number | The electron affinity of the material. Units: electron volts Default: the current value for this material |
| BANDGAP | number | The energy band gap of the material. Units: electron volts Default: the current value for this material |
| EGALPH | number | The value of alpha used in calculating the energy bandgap as a function of temperature. Units: eV/Kelvin Default: the current value for this material |
| EGBETA | number | The value of beta used in calculating the energy bandgap as a function of temperature. Units: Kelvins Default: the current value for this material |
| N.CONDUC | number | The effective density of electron states in the conduction band of the material at 300°K. Units: #/cm ³ Default: the current value for this material |
| N.VALENC | number | The effective density of hole states in the valence band of the material at 300°K. Units: #/cm ³ Default: none |

| Parameter | Type | Definition |
|-----------|---------|--|
| NC.F | number | The exponent of the temperature dependence for the effective density of electron states in the conduction band of the material Units: none Default: the current value for this material |
| NV.F | number | The exponent of the temperature dependence for the effective density of hole states in the valence band of the material Units: none Default: the current value for this material |
| G.DONOR | number | The donor impurity degeneracy factor of the material. Units: none Default: the current value for this material |
| E.DONOR | number | The donor impurity ionization energy of the material. Units: electron volts Default: the current value for this material |
| G.ACCEP | number | The acceptor impurity degeneracy factor of the material. Units: none Default: the current value for this material |
| E.ACCEP | number | The acceptor impurity ionization energy of the material. Units: electron volts Default: the current value for this material |
| BOLTZMAN | logical | Specifies that Boltzmann carrier statistics are used for this material during electrical calculations. If this parameter is false, Fermi-Dirac statistics are used. Default: the current value for this material |
| IONIZATI | logical | Specifies that complete impurity ionization is used for this material during electrical calculations. If this parameter is false, the impurities are considered to be incompletely ionized. Default: the current value for this material |
| QM.BETA | number | The proportional factor for van Dort's QM modeling. Units: eVcm Default: the current value for this material |
| QM.YCRIT | number | The critical depth of quantization for van Dort's QM modeling. Default: the current value for this material |
| CONDUCTO | logical | Specifies that the material is a conductor. Default: false |
| WORKFUNC | number | The work function of the material. Units: electron volts Default: the current value for this material |
| MAX.DAMA | number | The damage threshold for amorphization of this material. Units: #/cm ³ Default: the current value for this material; initially 1.15×10^{22} for silicon |
| DAM.GRAD | number | The sharpness of the amorphous/single-crystal interface in this material following solid-phase epitaxial regrowth Units: none Default: the current value for this material; initially 10 for silicon |

| Parameter | Type | Definition |
|-----------|---------|---|
| D.PCONST | number | An empirical constant for the effective +n implant damage model. Units: none Default: the current value for this material; initially 3.25 for silicon |
| D.PDME | number | The mass exponent in the low dose factor for the effective +n implant damage model. Units: none Default: the current value for this material; initially -0.5 for silicon |
| D.PDLF | number | The prefactor for lambda in the low dose factor for the effective +n implant damage model. Units: none Default: the current value for this material; initially -2 for silicon |
| D.PDLME | number | The mass exponent in lambda in the low dose factor for the effective +n implant damage model. Units: none Default: the current value for this material; initially -0.5 for silicon |
| D.PDRE | number | The projected range exponent in the low dose factor for the effective +n implant damage model. Units: none Default: the current value for this material; initially 0 for silicon |
| D.PNORM | number | The normalization for dose for the effective +n implant damage model. Units: #/cm ³ Default: the current value for this material; initially 3e12 for silicon |
| D.PDE | number | The dose exponent for the effective +n implant damage model. Units: none Default: the current value for this material; initially 1.0 for silicon |
| D.PHDF | number | The constant in the high dose factor for the effective +n implant damage model. Units: none Default: the current value for this material; initially 0.0905 for silicon |
| D.PME | number | The mass exponent in the high dose factor for the effective +n implant damage model. Units: none Default: the current value for this material; initially 0.85 for silicon |
| D.PLF | number | The prefactor for lambda in the high dose factor for the effective +n implant damage model. Units: none Default: the current value for this material; initially -2 for silicon |
| D.PLME | number | The mass exponent in lambda in the high dose factor for the effective +n implant damage model. Units: none Default: the current value for this material; initially -0.5 for silicon |
| POLYCRYS | logical | Specifies that the polycrystalline diffusion and grain-growth models are to be used for this material. Default: the current value for this material; initially true for polysilicon |

| Parameter | Type | Definition |
|------------|--------|---|
| GRASZ . 0 | number | The pre-exponential factor for the as-deposited grain size during polycrystalline (high-temperature) deposition. Units: none Default: the current value for this material |
| GRASZ . E | number | The activation energy for the as-deposited grain size during polycrystalline (high-temperature) deposition. Units: electron volts Default: the current value for this material |
| TEMP . BRE | number | The threshold temperature for amorphous deposition; deposition is amorphous at lower temperatures and polycrystalline at higher temperatures. Units: degrees Celsius Default: the current value for this material |
| MIN . GRAI | number | The minimum as-deposited grain size. Units: microns Default: the current value for this material |
| FRAC . TA | number | The geometrical factor for recrystallized grain size in amorphous regions. Units: none Default: the current value for this material |
| G . DENS | number | The geometrical factor for the density of grain boundaries. Units: none Default: the current value for this material; initially 2.0 for polysilicon |
| F11 | number | The geometrical factor for grain-boundary diffusion perpendicular to the columnar direction. Units: none Default: the current value for this material; initially 1.0 for polysilicon |
| F22 | number | The geometrical factor for grain-boundary diffusion parallel to the columnar direction. Units: none Default: the current value for this material; initially 2.0 for polysilicon |
| ALPHA | number | The geometrical factor relating the average grain boundary velocity to the grain growth rate. Units: none Default: the current value for this material; initially 1.33 for polysilicon |
| GEOM | number | The geometrical factor for the grain growth rate. Units: none Default: the current value for this material; initially 6.0 for polysilicon |
| GAMMA . 0 | number | Pre-exponential factor for the enhancement of silicon self-diffusivity at grain boundaries. Units: none Default: the current value for this material |
| GAMMA . E | number | Activation energy for the enhancement of silicon self-diffusivity at grain boundaries. Units: electron volts Default: the current value for this material |

| Parameter | Type | Definition |
|-----------|--------|---|
| DSIX.0 | number | Pre-exponential factor for the neutral component of silicon self diffusivity. Units: cm ² /sec Default: the current value for this material |
| DSIX.E | number | Activation energy for the neutral component of silicon self diffusivity. Units: electron volts Default: the current value for this material |
| DSIM.0 | number | Pre-exponential factor for the singly negative component of silicon self-diffusivity. Units: cm ² /sec Default: the current value for this material |
| DSIM.E | number | Activation energy for the singly negative component of silicon self-diffusivity. Units: electron volts Default: the current value for this material |
| DSIMM.0 | number | Pre-exponential factor for the doubly negative component of silicon self-diffusivity. Units: cm ² /sec Default: the current value for this material |
| DSIMM.E | number | Activation energy for the doubly negative component of silicon self-diffusivity. Units: electron volts Default: the current value for this material |
| DSIP.0 | number | Pre-exponential factor for the singly positive component of silicon self-diffusivity. Units: cm ² /sec Default: the current value for this material |
| DSIP.E | number | Activation energy for the singly positive component of silicon self-diffusivity. Units: electron volts Default: the current value for this material |
| GBE.0 | number | The grain boundary energy for normal grain growth. Units: electron volts Default: the current value for this material |
| GBE.H | number | The geometrical factor for the reduction of grain growth rate at large grain sizes. Units: none Default: the current value for this material |
| GBE.1 | number | The grain boundary energy for secondary grain growth. Units: electron volts Default: the current value for this material |
| NSEG | number | The exponent for solute drag on grain growth. Units: none Default: the current value for this material |
| TBU.0 | number | The prefactor in the expression for the time constant for polysilicon/silicon interfacial oxide break-up. Units: seconds Default: the current value for this material |

| Parameter | Type | Definition |
|-----------|--------|---|
| TBU.E | number | The activation energy for the time constant for polysilicon/silicon interfacial oxide break-up. Units: electron volts Default: the current value for this material |
| TOXIDE | number | The thickness of the interfacial oxide between deposited polysilicon layers and single-crystal silicon. Units: microns Default: the current value for this material |
| EAVEL.O | number | The prefactor in the expression for the velocity driving epitaxial regrowth of a polycrystalline layer. Units: cm/sec Default: the current value for this material |
| EAVEL.E | number | The activation energy for the velocity driving epitaxial regrowth of a polycrystalline layer. Units: electron volts Default: the current value for this material |
| DLGX.O | number | The prefactor in the expression for the grain size diffusivity. Units: cm ² /sec Default: the current value for this material |
| DLGX.E | number | The activation energy for the grain size diffusivity. Units: electron volts Default: the current value for this material |
| POLY.FAC | number | The adjustable parameter for polysilicon Monte Carlo implant model. Valid for polysilicon material only. Units: none Default: 0.1 |

Description

This statement is used to define materials and specify their properties. A material can be specified either by name, using the **MATERIAL** parameter, or with one of the **SILICON**, **OXIDE**, **OXYNITRI**, **NITRIDE**, **POLYSILI**, **PHOTORES**, **ALUMINUM**, or **AMBIENT** parameters. A new material can be defined by specifying the **NEW** parameter, the material name, and the properties of the material.



Note:

It is the responsibility of the user to ensure that all material properties required for a particular simulation step (e.g., viscosity and compressibility for oxidation or density and atomic number and weight for Monte Carlo ion implantation) have been specified.

The density, average atomic number, and average atomic weight of the material are used by the Monte Carlo ion implantation model. The density and molecular weight are used to calculate the volume changes that occur when materials are produced or consumed during reactions with other materials.

Viscosity and Compressibility

The **VISC.0**, **VISC.E**, and **VISC.X** parameters specify the viscosity as a function of temperature of the material for use with the **VISCOELA** and **VISCOUS** models for material flow. **VC** specifies the activation volume for the dependence of the material viscosity on shear stress when the stress-dependent **VISCOELA** or **VISCOUS** model is used. **YOUNG.M** and **POISS.R** give the viscosity and compressibility for use with the **COMPRESS** model and elasticity for the **VISCOELA** model. See [Reference \[28\]](#) in [Chapter 2](#) and [Reference \[42\]](#) in [Appendix A](#).

The only material properties that apply to **AMBIENT** are the viscosity parameters **VISC.0**, **VISC.E**, and **VISC.X**. These parameters apply to included voids, which may be formed during oxidation.

Stress Dependence

The parameters **VC** and **TEMPERAT** are used to specify the activation volume for dependence of viscosity on shear stress as functions of temperature during inert-ambient diffusions. A separate table is maintained for each material. **VC** specifies a list of values corresponding to the temperatures given by **TEMPERAT**. The portion of the table spanned by the specified temperatures is replaced by the specified values; the number of values must be the same as the number of temperatures, and the temperatures must be given in order, lowest to highest. The **CLEAR** parameter is used to clear the table before setting any values.

If **V.COMPAT** (on the **OPTION** statement) is less than 6.6, the specified values of the activation volumes apply to the specified material in all ambients, including oxidizing ambients.

An Arrhenius interpolation is used between values in the table. For temperatures outside the range of the table, the nearest value is used.

Examples

1. The following statement specifies the density of silicon:

```
MATERIAL MATERIAL=SILICON DENSITY=2.33
```

2. The following statement gives the thermal expansion coefficient of nitride as a function of absolute temperature T :

```
MATERIAL NITRIDE LCTE=(3e-6 + 2e-10 * T) +  
INTRIN.S=1.4e10
```

Thus, at 0K the coefficient is .0003%/K. The initial stress in the nitride film is $1.4e10$ dynes/cm².

3. The following statement specifies an oxide viscosity of 10^9 poise and a surface tension of 20 dynes/cm:

```
MATERIAL MATERIAL=OXIDE VISC.0=1e9 VISC.e=0 +  
SURF.TEN=20
```

These are values that might be used to simulate reflow of phosphosilicate or borophosphosilicate glass.

4. The following statements define a new material, tungsten disilicide, and specify some of its properties:

```
MATERIAL NEW MAT=WSi2 DENSITY=9.857 AT.NUM=34 +  
AT.WT=80.01  
MATERIAL MAT=WSi2 DY.DEFAU=0.1
```

A grid spacing of 0.1 microns (scaled by the value of **GRID.FAC**) is used when growing tungsten silicide.

IMPURITY

The **IMPURITY** statement is used to define impurities or modify their characteristics.

IMPURITY

```

IMPURITY=<c> [ ( NEW [MODEL=<c>] [C.INIT=<n>] ) ] [TIF.NAME=<c>]
[IMP.ACT=<c>] [IMP.GB=<c>][IMP.IT=<c>]
[ { DONOR | ACCEPTOR } ] [AT.NUM=<n>] [AT.WT=<n>]
[SOLVE] [STEADY] [IMPL.TAB=<c>]
[ MATERIAL=<c>
  [DIP.0=<n>] [DIP.E=<n>] [DIX.0=<n>] [DIX.E=<n>]
  [DIM.0=<n>] [DIM.E=<n>] [DIMM.0=<n>] [DIMM.E=<n>]
  [DVP.0=<n>] [DVP.E=<n>] [DVX.0=<n>] [DVX.E=<n>]
  [DVM.0=<n>] [DVM.E=<n>] [DVMM.0=<n>] [DVMM.E=<n>]
  [C.STATE=<n>] [DIC.0=<n>] [DIC.E=<n>] [DVC.0=<n>] [DVC.E=<n>] ]
  [D.MODEL=<c>] [DI.FAC=<c>] [DV.FAC=<c>]
  [DIFFUSE]
  [FGB=<n>]
  [DIPAIR.0=<n>] [DIPAIR.E=<n>] [DVPAIR.0=<n>] [DVPAIR.E=<n>]
  [R.I.S=<n>] [E.I.S=<n>] [R.V.S=<n>] [E.V.S=<n>]
  [R.IP.V=<n>] [E.IP.V=<n>] [R.VP.I=<n>] [E.VP.I=<n>]
  [SS.CLEAR] [SS.TEMP=<n>] [SS.CONC=<n>]
  [CTN.0=<n>] [CTN.E=<n>] [CTN.F=<n>] [CL.INI.A]
  [DDC.F.0=<n>] [DDC.F.E=<n>] [DDC.T.0=<n>] [DDC.T.E=<n>]
  [DDCF.D.N=<n>] [DDCF.N.N=<n>] [DDCF.I.N=<n>]
  [DDCR.N.N=<n>] [DDCR.I.N=<n>] [IFRACM=<n>]
  [DDCS.0=<n>] [DDCS.E=<n>] [DDCS.N=<n>] [IFRACS=<n>]
  [Q.SITES=<n>] [CG.MAX=<n>] [GSEG.0=<n>] [GSEG.E=<n>]
  [GSEG.INI=<n>] [VELIF.0=<n>] [VELIF.E=<n>]
  [ /MATERIA=<c>
    [SEG.0=<n>] [SEG.E=<n>] [TRANS.0=<n>] [TRANS.E=<n>]
    [RATIO.0=<n>] [RATIO.E=<n>] [SEG.SS]
    [ /SEG.0=<n>] [ /SEG.E=<n>] [ /TRANS.0=<n>] [ /TRANS.E=<n>]
    [ /RATIO.0=<n>] [ /RATIO.E=<n>] [ /SEG.SS]
    {SEG.EQ3 | SEG.EQ2 | /SEG.EQ2}
    [Q.INI.0=<n>] [Q.INI.E=<n>] [Q.MAX.0=<n>] [Q.MAX.E=<n>]
    [TWO.PHAS]
  ]
  [ES.RAND=<n>] [ES.F.RAN=<n>] [ES.BREAK=<n>] [ES.F.H=<n>]
  [ES.100=<n>] [ES.F.100=<n>] [ES.110=<n>] [ES.F.110=<n>]
  [NLOC.PRE=<n>] [NLOC.EXP=<n>] [NLOC.MAX=<n>] [NLOC.K=<n>]
  [LOC.FAC=<n>] [CHAN.CRI=<n>] [CHAN.FAC=<n>] [DISP.FAC=<n>]
]
[T.ACT.0=<n>] [T.ACT.E=<n>] [ACT.MIN=<n>]
[D.PNORM=<n>] [D.PDE=<n>]
[CM.SEC]

```

| Parameter | Type | Definition |
|-------------------|-----------|---|
| IMPURITY | character | The name of the impurity to be defined or modified. Default: none Synonym: IMP, NAME |
| NEW | logical | Specifies that a new impurity is being defined. Default: false |
| MODEL | character | The name of the model to which the diffusion equations for the new impurity are registered. The ENABLE parameter with the MODEL parameter given by this name in the METHOD statement specifies the diffusion equation for this impurity is to be solved. Default: none |
| C . INIT | number | The initial background concentration of the impurity. Units: atoms/cm ³ Default: the current value for this impurity; initially 0 |
| TIF . NAME | character | The name by which this impurity is known in the TIF materials database. Note: This value of this parameter is case-sensitive. Default: the name of the impurity |
| IMP . ACT | character | The name of the solution value representing the active concentration of the impurity named in the IMPURITY parameter. Required only when the ACT . TRAN model is to be applied to the impurity. Default: none |
| IMP . GB | character | The name of the solution value representing the grain boundary concentration of the impurity named in the IMPURITY parameter. Indicates that the polycrystalline diffusion model applies to this impurity. Default: none |
| IMP . IT | character | The name of the solution value representing the density of the trapped impurity at interface named in the IMPURITY parameter. Required only when the ITRAP model is to be applied to the impurity. Default: none |
| DONOR | logical | Specifies that the impurity is a donor. Default: the current value for this impurity |
| ACCEPTOR | logical | Specifies that the impurity is an acceptor. Default: the current value for this impurity |
| AT . NUM | number | The atomic number of the impurity. Units: none Default: the current value for this impurity |
| AT . WT | number | The atomic weight of the impurity. Units: atomic mass units Default: the current value for this impurity |
| SOLVE | logical | Specifies that the diffusion equations should be solved for this impurity. Default: true if NEW is specified; the current value for this impurity otherwise |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| STEADY | logical | Specifies that the steady-state conditions should be assumed when solving the diffusion equations for this impurity. Default: false if NEW is specified; the current value for this impurity otherwise |
| IMPL.TAB | character | The name of the implant moment table to use for this impurity. Default: the current implant moment table for this impurity |
| MATERIAL | character | The name of the material in which the diffusion, activation, and implant parameters apply and material 1 for the segregation and transport parameters. Default: none |
| DIP.0 | number | The pre-exponential constant for diffusion with positively charged interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIP.E | number | The activation energy for diffusion with positively charged interstitials. Units: electron volts Default: the current value for this material |
| DIX.0 | number | The pre-exponential constant for diffusion with neutral interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIX.E | number | The activation energy for diffusion with neutral interstitials. Units: electron volts Default: the current value for this material |
| DIM.0 | number | The pre-exponential constant for diffusion with singly negative interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIM.E | number | The activation energy for diffusion with singly negative interstitials. Units: electron volts Default: the current value for this material |
| DIMM.0 | number | The pre-exponential constant for diffusion with doubly negative interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIMM.E | number | The activation energy for diffusion with doubly negative interstitials. Units: electron volts Default: the current value for this material |
| DVP.0 | number | The pre-exponential constant for diffusion with positively charged vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVP.E | number | The activation energy for diffusion with positively charged vacancies. Units: electron volts Default: the current value for this material |
| DVX.0 | number | The pre-exponential constant for diffusion with neutral vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |

| Parameter | Type | Definition |
|----------------|-----------|--|
| DVX.E | number | The activation energy for diffusion with neutral vacancies. Units: electron volts Default: the current value for this material |
| DVM.0 | number | The pre-exponential constant for diffusion with singly negative vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVM.E | number | The activation energy for diffusion with singly negative vacancies. Units: electron volts Default: the current value for this material |
| DVMM.0 | number | The pre-exponential constant for diffusion with doubly negative vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVMM.E | number | The activation energy for diffusion with doubly negative vacancies. Units: electron volts Default: the current value for this material |
| C.STATE | number | The point defect charge state to which the DIC.0 , DIC.E , DVC.0 , and DVC.E parameters apply. The value must be in the range of -6 to +6. Units: none Default: none |
| DIC.0 | number | The pre-exponential constant for diffusion with interstitials in the charge state C.STATE . Units: none Default: the current value for this material |
| DIC.E | number | The activation energy for diffusion with interstitials in the charge state C.STATE . Units: electron volts Default: the current value for this material |
| DVC.0 | number | The pre-exponential constant for diffusion with vacancies in the charge state C.STATE . Units: none Default: the current value for this material |
| DVC.E | number | The activation energy for diffusion with vacancies in the charge state C.STATE . Units: electron volts Default: the current value for this material |
| D.MODEL | character | The name of the model to which the diffusivity modifiers of DI.FAC and DV.FAC are registered. Default: none |
| DI.FAC | character | The formula of multiplication factor to the diffusion with interstitials. Default: none |
| DV.FAC | character | The formula of multiplication factor to the diffusion with vacancies. Default: none |
| DIFFUSE | logical | Specifies that the diffusion equation is to be solved. Default: true |

| Parameter | Type | Definition |
|-----------------|--------|---|
| FGB | number | The factor by which the diffusivity is increased in polycrystalline materials when the polycrystalline model is not enabled. Units: none Default: the current value for this material |
| DIPAIR.0 | number | The pre-exponential constant for the diffusivity of dopant-interstitial pairs. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIPAIR.E | number | The activation energy for the diffusivity of dopant-interstitial pairs. Units: electron volts Default: the current value for this material |
| DVPAIR.0 | number | The pre-exponential constant for the diffusivity of dopant-vacancy pairs. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVPAIR.E | number | The activation energy for the diffusivity of dopant-vacancy pairs. Units: electron volts Default: the current value for this material |
| R.I.S | number | The capture radius for the reaction between interstitials and substitutional dopant atoms. Units: Å Default: the current value for this material |
| E.I.S | number | The barrier energy for the reaction between interstitials and substitutional dopant atoms. Units: electron volts Default: the current value for this material |
| R.V.S | number | The capture radius for the reaction between vacancies and substitutional dopant atoms. Units: Å Default: the current value for this material |
| E.V.S | number | The barrier energy for the reaction between vacancies and substitutional dopant atoms. Units: electron volts Default: the current value for this material |
| R.IP.V | number | The capture radius for the reaction between dopant-interstitial pairs and vacancies. Units: Å Default: the current value for this material |
| E.IP.V | number | The barrier energy for the reaction between dopant-interstitial pairs and vacancies. Units: electron volts Default: the current value for this material |
| R.VP.I | number | The capture radius for the reaction between dopant-vacancy pairs and interstitials. Units: Å Default: the current value for this material |

| Parameter | Type | Definition |
|-----------------|---------|--|
| E.VP.I | number | The barrier energy for the reaction between dopant-vacancy pairs and interstitials. Units: electron volts Default: the current value for this material |
| SS.CLEAR | logical | Clears the solid solubility vs. temperature table. Default: false |
| SS.TEMP | number | The temperature at which the solid solubility in the material is SS.CONC . Units: degrees Celsius Default: none |
| SS.CONC | number | The solid solubility in the material at temperature SS.TEMP . Units: atoms/cm ³ Default: none |
| CTN.O | number | The pre-exponential constant for impurity clustering. Units: (atoms/cm ³) ^(1/CTN.F-1) Default: the current value for this material |
| CTN.E | number | The activation energy for impurity clustering. Units: electron volts Default: the current value for this material |
| CTN.F | number | The exponent of concentration for impurity clustering. Units: none Default: the current value for this material |
| CL.INI.A | logical | Specifies that impurities in an amorphized region are initially clustered when ACT.TRAN is used. Default: the current value for this impurity and material; initially true for built-in impurities in silicon |
| DDC.F.O | number | The pre-exponential constant for dopant-defect clustering. Units: none Default: the current value for this material |
| DDC.F.E | number | The activation energy for dopant-defect clustering. Units: electron volts Default: the current value for this material |
| DDC.T.O | number | The pre-exponential constant for the dopant-defect cluster dissolution time. Units: seconds Default: the current value for this material |
| DDC.T.E | number | The activation energy for the dopant-defect cluster dissolution time. Units: electron volts Default: the current value for this material |
| DDCF.D.N | number | The power of the dopant concentration in the dopant-defect cluster formation reaction. Units: none Default: the current value for this material |

| Parameter | Type | Definition |
|-----------|--------|---|
| DDCF.N.N | number | The power of the electron concentration the dopant-defect cluster formation reaction. Units: none Default: the current value for this material |
| DDCF.I.N | number | The power of the interstitial concentration the dopant-defect cluster formation reaction. Units: none Default: the current value for this material |
| DDCR.N.N | number | The power of the electron concentration the dopant-defect cluster dissolution reaction. Units: none Default: the current value for this material |
| DDCR.I.N | number | The power of the interstitial concentration the dopant-defect cluster dissolution reaction. Units: none Default: the current value for this material |
| IFRACM | number | The number of interstitials per boron atom trapped in dopant-defect clusters. Units: none Default: the current value for this material |
| DDCS.O | number | The pre-exponential constant for the dopant contained in small dopant-defect clusters at equilibrium. Units: none Default: the current value for this material |
| DDCS.E | number | The activation energy for the dopant contained in small dopant-defect clusters at equilibrium. Units: electron volts Default: the current value for this material |
| DDCS.N | number | The number of dopant atoms contained in each small dopant-defect cluster. Only values of 1 and 2 are meaningful, and only the value in silicon is used. Units: none Default: the current value for this material |
| IFRACS | number | The number of interstitials per dopant atom trapped in small dopant-defect clusters. Units: none Default: the current value for this material |
| Q.SITES | number | The density of dopant sites in grain boundaries (the maximum allowable value for the grain-boundary concentration) in a polycrystalline material. Units: #/cm ² Default: the current value for this material |
| CG.MAX | number | The density of dopant sites in the grain interior of a polycrystalline material. Units: #/cm ³ Default: the current value for this material; initially 5e22 for polysilicon |

| Parameter | Type | Definition |
|-------------------|-----------|--|
| GSEG . 0 | number | The entropy for segregation between grain interior and boundaries in a polycrystalline material. Units: none Default: the current value for this material Synonym: A . SEG |
| GSEG . E | number | The activation energy of the segregation entropy between grain interior and boundaries in a polycrystalline material. Units: electron volts Default: the current value for this material Synonym: E . SEG |
| GSEG . INI | number | The entropy for the initial segregation between grain interior and boundaries in a polycrystalline material. Units: none Default: the current value for this material Synonym: A0 . SEG |
| VELIF . 0 | number | The pre-exponential factor for the segregation velocity at the boundaries of a polycrystalline material. Units: cm/sec Default: the current value for this material |
| VELIF . E | number | The activation energy for the segregation velocity at the boundaries of a polycrystalline material. Units: electron volts Default: the current value for this material |
| /MATERIA | character | The name of material 2 for the segregation and transport parameters. Default: none |
| SEG . 0 | number | The pre-exponential factor for segregation from material 1 to material 2. For the trapped impurity at interface, the pre-exponential factor for segregation from material 1 to interface adjacent to material 2. Units: none for the segregation from material 1 to material 2, cm^{-1} for the segregation from material 1 to interface Default: the current value for these materials |
| SEG . E | number | The activation energy for segregation from material 1 to material 2. For the trapped impurity at interface, the activation energy for segregation from material 1 to interface adjacent to material 2. Units: electron volts Default: the current value for these materials |
| TRANS . 0 | number | The pre-exponential factor for transport from material 1 to material 2. For the trapped impurity at interface, the pre-exponential factor for transport from material 1 to interface adjacent to material 2. Units: microns/min or cm/sec Default: the current value for these materials Synonym: TRN . 0 |

| Parameter | Type | Definition |
|-----------------|---------|---|
| TRANS.E | number | The activation energy for transport from material 1 to material 2. For the trapped impurity at interface, the activation energy for transport from material 1 to interface adjacent to material 2. Units: electron volts Default: the current value for these materials Synonym: TRN.E |
| RATIO.0 | number | The pre-exponential factor for ratio of detrapping rate to trapping rate for material 1 at interface adjacent to material 2. Used only with the ITRAP model. Units: none Default: the current value for these materials |
| RATIO.E | number | The activation energy for ratio of detrapping rate to trapping rate for material 1 at interface adjacent to material 2. Used only with the ITRAP model. Units: none Default: the current value for these materials |
| SEG.SS | logical | Specifies that the segregation from material 1 to interface adjacent to material 2 is calculated from solid solubility instead of SEG.0 and SEG.E . Used only with the ITRAP model. Default: false |
| /SEG.0 | number | The pre-exponential factor for segregation from material 2 to interface adjacent to material 1 for the trapped impurity at interface. Used only with the ITRAP model. Units: cm ⁻¹ Default: the current value for these materials |
| /SEG.E | number | The activation energy for segregation from material 2 to interface adjacent to material 1 for the trapped impurity at interface. Used only with the ITRAP model. Units: electron volts Default: the current value for these materials |
| /TRANS.0 | number | The pre-exponential factor for transport from material 2 to interface adjacent to material 1 for the trapped impurity at interface. Used only with the ITRAP model. Units: microns/min or cm/sec Default: the current value for these materials Synonym: /TRN.0 |
| /TRANS.E | number | The activation energy for transport from material 2 to interface adjacent to material 1 for the trapped impurity at interface. Used only with the ITRAP model. Units: electron volts Default: the current value for these materials Synonym: /TRN.E |
| /RATIO.0 | number | The pre-exponential factor for ratio of detrapping rate to trapping rate for material 2 at interface adjacent to material 1. Used only with the ITRAP model. Units: none Default: the current value for these materials |

| Parameter | Type | Definition |
|-----------------|---------|---|
| /RATIO.E | number | The activation energy for ratio of detrapping rate to trapping rate for material 2 at interface adjacent to material 1. Used only with the ITRAP model. Units: none Default: the current value for these materials |
| /SEG.SS | logical | Specifies that the segregation from material 2 to interface adjacent to material 1 is calculated from solid solubility instead of /SEG.0 and /SEG.E . Used only with the ITRAP model. Default: false |
| SEG.EQ3 | logical | Specifies that both of segregations from material 1 to interface and from material 2 to interface are determined from the values of the parameters, SEG.0 , SEG.E , /SEG.0 and /SEG.E . Used only with the ITRAP model. Default: true |
| SEG.EQ2 | logical | Specifies that the segregation from material 1 to interface is determined so as to make the concentration ratio in materials 1 and 2 in 3-phase equilibrium equal to the one in 2-phase equilibrium. Used only with the ITRAP model. Default: false |
| /SEG.EQ2 | logical | Specifies that the segregation from material 2 to interface is determined so as to make the concentration ratio in materials 1 and 2 in 3-phase equilibrium equal to the one in 2-phase equilibrium. Used only with the ITRAP model. Default: false |
| Q.INI.0 | number | The pre-exponential factor for the initial occupied trap density at interface. Used only with the ITRAP model. Units: atoms/cm ² Default: the current value for these materials |
| Q.INI.E | number | The activation energy for the initial occupied trap density at interface. Used only with the ITRAP model. Units: electron volts Default: the current value for these materials |
| Q.MAX.0 | number | The pre-exponential factor for the maximum trap density at interface. Used only with the ITRAP model. Units: atoms/cm ² Default: the current value for these materials |
| Q.MAX.E | number | The activation energy for the maximum trap density at interface. Used only with the ITRAP model. Units: electron volts Default: the current value for these materials |
| TWO.PHAS | logical | Specifies that the flux of 2-phase segregation is added to the one of 3-phase segregation flux. Used only with the ITRAP model. Default: false |
| ES.RAND | number | The electronic stopping power coefficient of implanted atoms in the specified material for materials other than silicon and for a nonchanneled direction in silicon. This value is used for the Monte Carlo ion implant calculation only. Units: angstrom ² *eV ^(1-ES.F.RAN) Default: the current value for the specified materials |

| Parameter | Type | Definition |
|-----------|--------|--|
| ES.F.RAN | number | The exponent of the electronic stopping power of implanted atoms in the specified material for materials other than silicon and for a nonchanneled direction in silicon. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for the specified materials |
| ES.BREAK | number | The energy above which ES.F.H is used instead of ES.F.RAN, ES.F.100, or ES.F.110 in calculating the electronic stopping power coefficient of implanted atoms. This value is used for the Monte Carlo ion implant calculation only. Units: keV Default: the current value for the specified materials |
| ES.F.H | number | The exponent of the electronic stopping power of implanted atoms at energies above ES.BREAK. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for the specified materials |
| ES.100 | number | The electronic stopping power for implanted atoms in silicon along the <100> channeling axes. This value is used for the Monte Carlo ion implant calculation only. Units: angstrom ² *eV ^(1-ES.F.100) Default: the current value for the specified materials |
| ES.F.100 | number | The exponent of the electronic stopping power for implanted atoms in silicon along the <100> crystal axes. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for the specified material |
| ES.110 | number | The electronic stopping power for implanted atoms in silicon along the <110> channeling axes. This value is used for the Monte Carlo ion implant calculation only. Units: angstrom ² *eV ^(1-ES.F.100) Default: the current value for the specified material |
| ES.F.110 | number | The exponent of the electronic stopping power for implanted atoms in silicon along the <110> crystal axes. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for the specified material |
| NLOC.PRE | number | The prefactor for the nonlocal electronic stopping power formula. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: The current value for the specified materials |
| NLOC.EXP | number | The exponent for the nonlocal electronic stopping power formula. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: The current value for the specified materials |

| Parameter | Type | Definition |
|-----------------|--------|---|
| NLOC.MAX | number | The maximum value allowed for the nonlocal part of electronic stopping power. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: The current value for the specified materials |
| NLOC.K | number | A correction factor for the LSS electron stopping power. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: The current value for the specified materials |
| LOC.FAC | number | A correction factor for the local part of electronic stopping power. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: The current value for the specified materials |
| CHAN.CRI | number | Specifies the critical angle below which the scattering angle is automatically reduced. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: The current value for the specified materials |
| CHAN.FAC | number | Specifies the factor by which the scattering angle is reduced. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: The current value for the specified materials |
| DISP.FAC | number | This is a correction factor for the probability of selecting the amorphous model. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: The current value for the specified materials |
| T.ACT.O | number | The prefactor in the expression for the activation time constant for the ACT.TRAN model for this impurity. Units: min or sec Default: the current value for this impurity. |
| T.ACT.E | number | The activation energy in the expression for the activation time constant for the ACT.TRAN model for this impurity. Units: electron volts Default: the current value for this impurity. |
| ACT.MIN | number | The factor by which the intrinsic carrier concentration is multiplied to obtain the minimum activation level used by the ACT.TRAN model for this impurity. Units: none Default: the current value for this impurity. |
| D.PNORM | number | The normalizing dose for the effective +n implant damage model. Units: #/cm ² Default: the current value for this impurity. |
| D.PDE | number | The power of the dose factor in the effective +n implant damage model. Units: none Default: the current value for this impurity. Synonym: D.PF |

| Parameter | Type | Definition |
|-----------------|---------|---|
| CM . SEC | logical | If true, parameters involving time are specified in centimeters and seconds; if false, parameters involving time are in microns and minutes. Default: false |

Description

This statement is used to define new impurities and specify their properties. If **NEW** is specified, a new impurity is defined; otherwise the properties of an existing impurity are set.

The coefficients for standard impurities are given in the *s4init* file (which is read at the start of each TSUPREM-4 execution), but these can be changed by you at any time. Coefficients that are *not* given in the *s4init* file or set by the user, default to 0.0, except for **SEG . 0**, which defaults to 1.0.

Parameters whose units include time are specified in units of microns and minutes, unless **CM . SEC** is true, in which case units of centimeters and seconds are assumed.

The **IMPURITY** statement replaces the **ANTIMONY**, **ARSENIC**, **BORON**, and **PHOSPHORUS** statements; the old statements are still available, however, for compatibility with existing input files.

Impurity Type

DONOR and **ACCEPTOR** specify whether the impurity is a donor or an acceptor. By default, newly defined impurities are electrically inactive. Once they have been specified to be active (with the **DONOR** or **ACCEPTOR** parameters), they cannot be returned to their electrically inactive state.

Solution Options

SOLVE specifies whether diffusion equations are solved for this impurity; if **SOLVE** is false, the impurity is assumed to be immobile. If the **MODEL** name is given, turning off the given model in **METHOD** statement is the same as turning off **SOLVE** in **IMPURITY** statement. Turning off **DIFFUSE** parameter specifies that the diffusion equation is not solved for this impurity in the specified material. **STEADY** specifies whether a steady-state or full transient solution to the diffusion equations should be computed. The steady-state solution is preferred for impurities that diffuse rapidly or react at material interfaces to produce material growth or consumption. The full transient solution is preferred for impurities that diffuse slowly or interact with other impurities.

Other Parameters

AT . NUM and **AT . WT** are used by the Monte Carlo implantation model. **IMPL . TAB** specifies the name of the implant moment table to be used for the impurity.

The initial background concentration of a user-defined impurity when it is first added to a structure is given by **C.INIT**. For normal impurities **C.INIT** should be greater than zero; a value of $1e5$ (the value used for built-in impurities) is suggested.

Multiplication to Diffusivity

DI.FAC and **DV.FAC** describe the formulas for multiplication factors to diffusivities. **D.MODEL** names the **DI.FAC** and/or **DV.FAC** so that its application can be turned *on* or *off*. For example:

```
IMPURITY IMP=BORON MAT=OXIDE D.MODEL=myDiff +
          DI.FAC=1+1e-21*fluorine
METHOD   MODEL=myDiff !ENABLE
```

Further Reading

For further reading and additional information see the following sections:

- Use of the diffusion parameters is described in [Chapter 2, “Diffusion of Impurities” on page 2-17](#).
- The solid solubility and clustering parameters are described in [Chapter 2, “Activation of Impurities” on page 2-24](#).
- The segregation parameters are described in [Chapter 2, “Segregation of Impurities” on page 2-28](#).
- The electronic stopping power parameters are described in [Chapter 2, “Monte Carlo Ion Implant Model” on page 2-87](#).

Examples

1. The following statement specifies the diffusivity of antimony diffusing with neutral vacancies in silicon:

```
IMPURITY IMP=ANTIMONY MAT=SILICON DVX.0=1.22e9 +
          DVX.E=3.65
```

2. The following statement specifies the diffusivity of phosphorus with triply-negative interstitials:

```
IMPURITY IMP=P MAT=SILI C.STATE=-3 DIC.0=2e11 +
          DIC.E=4.37
```

3. The following statement specifies the segregation parameters at the Si/SiO₂ interface:

```
IMPURITY IMP=ANTIMONY MAT=SILICON /MAT=OXIDE +
          SEG.0=30.0 TRANS.0=0.1
```


The concentration in silicon is 30.0 times the concentration in oxide, at equilibrium.

4. The following statements define a new impurity and set some of its properties:

```
IMPURITY NEW IMPURITY=CESIUM AT.NUM=55 +  
          AT.WT=132.9 C.INIT=1e5  
IMPURITY IMP=CESIUM MAT=OXIDE DIX.0=0.5 +  
          DIX.E=2.9
```

REACTION

The **REACTION** statement defines the reactions that occur at material interfaces.

REACTION

```

{ MAT.BULK=<c> | ( MAT.R=<c> /MAT.L=<c> ) }
[MODEL=<c>] [NAME=<c>]
[ { DELETE | REPLACE } ]
{ ( [IMP.L=<c>] [NI.L=<n>] [EI.L=<n> ]
  [ /IMP.L=<c>] [ /NI.L=<n>] [ /EI.L=<n> ]
  [IMP.R=<c>] [NI.R=<n>] [EI.R=<n> ]
  [ /IMP.R=<c>] [ /NI.R=<n>] [ /EI.R=<n> ]
  [NM.R=<n>] [ /NM.L=<n>]
  [RATE.0=<n>] [RATE.E=<n>] [EQUIL.0=<n>] [EQUIL.E=<n>]
  )
  | ( MAT.NEW=<c> THICKNES=<n> )
}

```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| MAT.BULK | character | A name of bulk material, in which the reaction equation is applied. Default: none Synonym: MAT |
| MAT.R | character | Material 1, which appears on the right side of the reaction equation. Default: none Synonym: MAT |
| /MAT.L | character | Material 2, which appears on the left side of the reaction equation. Default: none |
| MODEL | character | A name used to refer to this reaction. By turning on/off the ENABLE parameter with the MODEL parameter in the METHOD statement, the reaction can be enabled or disabled. Default: none |
| NAME | character | A name used to refer to this reaction in subsequent REACTION statements. Default: "<none>" |
| DELETE | logical | Deletes the specified reaction. Default: false |
| REPLACE | logical | Replaces the specified reaction. Default: false |
| IMP.L | character | The impurity in material 1 that appears on the left side of the reaction equation. Default: none |
| NI.L | number | The number of molecules of IMP.L that participate in the reaction. Units: none Default: 1.0 |

| Parameter | Type | Definition |
|---------------|-----------|---|
| EI.L | number | The exponent describing the dependence of the forward reaction rate on the concentration of IMP.L . Units: none Default: NI.L |
| /IMP.L | character | The impurity in material 2 that appears on the left side of the reaction equation. Default: none |
| /NI.L | number | The number of molecules of /IMP.L that participate in the reaction. Units: none Default: 1.0 |
| /EI.L | number | The exponent describing the dependence of the forward reaction rate on the concentration of /IMP.L . Units: none Default: /NI.L |
| IMP.R | character | The impurity in material 1 that appears on the right side of the reaction equation. Default: none |
| NI.R | number | The number of molecules of IMP.R that participate in the reaction. Units: none Default: 1.0 |
| EI.R | number | The exponent describing the dependence of the reverse reaction rate on the concentration of IMP.R . Units: none Default: NI.R |
| /IMP.R | character | The impurity in material 2 that appears on the right side of the reaction equation. Default: none |
| /NI.R | number | The number of molecules of /IMP.R that participate in the reaction. Units: none Default: 1.0 |
| /EI.R | number | The exponent describing the dependence of the reverse reaction rate on the concentration of /IMP.R . Units: none Default: /NI.R |
| NM.R | number | The number of molecules of MAT.R that participate in the reaction. Units: none Default: 0.0 |
| /NM.L | number | The number of molecules of /MAT.L that participate in the reaction. Units: none Default: 0.0 |
| RATE.O | number | The prefactor in the expression for the forward reaction rate. Units: varies Default: 0.0 |
| RATE.E | number | The activation energy for the forward reaction rate. Units: none Default: 0.0 |

| Parameter | Type | Definition |
|-----------------|-----------|---|
| EQUIL.O | number | The prefactor in the expression for the ratio of the reverse reaction rate to the forward reaction rate. Units: varies Default: 0.0 |
| EQUIL.E | number | The activation energy for the ratio of the reverse reaction rate to the forward reaction rate. Units: none Default: 0.0 |
| MAT.NEW | character | The name of the material to be added between layers of MAT.R and /MAT.L when one is deposited on the other. Default: none |
| THICKNES | number | The thickness of MAT.NEW to be added between layers of MAT.R and /MAT.L when one is deposited on the other. Units: microns Default: none |

Description

The **REACTION** statement can specify either a reaction that takes place at the interface between two materials or the addition of a *native* layer when one material is deposited on another. Each reaction is identified by its name and the materials on either side of the interface. The name is optional: if no name is specified, the name “<none>” is assumed.

Reactions defined with the **REACTION** statement take place during **DIFFUSION** steps with either oxidizing or inert ambients. The reactions occur only when the **VERTICAL**, **COMPRESS**, or **VISCOELA** oxidation model has been selected.

Defining and Deleting

A new reaction is defined by specifying a name (optional), the materials on either side of the interface, and the impurity and reaction rate parameters. The parameters for a previously defined reaction can be changed by specifying the name and materials for the reaction along with any parameters that need to be changed. A previously defined reaction can be deleted with the **DELETE** parameter. The **REPLACE** parameter deletes a previously defined reaction then replaces it with a new reaction defined by the parameters on the **REACTION** statement.

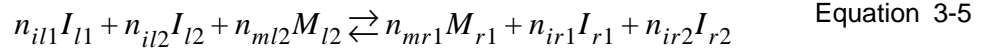
Insertion of Native Layers

The **MAT.NEW** and **THICKNES** parameters specify a new material to be inserted between **MAT.R** and **/MAT.L** when one is deposited on the other. A layer of **MAT.NEW** with thickness **THICKNES** separates **MAT.R** and **/MAT.L**. The pro-

cess is analogous to the insertion of a native oxide layer between exposed silicon and an oxidizing ambient.

Reaction Equation

The general form of the reaction is



where the subscripts *l* and *r* denote terms on the left and right sides of the reaction and subscripts 1 and 2 refer to materials 1 and 2. The forward and reverse reaction rates are given by

$$R_f = k_f [I_{l1}]^{e_{l1}} [I_{l2}]^{e_{l2}} \quad \text{Equation 3-6}$$

$$R_r = k_r [I_{r1}]^{e_{r1}} [I_{r2}]^{e_{r2}} \quad \text{Equation 3-7}$$

where $[I_x]$ denotes the concentration of impurity I_x .

Parameters

Specify the parameters of this reaction as follows:

$$I_{l1} = \text{IMP.L}, \quad I_{l2} = \text{/IMP.L}, \quad I_{r1} = \text{IMP.R}, \quad I_{r2} = \text{/IMP.R} \quad \text{Equation 3-8}$$

$$n_{il1} = \text{NI.L}, \quad n_{il2} = \text{/NI.L}, \quad n_{ir1} = \text{NI.R}, \quad n_{ir2} = \text{/NI.R} \quad \text{Equation 3-9}$$

$$n_{mr1} = \text{NM.R}, \quad n_{ml2} = \text{/NM.L} \quad \text{Equation 3-10}$$

$$e_{il1} = \text{EI.L}, \quad e_{il2} = \text{/EI.L}, \quad e_{ir1} = \text{EI.R}, \quad e_{ir2} = \text{/EI.R} \quad \text{Equation 3-11}$$

$$k_f = \text{RATE.0} \cdot \exp\left(\frac{-\text{RATE.E}}{kT}\right) \quad \text{Equation 3-12}$$

$$\frac{k_r}{k_f} = \frac{[I_{l1}]^{e_{l1}} [I_{l2}]^{e_{l2}}}{[I_{r1}]^{e_{r1}} [I_{r2}]^{e_{r2}}} = \text{EQUIL.0} \cdot \exp\left(\frac{-\text{EQUIL.E}}{kT}\right) \quad \text{Equation 3-13}$$

- **NI.L**, **/NI.L**, **NI.R**, **/NI.R**, **NM.R**, and **/NM.L** determine the number of molecules of each reactant that participate in the reaction.
- **EI.L**, **/EI.L**, **EI.R**, **/EI.R**, **RATE.0**, **RATE.E**, **EQUIL.0**, and **EQUIL.E** determine the rate of the reaction.

In theory the n_i and e_i for each reaction would be equal, but this is not required by TSUPREM-4. This allows the rate to depend on the concentration of an impurity without the concentration being affected by the reaction ($e_i \neq 0, n_i = 0$), or for the concentration to change without affecting the rate ($e_i = 0, n_i \neq 0$).

The units of R_f and R_r are #/cm²/sec; the units of k_f and k_r depend on the values of the n_i and e_i .

Effects

This reaction has two effects on the simulation:

1. As a boundary condition for diffusing species, it can result in generation, consumption, or transport of impurities across a material interface. The impurity generation fluxes (F , in #/cm²/sec) at the interface are given by

$$-\frac{1}{n_{il1}}F_{l1} = -\frac{1}{n_{il2}}F_{l2} = \frac{1}{n_{ir1}}F_{r1} = \frac{1}{n_{ir2}}F_{r2} = R_f - R_r \quad \text{Equation 3-14}$$

(Note that when n_i is zero, the corresponding flux is also zero.)

2. It can produce growth or consumption of material regions. This occurs for materials for which $n_m \neq 0$. The growth rate of M_{rl} , in cm/sec, is given by

$$V = \frac{n_{mr1} \times \text{MOL. WT}}{A \times \text{DENSITY}} (R_f - R_r) \quad \text{Equation 3-15}$$

where V is the growth velocity in cm/sec, $A = 6.022 \times 10^{23}$ is Avogadro's number, and **MOL. WT** and **DENSITY** are material parameters specified on the **MATERIAL** statement. The material is consumed ($V < 0$) if $R_f < R_r$.

MOBILITY

The **MOBILITY** statement is used to define or modify the dependence of carrier mobility on impurity concentration and temperature within a semiconductor.

MOBILITY

```
[ {TAB.TEMP=<n> [KELVIN]
  TAB.CONC=<c> TAB.E.MU=<c> TAB.H.MU=<c> [TAB.CLEA]} ]
[ECN.MU=<n>] [ECP.MU=<n>]
[GSURFN=<n>] [GSURFP=<n>]
[MUN1=<n>] [MUN2=<n>] [AN=<n>] [CN=<n>]
[EXN1=<n>] [EXN2=<n>] [EXN3=<n>] [EXN4=<n>]
[MUP1=<n>] [MUP2=<n>] [AP=<n>] [CP=<n>]
[EXP1=<n>] [EXP2=<n>] [EXP3=<n>] [EXP4=<n>]
[MUN.MIN=<n>] [MUN.MAX=<n>] [NREFN=<n>]
[NUN=<n>] [XIN=<n>] [ALPHAN=<n>]
[MUP.MIN=<n>] [MUP.MAX=<n>] [NREFP=<n>]
[NUP=<n>] [XIP=<n>] [ALPHAP=<n>]
```

| Parameter | Type | Definition |
|-----------------|-----------|--|
| TAB.TEMP | number | The temperature at which the mobility table values were measured. Units: Kelvins if the KELVIN parameter is specified, otherwise, degrees Celsius Default: none |
| KELVIN | logical | Specifies that the units of TAB.TEMP is Kelvins. Default: false |
| TAB.CONC | character | This parameter is interpreted as a series of numeric values, separated by spaces or commas. The parameter defines the concentrations associated with the mobilities defined in TAB.E.MU or TAB.H.MU . Units: atoms/cm ³ Default: none |
| TAB.E.MU | character | This parameter is interpreted as a series of numeric values, separated by spaces or commas. The parameter defines the electron mobilities associated with the concentrations defined in TAB.CONC . Units: cm ² /V/sec Default: none |
| TAB.H.MU | character | This parameter is interpreted as a series of numeric values, separated by spaces or commas. The parameter defines the hole mobilities associated with the concentrations defined in TAB.CONC . Units: cm ² /V/sec Default: none |
| TAB.CLEA | logical | Specifies that the current table is to be cleared. Default: false |

| Parameter | Type | Definition |
|---------------|--------|--|
| ECN.MU | number | The critical electric field used to calculate the electron mobility degradation by perpendicular electric field. Units: V/cm Default: 6.49e4 |
| ECP.MU | number | The critical electric field used to calculate the hole mobility degradation by perpendicular electric field. Units: V/cm Default: 1.87e4 |
| GSURFN | number | The low-field surface reduction factor for electron mobility. Units: none Default: 1.0 |
| GSURFP | number | The low-field surface reduction factor for hole mobility. Units: none Default: 1.0 |
| MUN1 | number | The minimum electron mobility used in Arora mobility model. Units: cm ² /V/sec Default: 88.0 |
| MUN2 | number | The maximum electron mobility used in Arora mobility model. Units: cm ² /V/sec Default: 1252.0 |
| AN | number | Parameter used in the exponent of normalized impurity concentration in the Arora mobility model for electrons. Units: none Default: 0.88 |
| CN | number | The reference impurity concentration used in the Arora mobility model for electrons. Units: atoms/cm ³ Default: 1.26e17 |
| EXN1 | number | Exponent of normalized temperature used in the Arora mobility model for electrons. Units: none Default: -0.57 |
| EXN2 | number | Exponent of normalized temperature used in the Arora mobility model for electrons. Units: none Default: -2.33 |
| EXN3 | number | Exponent of normalized temperature used in the Arora mobility model for electrons. Units: none Default: 2.4 |
| EXN4 | number | Exponent of normalized temperature used in the Arora mobility model for electrons. Units: none Default: -0.146 |

| Parameter | Type | Definition |
|------------------|--------|---|
| MUP1 | number | The minimum hole mobility used in Arora mobility model. Units: cm ² /V/sec Default: 54.3 |
| MUP2 | number | The maximum hole mobility used in Arora mobility model. Units: cm ² /V/sec Default: 407.0 |
| AP | number | Parameter used in the exponent of normalized impurity concentration in the Arora mobility model for holes. Units: none Default: 0.88 |
| CP | number | The reference impurity concentration used in the Arora mobility model for holes. Units: atoms/cm ³ Default: 2.35e17 |
| EXP1 | number | Exponent of normalized temperature used in the Arora mobility model for holes. Units: none Default: -0.57 |
| EXP2 | number | Exponent of normalized temperature used in the Arora mobility model for holes. Units: none Default: -2.23 |
| EXP3 | number | Exponent of normalized temperature used in the Arora mobility model for holes. Units: none Default: 2.4 |
| EXP4 | number | Exponent of normalized temperature used in the Arora mobility model for holes. Units: none Default: -0.146 |
| MUN . MIN | number | The minimum electron mobility in the Caughey mobility model. Units: cm ² /V/sec Default: 55.24 |
| MUN . MAX | number | The maximum electron mobility in the Caughey mobility model. Units: cm ² /V/sec Default: 1429.23 |
| NREFN | number | The reference impurity concentration used in the Caughey mobility model for electrons. Units: atoms/cm ³ Default: 1.072e17 |
| NUN | number | Exponent of normalized temperature used in the numerator of the Caughey mobility model for electrons. Units: none Default: -2.3 |
| XIN | number | Exponent of normalized temperature used in the denominator of the Caughey mobility model for electrons. Units: none Default: -3.8 |

| Parameter | Type | Definition |
|------------------|--------|--|
| ALPHAN | number | Exponent of the ratio of the total impurity concentration to NREFN used in the Caughey mobility model for electrons. Units: none Default: 0.733 |
| MUP . MIN | number | The minimum hole mobility in the Caughey mobility model. Units: cm ² /V/sec Default: 49.705 |
| MUP . MAX | number | The maximum hole mobility in the Caughey mobility model. Units: cm ² /V/sec Default: 479.37 |
| NREFP | number | The reference impurity concentration used in the Caughey mobility model for holes. Units: atoms/cm ³ Default: 1.606e17 |
| NUP | number | Exponent of normalized temperature used in the numerator of the Caughey mobility model for holes. Units: none Default: -2.2 |
| XIP | number | Exponent of normalized temperature used in the denominator of the Caughey mobility model for holes. Units: none Default: -3.7 |
| ALPHAP | number | Exponent of the ratio of the total impurity concentration to NREFN used in the Caughey mobility model for holes. Units: none Default: 0.70 |

Description

This statement is used to modify parameters associated with the various carrier mobility models which are available in the program.

Tables and Analytic Models

The dependencies of electron and hole mobilities on impurity concentration and temperature are represented by tables or analytic expressions.

Analytic Models

There are two analytic models:

- Arora's model
- Caughey's model

The analytic expressions are defined in [Chapter 2, "Carrier Mobility" on page 2-123](#). Since the default parameters of analytic models are based on the data mea-

sured on concentrations less than about $10^{20}/\text{cm}^3$, the analytic mobilities on heavily doped concentration might be not accurate.

Tables or Model Selection

The table is two-dimensional with at most 100 rows representing impurity concentrations and with columns representing temperatures. Interpolation is used to obtain values for impurity concentration and temperature between the values in the table. During interpolation, the impurity concentration is assumed to vary exponentially and the temperature and the mobility are assumed to vary linearly. If the number of table data is too few to interpolate, a warning message is displayed and the Arora analytic model is used. For example, by default, TSUPREM-4 only has tabulated data at 300K. Therefore, if another temperature is input in the **ELECTRICAL** statement, the Arora model is selected to calculate the mobility.

The new table values can be added to the current table and also the current values can be modified. If the **TAB.CLEA** parameter is specified, the current table values are totally replaced with the new values. Otherwise, the current table values for concentrations other than those defined **TAB.CONC** are retained.

Example

Consider the following table of hole mobilities with four rows of impurity concentrations at 300K, and suppose that these values need to be replaced with the current table values.

| Concentration (#/cm ³) | Hole Mobility (cm ² /V sec) |
|---------------------------------------|---|
| 1e20 | 49.9 |
| 2e20 | 45.6 |
| 5e20 | 35.1 |
| 1e21 | 24.9 |

Compare with the default table.

| Concentration (#/cm ³) | Hole Mobility (cm ² /V sec) | |
|---------------------------------------|---|------|
| | Default | New |
| 1e20 | 52.0 | 49.9 |
| 2e20 | 50.8 | 45.6 |
| 4e20 | 49.6 | none |
| 5e20 | none | 35.1 |
| 6e20 | 48.9 | none |
| 8e20 | 48.4 | none |
| 1e21 | 48.0 | 24.9 |

The following statement makes the modified table, on which some values are replaced with new ones.

```
MOBILITY  TAB.TEMP=300 KELVIN +
          TAB.CONC="1e20, 2e20, 5e20, 1e21" +
          TAB.H.MU="49.9, 45.6, 35.1, 24.9"
```

| Concentration (#/cm ³) | Hole Mobility (cm ² /V sec) |
|---------------------------------------|---|
| 1e20 | 49.9 |
| 2e20 | 45.6 |
| 4e20 | 49.6 |
| 5e20 | 35.1 |
| 6e20 | 48.9 |
| 8e20 | 48.4 |
| 1e21 | 24.9 |

However, since the mobilities associated with concentrations not represented in the **TAB.CONC** are left, the interpolation might result in undesirable mobilities. This can be avoided by the following statement.

```
MOBILITY  TAB.TEMP=300 KELVIN +
          TAB.CONC="1e20, 2e20, 4e20, 5e20, 6e20, 8e20, 1e21" +
          TAB.H.MU="49.9, 45.6, 0, 35.1, 0, 0, 24.9"
```

The mobilities set to zero are interpolated automatically, and then the interpolated values replace the default ones.

INTERSTITIAL

The **INTERSTITIAL** statement sets the coefficients for interstitial kinetics.

INTERSTITIAL

```
{MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
 | AMBIENT}[D.0=<n>] [D.E=<n>]
 [KB.0=<n>] [KB.E=<n>] [{KB.LOW | KB.MED | KB.HIGH}] [KIV.NORM]
 [CEQUIL.0=<n>] [CEQUIL.E=<n>] [CL.MODEL] [VMOLE=<n>]
 [NEU.0=<n>] [NEU.E=<n>] [NEG.0=<n>] [NEG.E=<n>] [DNEG.0=<n>]
 [DNEG.E=<n>] [POS.0=<n>] [POS.E=<n>] [DPOS.0=<n>] [DPOS.E=<n>]
 [ { ( C.STATE=<n> [C.V=<n>] ) | C.ALL }
 [DC.0=<n>] [DC.E=<n>] [FRAC.0=<n>] [FRAC.E=<n>]
 [KIV.0=<n>] [KIV.E=<n>] [KCV.0=<n>] [KCV.E=<n>]
 [ECLUST.0=<n>] [ECLUST.E=<n>] ]
 [TRAP.CON=<n>] [K.TRAP.0=<n>] [K.TRAP.E=<n>]
 [F.TRAP.0=<n>] [F.TRAP.E=<n>]
 [CL.KFI.0=<n>] [CL.KFI.E=<n>] [CL.IFI=<n>] [CL.ISFI=<n>]
 [CL.KFC.0=<n>] [CL.KFC.E=<n>] [CL.IFC=<n>] [CL.ISFC=<n>]
 [CL.KFCI=<n>] [CL.CF=<n>] [CL.KR.0=<n>] [CL.KR.E=<n>] [CL.CR=<n>]
 [ECLUST.N=<n>] [KLOOP.0=<n>] [KLOOP.E=<n>][RLMIN=<n>]
 [{/MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE | /POLYSIL
 | /AMBIENT}]
 [{V.MAXOX | V.INITOX | V.NORM}]
 [KSURF.0=<n>] [KSURF.E=<n>] [KSVEL.0=<n>] [KSVEL.E=<n>]
 [KSRAT.0=<n>] [KSRAT.E=<n>] [VNORM.0=<n>] [VNORM.E=<n>]
 [GROWTH] [THETA.0=<n>] [THETA.E=<n>]
 [A.0=<n>] [A.E=<n>] [T0.0=<n>] [T0.E=<n>]
 [KPOW.0=<n>] [KPOW.E=<n>] [GPOW.0=<n>] [GPOW.E=<n>]
 [N.MAX.0=<n>][N.MAX.E=<n>]
 [CM.SEC]
```

| Parameter | Type | Definition |
|-----------------|-----------|--|
| MATERIAL | character | The name of the material to which the other parameters apply. Default: none |
| SILICON | logical | The other parameters apply to silicon. Default: True if no other material specified. |
| OXIDE | logical | The other parameters apply to oxide. Default: false |
| OXYNITRI | logical | The other parameters apply to oxynitride. Default: false |
| NITRIDE | logical | The other parameters apply to nitride. Default: false |

| Parameter | Type | Definition |
|-------------------|---------|---|
| POLYSILI | logical | The other parameters apply to polysilicon. Default: false |
| AMBIENT | logical | The other parameters apply to the ambient gas. Default: false Synonym: GAS |
| D . 0 | number | The pre-exponential constant for the diffusivity of interstitials in the specified material. Only used with SILICON . Units: microns ² /min or cm ² /sec or none Default: the current value for this material Synonym: DI |
| D . E | number | The activation energy for the diffusivity of interstitials in the specified material. Only used with SILICON . Units: electron volts Default: the current value for this material |
| KB . 0 | number | The pre-exponential constant for bulk recombination rate in the specified material. Only used with SILICON . Units: microns ³ /min or cm ³ /sec Default: the current value for this material Synonym: KR . 0 |
| KB . E | number | The activation energy for bulk recombination rate in the specified material. Only used with SILICON . Units: electron volts Default: the current value for this material |
| KB . LOW | logical | Assume that only interstitials and vacancies having opposite charges of equal magnitude recombine in bulk material. Default: the current value for this material; initially true |
| KB . MED | logical | Assume that uncharged point defects can recombine with defects in any charge state, and that oppositely charged point defects can recombine. Default: the current value for this material; initially false |
| KB . HIGH | logical | Assume the charge state of a point defect doesn't affect its rate of recombination. Default: the current value for this material; initially false |
| KIV . NORM | logical | Normalize the charge state dependence in the point defect recombination model. Default: the current value for this material; initially false |
| CEQUIL . 0 | number | The pre-exponential constant for equilibrium concentration of interstitials in the specified material. Only used with SILICON . Units: interstitials/cm ³ Default: the current value for this material Synonym: CI |
| CEQUIL . E | number | The activation energy for equilibrium concentration of interstitials in the specified material. Only used with SILICON . Units: electron volts Default: the current value for this material |
| CL . MODEL | logical | Enables the interstitial clustering model for the specified material. Default: The current value; initially true for silicon and false for other materials. |

| Parameter | Type | Definition |
|-----------------|--------|--|
| VMOLE | number | The concentration of silicon atoms, used in the GROWTH model. Only used with SILICON . Units: atoms/cm ³ Default: the current value for this material |
| NEU . 0 | number | The pre-exponential constant for the relative concentration of neutral interstitials. Units: none Default: the current value for this material |
| NEU . E | number | The activation energy for the relative concentration of neutral interstitials. Units: electron volts Default: the current value for this material |
| NEG . 0 | number | The pre-exponential constant for the relative concentration of negatively charged interstitials. Units: none Default: the current value for this material |
| NEG . E | number | The activation energy for the relative concentration of negatively charged interstitials. Units: electron volts Default: the current value for this material |
| DNEG . 0 | number | The pre-exponential constant for the relative concentration of doubly negative charged interstitials. Units: none Default: the current value for this material |
| DNEG . E | number | The activation energy for the relative concentration of doubly negative charged interstitials. Units: electron volts Default: the current value for this material |
| POS . 0 | number | The pre-exponential constant for the relative concentration of positively charged interstitials. Units: none Default: the current value for this material |
| POS . E | number | The activation energy for the relative concentration of positively charged interstitials. Units: electron volts Default: the current value for this material |
| DPOS . 0 | number | The pre-exponential constant for the relative concentration of doubly positive charged interstitials. Units: none Default: the current value for this material |
| DPOS . E | number | The activation energy for the relative concentration of doubly positive charged interstitials. Units: electron volts Default: the current value for this material |

| Parameter | Type | Definition |
|----------------|--------|--|
| C.STATE | number | The interstitial charge state to which the DC.0 , DC.E , FRAC.0 , FRAC.E , KIV.0 , KIV.E , KCV.0 , KCV.E , ECLUST.0 , and ECLUST.E parameters apply. The value must be in the range of -6 to +6. Units: none Default: none Synonym: C.I |
| C.V | number | The vacancy charge state to which the KIV.0 , KIV.E , KCV.0 , and KCV.E parameters apply. The value must be in the range of -6 to +6. Units: none Default: none |
| C.ALL | number | Specifies that the DC.0 , DC.E , FRAC.0 , FRAC.E , KIV.0 , KIV.E , KCV.0 , KCV.E , ECLUST.0 , and ECLUST.E parameters should be set for all charge states. Default: false |
| DC.0 | number | The pre-exponential constant for diffusion of interstitials in the charge state(s) specified by C.STATE (or C.ALL) in the specified material. Only used with SILICON . Units: none or microns ² /min or cm ² /sec Default: the current value for this material; initially 1.0 for all charge states |
| DC.E | number | The activation energy for diffusion of interstitials in the charge state(s) specified by C.STATE (or C.ALL) in the specified material. Only used with SILICON . Units: electron volts Default: the current value for this material; initially 0.0 for all charge states |
| FRAC.0 | number | The pre-exponential constant for the relative concentration of interstitials in the charge state C.STATE (or C.ALL). Units: none Default: the current value for this material |
| FRAC.E | number | The activation energy for the relative concentration of interstitials in the charge state C.STATE (or C.ALL). Units: electron volts Default: the current value for this material |
| KIV.0 | number | The pre-exponential constant for the bulk recombination factor for interstitials in charge state C.I (C.STATE) and vacancies in charge state C.V (or for all combinations of charge states if C.ALL is set) in the specified material. Only used with SILICON . Units: none Default: the current value for this material |
| KIV.E | number | The activation energy for the bulk recombination factor for interstitials in charge state C.I (C.STATE) and vacancies in charge state C.V (or for all combinations of charge states if C.ALL is set) in the specified material. Only used with SILICON . Units: electron volts Default: the current value for this material |

| Parameter | Type | Definition |
|-----------------|--------|---|
| KCV.0 | number | The pre-exponential constant for the bulk recombination factor for small interstitial clusters in charge state C.I (C.STATE) and vacancies in charge state C.V (or for all combinations of charge states if C.ALL is set) in the specified material. Only used with SILICON . Units: none Default: the current value for this material |
| KCV.E | number | The activation energy for the bulk recombination factor for small interstitial clusters in charge state C.I (C.STATE) and vacancies in charge state C.V (or for all combinations of charge states if C.ALL is set) in the specified material. Only used with SILICON . Units: electron volts Default: the current value for this material |
| ECLUST.0 | number | The pre-exponential constant for the concentration of interstitials in small clusters in charge state C.STATE at thermal equilibrium in the specified material. Only used with SILICON . Units: interstitials/cm ³ Default: the current value for this material |
| ECLUST.E | number | The activation energy for the concentration of interstitials in small clusters in charge state C.STATE at thermal equilibrium in the specified material. Only used with SILICON . Units: electron volts Default: the current value for this material |
| TRAP.CON | number | The concentration of interstitial traps. Units: traps/cm ³ Default: the current value; initially 0.0 |
| K.TRAP.0 | number | The pre-exponential constant in the expression for the forward trap time constant. Units: cm ³ /sec Default: the current value; initially 0.0 |
| K.TRAP.E | number | The activation energy in the expression for the forward trap time constant. Units: electron volts Default: the current value; initially 0.0 |
| F.TRAP.0 | number | The pre-exponential constant in the expression for the fraction of empty interstitial traps at equilibrium. Units: none Default: the current value; initially 0.0 |
| F.TRAP.E | number | The activation energy in the expression for the fraction of empty interstitial traps at equilibrium. Units: electron volts Default: the current value; initially 0.0 |
| CL.KFI.0 | number | The prefactor for the $K_{fi} (I+I \rightarrow C)$ term in the interstitial clustering model. Units: cm ^{-(3*(1+CL.ISFI-CL.IFI))} /sec Default: the current value for this material |

| Parameter | Type | Definition |
|-----------|--------|---|
| CL.KFI.E | number | The activation energy for the K_{fi} ($I+I \rightarrow C$) term in the interstitial clustering model. Units: electron volts Default: the current value for this material |
| CL.IFI | number | The power of the interstitial concentration in the K_{fi} ($I+I \rightarrow C$) term in the interstitial clustering model. Units: none Default: the current value for this material |
| CL.ISFI | number | The power of the equilibrium interstitial concentration in the K_{fi} ($I+I \rightarrow C$) term in the interstitial clustering model. Units: none Default: the current value for this material |
| CL.KFC.O | number | The prefactor for the K_{fc} ($I+C \rightarrow C$) term in the interstitial clustering model. Units: $\text{cm}^{-3*(1-\text{CL.CF}+\text{CL.ISFC}-\text{CL.IFC})}/\text{sec}$ Default: the current value for this material |
| CL.KFC.E | number | The activation energy for the K_{fc} ($I+C \rightarrow C$) term in the interstitial clustering model. Units: electron volts Default: the current value for this material |
| CL.IFC | number | The power of the interstitial concentration in the K_{fc} ($I+C \rightarrow C$) term in the interstitial clustering model. Units: none Default: the current value for this material |
| CL.ISFC | number | The power of the equilibrium interstitial concentration in the K_{fc} ($I+C \rightarrow C$) term in the interstitial clustering model. Units: none Default: the current value for this material |
| CL.KFCI | number | The fraction of the interstitial concentration to be included in the K_{fc} ($I+C \rightarrow C$) term in the interstitial clustering model. Units: none Default: the current value for this material; initially 1.0 |
| CL.CF | number | The power of the clustered interstitial concentration in the K_{fc} ($I+C \rightarrow C$) term in the interstitial clustering model. Units: none Default: the current value for this material |
| CL.KR.O | number | The prefactor for the K_r (cluster dissolution) term in the interstitial clustering model. Units: $\text{cm}^{-3*(1-\text{CL.CR})}/\text{sec}$ Default: the current value for this material |
| CL.KR.E | number | The activation energy for the K_r (cluster dissolution) term in the interstitial clustering model. Units: electron volts Default: the current value for this material |

| Parameter | Type | Definition |
|-----------|-----------|--|
| CL.CR | number | The power of the clustered interstitial concentration in the K_r (cluster dissolution) term in the interstitial clustering model. Units: none Default: the current value for this material |
| ECLUST.N | number | The number of interstitials in each small cluster. Units: none Default: the current value for this material |
| KLOOP.O | number | The prefactor for K_L in the dislocation loop model. Units: none Default: the current value for this material |
| KLOOP.E | number | The activation energy for K_L in the dislocation loop model. Units: electron volts Default: the current value for this material |
| RLMIN | number | The minimum radius for a dislocation loop; loops smaller than this size are removed from the simulation. Units: cm Default: the current value for this material; initially 100e-8 cm |
| /MATERIA | character | The name of the second material for specifying interface injection and recombination parameters. Default: none |
| /SILICON | logical | The interface injection and recombination parameters apply to the interface between the specified material and silicon. Default: false |
| /OXIDE | logical | The interface injection and recombination parameters apply to the interface between the specified material and oxide. Default: false |
| /OXYNITR | logical | The interface injection and recombination parameters apply to the interface between the specified material and oxynitride. Default: false |
| /NITRIDE | logical | The interface injection and recombination parameters apply to the interface between the specified material and nitride. Default: false |
| /POLYSIL | logical | The interface injection and recombination parameters apply to the interface between the specified material and polysilicon. Default: false |
| /AMBIENT | logical | The interface injection and recombination parameters apply to the interface between the specified material and the ambient gas. Default: false Synonym: /GAS |
| V.MAXOX | logical | Use the injection/recombination model in which the interface velocity is normalized by the maximum interface velocity in the structure (the model used in older versions of TSUPREM-4). Default: the current value for these materials |

| Parameter | Type | Definition |
|-----------------|---------|---|
| V.INITOX | logical | Use the injection/recombination model in which the interface velocity is normalized by the initial growth velocity for a bare silicon surface. Default: the current value for these materials |
| V.NORM | logical | Use the injection/recombination model in which the interface velocity is normalized by the value specified by VNORM.0 and VNORM.E parameters. Default: the current value for these materials |
| KSURF.0 | number | The pre-exponential constant for the surface recombination velocity at the interface between the specified materials under inert conditions. Units: microns/min or cm/sec Default: the current value for these materials Synonyms: KSMIN.0 , KI.MIN |
| KSURF.E | number | The activation energy for the surface recombination velocity at the interface between the specified materials under inert conditions. Units: electron volts Default: the current value for these materials Synonym: KSMIN.E |
| KSVEL.0 | number | The pre-exponential constant for the growth-rate-dependent component of the surface recombination velocity at the interface between the specified materials. Used only with the V.MAXOX and V.NORM models. Units: microns/min or cm/sec Default: the current value for these materials Synonyms: KSMAX.0 , KI.MAX |
| KSVEL.E | number | The activation energy for the growth-rate-dependent component of the surface recombination velocity at the interface between the specified materials. Used only with the V.MAXOX and V.NORM models. Units: electron volts Default: the current value for these materials Synonym: KSMAX.E |
| KSRAT.0 | number | The pre-exponential constant for the ratio of the growth-rate-dependent component of the surface recombination velocity to the inert component at the interface between the specified materials. Used only with the V.INITOX model. Units: none Default: the current value for these materials |
| KSRAT.E | number | The activation energy for the ratio of the growth-rate-dependent component of the surface recombination velocity to the inert component at the interface between the specified materials. Used only with the V.INITOX model. Units: electron volts Default: the current value for these materials |
| VNORM.0 | number | The pre-exponential constant for the normalization velocity in the V.NORM model. Units: microns/min or cm/sec Default: the current value for these materials |
| VNORM.E | number | The activation energy for the normalization velocity in the V.NORM model. Units: electron volts Default: the current value for these materials |

| Parameter | Type | Definition |
|------------------|---------|---|
| GROWTH | logical | Specifies that interstitial injection should be calculated using the computed velocities of the interface rather than the analytic model. Default: the current value for these materials |
| THETA . 0 | number | The pre-exponential constant for the fraction of consumed silicon atoms that are injected into the bulk as interstitials during oxidation. Units: none, for V . INITOX and V . NORM models; (microns/min) ^(1-KPOW) or (cm/sec) ^(1-KPOW) for V . MAXOX model Default: the current value for these materials Synonym: THETA |
| THETA . E | number | The activation energy for the fraction of consumed silicon atoms that are injected into the bulk as interstitials during oxidation. Units: electron volts Default: the current value for these materials |
| A . 0 | number | The pre-exponential constant for the injection rate of interstitials at the interface between the specified materials. Units: #/micron ² /min ^(1+GPOW) or #/cm ² /sec ^(1+GPOW) for V . INITOX and V . NORM models; #/micron ² /min ^(1-KPOW) or #/cm ² /sec ^(1-KPOW) for V . MAXOX model Default: the current value for these materials Synonym: A |
| A . E | number | The activation energy for the injection rate of interstitials at the interface between the specified materials. Units: electron volts Default: the current value for these materials |
| T0 . 0 | number | The pre-exponential constant for the time constant for injection at the interface between the specified materials. Units: minutes or seconds Default: the current value for these materials Synonyms: T0 |
| T0 . E | number | The activation energy for the time constant for injection at the interface between the specified materials. Units: electron volts Default: the current value for these materials |
| KPOW . 0 | number | The pre-exponential constant in the expression for the exponent in the surface recombination models. (Also used for injection with the V . MAXOX model.) Units: none Default: the current value for these materials Synonyms: TPOW . 0 , POWER |
| KPOW . E | number | The activation energy in the expression for the exponent in the surface recombination models. (Also used for injection with the V . MAXOX model.) Units: electron volts Default: the current value for these materials Synonym: TPOW . E |

| Parameter | Type | Definition |
|----------------|---------|---|
| GPOW.O | number | The pre-exponential constant in the expression for the exponent in the V.INITOX and V.NORM models for interstitial injection. Units: none Default: the current value for these materials |
| GPOW.E | number | The activation energy in the expression for the exponent in the V.INITOX and V.NORM models for interstitial injection. Units: electron volts Default: the current value for these materials |
| N.MAX.O | number | The pre-exponential constant in the expression for the exponent in the maximum trapping site of nitrogen for interstitial recombination. Units: /cm ² Default: the current value for these materials |
| N.MAX.E | number | The activation energy in the expression for the exponent in the maximum trapping site of nitrogen for interstitial recombination. Units: electron volts Default: the current value for these materials |
| CM.SEC | logical | If true, parameters involving time are specified in centimeters and seconds; if false, parameters involving time are in microns and minutes. Default: false |

Description

This statement specifies values for coefficients of interstitial diffusion, recombination, injection, equilibrium concentration, and interstitial trap parameters. These parameters are normally specified in the *s4init* file (which is read when the program starts up) but can be changed by you at any time. Values that have *not* been set in *s4init* or by the user, default to 0.0.

Parameters whose units include time are specified in microns and minutes, unless **CM.SEC** is true, in which case units of centimeters and seconds are assumed.

The interpretation of the various parameters on the **INTERSTITIAL** statement is described in [Chapter 2, “Diffusion of Point Defects”](#) on page 2-32.

Bulk and Interface Parameters

If bulk parameters (e.g., **D.O** or **CEQUIL.E**) are specified but no material is given, the parameters are assumed to apply to silicon. If interface parameters (e.g., **KSVEL.O** or **A.E**) are specified and only a “first” material (e.g., **OXIDE**) is given, the parameters are assumed to apply to the interface between silicon and the specified material. This usage is not recommended, and is intended only for compatibility with older releases of TSUPREM-4.

Examples

1. The following statement specifies the silicon diffusivity and equilibrium values for interstitials:

```
INTERSTITIAL SILICON D.0=3000 CEQUIL.0=1.0e13
```

2. The following statement causes the Si/SiO₂ interface injection is to be computed using the oxide growth velocity, with 1% of consumed silicon injected as interstitials:

```
INTERSTITIAL SILICON /OXIDE GROWTH THETA.0=0.01
```

3. The following statement specifies that the surface recombination velocity in the silicon at the interface with nitride is 3.5×10^{-3} cm/s:

```
INTERSTITIAL SILICON /NITRIDE CM.SEC +
KSURF.0=3.5e-3 KSURF.E=0.0
```

4. The following statement specifies that the ratio of doubly-positive interstitials to neutral interstitials is 0.1 under intrinsic conditions:

```
INTERSTITIAL SILICON C.STATE=2 FRAC.0=0.1 +
FRAC.E=0.0
```

Note that the **C.STATE**, **FRAC.0**, and **FRAC.E** parameters can be used to specify the fraction of interstitials in any charge state from -6 to +6.

Additional INTERSTITIAL Notes

1. The model for interstitial traps has not been calibrated. Typical values of **TRAP.CON** are in the range 10^{15} to 10^{18} . The combination of **F.TRAP.0** and **F.TRAP.E** should give a value between 0.0 and 1.0.
2. Coefficients can be specified for each of the materials and interfaces, but at present only the coefficients for silicon and interfaces between silicon and other materials are used.
3. **D.0** and **DC.0** are used together to specify the diffusivity of interstitials in each charge state. One of these should have the units of cm²/sec or microns²/min while the other should be unitless. See [“Interstitial and Vacancy Diffusivities” on page 2-34.](#)

VACANCY

The **VACANCY** statement sets the coefficients for vacancy kinetics.

VACANCY

```
{MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
| AMBIENT}
[D.0=<n>] [D.E=<n>]
[CEQUIL.0=<n>] [CEQUIL.E=<n>] [VMOLE=<n>]
[NEU.0=<n>] [NEU.E=<n>] [NEG.0=<n>] [NEG.E=<n>] [DNEG.0=<n>]
[DNEG.E=<n>] [POS.0=<n>] [POS.E=<n>] [DPOS.0=<n>] [DPOS.E=<n>]
[ { ( [C.I=<n>] C.STATE=<n> ) | C.ALL }
[DC.0=<n>] [DC.E=<n>] [FRAC.0=<n>] [FRAC.E=<n>]
[KIC.0=<n>] [KIC.E=<n>] [ECLUST.0=<n>] [ECLUST.E=<n>] ]

[ECLUST.N=<n>]
[ { /MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE | /POLYSIL
| /AMBIENT } ]
[ { V.MAXOX | V.INITOX | V.NORM } ]
[KSURF.0=<n>] [KSURF.E=<n>] [KSVEL.0=<n>] [KSVEL.E=<n>]
[KSRAT.0=<n>] [KSRAT.E=<n>] [VNORM.0=<n>] [VNORM.E=<n>]
[GROWTH] [THETA.0=<n>] [THETA.E=<n>]
[A.0=<n>] [A.E=<n>] [T0.0=<n>] [T0.E=<n>]
[KPOW.0=<n>] [KPOW.E=<n>] [GPOW.0=<n>] [GPOW.E=<n>]
[CM.SEC]
```

| Parameter | Type | Definition |
|-----------------|-----------|--|
| MATERIAL | character | The name of the material to which the other parameters apply. Default: none |
| SILICON | logical | The other parameters apply to silicon. Default: true if no other material specified |
| OXIDE | logical | The other parameters apply to oxide. Default: false |
| OXYNITRI | logical | The other parameters apply to oxynitride. Default: false |
| NITRIDE | logical | The other parameters apply to nitride. Default: false |
| POLYSILI | logical | The other parameters apply to polysilicon. Default: false |
| AMBIENT | logical | The other parameters apply to the ambient gas. Default: false Synonym: GAS |

| Parameter | Type | Definition |
|-------------------|--------|---|
| D . 0 | number | The pre-exponential constant for the diffusivity of vacancies in the specified material. Only used with SILICON . Units: microns ² /min or cm ² /sec or none Default: the current value for this material Synonym: DI |
| D . E | number | The activation energy for the diffusivity of vacancies in the specified material. Only used with SILICON . Units: electron volts Default: the current value for this material |
| CEQUIL . 0 | number | The pre-exponential constant for equilibrium concentration of vacancies in the specified material. Only used with SILICON . Units: vacancies/cm ³ Default: the current value for this material Synonym: CV |
| CEQUIL . E | number | The activation energy for equilibrium concentration of vacancies in the specified material. Only used with SILICON . Units: electron volts Default: the current value for this material |
| VMOLE | number | The concentration of silicon atoms, used in the GROWTH model. Only used with SILICON . Units: atoms/cm ³ Default: the current value for this material |
| NEU . 0 | number | The pre-exponential constant for the relative concentration of neutral vacancies. Units: none Default: the current value for this material |
| NEU . E | number | The activation energy for the relative concentration of neutral vacancies. Units: electron volts Default: the current value for this material |
| NEG . 0 | number | The pre-exponential constant for the relative concentration of negatively charged vacancies. Units: none Default: the current value for this material |
| NEG . E | number | The activation energy for the relative concentration of negatively charged vacancies. Units: electron volts Default: the current value for this material |
| DNEG . 0 | number | The pre-exponential constant for the relative concentration of doubly negative charged vacancies. Units: none Default: the current value for this material |
| DNEG . E | number | The activation energy for the relative concentration of doubly negative charged vacancies. Units: electron volts Default: the current value for this material |

| Parameter | Type | Definition |
|------------------|--------|---|
| POS . 0 | number | The pre-exponential constant for the relative concentration of positively charged vacancies. Units: none Default: the current value for this material |
| POS . E | number | The activation energy for the relative concentration of positively charged vacancies. Units: electron volts Default: the current value for this material |
| DPOS . 0 | number | The pre-exponential constant for the relative concentration of doubly positive charged vacancies. Units: none Default: the current value for this material |
| DPOS . E | number | The activation energy for the relative concentration of doubly positive charged vacancies. Units: electron volts Default: the current value for this material |
| C . I | number | The interstitial charge state to which the KIC . 0 and KIC . E parameters apply. The value must be in the range of -6 to +6. Units: none Default: none |
| C . STATE | number | The vacancy charge state to which the DC . 0 , DC . E , FRAC . 0 , FRAC . E , KIC . 0 , KIC . E , ECLUST . 0 , and ECLUST . E parameters apply. The value must be in the range of -6 to +6. Units: none Default: none Synonym: C . V |
| C . ALL | number | Specifies that the DC . 0 , DC . E , FRAC . 0 , FRAC . E , KIC . 0 , KIC . E , ECLUST . 0 , and ECLUST . E parameters should be used for all charge states. Default: false |
| DC . 0 | number | The pre-exponential constant for diffusion of vacancies in the charge state(s) specified by C . STATE (or C . ALL) in the specified material. Only used with SILICON . Units: none or microns ² /min or cm ² /sec Default: the current value for this material; initially 1.0 for all charge states |
| DC . E | number | The activation energy for diffusion of vacancies in the charge state(s) specified by C . STATE (or C . ALL) in the specified material. Only used with SILICON . Units: electron volts Default: the current value for this material; initially 0.0 for all charge states |
| FRAC . 0 | number | The pre-exponential constant for the relative concentration of vacancies in the charge state C . STATE (or C . ALL). Units: none Default: the current value for this material |

| Parameter | Type | Definition |
|-----------------|-----------|--|
| FRAC.E | number | The activation energy for the relative concentration of vacancies in the charge state C.STATE (or C.ALL). Units: electron volts Default: the current value for this material |
| KIC.0 | number | The pre-exponential constant for the bulk recombination factor for interstitials in charge state C.I and small vacancy clusters in charge state C.V (C.STATE) (or for all combinations of charge states if C.ALL is set) in the specified material. Only used with SILICON . Units: none Default: the current value for this material |
| KIC.E | number | The activation energy for the bulk recombination factor for interstitials in charge state C.I and small vacancy clusters in charge state C.V (C.STATE) (or for all combinations of charge states if C.ALL is set) in the specified material. Only used with SILICON . Units: electron volts Default: the current value for this material |
| ECLUST.0 | number | The pre-exponential constant for the concentration of vacancies in small clusters in charge state C.STATE at thermal equilibrium in the specified material. Only used with SILICON . Units: vacancies/cm ³ Default: the current value for this material |
| ECLUST.E | number | The activation energy for the concentration of vacancies in small clusters in charge state C.STATE at thermal equilibrium in the specified material. Only used with SILICON . Units: electron volts Default: the current value for this material |
| ECLUST.N | number | The number of vacancies in each small cluster. Units: none Default: the current value for this material |
| /MATERIA | character | The name of the second material for specifying interface injection and recombination parameters. Default: none |
| /SILICON | logical | The interface injection and recombination parameters apply to the interface between the specified material and silicon. Default: false |
| /OXIDE | logical | The interface injection and recombination parameters apply to the interface between the specified material and oxide. Default: false |
| /OXYNITR | logical | The interface injection and recombination parameters apply to the interface between the specified material and oxynitride. Default: false |
| /NITRIDE | logical | The interface injection and recombination parameters apply to the interface between the specified material and nitride. Default: false |

| Parameter | Type | Definition |
|-----------------|---------|---|
| /POLYSIL | logical | The interface injection and recombination parameters apply to the interface between the specified material and polysilicon. Default: false |
| /AMBIENT | logical | The interface injection and recombination parameters apply to the interface between the specified material and the ambient gas. Default: false Synonym: /GAS |
| V.MAXOX | logical | Use the injection/recombination model in which the interface velocity is normalized by the maximum interface velocity in the structure (the model used in older versions of TSUPREM-4). Default: the current value for these materials |
| V.INITOX | logical | Use the injection/recombination model in which the interface velocity is normalized by the initial growth velocity for a bare silicon surface. Default: the current value for these materials |
| V.NORM | logical | Use the injection/recombination model in which the interface velocity is normalized by the value specified by VNORM.0 and VNORM.E parameters. Default: the current value for these materials |
| KSURF.0 | number | The pre-exponential constant for the surface recombination velocity at the interface between the specified materials under inert conditions. Units: microns/min or cm/sec Default: the current value for these materials Synonyms: KSMIN.0 , KV.MIN |
| KSURF.E | number | The activation energy for the surface recombination velocity at the interface between the specified materials under inert conditions. Units: electron volts Default: the current value for these materials Synonym: KSMIN.E |
| KSVEL.0 | number | The pre-exponential constant for the growth-rate-dependent component of the surface recombination velocity at the interface between the specified materials. Used only with the V.MAXOX and V.NORM models. Units: microns/min or cm/sec Default: the current value for these materials Synonyms: KSMAX.0 , KV.MAX |
| KSVEL.E | number | The activation energy for the growth-rate-dependent component of the surface recombination velocity at the interface between the specified materials. Used only with the V.MAXOX and V.NORM models. Units: electron volts Default: the current value for these materials Synonym: KSMAX.E |
| KSRAT.0 | number | The pre-exponential constant for the ratio of the growth-rate-dependent component of the surface recombination velocity to the inert component at the interface between the specified materials. Used only with the V.INITOX model. Units: none Default: the current value for these materials |

| Parameter | Type | Definition |
|----------------|---------|---|
| KSRAT.E | number | The activation energy for the ratio of the growth-rate-dependent component of the surface recombination velocity to the inert component at the interface between the specified materials. Used only with the V.INITOX model. Units: electron volts Default: the current value for these materials |
| VNORM.0 | number | The pre-exponential constant for the normalization velocity in the V.NORM model. Units: microns/min or cm/sec Default: the current value for these materials |
| VNORM.E | number | The activation energy for the normalization velocity in the V.NORM model. Units: electron volts Default: the current value for these materials |
| GROWTH | logical | Specifies that vacancy injection should be calculated using the computed velocities of the interface rather than the analytic model. Default: the current value for these materials |
| THETA.0 | number | The pre-exponential constant for the number of vacancies injected into the bulk per consumed silicon atom during oxidation. Units: none, for V.INITOX and V.NORM models; (microns/min) ^(1-KPOW) or (cm/sec) ^(1-KPOW) for V.MAXOX model Default: the current value for these materials Synonyms: THETA |
| THETA.E | number | The activation energy for the number of vacancies injected into the bulk per consumed silicon atom during oxidation. Units: electron volts Default: the current value for these materials |
| A.0 | number | The pre-exponential constant for the injection rate of vacancies at the interface between the specified materials. Units: #/micron ² /min ^(1+GPOW) or #/cm ² /sec ^(1+GPOW) for V.INITOX and V.NORM models; #/micron ² /min ^(1-KPOW) or #/cm ² /sec ^(1-KPOW) for V.MAXOX model Default: the current value for these materials Synonym: A |
| A.E | number | The activation energy for the injection rate of vacancies at the interface between the specified materials. Units: electron volts Default: the current value for these materials |
| T0.0 | number | The pre-exponential constant for the time constant for injection at the interface between the specified materials. Units: minutes or seconds Default: the current value for these materials Synonym: T0 |
| T0.E | number | The activation energy for the time constant for injection at the interface between the specified materials. Units: electron volts Default: the current value for these materials |

| Parameter | Type | Definition |
|---------------|---------|---|
| KPOW.0 | number | The pre-exponential constant in the expression for the exponent in the surface recombination models. (Also used for injection with the V.MAXOX model.) Units: none Default: the current value for these materials Synonyms: TPOW.0 , POWER |
| KPOW.E | number | The activation energy in the expression for the exponent in the surface recombination models. (Also used for injection with the V.MAXOX model.) Units: electron volts Default: the current value for these materials Synonym: TPOW.E |
| GPOW.0 | number | The pre-exponential constant in the expression for the exponent in the V.INITOX and V.NORM models for vacancy injection. Units: none Default: the current value for these materials |
| GPOW.E | number | The activation energy in the expression for the exponent in the V.INITOX and V.NORM models for vacancy injection. Units: electron volts Default: the current value for these materials |
| CM.SEC | logical | If true, parameters involving time are specified in centimeters and seconds; if false, parameters involving time are in microns and minutes. Default: false |

Description

This statement specifies values for coefficients of vacancy diffusion, recombination, injection, and equilibrium concentration. These parameters are normally specified in the *s4init* file (which is read when the program starts up) but can be changed by you at any time. Values that have *not* been set in *s4init* or by the user, default to 0.0.

Parameters whose units include time are specified in units of microns and minutes, unless **CM.SEC** is true, in which case units of centimeters and seconds are assumed.

The interpretation of the various parameters on the **VACANCY** statement is described in [Chapter 2, “Diffusion of Point Defects”](#) on page 2-32.

Bulk and Interface Parameters

If bulk parameters (e.g., **D.0** or **CEQUIL.E**) are specified but no material is given, the parameters are assumed to apply to silicon. If interface parameters (e.g., **KSVEL.0** or **A.E**) are specified and only a “first” material (e.g., **OXIDE**) is given, the parameters are assumed to apply to the interface between silicon and the specified material. This usage is not recommended, and is intended only for compatibility with older releases of TSUPREM-4.

Examples

1. The following statement specifies the silicon diffusivity and equilibrium values for vacancies:

```
VACANCY SILICON D.0=3000 CEQUIL.0=1.0e13
```

2. The following statement causes the Si/SiO₂ interface injection to be computed using the oxide growth velocity, with 1% of consumed silicon injected as vacancies:

```
VACANCY SILICON /OXIDE GROWTH THETA.0=0.01
```

3. The following statement specifies that the surface recombination velocity in the silicon at the interface with nitride is 3.5×10^{-3} cm/s:

```
VACANCY SILICON /NITRIDE CM.SEC +
KSURF.0=3.5e-3 KSURF.E=0.0
```

4. The following statement specifies that the ratio of doubly-positive vacancies to neutral vacancies is 0.1 under intrinsic conditions:

```
VACANCY SILICON C.STATE=2 FRAC.0=0.1 FRAC.E=0.0
```

The **C.STATE**, **FRAC.0**, and **FRAC.E** parameters can be used to specify the fraction of interstitials in any charge state from -6 to +6.

Additional VACANCY Notes

- Coefficients can be specified for each of the materials, but some coefficients are only used for silicon and others apply to interfaces between silicon and other materials.
- **D.0** and **DC.0** are used together to specify the diffusivity of vacancies in each charge state. One of these should have the units of cm²/sec or microns²/min while the other should be unitless. See [“Interstitial and Vacancy Diffusivities” on page 2-34](#).

ANTIMONY

The **ANTIMONY** statement sets some of the properties of antimony.

ANTIMONY

```
{MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
 | AMBIENT}
[DIX.0=<n>] [DIX.E=<n>] [DIM.0=<n>] [DIM.E=<n>]
[DVX.0=<n>] [DVX.E=<n>] [DVM.0=<n>] [DVM.E=<n>]
[DIPAIR.0=<n>] [DIPAIR.E=<n>] [DVPAIR.0=<n>] [DVPAIR.E=<n>]
[R.I.S=<n>] [E.I.S=<n>] [R.V.S=<n>] [E.V.S=<n>]
[R.IP.V=<n>] [E.IP.V=<n>] [R.VP.I=<n>] [E.VP.I=<n>]
[SS.CLEAR] [SS.TEMP=<n>] [SS.CONC=<n>]
[ {/MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE | /POLYSIL
 | /AMBIENT}
 [SEG.0=<n>] [SEG.E=<n>] [TRANS.0=<n>] [TRANS.E=<n>]
]
[ES.RAND=<n>] [ES.F.RAN=<n>]] [ES.100=<n>] [ES.F.100=<n>]]
[ES.110=<n>] [ES.F.110=<n>]]
[CM.SEC]
```

| Parameter | Type | Definition |
|-----------------|-----------|--|
| MATERIAL | character | The name of the material to which the other parameters apply (material 1 for the segregation terms). Default: none |
| SILICON | logical | Specifies that other parameters in this statement apply to antimony in silicon, and that silicon is material 1 for the segregation terms. Default: true if no other first material is specified. |
| OXIDE | logical | Specifies that other parameters in this statement apply to antimony in oxide, and that oxide is material 1 for the segregation terms. Default: false |
| OXYNITRI | logical | Specifies that other parameters in this statement apply to antimony in oxynitride, and that oxynitride is material 1 for the segregation terms. Default: false |
| NITRIDE | logical | Specifies that other parameters in this statement apply to antimony in nitride, and that nitride is material 1 for the segregation terms. Default: false |
| POLYSILI | logical | Specifies that other parameters in this statement apply to antimony in polysilicon, and that polysilicon is material 1 for the segregation terms. Default: false |
| AMBIENT | logical | Specifies that the ambient gas is material 1 for the segregation terms. Default: false Synonym: GAS |

| Parameter | Type | Definition |
|-----------------|--------|--|
| DIX.0 | number | The pre-exponential constant for diffusion of antimony with neutral interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIX.E | number | The activation energy for diffusion of antimony with neutral interstitials. Units: electron volts Default: the current value for this material |
| DIM.0 | number | The pre-exponential constant for diffusion of antimony with singly negative interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIM.E | number | The activation energy for diffusion of antimony with singly negative interstitials. Units: electron volts Default: the current value for this material |
| DVX.0 | number | The pre-exponential constant for diffusion of antimony with neutral vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVX.E | number | The activation energy for diffusion of antimony with neutral vacancies. Units: electron volts Default: the current value for this material |
| DVM.0 | number | The pre-exponential constant for diffusion of antimony with singly negative vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVM.E | number | The activation energy for diffusion of antimony with singly negative vacancies. Units: electron volts Default: the current value for this material |
| DIPAIR.0 | number | The pre-exponential constant for the diffusivity of antimony-interstitial pairs. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIPAIR.E | number | The activation energy for the diffusivity of antimony-interstitial pairs. Units: electron volts Default: the current value for this material |
| DVPAIR.0 | number | The pre-exponential constant for the diffusivity of antimony-vacancy pairs. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVPAIR.E | number | The activation energy for the diffusivity of antimony-vacancy pairs. Units: electron volts Default: the current value for this material |
| R.I.S | number | The capture radius for the reaction between interstitials and substitutional antimony atoms. Units: Å Default: the current value for this material |

| Parameter | Type | Definition |
|-------------------|-----------|---|
| E . I . S | number | The barrier energy for the reaction between interstitials and substitutional antimony atoms. Units: electron volts Default: the current value for this material |
| R . V . S | number | The capture radius for the reaction between vacancies and substitutional antimony atoms. Units: Å Default: the current value for this material |
| E . V . S | number | The barrier energy for the reaction between vacancies and substitutional antimony atoms. Units: electron volts Default: the current value for this material |
| R . IP . V | number | The capture radius for the reaction between antimony-interstitial pairs and vacancies. Units: Å Default: the current value for this material |
| E . IP . V | number | The barrier energy for the reaction between antimony-interstitial pairs and vacancies. Units: electron volts Default: the current value for this material |
| R . VP . I | number | The capture radius for the reaction between antimony-vacancy pairs and interstitials. Units: Å Default: the current value for this material |
| E . VP . I | number | The barrier energy for the reaction between antimony-vacancy pairs and interstitials. Units: electron volts Default: the current value for this material |
| SS . CLEAR | logical | Clears the solid solubility vs. temperature table. Default: false |
| SS . TEMP | number | The temperature at which the solid solubility in material 1 is SS . CONC . Units: degrees Celsius Default: none |
| SS . CONC | number | The solid solubility in material 1 at temperature SS . TEMP . Units: atoms/cm ³ Default: none |
| /MATERIA | character | The name of material 2 for the segregation parameters. Default: none |
| /SILICON | logical | Specifies that segregation parameters given on this statement apply to silicon as material 2. Default: false |
| /OXIDE | logical | Specifies that segregation parameters given on this statement apply to oxide as material 2. Default: True if no other second material is specified. |

| Parameter | Type | Definition |
|---------------------|---------|--|
| /OXYNITR | logical | Specifies that segregation parameters given on this statement apply to oxynitride as material 2. Default: true |
| /NITRIDE | logical | Specifies that segregation parameters given on this statement apply to nitride as material 2. Default: false |
| /POLYSIL | logical | Specifies that segregation parameters given on this statement apply to polysilicon as material 2. Default: false |
| /AMBIENT | logical | Specifies that segregation parameters given on this statement apply to the ambient gas as material 2. Default: false Synonym: /GAS |
| SEG . 0 | number | The pre-exponential factor for segregation from material 1 to material 2. Units: none Default: the current value for these materials |
| SEG . E | number | The activation energy for segregation from material 1 to material 2. Units: electron volts Default: the current value for these materials |
| TRANS . 0 | number | The pre-exponential factor for transport from material 1 to material 2. Units: microns/min or cm/sec Default: the current value for these materials Synonym: TRN . 0 |
| TRANS . E | number | The activation energy for transport from material 1 to material 2. Units: electron volts Default: the current value for these materials Synonym: TRN . E |
| ES . RAND | number | The electronic stopping power coefficient of implanted antimony in the specified material for materials other than silicon and for a nonchanneled direction in silicon. This value is used for the Monte Carlo ion implant calculation only. Units: $\text{angstrom}^2 * \text{eV}^{(1-\text{ES} . \text{F} . \text{RAN})}$ Default: the current value for antimony and the specified material |
| ES . F . RAN | number | The exponent of the electronic stopping power of implanted antimony in the specified material for materials other than silicon and for a nonchanneled direction in silicon. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for antimony and the specified material |
| ES . 100 | number | The electronic stopping power of antimony in silicon along the <100> channeling axes. This value is used for the Monte Carlo ion implant calculation only. Units: $\text{angstrom}^2 * \text{eV}^{(1-\text{ES} . \text{F} . 100)}$ Default: the current value for antimony and the specified material |

| Parameter | Type | Definition |
|-----------------|---------|--|
| ES.F.100 | number | The exponent of the electronic stopping power of antimony in silicon along the <100> crystal axes. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for antimony and the specified material |
| ES.110 | number | The electronic stopping power of antimony in silicon along the <110> channeling axes. This value is used for the Monte Carlo ion implant calculation only. Units: $\text{angstrom}^2 * eV^{(1-ES.F.100)}$ Default: the current value for antimony and the specified material |
| ES.F.110 | number | The exponent of the electronic stopping power of antimony in silicon along the <110> crystal axes. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for antimony and the specified material |
| CM.SEC | logical | If true, parameters involving time are specified in centimeters and seconds; if false, parameters involving time are in microns and minutes. Default: false |

Description

This statement specifies properties and model coefficients for antimony. The values of the diffusivity, reaction constant, solid solubility, and electronic stopping parameters apply in material 1 (specified without the "/"), while **SEG.0**, **SEG.E**, **TRANS.0**, and **TRANS.E** apply at the interface between material 1 and material 2 (specified with the "/"). These coefficients are normally given in the *s4init* file (which is read at the start of each TSUPREM-4 execution) but can be changed by you at any time. Coefficients that are *not* given in the *s4init* file or set by the user, default to 0.0, except for **SEG.0** that defaults to 1.0.

The newer **IMPURITY** statement can be used to set all of the properties of antimony, including some that cannot be set with the **ANTIMONY** statement.

Parameters whose units include time are specified in units of microns and minutes, unless **CM.SEC** is true, in which case units of centimeters and seconds are assumed.

For additional information see the following sections:

- The diffusion and segregation parameters are described in [Chapter 2, "Diffusion" on page 2-13](#).
- The electronic stopping power parameters are described in [Chapter 2, "Monte Carlo Ion Implant Model" on page 2-87](#).

Examples

1. The following statement specifies the diffusivity of antimony diffusing with neutral vacancies in silicon:

```
ANTIMONY SILICON DVX.0=1.22e9 DVX.E=3.65
```

2. The following statement specifies the segregation parameters at the Si/SiO₂ interface:

```
ANTIMONY SILICON /OXIDE SEG.0=30.0 TRANS.0=0.1
```

The concentration in silicon is 30.0 times the concentration in oxide, at equilibrium.

Additional ANTIMONY Notes

1. The **ANTIMONY** statement has been made obsolete by the **IMPURITY** statement, but remains available for compatibility with existing input files. Note that some properties of antimony can only be set on the **IMPURITY** statement.
2. The fractional interstitial parameter **FI** that was used in TSUPREM-4 prior to version 6.0 is no longer supported. Instead, it is now necessary to specify the diffusivities with interstitials and vacancies separately.

ARSENIC

The **ARSENIC** statement sets some of the properties of arsenic.

ARSENIC

```
{MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
 | AMBIENT}
[DIX.0=<n>] [DIX.E=<n>] [DIM.0=<n>] [DIM.E=<n>]
[DVX.0=<n>] [DVX.E=<n>] [DVM.0=<n>] [DVM.E=<n>]
[DIPAIR.0=<n>] [DIPAIR.E=<n>] [DVPAIR.0=<n>] [DVPAIR.E=<n>]
[R.I.S=<n>] [E.I.S=<n>] [R.V.S=<n>] [E.V.S=<n>]
[R.IP.V=<n>] [E.IP.V=<n>] [R.VP.I=<n>] [E.VP.I=<n>]
[CTN.0=<n>] [CTN.E=<n>] [CTN.F=<n>]
[ {/MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE | /POLYSIL
 | /AMBIENT}
 [SEG.0=<n>] [SEG.E=<n>] [TRANS.0=<n>] [TRANS.E=<n>]
]
[ES.RAND=<n> [ES.F.RAN=<n>]] [ES.100=<n> [ES.F.100=<n>]]
[ES.110=<n> [ES.F.110=<n>]]
[CM.SEC]
```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| MATERIAL | character | The name of the material to which the other parameters apply (material 1 for the segregation terms). Default: none |
| SILICON | logical | Specifies that other parameters in this statement apply to arsenic in silicon, and that silicon is material 1 for the segregation terms. Default: true if no other first material is specified. |
| OXIDE | logical | Specifies that other parameters in this statement apply to arsenic in oxide, and that oxide is material 1 for the segregation terms. Default: false |
| OXYNITRI | logical | Specifies that other parameters in this statement apply to arsenic in oxynitride, and that oxynitride is material 1 for the segregation terms. Default: false |
| NITRIDE | logical | Specifies that other parameters in this statement apply to arsenic in nitride, and that nitride is material 1 for the segregation terms. Default: false |
| POLYSILI | logical | Specifies that other parameters in this statement apply to arsenic in polysilicon, and that polysilicon is material 1 for the segregation terms. Default: false |
| AMBIENT | logical | Specifies that the ambient gas is material 1 for the segregation terms. Default: false Synonym: GAS |

| Parameter | Type | Definition |
|-----------------|--------|---|
| DIX.0 | number | The pre-exponential constant for diffusion of arsenic with neutral interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIX.E | number | The activation energy for diffusion of arsenic with neutral interstitials. Units: electron volts Default: the current value for this material |
| DIM.0 | number | The pre-exponential constant for diffusion of arsenic with singly negative interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIM.E | number | The activation energy for diffusion of arsenic with singly negative interstitials. Units: electron volts Default: the current value for this material |
| DVX.0 | number | The pre-exponential constant for diffusion of arsenic with neutral vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVX.E | number | The activation energy for diffusion of arsenic with neutral vacancies. Units: electron volts Default: the current value for this material |
| DVM.0 | number | The pre-exponential constant for diffusion of arsenic with singly negative vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVM.E | number | The activation energy for diffusion of arsenic with singly negative vacancies. Units: electron volts Default: the current value for this material |
| DIPAIR.0 | number | The pre-exponential constant for the diffusivity of arsenic-interstitial pairs. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIPAIR.E | number | The activation energy for the diffusivity of arsenic-interstitial pairs. Units: electron volts Default: the current value for this material |
| DVPAIR.0 | number | The pre-exponential constant for the diffusivity of arsenic-vacancy pairs. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVPAIR.E | number | The activation energy for the diffusivity of arsenic-vacancy pairs. Units: electron volts Default: the current value for this material |
| R.I.S | number | The capture radius for the reaction between interstitials and substitutional arsenic atoms. Units: Å Default: the current value for this material |

| Parameter | Type | Definition |
|-----------------|-----------|--|
| E.I.S | number | The barrier energy for the reaction between interstitials and substitutional arsenic atoms. Units: electron volts Default: the current value for this material |
| R.V.S | number | The capture radius for the reaction between vacancies and substitutional arsenic atoms. Units: Å Default: the current value for this material |
| E.V.S | number | The barrier energy for the reaction between vacancies and substitutional arsenic atoms. Units: electron volts Default: the current value for this material |
| R.IP.V | number | The capture radius for the reaction between arsenic-interstitial pairs and vacancies. Units: Å Default: the current value for this material |
| E.IP.V | number | The barrier energy for the reaction between arsenic-interstitial pairs and vacancies. Units: electron volts Default: the current value for this material |
| R.VP.I | number | The capture radius for the reaction between arsenic-vacancy pairs and interstitials. Units: Å Default: the current value for this material |
| E.VP.I | number | The barrier energy for the reaction between arsenic-vacancy pairs and interstitials. Units: electron volts Default: the current value for this material |
| CTN.O | number | The pre-exponential constant for clustering of arsenic. Units: (atoms/cm ³) ^(1/CTN.F-1) Default: the current value for this material |
| CTN.E | number | The activation energy for clustering of arsenic. Units: electron volts Default: the current value for this material |
| CTN.F | number | The exponent of concentration for clustering of arsenic. Units: none Default: the current value for this material |
| /MATERIA | character | The name of material 2 for the segregation parameters. Default: none |
| /SILICON | logical | Specifies that segregation parameters given on this statement apply to silicon as material 2. Default: false |

| Parameter | Type | Definition |
|---------------------|---------|--|
| /OXIDE | logical | Specifies that segregation parameters given on this statement apply to oxide as material 2. Default: True if no other second material is specified. |
| /OXYNITR | logical | Specifies that segregation parameters given on this statement apply to oxynitride as material 2. Default: false |
| /NITRIDE | logical | Specifies that segregation parameters given on this statement apply to nitride as material 2. Default: false |
| /POLYSIL | logical | Specifies that segregation parameters given on this statement apply to polysilicon as material 2. Default: false |
| /AMBIENT | logical | Specifies that segregation parameters given on this statement apply to the ambient gas as material 2. Default: false Synonyms: /GAS |
| SEG . 0 | number | The pre-exponential factor for segregation from material 1 to material 2. Units: none Default: the current value for these materials |
| SEG . E | number | The activation energy for segregation from material 1 to material 2. Units: electron volts Default: the current value for these materials |
| TRANS . 0 | number | The pre-exponential factor for transport from material 1 to material 2. Units: microns/min or cm/sec Default: the current value for these materials Synonym: TRN . 0 |
| TRANS . E | number | The activation energy for transport from material 1 to material 2. Units: electron volts Default: the current value for these materials Synonym: TRN . E |
| ES . RAND | number | The electronic stopping power coefficient of implanted arsenic in the specified material for materials other than silicon and for a nonchanneled direction in silicon. This value is used for the Monte Carlo ion implant calculation only. Units: $\text{angstrom}^2 * eV^{(1-ES.F.RAN)}$ Default: the current value for arsenic and the specified material |
| ES . F . RAN | number | The exponent of the electronic stopping power of implanted arsenic in the specified material for materials other than silicon and for a nonchanneled direction in silicon. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for arsenic and the specified material |
| ES . 100 | number | The electronic stopping power of arsenic in silicon along the <100> channeling axes. This value is used for the Monte Carlo ion implant calculation only. Units: $\text{angstrom}^2 * eV^{(1-ES.F.100)}$ Default: the current value for arsenic and the specified material |

| Parameter | Type | Definition |
|-----------------|---------|--|
| ES.F.100 | number | The exponent of the electronic stopping power of arsenic in silicon along the <100> crystal axes. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for arsenic and the specified material |
| ES.110 | number | The electronic stopping power of arsenic in silicon along the <110> channeling axes. This value is used for the Monte Carlo ion implant calculation only. Units: $\text{angstrom}^2 * eV^{(1-ES.F.100)}$ Default: the current value for arsenic and the specified material |
| ES.F.110 | number | The exponent of the electronic stopping power of arsenic in silicon along the <110> crystal axes. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for arsenic and the specified material |
| CM.SEC | logical | If true, parameters involving time are specified in centimeters and seconds; if false, parameters involving time are in microns and minutes. Default: false |

Description

This statement specifies properties and model coefficients for arsenic. The values of the diffusivity, reaction constant, clustering, and electronic stopping parameters apply in material 1 (specified without the "/"), while **SEG.0**, **SEG.E**, **TRANS.0** and **TRANS.E** apply at the interface between material 1 and material 2 (specified with the "/"). These coefficients are normally given in the *s4init* file (which is read at the start of each TSUPREM-4 execution), but can be changed by you at any time. Coefficients that are *not* given in the *s4init* file or set by the user, default to 0.0, except for **SEG.0** that defaults to 1.0.

The newer **IMPURITY** statement can be used to set all of the properties of arsenic, including some that cannot be set with the **ARSENIC** statement.

Parameters whose units include time are specified in units of microns and minutes, unless **CM.SEC** is true, in which case units of centimeters and seconds are assumed.

For additional information see the following sections:

- The diffusion and segregation parameters are described in [Chapter 2, "Diffusion" on page 2-13](#).
- The electronic stopping power parameters are described in [Chapter 2, "Monte Carlo Ion Implant Model" on page 2-87](#).

Examples

1. The following statement specifies the diffusivity of arsenic diffusing with negative vacancies in silicon:

```
ARSENIC SILICON DVM.0=1.49e11 DVM.E=4.15
```

2. The following statement specifies the segregation parameters at the Si/SiO₂ interface:

```
ARSENIC SILICON /OXIDE SEG.0=30.0 TRANS.0=0.1
```

The concentration in silicon is 30.0 times the concentration in oxide, at equilibrium.

Additional ARSENIC Notes

1. The **ARSENIC** statement has been obsoleted by the **IMPURITY** statement, but remains available for compatibility with existing input files. Note that some properties of arsenic can only be set on the **IMPURITY** statement.
2. The fractional interstitial parameter **FI** that was used in TSUPREM-4 prior to version 6.0 is no longer supported. Instead, it is now necessary to specify the diffusivities with interstitials and vacancies separately.

BORON

The **BORON** statement sets some of the properties of boron.

BORON

```
{MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
 | AMBIENT}
[DIX.0=<n>] [DIX.E=<n>] [DIP.0=<n>] [DIP.E=<n>]
[DVX.0=<n>] [DVX.E=<n>] [DVP.0=<n>] [DVP.E=<n>]
[DIPAIR.0=<n>] [DIPAIR.E=<n>] [DVPAIR.0=<n>] [DVPAIR.E=<n>]
[R.I.S=<n>] [E.I.S=<n>] [R.V.S=<n>] [E.V.S=<n>]
[R.IP.V=<n>] [E.IP.V=<n>] [R.VP.I=<n>] [E.VP.I=<n>]
[SS.CLEAR] [SS.TEMP=<n>] [SS.CONC=<n>]
[ {/MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE | /POLYSIL
 | /AMBIENT}
 [SEG.0=<n>] [SEG.E=<n>] [TRANS.0=<n>] [TRANS.E=<n>]
]
[ES.RAND=<n>] [ES.F.RAN=<n>]] [ES.100=<n>] [ES.F.100=<n>]]
[ES.110=<n>] [ES.F.110=<n>]]
[CM.SEC]
```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| MATERIAL | character | The name of the material to which the other parameters apply (material 1 for the segregation terms). Default: none |
| SILICON | logical | Specifies that other parameters in this statement apply to boron in silicon, and that silicon is material 1 for the segregation terms. Default: True if no other first material is specified. |
| OXIDE | logical | Specifies that other parameters in this statement apply to boron in oxide, and that oxide is material 1 for the segregation terms. Default: false |
| OXYNITRI | logical | Specifies that other parameters in this statement apply to boron in oxynitride, and that oxynitride is material 1 for the segregation terms. Default: false |
| NITRIDE | logical | Specifies that other parameters in this statement apply to boron in nitride, and that nitride is material 1 for the segregation terms. Default: false |
| POLYSILI | logical | Specifies that other parameters in this statement apply to boron in polysilicon, and that polysilicon is material 1 for the segregation terms. Default: false |
| AMBIENT | logical | Specifies that the ambient gas is material 1 for the segregation terms. Default: false Synonym: GAS |

| Parameter | Type | Definition |
|-----------|--------|---|
| DIX.0 | number | The pre-exponential constant for diffusion of boron with neutral interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIX.E | number | The activation energy for diffusion of boron with neutral interstitials. Units: electron volts Default: the current value for this material |
| DIP.0 | number | The pre-exponential constant for diffusion of boron with singly positive interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIP.E | number | The activation energy for diffusion of boron with singly positive interstitials. Units: electron volts Default: the current value for this material |
| DVX.0 | number | The pre-exponential constant for diffusion of boron with neutral vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVX.E | number | The activation energy for diffusion of boron with neutral vacancies. Units: electron volts Default: the current value for this material |
| DVP.0 | number | The pre-exponential constant for diffusion of boron with singly positive vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVP.E | number | The activation energy for diffusion of boron with singly positive vacancies. Units: electron volts Default: the current value for this material |
| DIPAIR.0 | number | The pre-exponential constant for the diffusivity of boron-interstitial pairs. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIPAIR.E | number | The activation energy for the diffusivity of boron-interstitial pairs. Units: electron volts Default: the current value for this material |
| DVPAIR.0 | number | The pre-exponential constant for the diffusivity of boron-vacancy pairs. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVPAIR.E | number | The activation energy for the diffusivity of boron-vacancy pairs. Units: electron volts Default: the current value for this material |
| R.I.S | number | The capture radius for the reaction between interstitials and substitutional boron atoms. Units: Å Default: the current value for this material |

| Parameter | Type | Definition |
|-----------------|-----------|--|
| E.I.S | number | The barrier energy for the reaction between interstitials and substitutional boron atoms. Units: electron volts Default: the current value for this material |
| R.V.S | number | The capture radius for the reaction between vacancies and substitutional boron atoms. Units: Å Default: the current value for this material |
| E.V.S | number | The barrier energy for the reaction between vacancies and substitutional boron atoms. Units: electron volts Default: the current value for this material |
| R.IP.V | number | The capture radius for the reaction between boron-interstitial pairs and vacancies. Units: Å Default: the current value for this material |
| E.IP.V | number | The barrier energy for the reaction between boron-interstitial pairs and vacancies. Units: electron volts Default: the current value for this material |
| R.VP.I | number | The capture radius for the reaction between boron-vacancy pairs and interstitials. Units: Å Default: the current value for this material |
| E.VP.I | number | The barrier energy for the reaction between boron-vacancy pairs and interstitials. Units: electron volts Default: the current value for this material |
| SS.CLEAR | logical | Clears the solid solubility vs. temperature table. Default: false |
| SS.TEMP | number | The temperature at which the solid solubility in material 1 is SS.CONC . Units: degrees Celsius Default: none |
| SS.CONC | number | The solid solubility in material 1 at temperature SS.TEMP . Units: atoms/cm ³ Default: none |
| /MATERIA | character | The name of material 2 for the segregation parameters. Default: none |
| /SILICON | logical | Specifies that segregation parameters given on this statement apply to silicon as material 2. Default: false |
| /OXIDE | logical | Specifies that segregation parameters given on this statement apply to oxide as material 2. Default: true if no other second material is specified |

| Parameter | Type | Definition |
|--------------|---------|--|
| /OXYNITR | logical | Specifies that segregation parameters given on this statement apply to oxynitride as material 2. Default: false |
| /NITRIDE | logical | Specifies that segregation parameters given on this statement apply to nitride as material 2. Default: false |
| /POLYSIL | logical | Specifies that segregation parameters given on this statement apply to polysilicon as material 2. Default: false |
| /AMBIENT | logical | Specifies that segregation parameters given on this statement apply to the ambient gas as material 2. Default: false Synonym: /GAS |
| SEG . 0 | number | The pre-exponential factor for segregation from material 1 to material 2. Units: none Default: the current value for these materials |
| SEG . E | number | The activation energy for segregation from material 1 to material 2. Units: electron volts Default: the current value for these materials |
| TRANS . 0 | number | The pre-exponential factor for transport from material 1 to material 2. Units: microns/min or cm/sec Default: the current value for these materials Synonym: TRN . 0 |
| TRANS . E | number | The activation energy for transport from material 1 to material 2. Units: electron volts Default: the current value for these materials Synonym: TRN . E |
| ES . RAND | number | The electronic stopping power coefficient of implanted boron in the specified material for materials other than silicon and for a nonchanneled direction in silicon. This value is used for the Monte Carlo ion implant calculation only. Units: $\text{angstrom}^2 * eV^{(1-ES.F.RAN)}$ Default: the current value for boron and the specified material |
| ES . F . RAN | number | The exponent of the electronic stopping power of implanted boron in the specified material for materials other than silicon and for a nonchanneled direction in silicon. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for boron and the specified material |
| ES . 100 | number | The electronic stopping power of boron in silicon along the <100> channeling axes. This value is used for the Monte Carlo ion implant calculation only. Units: $\text{angstrom}^2 * eV^{(1-ES.F.100)}$ Default: the current value for boron and the specified material |

| Parameter | Type | Definition |
|-----------------|---------|--|
| ES.F.100 | number | The exponent of the electronic stopping power of boron in silicon along the <100> crystal axes. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for boron and the specified material |
| ES.110 | number | The electronic stopping power of boron in silicon along the <110> channeling axes. This value is used for the Monte Carlo ion implant calculation only. Units: $\text{angstrom}^2 * eV^{(1-ES.F.100)}$ Default: the current value for boron and the specified material |
| ES.F.110 | number | The exponent of the electronic stopping power of boron in silicon along the <110> crystal axes. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for boron and the specified material |
| CM.SEC | logical | If true, parameters involving time are specified in centimeters and seconds; if false, parameters involving time are in microns and minutes. Default: false |

Description

This statement specifies properties and model coefficients for boron. The values of the diffusivity, reaction constant, solid solubility, and electronic stopping parameters apply in material 1 (specified without the "/"), while **SEG.0**, **SEG.E**, **TRANS.0**, and **TRANS.E** apply at the interface between material 1 and material 2 (specified with the "/"). These coefficients are normally given in the *s4init* file (which is read at the start of each TSUPREM-4 execution) but can be changed by you at any time. Coefficients that are *not* given in the *s4init* file or set by the user, default to 0.0, except for **SEG.0** that defaults to 1.0.

The newer **IMPURITY** statement can be used to set all of the properties of boron, including some that cannot be set with the **BORON** statement.

Parameters whose units include time are specified in units of microns and minutes, unless **CM.SEC** is true, in which case units of centimeters and seconds are assumed.

For additional information see the following sections:

- The diffusion and segregation parameters are described in [Chapter 2, "Diffusion" on page 2-13](#).
- The electronic stopping power parameters are described in [Chapter 2, "Monte Carlo Ion Implant Model" on page 2-87](#).

Examples

1. The following statement specifies the diffusivity of boron diffusing with neutral interstitials in silicon:

```
BORON SILICON DIX.0=2.09e8 DIX.E=3.46
```

2. The following statement specifies the segregation parameters at the Si/SiO₂ interface:

```
BORON SILICON /OXIDE SEG.0=0.91 TRANS.0=0.1
```

The concentration in silicon is 0.91 times the concentration in oxide, at equilibrium.

Additional BORON Notes

1. The **BORON** statement has been made obsolete by the **IMPURITY** statement, but remains available for compatibility with existing input files. Note that some properties of boron can only be set on the **IMPURITY** statement.
2. The fractional interstitial parameter **FI** that was used in TSUPREM-4 prior to version 6.0 is no longer supported. Instead, it is now necessary to specify the diffusivities with interstitials and vacancies separately.

PHOSPHORUS

The **PHOSPHORUS** statement sets some of the properties of phosphorus.

PHOSPHORUS

```
{MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
 | AMBIENT}
[DIX.0=<n>] [DIX.E=<n>] [DIM.0=<n>] [DIM.E=<n>]
[DIMM.0=<n>] [DIMM.E=<n>] [DVX.0=<n>] [DVX.E=<n>]
[DVM.0=<n>] [DVM.E=<n>] [DVMM.0=<n>] [DVMM.E=<n>]
[DIPAIR.0=<n>] [DIPAIR.E=<n>] [DVPAIR.0=<n>] [DVPAIR.E=<n>]
[R.I.S=<n>] [E.I.S=<n>] [R.V.S=<n>] [E.V.S=<n>]
[R.IP.V=<n>] [E.IP.V=<n>] [R.VP.I=<n>] [E.VP.I=<n>]
[SS.CLEAR] [SS.TEMP=<n>] [SS.CONC=<n>]
[ {/MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE | /POLYSIL
 | /AMBIENT}
 [SEG.0=<n>] [SEG.E=<n>] [TRANS.0=<n>] [TRANS.E=<n>]
]
[ES.RAND=<n>] [ES.F.RAN=<n>]] [ES.100=<n>] [ES.F.100=<n>]]
[ES.110=<n>] [ES.F.110=<n>]]
[CM.SEC]
```

| Parameter | Type | Definition |
|-----------------|-----------|---|
| MATERIAL | character | The name of the material to which the other parameters apply (material 1 for the segregation terms). Default: none |
| SILICON | logical | Specifies that other parameters in this statement apply to phosphorus in silicon, and that silicon is material 1 for the segregation terms. Default: True if no other first material is specified |
| OXIDE | logical | Specifies that other parameters in this statement apply to phosphorus in oxide, and that oxide is material 1 for the segregation terms. Default: false |
| OXYNITRI | logical | Specifies that other parameters in this statement apply to phosphorus in oxynitride, and that oxynitride is material 1 for the segregation terms. Default: false |
| NITRIDE | logical | Specifies that other parameters in this statement apply to phosphorus in nitride, and that nitride is material 1 for the segregation terms. Default: false |
| POLYSILI | logical | Specifies that other parameters in this statement apply to phosphorus in polysilicon, and that polysilicon is material 1 for the segregation terms. Default: false |
| AMBIENT | logical | Specifies that the ambient gas is material 1 for the segregation terms. Default: false Synonym: GAS |

| Parameter | Type | Definition |
|---------------|--------|--|
| DIX.0 | number | The pre-exponential constant for diffusion of phosphorus with neutral interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIX.E | number | The activation energy for diffusion of phosphorus with neutral interstitials. Units: electron volts Default: the current value for this material |
| DIM.0 | number | The pre-exponential constant for diffusion of phosphorus with singly negative interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIM.E | number | The activation energy for diffusion of phosphorus with singly negative interstitials. Units: electron volts Default: the current value for this material |
| DIMM.0 | number | The pre-exponential constant for diffusion of phosphorus with doubly negative interstitials. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIMM.E | number | The activation energy for diffusion of phosphorus with doubly negative interstitials. Units: electron volts Default: the current value for this material |
| DVX.0 | number | The pre-exponential constant for diffusion of phosphorus with neutral vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVX.E | number | The activation energy for diffusion of phosphorus with neutral vacancies. Units: electron volts Default: the current value for this material |
| DVM.0 | number | The pre-exponential constant for diffusion of phosphorus with singly negative vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVM.E | number | The activation energy for diffusion of phosphorus with singly negative vacancies. Units: electron volts Default: the current value for this material |
| DVMM.0 | number | The pre-exponential constant for diffusion of phosphorus with doubly negative vacancies. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVMM.E | number | The activation energy for diffusion of phosphorus with doubly negative vacancies. Units: electron volts Default: the current value for this material |

| Parameter | Type | Definition |
|-----------|--------|--|
| DIPAIR.O | number | The pre-exponential constant for the diffusivity of phosphorus-interstitial pairs. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DIPAIR.E | number | The activation energy for the diffusivity of phosphorus-interstitial pairs. Units: electron volts Default: the current value for this material |
| DVPAIR.O | number | The pre-exponential constant for the diffusivity of phosphorus-vacancy pairs. Units: microns ² /min or cm ² /sec Default: the current value for this material |
| DVPAIR.E | number | The activation energy for the diffusivity of phosphorus-vacancy pairs. Units: electron volts Default: the current value for this material |
| R.I.S | number | The capture radius for the reaction between interstitials and substitutional phosphorus atoms. Units: Å Default: the current value for this material |
| E.I.S | number | The barrier energy for the reaction between interstitials and substitutional phosphorus atoms. Units: electron volts Default: the current value for this material |
| R.V.S | number | The capture radius for the reaction between vacancies and substitutional phosphorus atoms. Units: Å Default: the current value for this material |
| E.V.S | number | The barrier energy for the reaction between vacancies and substitutional phosphorus atoms. Units: electron volts Default: the current value for this material |
| R.IP.V | number | The capture radius for the reaction between phosphorus-interstitial pairs and vacancies. Units: Å Default: the current value for this material |
| E.IP.V | number | The barrier energy for the reaction between phosphorus-interstitial pairs and vacancies. Units: electron volts Default: the current value for this material |
| R.VP.I | number | The capture radius for the reaction between phosphorus-vacancy pairs and interstitials. Units: Å Default: the current value for this material |
| E.VP.I | number | The barrier energy for the reaction between phosphorus-vacancy pairs and interstitials. Units: electron volts Default: the current value for this material |

| Parameter | Type | Definition |
|-------------------|-----------|--|
| SS . CLEAR | logical | Clears the solid solubility vs. temperature table. Default: false |
| SS . TEMP | number | The temperature at which the solid solubility in material 1 is SS . CONC . Units: degrees Celsius Default: none |
| SS . CONC | number | The solid solubility in material 1 at temperature SS . TEMP . Units: atoms/cm ³ Default: none |
| /MATERIA | character | The name of material 2 for the segregation parameters. Default: none |
| /SILICON | logical | Specifies that segregation parameters given on this statement apply to silicon as material 2. Default: false |
| /OXIDE | logical | Specifies that segregation parameters given on this statement apply to oxide as material 2. Default: false |
| /OXYNITR | logical | Specifies that segregation parameters given on this statement apply to oxynitride as material 2. Default: false |
| /NITRIDE | logical | Specifies that segregation parameters given on this statement apply to nitride as material 2. Default: false |
| /POLYSIL | logical | Specifies that segregation parameters given on this statement apply to polysilicon as material 2. Default: false |
| /AMBIENT | logical | Specifies that segregation parameters given on this statement apply to the ambient gas as material 2. Default: false Synonym: /GAS |
| SEG . 0 | number | The pre-exponential factor for segregation from material 1 to material 2. Units: none Default: the current value for these materials |
| SEG . E | number | The activation energy for segregation from material 1 to material 2. Units: electron volts Default: the current value for these materials |
| TRANS . 0 | number | The pre-exponential factor for transport from material 1 to material 2. Units: microns/min or cm/sec Default: the current value for these materials Synonym: TRN . 0 |
| TRANS . E | number | The activation energy for transport from material 1 to material 2. Units: electron volts Default: the current value for these materials Synonym: TRN . E |

| Parameter | Type | Definition |
|---------------------|---------|--|
| ES . RAND | number | The electronic stopping power coefficient of implanted phosphorus in the specified material for materials other than silicon and for a nonchanneled direction in silicon. This value is used for the Monte Carlo ion implant calculation only. Units: angstrom ² *eV ^(1-ES . F . RAN) Default: the current value for phosphorus and the specified material |
| ES . F . RAN | number | The exponent of the electronic stopping power of implanted phosphorus in the specified material for materials other than silicon and for a nonchanneled direction in silicon. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for phosphorus and the specified material |
| ES . 100 | number | The electronic stopping power of phosphorus in silicon along the <100> channeling axes. This value is used for the Monte Carlo ion implant calculation only. Units: angstrom ² *eV ^(1-ES . F . 100) Default: the current value for phosphorus and the specified material |
| ES . F . 100 | number | The exponent of the electronic stopping power of phosphorus in silicon along the <100> crystal axes. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for phosphorus and the specified material |
| ES . 110 | number | The electronic stopping power of phosphorus in silicon along the <110> channeling axes. This value is used for the Monte Carlo ion implant calculation only. Units: angstrom ² *eV ^(1-ES . F . 100) Default: the current value for phosphorus and the specified material |
| ES . F . 110 | number | The exponent of the electronic stopping power of phosphorus in silicon along the <110> crystal axes. This value is used for the Monte Carlo ion implant calculation only. Units: none Default: the current value for phosphorus and the specified material |
| CM . SEC | logical | If true, parameters involving time are specified in centimeters and seconds; if false, parameters involving time are in microns and minutes. Default: false |

Description

This statement specifies properties and model coefficients for phosphorus. The values of the diffusivity, reaction constant, solid solubility, and electronic stopping parameters apply in material 1 (specified without the "/"), while **SEG . 0**, **SEG . E**, **TRANS . 0**, and **TRANS . E** apply at the interface between material 1 and material 2 (specified with the "/"). These coefficients are normally given in the *s4init* file (which is read at the start of each TSUPREM-4 execution) but can be changed by you at any time. Coefficients that are *not* given in the *s4init* file or set by the user, default to 0.0, except for **SEG . 0** that defaults to 1.0.

The newer **IMPURITY** statement can be used to set all of the properties of phosphorus, including some that cannot be set with the **PHOSPHOR** statement.

Parameters whose units include time are specified in units of microns and minutes, unless **CM.SEC** is true, in which case units of centimeters and seconds are assumed.

For additional information see the following sections:

- Use of the diffusion and segregation parameters is described in [Chapter 2, “Diffusion” on page 2-13](#).
- The electronic stopping power parameters are described in [Chapter 2, “Monte Carlo Ion Implant Model” on page 2-87](#).

Examples

1. The following statement specifies the diffusion of phosphorus diffusing with doubly negative interstitials in silicon:

```
PHOSPHORUS SILICON DIMM.0=2.652e11 DIMM.E=4.37
```

2. The following statement specifies the segregation parameters at the Si/SiO₂ interface:

```
PHOS SILICON /OXIDE SEG.0=30.0 TRANS.0=0.1
```

The concentration in silicon is 30.0 times the concentration in oxide, at equilibrium.

Additional PHOSPHORUS Notes

1. The **PHOSPHORUS** statement has been made obsolete by the **IMPURITY** statement, but remains available for compatibility with existing input files. Note that some properties of phosphorus can only be set on the **IMPURITY** statement.
2. The fractional interstitial parameter **FI** that was used in TSUPREM-4 prior to version 6.0 is no longer supported. Instead, it is now necessary to specify the diffusivities with interstitials and vacancies separately.

3.6 Summary

This section summarizes the input statements recognized by TSUPREM-4. The format used for the parameter list associated with a statement is identical to that used in the detailed statement descriptions. The special characters < >, [], |, { }, and () are used to indicate parameter types, optional parameters, and valid parameter combinations. (For more information on the use of special characters see [“Syntax of Parameter Lists” on page 3-4.](#)) The summary is organized alphabetically by statement name and includes references to the page of the manual where a detailed description of the statement can be found.

AMBIENT Page 3-207

```

[ { DRYO2 | WETO2 | STEAM | N2O | INERT | AMB.1 | AMB.2 | AMB.3 | AMB.4
  | AMB.5}
  [F.O2=<n>] [F.H2O=<n>] [F.N2O=<n>] [F.H2=<n>] [F.N2=<n>]
  [F.HCL=<n>][PRESSURE=<n>] [HCL=<n>]
]
[ {O2 | H2O | N2O }
  [ {<111> | <110> | <100> | ORIENTAT=<n> | POLYSILI}
    [THINOX.0=<n>] [THINOX.E=<n>] [THINOX.L=<n>]
    [L.LIN.0=<n>] [L.LIN.E=<n>] [H.LIN.0=<n>] [H.LIN.E=<n>]
    [N.0=<n>] [N.E=<n>] [N.THIN.0=<n>] [N.THIN.E=<n>]
  ]
  [L.PAR.0=<n>] [L.PAR.E=<n>] [H.PAR.0=<n>] [H.PAR.E=<n>]
  [LIN.BREA=<n>] [PAR.BREA=<n>] [LIN.PDEP=<n>] [PAR.PDEP=<n>]
  [GAMMA.0=<n>] [GAMMA.E=<n>]
  [ { LIN.PCT | PAR.PCT
    | ( {LIN.CLDE | PAR.CLDE} COLUMN=<n> )
  }
  ]
  TABLE=<c>
]
[ { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
  | AMBIENT}
  [D.0=<n>] [D.E=<n>] [VC=<c>]
  [HENRY.CO=<n>] [THETA=<n>]
  [ { /MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR
    | /NITRIDE | /POLYSIL | /AMBIENT
  }
  ]
  [SEG.0=<n>] [SEG.E=<n>] [TRANS.0=<n>] [TRANS.E=<n>]
  [ALPHA=<n>]
  [N.G.0=<n>] [N.G.E=<n>] [N.GPOW.0=<n>] [N.GPOW.E=<n>]
  [N.V.0=<n>] [N.V.E=<n>]
]
]
]
[STRESS.D] [VR=<c>] [VT=<c>] [VD=<c>] [VDLIM=<n>]
[INITIAL=<n>] [SPREAD=<n>] [MASK.EDG=<n>]
[ERF.Q=<n>] [ERF.DELT=<n>] [ERF.LBB=<c>] [ERF.H=<c>]
[NIT.THIC=<n>]
[CLEAR]
[TEMPERAT=<c>]
[CM.SEC]

```

ANTIMONY Page 3-284

```
{MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
| AMBIENT}
[DIX.0=<n>] [DIX.E=<n>] [DIM.0=<n>] [DIM.E=<n>]
[DVX.0=<n>] [DVX.E=<n>] [DVM.0=<n>] [DVM.E=<n>]
[DIPAIR.0=<n>] [DIPAIR.E=<n>] [DVPAIR.0=<n>] [DVPAIR.E=<n>]
[R.I.S=<n>] [E.I.S=<n>] [R.V.S=<n>] [E.V.S=<n>]
[R.IP.V=<n>] [E.IP.V=<n>] [R.VP.I=<n>] [E.VP.I=<n>]
[SS.CLEAR] [SS.TEMP=<n>] [SS.CONC=<n>]
[ {/MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE | /POLYSIL
| /AMBIENT}
[SEG.0=<n>] [SEG.E=<n>] [TRANS.0=<n>] [TRANS.E=<n>]
]
[ES.RAND=<n> [ES.F.RAN=<n>]] [ES.100=<n> [ES.F.100=<n>]]
[ES.110=<n> [ES.F.110=<n>]]
[CM.SEC]
```

ARSENIC Page 3-290

```
{MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
| AMBIENT}
[DIX.0=<n>] [DIX.E=<n>] [DIM.0=<n>] [DIM.E=<n>]
[DVX.0=<n>] [DVX.E=<n>] [DVM.0=<n>] [DVM.E=<n>]
[DIPAIR.0=<n>] [DIPAIR.E=<n>] [DVPAIR.0=<n>] [DVPAIR.E=<n>]
[R.I.S=<n>] [E.I.S=<n>] [R.V.S=<n>] [E.V.S=<n>]
[R.IP.V=<n>] [E.IP.V=<n>] [R.VP.I=<n>] [E.VP.I=<n>]
[CTN.0=<n>] [CTN.E=<n>] [CTN.F=<n>]
[ {/MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE | /POLYSIL
| /AMBIENT}
[SEG.0=<n>] [SEG.E=<n>] [TRANS.0=<n>] [TRANS.E=<n>]
]
[ES.RAND=<n> [ES.F.RAN=<n>]] [ES.100=<n> [ES.F.100=<n>]]
[ES.110=<n> [ES.F.110=<n>]]
[CM.SEC]
```

ASSIGN Page 3-25

```
{ ( NAME=<c> [PRINT] [DELETE] [PROMPT=<c>
[ { ( {N.EXPRES=<n> | N.VALUE=<c>}
[ { DELTA=<n> | RATIO=<n> | (LOWER=<n> UPPER=<n> [LOG]) } ] )
| C.VALUE=<c>
| ( C.FILE=<c> [LINE=<n>] )
| ( [C1=<c>] [C2=<c>] [C3=<c>] [C4=<c>] [C5=<c>]
[C6=<c>] [C7=<c>] [C8=<c>] [C9=<c>] [C10=<c>] )
}]
| ( ARRAY=<c>
{ ( IN.FILE=<c> DATA=<c> [{TIF | ROW | COLUMN}] )
| IN.NVALU=<c> | IN.CVALU=<c> }
[C.COUNT=<c>] )
}
```

BORON [Page 3-296](#)

```

{MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
| AMBIENT}
[DIX.0=<n>] [DIX.E=<n>] [DIP.0=<n>] [DIP.E=<n>]
[DVX.0=<n>] [DVX.E=<n>] [DVP.0=<n>] [DVP.E=<n>]
[DIPAIR.0=<n>] [DIPAIR.E=<n>] [DVPAIR.0=<n>] [DVPAIR.E=<n>]
[R.I.S=<n>] [E.I.S=<n>] [R.V.S=<n>] [E.V.S=<n>]
[R.IP.V=<n>] [E.IP.V=<n>] [R.VP.I=<n>] [E.VP.I=<n>]
[SS.CLEAR] [SS.TEMP=<n>] [SS.CONC=<n>]
[ {/MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE | /POLYSIL
| /AMBIENT}
[SEG.0=<n>] [SEG.E=<n>] [TRANS.0=<n>] [TRANS.E=<n>]
]
[ES.RAND=<n>] [ES.F.RAN=<n>]] [ES.100=<n>] [ES.F.100=<n>]]
[ES.110=<n>] [ES.F.110=<n>]]
[CM.SEC]

```

BOUNDARY [Page 3-60](#)

```

{REFLECTI | EXPOSED}
XLO=<c> XHI=<c> YLO=<c> YHI=<c>

```

COLOR [Page 3-150](#)

```

[COLOR=<n>] [MIN.VALU=<n>] [MAX.VALU=<n>]
[ { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
| ALUMINUM | PHOTORES
}
]

```

COMMENT [Page 3-8](#)

```

[<c>]
or
$
[<c>]

```

CONTOUR [Page 3-148](#)

```

VALUE=<n> [LINE.TYP=<n>] [COLOR=<n>] [SYMBOL=<n>]

```

CPULOG [Page 3-46](#)

[LOG] [OUT.FILE=<c>]

DEFINE [Page 3-42](#)

[<name> <body>]

DEPOSITION [Page 3-90](#)

```
{ MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE
  | POLYSILI | ALUMINUM
  | ( PHOTORES [ { POSITIVE | NEGATIVE } ] )
}
[ IMPURITY=<c> { I.CONC=<n> | I.RESIST=<n> } ]
[ANTIMONY=<n>] [ARSENIC=<n>] [BORON=<n>] [PHOSPHOR=<n>]
[ {CONCENTR | RESISTIV} ]
THICKNES=<n> [SPACES=<n>] [DY=<n>] [YDY=<n>] [ARC.SPAC=<n>]
[TEMPERAT=<n>] [GSZ.LIN]
TOPOGRAP=<c>
```

DEVELOP [Page 3-97](#)

[<c>]

DIFFUSION [Page 3-115](#)

```
TIME=<n> [CONTINUE]
TEMPERAT=<n> [ {T.RATE=<n> | T.FINAL=<n>} ]
[ { DRYO2 | WETO2 | STEAM | INERT
  | AMB.1 | AMB.2 | AMB.3 | AMB.4 | AMB.5
  | ( [F.O2=<n>] [F.H2O=<n>] [F.H2=<n>] [F.N2=<n>] [F.HCL=<n>] )
}
]
[IMPURITY=<c> I.CONC=<n>]
[ANTIMONY=<n>] [ARSENIC=<n>] [BORON=<n>] [PHOSPHOR=<n>]
[PRESSURE=<n>] [ {P.RATE=<n> | P.FINAL=<n>} ] [HCL=<n>]
[D.RECOMB=<n>] [MOVIE=<c>] [DUMP=<n>]
```

ECHO [Page 3-38](#)

<string>

ELECTRICAL Page 3-174

```

[X=<n>]
[ { ( SRP
    [ANGLE=<n>] [PITCH=<n>] [ {POINT=<n> | DEPTH=<n>} ]
    [Y.SURFAC=<n>]
  )
  | ( {V=<c> | (VSTART=<n> VSTOP=<n> VSTEP=<n>)}
    { ( RESISTAN [EXT.REG=<n>] [BIAS.REG=<n>] )
      | ( JCAP [JUNCTION=<n>] )
      | ( { ( MOSCAP [HIGH] [LOW] [DEEP] )
          | ( THRESHOL [VB=<n>] )
          }
        {NMOS | PMOS} [QM]
        [QSS=<n>] [GATE.WF=<n>] [GATE.ELE]
        [BULK.REG=<n>] )
    }
    [BULK.LAY=<n>] [PRINT] [DISTRIB]
  )
}
]
[TEMPERAT=<n>]
[OUT.FILE=<c>]
[NAME=<c> [V.SELECT=<n>]
 { TARGET=<n> [SENSITIV]
 | T.FILE=<c> [V.COLUMN=<n>] [V.LOWER=<n>] [V.UPPER=<n>]
   [T.COLUMN=<n>] [T.LOWER=<n>][T.UPPER=<n>]
   [V.TRANSF=<c>] [T.TRANSF=<c>] )
 [Z.VALUE]
}
[TOLERANC=<n>] [WEIGHT=<n>] [MIN.REL=<n>][MIN.ABS=<n>]
]

```

ELECTRODE Page 3-87

```

[NAME=<c>] [ { ( X=<n> [Y=<n>] ) | BOTTOM } ]
 [CLEAR [ALL]]
 [MERGE]
 [PRINT]

```

ELIMINATE Page 3-57

```

{ROWS | COLUMNS}
[X.MIN=<n>] [X.MAX=<n>] [Y.MIN=<n>] [Y.MAX=<n>]

```

EPITAXY Page 3-121

```

TIME=<n> TEMPERAT=<n> [ {T.RATE=<n> | T.FINAL=<n>} ]
[ IMPURITY=<c> {I.CONC=<n> | I.RESIST=<n>} ]
[ ANTIMONY=<n> ] [ ARSENIC=<n> ] [ BORON=<n> ] [ PHOSPHOR=<n> ]
[ {CONCENTR | RESISTIV} ]
THICKNES=<n> [ SPACES=<n> ] [ DY=<n> ] [ YDY=<n> ] [ ARC.SPAC=<n> ]
    
```

EQUATION Page 3-204

EQUATION

```

VARIABLE=<c> MATERIAL=<c> [ MODEL=<c> ] [ TIF.NAME=<c> ]
[ ADDTOEXP=<c> ] [ INITIAL=<c> ] [ UPDATE=<c> ]
[ /MATERIA=<c> INTERFAC=<c> [ BOUNDARY ] [ TRANSPOR ] ]
    
```

ETCH Page 3-98

```

[ { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE
  | POLYSILI | PHOTORES | ALUMINUM
  }
]
{ ( TRAPEZOI [THICKNES=<n>] [ANGLE=<n>] [UNDERCUT=<n>] )
  | ( {LEFT | RIGHT} [P1.X=<n>] [P1.Y=<n>] [P2.X=<n>] [P2.Y=<n>] )
  | ( {START | CONTINUE | DONE} X=<n> Y=<n> )
  | ISOTROPI
  | ( OLD.DRY THICKNES=<n> )
  | ALL
  | TOPOGRAP=<c>
}
    
```

EXPOSE Page 3-95

```

MASK=<c> [ SHRINK=<n> ] [ OFFSET=<n> ]
    
```

EXTRACT Page 3-160

```

[ { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI |
  PHOTORES | ALUMINUM } ]
[P1.X=<n>] [P1.Y=<n>] [P2.X=<n>] [P2.Y=<n>]
[ { /MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE |
  /POLYSIL | /PHOTORE | /ALUMINU | /AMBIENT } [CLOCKWIS] ]
[X=<n>] [Y=<n>] [ { DISTANCE=<n> | MINIMUM | MAXIMUM | VALUE=<n> } ]
{ ( [X.EXTRAC] [Y.EXTRAC] [D.EXTRAC] [VAL.EXTR] )
  | ( [INT.EXTR] [AREA.EXT] [AVG.EXTR] ) }
[PREFIX=<c>] [SEPARAT=<c>] [SUFFIX=<c>]
[WRITE] [PRINT]
[ NAME=<c> [ASSIGN]
  [ { TARGET=<n>
    | ( T.FILE=<c> [V.COLUMN=<n>] [V.LOWER=<n>] [V.UPPER=<n>]
      [T.COLUMN=<n>] [T.LOWER=<n>] [T.UPPER=<n>]
      [V.TRANSF=<c>] [T.TRANSF=<c>]
      [Z.VALUE=<c>] [SENSITIV])
    }
  [TOLERANC=<n>] [WEIGHT=<n>]
  [MIN.REL=<n>] [MIN.ABS=<n>]
  ]
]
[ OUT.FILE=<c> [APPEND] ] [CLOSE]

```

FOREACH/END Page 3-16

```
<name> <list>
```

HELP Page 3-47

```
[<name>]
```

IF/ELSEIF/ELSE/IF.END Page 3-23

```

IF ( condition )
[ ELSEIF ( condition ) ]
[ ELSE ]
IF.END

```


IMPLANT Page 3-103

```

DOSE=<n> ENERGY=<n> [TILT=<n>] [ROTATION=<n>]
{IMPURITY=<c> | ANTIMONY | ARSENIC | BORON | BF2 | PHOSPHOR}
{ ( [ {GAUSSIAN | PEARSON} ] [RP.EFF] [IN.FILE=<c>]
  [IMPL.TAB=<c>] [MOMENTS] [BACKSCAT]
  )
  | ( MONTECAR [N.ION=<n>] [BEAMWIDT=<n>] [SEED=<n>]
      [CRYSTAL [TEMPERAT=<n>] [VIBRATIO [X.RMS=<n>] [E.LIMIT=<n>] ]
      [THRESHOL=<n>] [REC.FRAC=<n>] [CRIT.PRE=<n>]
      [CRIT.F=<n>] [CRIT.110=<n>]
      ]
    [ {PERIODIC | REFLECT | VACUUM} ]
  )
}

[POLY.GSZ=<n>] [INTERST=<c>]
[DAMAGE] [ D.PLUS=<n> | D.P1=<n> ] [D.SCALE=<n>]
[MAX.DAMA=<n>] [D.RECOMB]
[L.RADIUS=<n> { (L.DENS=<n> [L.DMIN=<n>] [L.DMAX=<n>])
               | (L.THRESH=<n> [L.FRAC=<n>]) } ]
[PRINT]

```

IMPURITY Page 3-239

```

IMPURITY=<c> [( NEW [MODEL=<c>] [C.INIT=<n>] ) ] [TIF.NAME=<c>]
[IMP.ACT=<c>] [IMP.GB=<c>][IMP.IT=<c>]
[ { DONOR | ACCEPTOR } ] [AT.NUM=<n>] [AT.WT=<n>]
[SOLVE] [STEADY] [IMPL.TAB=<c>]
[ MATERIAL=<c>
  [DIP.0=<n>] [DIP.E=<n>] [DIX.0=<n>] [DIX.E=<n>]
  [DIM.0=<n>] [DIM.E=<n>] [DIMM.0=<n>] [DIMM.E=<n>]
  [DVP.0=<n>] [DVP.E=<n>] [DVX.0=<n>] [DVX.E=<n>]
  [DVM.0=<n>] [DVM.E=<n>] [DVMM.0=<n>] [DVMM.E=<n>]
  [C.STATE=<n>] [DIC.0=<n>] [DIC.E=<n>] [DVC.0=<n>] [DVC.E=<n>] ]
  [D.MODEL=<c>] [DI.FAC=<c>] [DV.FAC=<c>]
  [DIFFUSE]
  [FGB=<n>]
  [DIPAIR.0=<n>] [DIPAIR.E=<n>] [DVPAIR.0=<n>] [DVPAIR.E=<n>]
  [R.I.S=<n>] [E.I.S=<n>] [R.V.S=<n>] [E.V.S=<n>]
  [R.IP.V=<n>] [E.IP.V=<n>] [R.VP.I=<n>] [E.VP.I=<n>]
  [SS.CLEAR] [SS.TEMP=<n>] [SS.CONC=<n>]
  [CTN.0=<n>] [CTN.E=<n>] [CTN.F=<n>] [CL.INI.A]
  [DDC.F.0=<n>] [DDC.F.E=<n>] [DDC.T.0=<n>] [DDC.T.E=<n>]
  [DDCF.D.N=<n>] [DDCF.N.N=<n>] [DDCF.I.N=<n>]
  [DDCR.N.N=<n>] [DDCR.I.N=<n>] [IFRACM=<n>]
  [DDCS.0=<n>] [DDCS.E=<n>] [DDCS.N=<n>] [IFRACS=<n>]
  [Q.SITES=<n>] [CG.MAX=<n>] [GSEG.0=<n>] [GSEG.E=<n>]
  [GSEG.INI=<n>] [VELIF.0=<n>] [VELIF.E=<n>]
  [ /MATERIA=<c>
    [SEG.0=<n>] [SEG.E=<n>] [TRANS.0=<n>] [TRANS.E=<n>]
    [RATIO.0=<n>] [RATIO.E=<n>] [SEG.SS]
    [/SEG.0=<n>] [/SEG.E=<n>] [/TRANS.0=<n>] [/TRANS.E=<n>]
    [/RATIO.0=<n>] [/RATIO.E=<n>] [/SEG.SS]
    {SEG.EQ3 | SEG.EQ2 | /SEG.EQ2}
    [Q.INI.0=<n>] [Q.INI.E=<n>] [Q.MAX.0=<n>] [Q.MAX.E=<n>]
    [TWO.PHAS]
  ]
  [ES.RAND=<n>] [ES.F.RAN=<n>] [ES.BREAK=<n>] [ES.F.H=<n>]
  [ES.100=<n>] [ES.F.100=<n>] [ES.110=<n>] [ES.F.110=<n>]
  [NLOC.PRE=<n>] [NLOC.EXP=<n>] [NLOC.MAX=<n>] [NLOC.K=<n>]
  [LOC.FAC=<n>] [CHAN.CRI=<n>] [CHAN.FAC=<n>] [DISP.FAC=<n>]
]
[T.ACT.0=<n>] [T.ACT.E=<n>] [ACT.MIN=<n>]
[D.PNORM=<n>] [D.PDE=<n>]
[CM.SEC]

```

INITIALIZE [Page 3-64](#)

```

{ ( IN.FILE=<c> { ( [SCALE=<n>] [FLIP.Y] ) | TIF } )
  | ( [WIDTH=<n> [DX=<n>]] [ {<111> | <110> | <100> | ORIENTAT=<n>} ]
    [ {ROT.SUB=<n> | X.ORIENT=<n>} ] [RATIO=<n>] [LINE.DAT]
  )
}
[ IMPURITY=<c> { I.CONC=<n> | I.RESIST=<n> } ]
[ MATERIAL=<c> ][ANTIMONY=<n>] [ARSENIC=<n>] [BORON=<n>] [PHOSPHOR=<n>]
[ {CONCENTR | RESISTIV} ]

```

INTERACTIVE [Page 3-12](#)

[<c>]

INTERMEDIATE [Page 3-32](#)

INTERMEDIATE

```

NAME=<c> [EXPRESS=<c>] [SIZE=<c> [MININDEX=<c>][DEFAULT=<c>]]
[PRINT] [DELETE]

```

INTERSTITIAL Page 3-265

```

{MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
 | AMBIENT}[D.0=<n>] [D.E=<n>]
[KB.0=<n>] [KB.E=<n>] [ {KB.LOW | KB.MED | KB.HIGH} ] [KIV.NORM]
[CEQUIL.0=<n>] [CEQUIL.E=<n>] [CL.MODEL] [VMOLE=<n>]
[NEU.0=<n>] [NEU.E=<n>] [NEG.0=<n>] [NEG.E=<n>] [DNEG.0=<n>]
[DNEG.E=<n>] [POS.0=<n>] [POS.E=<n>] [DPOS.0=<n>] [DPOS.E=<n>]
[ { ( C.STATE=<n> [C.V=<n>] ) | C.ALL }
  [DC.0=<n>] [DC.E=<n>] [FRAC.0=<n>] [FRAC.E=<n>]
  [KIV.0=<n>] [KIV.E=<n>] [KCV.0=<n>] [KCV.E=<n>]
  [ECLUST.0=<n>] [ECLUST.E=<n>] ]
[TRAP.CON=<n>] [K.TRAP.0=<n>] [K.TRAP.E=<n>]
[F.TRAP.0=<n>] [F.TRAP.E=<n>]
[CL.KFI.0=<n>] [CL.KFI.E=<n>] [CL.IFI=<n>] [CL.ISFI=<n>]
[CL.KFC.0=<n>] [CL.KFC.E=<n>] [CL.IFC=<n>] [CL.ISFC=<n>]
[CL.KFCI=<n>] [CL.CF=<n>] [CL.KR.0=<n>] [CL.KR.E=<n>] [CL.CR=<n>]
[ECLUST.N=<n>] [KLOOP.0=<n>] [KLOOP.E=<n>][RLMIN=<n>]
[ /MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE | /POLYSIL
 | /AMBIENT} ]
[ {V.MAXOX | V.INITOX | V.NORM} ]
[KSURF.0=<n>] [KSURF.E=<n>] [KSVEL.0=<n>] [KSVEL.E=<n>]
[KSRAT.0=<n>] [KSRAT.E=<n>] [VNORM.0=<n>] [VNORM.E=<n>]
[GROWTH] [THETA.0=<n>] [THETA.E=<n>]
[A.0=<n>] [A.E=<n>] [T0.0=<n>] [T0.E=<n>]
[KPOW.0=<n>] [KPOW.E=<n>] [GPOW.0=<n>] [GPOW.E=<n>]
[CM.SEC]

```

LABEL Page 3-155

```

X=<n> Y=<n> [CM] | ( [X.CLICK=<c>] [Y.CLICK=<c>] ) }
[SIZE=<n>] [COLOR=<n>]
[ LABEL=<c> [ {LEFT | CENTER | RIGHT} ] ]
[LINE.TYP=<n>] [C.LINE=<n>] [LENGTH=<n>]
[ { ( [SYMBOL=<n>] [C.SYMBOL=<n>] )
  | ( [RECTANGL] [C.RECTAN=<n>] [W.RECTAN=<n>]
[H.RECTAN=<n>] )
}
]

```

LINE Page 3-55

```

{X | Y} LOCATION=<n> [SPACING=<n>] [TAG=<c>]

```

LOADFILE [Page 3-68](#)

```
IN.FILE=<c>
{ ( [SCALE=<n>] [FLIP.Y] ) | TIF | DEPICT }
```

LOOP/L.END [Page 3-18](#)

```
[STEPS=<c>] [INDEX=<c>] [START=<n>]]
[ OPTIMIZE [DSSQ=<n>] [DNORM=<n>] [PLOT] ]
```

L.MODIFY [Page 3-22](#)

```
[STEPS=<n>] [ {NEXT | BREAK} ]
```

MASK [Page 3-82](#)

```
[ IN.FILE=<c> [ SCALE=<n> ] [ GRID=<n> ] [ G.EXTENT=<n> ] ] [ PRINT ]
```

MATERIAL Page 3-228

```

{ ( MATERIAL=<c> [NEW]) | SILICON | OXIDE | OXYNITRI | NITRIDE
  | POLYSILI | PHOTORES | ALUMINUM | AMBIENT
}
[TIF.NAME=<c>] [MD.INDEX=<n>] [IMPL.TAB=<c>]
[DY.DEFAU=<n>] [E.FIELD] [ION.PAIR] [IP.OMEGA=<n>]
[NI.0=<n>] [NI.E=<n>] [NI.F=<n>] [EPSILON=<n>]
[DENSITY=<n>] [AT.NUM=<n>] [AT.WT=<n>] [MOL.WT=<n>]
[VISC.0=<n>] [VISC.E=<n>] [VISC.X=<n>] [VC=<c>] [TEMPERAT=<c>]
[YOUNG.M=<n>] [POISS.R=<n>] [LCTE=<c>] [INTRIN.S=<n>] [SURF.TEN=<n>]
[ { ( SEMICOND [AFFINITY=<n>] [BANDGAP=<n>] [EGALPH=<n>] [EGBETA=<n>]
      [N.CONDUC=<n>] [N.VALENC=<n>] [NC.F=<n>] [NV.F=<n>]
      [G.DONOR=<n>] [E.DONOR=<n>]
      [G.ACCEP=<n>] [E.ACCEP=<n>]
      [BOLTZMAN] [IONIZATI]
      [QM.BETA=<n>] [QM.YCRIT=<n>])
  | ( CONDUCTO [WORKFUNC=<n>] ) }
]
[MAX.DAMA=<n>] [DAM.GRAD=<n>] [D.PCONST=<n>] [D.PME=<n>] [D.PDLF=<n>]
[D.PDLME=<n>] [D.PDRE=<n>] [D.PNORM=<n>] [D.PDE=<n>] [D.PHDF=<n>]
[D.PME=<n>] [D.PLF=<n>] [D.PLME=<n>]
[POLYCRYS] [GRASZ.0=<n>] [GRASZ.E=<n>] [TEMP.BRE=<n>] [MIN.GRAI=<n>]
[FRAC.TA=<n>] [G.DENS=<n>] [F11=<n>] [F22=<n>] [ALPHA=<n>]
[GEOM=<n>] [GAMMA.0=<n>] [GAMMA.E=<n>] [DSIX.0=<n>] [DSIX.E=<n>]
[DSIM.0=<n>] [DSIM.E=<n>] [DSIMM.0=<n>] [DSIMM.E=<n>] [DSIP.0=<n>]
[DSIP.E=<n>] [GBE.0=<n>] [GBE.H=<n>] [GBE.E=<n>] [NSEG=<n>]
[TBU.0=<n>] [TBU.E=<n>] [TOXIDE=<n>] [EAVEL.0=<n>] [EAVEL.E=<n>]
[DLGX.0=<n>] [DLGX.E=<n>] [POLY.FAC=<n>]

```

MESH Page 3-50

```

[GRID.FAC=<n>]
[DX.MAX=<n>] [DX.MIN=<n>] [DX.RATIO=<n>]
[LY.SURF=<n>] [DY.SURF=<n>] [LY.ACTIV=<n>] [DY.ACTIV=<n>]
[LY.BOT=<n>] [DY.BOT=<n>] [DY.RATIO=<n>]
[FAST]

```

METHOD Page 3-188

```
[ {ERFC | ERF1 | ERF2 | ERF3 | VERTICAL
  | COMPRESS | VISCOELA | VISCOUS} ] [ST.HISTO]
[DY.OXIDE=<n>] [DY.EXACT] [DY.LOCAL] [GRID.OXI=<n>] [SKIP.SIL]
[ {PD.FERMI | PD.TRANS | PD.FULL} ]
[NSTREAMS=<n>] [PAIR.GRA] [PAIR.SAT] [PAIR.REC] [PD.PFLUX]
  [PD.PTIME] [PD.PREC]
[IMP.ADAP] [DIF.ADAP] [OX.ADAPT] [ERR.FAC=<n>] [UNREFINE=<n>]
[ {ACT.EQUI | ACT.TRAN | ACT.FULL} ]
[INIT.TIM=<n>] [ {TRBDF | MILNE | HYBRID | FORMULA=<c>} ]
[ {CG | GAUSS} ] [BACK=<n>] [BLK.ITLI=<n>]
[MIN.FILL] [MIN.FREQ=<n>] [MF.METH=<n>] [MF.DIST=<n>]
( [IMPURITY=<c> ] [VACANCY] [INTERSTI] [ANTIMONY] [ARSENIC] [BORON]
  [PHOSPHOR] [OXIDANT] [TRAP]
  [ {LU | SOR | SIP | ICCG} ]
  [ {FULL | PART | NONE} ] [SYMMETRY]
  [ {TIME.STE | ERROR | NEWTON} ]
  [REL.ERR=<n>] [ABS.ERR=<n>]
  ( [MATERIAL=<c>] [SILICON] [POLYSILI] [OXIDE] [OXYNITRI]
    [NITRIDE] [ALUMINUM] [PHOTORES]
    [REL.ADAP=<n>] [ABS.ADAP=<n>] [MIN.SPAC=<n>] [MAX.SPAC=<n>]
  )
)
)
[OX.REL=<n>] [CONTIN.M=<n>] [VE.SMOOT=<n>]
[E.ITMIN=<n>] [E.ITMAX=<n>] [E.RELERR=<n>] [E.RVCAP=<n>] [E.AVCAP=<n>]
  [E.USEAVC]
[E.REGRID] [E.TSURF=<n>] [E.DSURF=<n>] [E.RSURF=<n>]
[ {MOB.TABL | MOB.AROR | MOB.CAUG} ]
[ ITRAP [IT.CPL] [IT.ACT] {IT.ZERO | IT.THERM | IT.STEAD}
[MODEL=<c>] [ENABLE] ]
```

MOBILITY Page 3-259

```
[ {TAB.TEMP=<n> [KELVIN]
  TAB.CONC=<c> TAB.E.MU=<c> TAB.H.MU=<c> [TAB.CLEA]} ]
[ECN.MU=<n>] [ECP.MU=<n>]
[GSURFN=<n>] [GSURFP=<n>]
[MUN1=<n>] [MUN2=<n>] [AN=<n>] [CN=<n>]
[EXN1=<n>] [EXN2=<n>] [EXN3=<n>] [EXN4=<n>]
[MUP1=<n>] [MUP2=<n>] [AP=<n>] [CP=<n>]
[EXP1=<n>] [EXP2=<n>] [EXP3=<n>] [EXP4=<n>]
[MUN.MIN=<n>] [MUN.MAX=<n>] [NREFN=<n>]
[NUN=<n>] [XIN=<n>] [ALPHAN=<n>]
[MUP.MIN=<n>] [MUP.MAX=<n>] [NREFP=<n>]
[NUP=<n>] [XIP=<n>] [ALPHAP=<n>]
```

MOMENT [Page 3-224](#)

```

[ CLEAR ]
[ { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
  | PHOTORES } ]
[ RANGE=<n> SIGMA=<n> [ GAMMA=<n> ] [ KURTOSIS=<n> ] [ LSIGMA=<n> ]
[ LSLOPE=<n> ]
  [ D.FRAC=<n> D.RANGE=<n> D.SIGMA=<n> [ D.GAMMA=<n> ] [ D.KURTOS=<n> ]
    [ D.LSIGMA=<n> ] [ D.LSLOPE=<n> ]

    TILT=<n> ] ]
]

```

OPTION [Page 3-39](#)

```

[ DEVICE=<c> ] [ PLOT.OUT=<c> ]
[ { QUIET | NORMAL | VERBOSE | DEBUG } ]
[ INFORMAT ] [ DIAGNOST ]
[ ECHO ] [ EXECUTE ] [ V.COMPAT ]

```

PAUSE [Page 3-14](#)

```

[ <c> ]

```

PHOSPHORUS [Page 3-302](#)

```

{ MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
  | AMBIENT }
[ DIX.0=<n> ] [ DIX.E=<n> ] [ DIM.0=<n> ] [ DIM.E=<n> ]
[ DIMM.0=<n> ] [ DIMM.E=<n> ] [ DVX.0=<n> ] [ DVX.E=<n> ]
[ DVM.0=<n> ] [ DVM.E=<n> ] [ DVMM.0=<n> ] [ DVMM.E=<n> ]
[ DIPAIR.0=<n> ] [ DIPAIR.E=<n> ] [ DVPAIR.0=<n> ] [ DVPAIR.E=<n> ]
[ R.I.S=<n> ] [ E.I.S=<n> ] [ R.V.S=<n> ] [ E.V.S=<n> ]
[ R.IP.V=<n> ] [ E.IP.V=<n> ] [ R.VP.I=<n> ] [ E.VP.I=<n> ]
[ SS.CLEAR ] [ SS.TEMP=<n> ] [ SS.CONC=<n> ]
[ { /MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE | /POLYSIL
  | /AMBIENT } ]
  [ SEG.0=<n> ] [ SEG.E=<n> ] [ TRANS.0=<n> ] [ TRANS.E=<n> ]
]
[ ES.RAND=<n> [ ES.F.RAN=<n> ] ] [ ES.100=<n> [ ES.F.100=<n> ] ]
[ ES.110=<n> [ ES.F.110=<n> ] ]
[ CM.SEC ]

```


PLOT.1D Page 3-136

PLOT.1D

```

{ { [ { X.VALUE=<n> | Y.VALUE=<n> } ]
  | ( { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE
      | POLYSILI | PHOTORES | ALUMINUM
      }
    { /MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE
      | /POLYSIL | /PHOTORE | /ALUMINU | /AMBIENT | /REFLECT
      }
    )
  }
| { IN.FILE=<c>
  { (TIF X.AXIS=<c> Y.AXIS=<c>)
  | ( { (COLUMN [X.COLUMN=<n>] [Y.COLUMN=<n>])
      | (ROW [X.ROW=<n>] [Y.ROW=<n>]) }
    [X.LABEL=<c>] [Y.LABEL=<c>] )
  }
  [X.SHIFT=<n>] [Y.SHIFT=<n>]
  [X.SCALE=<n>] [Y.SCALE=<n>]
  [Y.LOG] [X.LOG]
}
| ELECTRIC
}
[BOUNDARY] [CLEAR] [AXES]
[SYMBOL=<n>] [CURVE] [LINE.TYP=<n>] [COLOR=<n>]
[LEFT=<n>] [RIGHT=<n>] [BOTTOM=<n>] [TOP=<n>]
[X.OFFSET=<n>] [X.LENGTH=<n>] [X.SIZE=<n>]
[Y.OFFSET=<n>] [Y.LENGTH=<n>] [Y.SIZE=<n>]
[T.SIZE=<n>]

```

PLOT.2D Page 3-144

```

[X.MIN=<n>] [X.MAX=<n>] [Y.MIN=<n>] [Y.MAX=<n>] [SCALE]
[CLEAR] [AXES] [BOUNDARY] [L.BOUND=<n>] [C.BOUND=<n>]
[GRID] [L.GRID=<n>] [C.GRID=<n>]
[ [STRESS] [FLOW] VLENG=<n> [VMAX=<n>]
  [L.COMPRE=<n>] [C.COMPRE=<n>] [L.TENSIO=<n>] [C.TENSIO=<n>]
]
[DIAMONDS]
[X.OFFSET=<n>] [X.LENGTH=<n>] [X.SIZE=<n>]
[Y.OFFSET=<n>] [Y.LENGTH=<n>] [Y.SIZE=<n>]
[T.SIZE=<n>]

```

PLOT.3D Page 3-152

```
[THETA=<n>] [PHI=<n>]
[ CLEAR ]
[X.MIN=<n>] [X.MAX=<n>] [Y.MIN=<n>] [Y.MAX=<n>]
[Z.MIN=<n>] [Z.MAX=<n>] [LINE.TYP=<n>] [COLOR=<n>]
[ NUM.CNTR=<n> ]
[ BOUNDARY ] [L.BOUND=<n>] [C.BOUND=<n>]
```

PRINT.1D Page 3-132

```
{ X.VALUE=<n> | Y.VALUE=<n>
  | ( { MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE
      | POLYSILI | PHOTORES | ALUMINUM
      }
      { /MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE
      | /POLYSIL | /PHOTORE | /ALUMINU | /AMBIENT | /REFLECT
      }
    )
}
[OUT.FILE=<c> [APPEND] ]
[SPOT=<n>] [LAYERS]
[X.MIN=<n>] [X.MAX=<n>]
```

PROFILE Page 3-84

```
{ IMPURITY=<c> | ANTIMONY | ARSENIC | BORON | PHOSPHOR }
IN.FILE=<c> OFFSET=<n> [REPLACE]
```

REACTION Page 3-254

```
{ MAT.BULK=<c> | ( MAT.R=<c> /MAT.L=<c> ) }
[MODEL=<c>] [NAME=<c>]
[ { DELETE | REPLACE } ]
{ ( [IMP.L=<c>] [NI.L=<n>] [EI.L=<n> ]
    [ /IMP.L=<c>] [ /NI.L=<n>] [ /EI.L=<n> ]
    [IMP.R=<c>] [NI.R=<n>] [EI.R=<n> ]
    [ /IMP.R=<c>] [ /NI.R=<n>] [ /EI.R=<n> ]
    [NM.R=<n>] [ /NM.L=<n>]
    [RATE.0=<n>] [RATE.E=<n>] [EQUIL.0=<n>] [EQUIL.E=<n>]
  )
  | ( MAT.NEW=<c> THICKNES=<n> )
}
```

REGION [Page 3-62](#)

```
{ MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
  | PHOTORES | ALUMINUM
}
XLO=<c> XHI=<c> YLO=<c> YHI=<c>
```

RETURN [Page 3-10](#)

```
<c>
```

SAVEFILE [Page 3-71](#)

```
OUT.FILE=<c> [TEMPERAT=<n>]
{ ( [SCALE=<n>] [FLIP.Y] [ACTIVE] )
  | (TIF [TIF.VERS=<c>]) | DEPICT
  | ( MEDICI [POLY.ELE] [ELEC.BOT] ] )
  | ( MINIMOS5 X.MASK.S=<n>
    { HALF.DEV | ( FULL.DEV X.MASK.D=<n> [X.CHANNE=<n>] ) }
    [X.MIN=<n>] [X.MAX=<n>] [Y.MIN=<n>] [Y.MAX=<n>]
    [DX.MIN=<n>] [DY.MIN=<n>]
  )
  | ( WAVE [ACTIVE] [CHEMICAL] [DEFECT] [OXID] [MISC] )
}
```

SELECT [Page 3-128](#)

```
[Z=<c>] [TEMPERAT=<n>] [LABEL=<c>] [TITLE=<c>]
```

SOURCE [Page 3-9](#)

```
<filename>
```

STOP [Page 3-15](#)

```
<c>
```

STRESS [Page 3-124](#)

```
[TEMP1=<n> TEMP2=<n>]
[NEL=<n>]
```

STRUCTURE [Page 3-77](#)

```
[ TRUNCATE { ( {RIGHT | LEFT} X=<n> ) | ( {BOTTOM | TOP} Y=<n> ) } ]
[ REFLECT [ {RIGHT | LEFT} ] ]
[ EXTEND [ {RIGHT | LEFT} ] WIDTH=<n>
  [SPACES=<n>] [DX=<n>] [XDX=<n>] [Y.ELIM=<c>] ]
[ UNREFINE [REPEAT=<n>] [ROWS] [COLUMNS]
  [X.MIN=<n>] [X.MAX=<n>] [Y.MIN=<n>] [Y.MAX=<n>] ]
[TEMPERAT=<n>]
```

UNDEFINE [Page 3-45](#)

```
<name>
```

VACANCY [Page 3-276](#)

```
{MATERIAL=<c> | SILICON | OXIDE | OXYNITRI | NITRIDE | POLYSILI
 | AMBIENT}
[D.0=<n>] [D.E=<n>]
[CEQUIL.0=<n>] [CEQUIL.E=<n>] [VMOLE=<n>]
[NEU.0=<n>] [NEU.E=<n>] [NEG.0=<n>] [NEG.E=<n>] [DNEG.0=<n>]
[DNEG.E=<n>] [POS.0=<n>] [POS.E=<n>] [DPOS.0=<n>] [DPOS.E=<n>]
[ { ( [C.I=<n>] C.STATE=<n> ) | C.ALL }
  [DC.0=<n>] [DC.E=<n>] [FRAC.0=<n>] [FRAC.E=<n>]
  [KIC.0=<n>] [KIC.E=<n>] [ECLUST.0=<n>] [ECLUST.E=<n>] ]
[ECLUST.N=<n>]
[ { /MATERIA=<c> | /SILICON | /OXIDE | /OXYNITR | /NITRIDE | /POLYSIL
 | /AMBIENT } ]
[ { v.MAXOX | v.INITOX | v.NORM } ]
[KSURF.0=<n>] [KSURF.E=<n>] [KSVEL.0=<n>] [KSVEL.E=<n>]
[KSRAT.0=<n>] [KSRAT.E=<n>] [VNORM.0=<n>] [VNORM.E=<n>]
[GROWTH] [THETA.0=<n>] [THETA.E=<n>]
[A.0=<n>] [A.E=<n>] [T0.0=<n>] [T0.E=<n>]
[KPOW.0=<n>] [KPOW.E=<n>] [GPOW.0=<n>] [GPOW.E=<n>]
[CM.SEC]
```

VIEWPORT [Page 3-184](#)

```
[X.MIN=<n>] [X.MAX=<n>] [Y.MIN=<n>] [Y.MAX=<n>]
```

Tutorial Examples

Overview

This chapter presents three short examples showing how to use TSUPREM-4 to do some simple simulations. New users may wish to study these examples in sequence. Each example introduces new TSUPREM-4 commands and concepts. The examples in this chapter are fairly short and execute relatively quickly. The simulations presented in [Chapter 5, Advanced Examples](#) are more typical of real-life applications, and take more execution time. The examples include:

- A one-dimensional simulation of a bipolar transistor structure. This example illustrates the basic simulation steps using a simple one-dimensional grid. (See [One-Dimensional Bipolar Example on page 4-2](#)).
- An example that shows how to set up a two-dimensional grid and how to activate the various oxidation and point defect models. (See [Local Oxidation on page 4-12](#)).
- An example illustrating the effect of the point defect models on impurity diffusion. (See [Point Defect Models on page 4-27](#)).

Input File Syntax and Format

This chapter uses lowercase file names with the *.inp* extension for the example input files. Note the following regarding the format of input files:

1. Each statement occupies one line, unless the statement is explicitly continued with a “+” character at the end of a line.
2. Statement and parameter names can be abbreviated.
3. On most statements, parameters can appear in any order.
4. Blank lines between statements and extra spaces between parameters are ignored.
5. Comment statements begin with the “\$” character and are used to document the input file.

6. The input can contain a mixture of upper and lower case; the case is ignored except in some character strings.

One-Dimensional Bipolar Example

This section presents a one-dimensional simulation of a bipolar transistor structure. The example simulates the active region of an oxide-isolated bipolar structure, from the buried collector region up through the emitter. The statements that direct the simulation are contained in the simulation input files *s4ex1a.inp* and *s4ex1b.inp*.

The purpose of this example is to illustrate the mechanics of using TSUPREM-4 as well as the use of particular statements for mesh setup, model specification, simulation of processing steps, and printing and plotting of results. A one-dimensional example is presented primarily to simplify the discussion, but it also serves to illustrate a useful technique for performing fast simulations of simple structures.

TSUPREM-4 Input File Sequence

This example illustrates the organization of a typical TSUPREM-4 input file. In general, the sequence is as follows:

1. Identify the problem with comments and set any necessary execution options (none are needed in most cases).
2. Generate an initial mesh or read in a previously saved structure.
3. Simulate the desired process sequence, and print and/or plot the results.

Note that there is considerable flexibility in this sequence; for example, alternating between simulation and plotting. The only strict requirement is that a mesh must be defined before any processing or output can be performed.

Initial Active Region Simulation

The input statements in the file *s4ex1a.inp* simulate the initial steps in the formation of the active region of a bipolar structure, including the formation of the buried collector and deposition of the epitaxial layer. These input statements are shown in [Figure 4-1](#).

Mesh Generation

Traditionally, generation of the simulation mesh has been one of the most difficult and time-consuming tasks required for process simulation. If the mesh is too coarse, accuracy of the simulation is poor, while making the mesh too fine wastes time and computational resources. TSUPREM-4 simplifies the problem of creat-

ing an appropriate mesh by providing automatic mesh generation and adaptive gridding.

```

$ TSUPREM-4 -- Example 1, Part A
$ Bipolar active device region: Buried layer and epitaxial deposition

$ Use automatic grid generation and adaptive grid
INITIALIZE BORON=1E15

$ Grow buried layer masking oxide
DIFFUSION TEMP=1150 TIME=120 STEAM

$ Etch the buried layer masking oxide
ETCH OXIDE ALL

$ Implant and drive in the antimony buried layer
IMPLANT ANTIMONY DOSE=1E15 ENERGY=75
DIFFUSION TEMP=1150 TIME=30 DRYO2
DIFFUSION TEMP=1150 TIME=360

$ Etch the oxide.
ETCH OXIDE ALL

$ Grow 1.8 micron of arsenic-doped epitaxy
EPITAXY THICKNES=1.8 SPACES=9 TEMP=1050 TIME=6 ARSENIC=5E15

$ Grow pad oxide and deposit nitride
DIFFUSION TEMP=1050 TIME=30 DRYO2
DEPOSITION NITRIDE THICKNES=0.12

$ Save initial active region results
SAVEFILE OUT.FILE=S4EX1AS

$ Plot results
SELECT Z=LOG10(BORON) TITLE="Active, Epitaxy" LABEL=LOG(CONCENTRATION)
PLOT.1D BOTTOM=13 TOP=21 RIGHT=5 LINE.TYP=5 COLOR=2
SELECT Z=LOG10(ARSENIC)
PLOT.1D ^AXES ^CLEAR LINE.TYP=2 COLOR=3
SELECT Z=LOG10(ANTIMONY)
PLOT.1D ^AX ^CL LINE.TYP=3 COLOR=3

$ Label plot
LABEL X=4.2 Y=15.1 LABEL=Boron
LABEL X=-.8 Y=15.8 LABEL=Arsenic
LABEL X=2.1 Y=18.2 LABEL=Antimony

$ Print layer information
SELECT Z=DOPING
PRINT.1D LAYERS

```

Figure 4-1 Input file *s4ex1a.inp*, for simulating the buried layer and epitaxial deposition for a bipolar transistor structure

Automatic Mesh Generation

A mesh is generated automatically when the **INITIALIZE** statement is processed. A boron concentration of $10^{15}/\text{cm}^3$ for the initial structure has been specified. By default, the mesh that has two vertical lines, one at $x=0$ and one at $x=1$ micron. (For a two-dimensional simulation it would also be necessary to specify the width of the initial structure.) The location of the horizontal grid lines is determined by defaults set in the *s4init* file. The mesh can be made finer or coarser by

including a **MESH** statement with the **GRID.FAC** parameter; the default spacing has been chosen for this example.

Adaptive Gridding

Adaptive gridding is done whenever an ion implantation or diffusion step is simulated to ensure that the grid is fine enough to maintain the required accuracy. The accuracy criteria can be adjusted with the **ERR.FAC** parameter on the **METHOD** statement; in this example the default values are used. By using adaptive gridding the need to predict the grid requirements of a simulation in advance is eliminated.

Model Selection

The choice of simulation models should be considered before performing any processing steps. The speed and accuracy of the simulation depends strongly on the choice of models. The default models have been carefully chosen to give the best results in many cases, but some choices depend on the structures being simulated and on individual requirements of the user.

Two model choices must be made in most simulations:

- Oxidation model
- Point defect model

These selections are made using the **METHOD** statement. In this example the default models are used, i.e., the **VERTICAL** oxidation model and the **PD.FERMI** point defect model.

The choice of models may be changed during the course of the simulation.

Oxidation Model

The **VERTICAL** oxidation model can be used in this case because the simulation is one-dimensional, so only planar surfaces are oxidized. The **ERFC** oxidation model could also be used in this example, but it does not model the dependence of the oxidation rate on the concentration of impurities in the silicon. Also, the **ERFC** model requires an additional statement (an **AMBIENT** statement with the **INITIAL** parameter) to specify the initial oxide thickness whenever a structure with an initial oxide layer is oxidized, and does not automatically recognize the presence of nitride masking layers.

Point Defect Model

For this example the default point defect model (**PD.FERMI**) is used. The **PD.TRANS** point defect model increases the simulation time significantly, and should be used only when required. In this example, the effects of nonequilibrium point defect concentrations are relatively small. The magnitude of the error can be checked by repeating the simulation using the **PD.TRANS** or **PD.FULL** model.

Processing Steps

The following processing steps are used:

1. Buried layer masking oxide growth and patterning
2. Buried layer implant and drive-in
3. Epitaxial layer growth
4. Pad oxide growth and nitride mask deposition

Buried Layer Masking Oxide

The first step in the process is to grow an oxide to mask the buried layer implant. Oxidation is accomplished by specifying an oxidizing ambient (**STEAM**, in this case) on the **DIFFUSION** statement. The time (in minutes) and temperature (in °Celsius) must be specified. Note that specifying an oxidizing ambient turns on oxidation, but does not disable any other aspects of a diffusion step. Impurity diffusion and segregation at interfaces still occur.

The oxide must be removed from the active region of the device before implanting the buried layer. This is accomplished with the **ETCH** statement. The simplest form of the **ETCH** statement, used here, removes all of a specified material. The **ETCH** statement can also be used to remove portions of a layer.

Buried Layer

The next step is to implant the buried layer and drive it in. This is done with the **IMPLANT** statement and the following **DIFFUSION** statements. The **IMPLANT** statement specifies the type of impurity, and the dose (per cm²) and energy (in keV). Use of adaptive gridding is generally sufficient to ensure that the mesh at the surface is fine enough to contain several grid points within the peak of the implanted distribution, but it is wise to plot the as-implanted distribution if there is any doubt.

The drive-in of the buried layer is done in two steps. In the first, a **DRYO2** ambient is used to grow a thin layer of oxide to prevent outdiffusion of the implanted anti-mony. No ambient is given for the second; the absence of an ambient specification implies that an inert ambient is used. Once the drive-in is finished, the oxide is removed with an **ETCH** statement.

Epitaxial Layer

Next, an epitaxial layer is grown. Epitaxial growth is simulated with the **EPITAXY** statement, which combines the effects of the **DEPOSITION** and **DIFFUSION** statements. The thickness of the deposited layer is specified to be 1.8 microns using the **THICKNES** parameter.

The grid density in the layer is determined by the **SPACES** parameter. This example specifies that 9 grid spaces be placed in the epitaxial layer. This produces a reasonable starting grid for the base and emitter processing that follows. Note that the specification of the number of spaces determines not only the grid in the resulting structure, but also the time discretization of the epitaxy step. The **EPITAXY** statement causes the deposition of one grid layer followed by a diffusion for some fraction of the total time. This process is repeated for each grid

layer. If **SPACES** is not specified, its value defaults to one and the **EPITAXY** is equivalent to a **DEPOSITION** followed by a **DIFFUSION** statement.

The time and temperature for the epitaxial growth step are given by the **TIME** and **TEMPERAT** parameters, respectively. These determine the amount of diffusion that occurs during the growth process. The doping of the epitaxial layer can be specified with the **ARSNIC** parameter on the **EPITAXY** statement.

Pad Oxide and Nitride Mask

Next, a pad oxide is grown (using the **DIFFUSION** statement with the **DRYO2** parameter) and a nitride mask is deposited to be used for the field-isolation oxidation. The deposition is specified by the **DEPOSITION** statement, which specifies the material to deposit and the thickness of the layer. Other optional parameters can be used to specify the number and spacing of grid spaces in the deposited layer; by default, a single grid space is used in the deposited layer.

Saving the Structure

The resulting structure is saved using the **SAVEFILE** statement. In this case, it is saved in the output file named *S4EXIAS*. The saved structure is used as the starting point for the remaining process steps, which is simulated with a separate TSUPREM-4 input file.

It is recommended that the structure be saved after any long (in terms of computer time) sequence of operations. This allows the simulation to be resumed at that point should the need arise. It is also a good idea to save the structure at the end of any simulation, so that the results can be examined further at a later time.

Plotting the Results

The results of the simulation to this point are now ready to be displayed. This example demonstrates the use of the **SELECT**, **PLOT .1D**, and **LABEL** statements; two-dimensional plots are demonstrated in the next section.

Specifying a Graphics Device

Before displaying a plot, the program must be told what sort of graphics device is being used. This can be done either with the **DEVICE** parameter on the **OPTION** statement or by using a default plot device. In this example, the default plot device is used. See [Chapter 3 OPTION on page 3-39](#) and [Appendix B: "Graphics Devices."](#)

The **SELECT** Statement

The value to be plotted is given by the **Z** expression on the **SELECT** statement. **Z** defines a mathematical expression that may contain a number of variables, functions, and mathematical operators. (See the description of the **SELECT** statement in [Chapter 3 SELECT on page 3-128](#)). This example plots the base-10 logarithm of the various impurity concentrations.

**Note:**

*The Z expression is evaluated when the **SELECT** statement is processed. Thus the **SELECT** statement should be specified after any process steps that affect the device structure or solution quantities (e.g., impurity concentrations) used in the expression.*

The **SELECT** statement can also be used to specify a title for the plot and/or the label to be used on the vertical axis. If no label is given, the **Z** expression is used.

The **PLOT . 1D** Statement

The **PLOT . 1D** statement plots the values of a quantity along a (one-dimensional) section through the device. The section can be a vertical line at the location specified by **X . VALUE**, a horizontal line at **Y . VALUE**, or along an interface between two materials. This default is to plot along the vertical line at $x=0$.

The first **PLOT . 1D** statement in this example plots the axes and title as well as the logarithm of the antimony concentration. In addition to the x coordinate, the minimum and maximum values for the y axis and the maximum value for the x axis have also been specified.

**Note:**

*The limits of the y axis are in the units of the Z expression on the **SELECT** statement; thus, 13 and 21 are used, (not 1e13 and 1e21).*

If no axis limits are given, default values based on the dimensions of the device and the values of the **Z** expression are used.

By default, a dashed vertical line is drawn at the interfaces between materials. **LINE . TYP=5** specifies that dashed line type 5 is to be used for plotting the boron profile; **COLOR=2** specifies that color 2 (usually red) is to be used on color graphics devices.

The next two **PLOT . 1D** statements add to the first plot. They do this by including the **^CLEAR** and **^AXES** specifications to prevent clearing of the screen and drawing of new axes, respectively. For each plot a new quantity to be plotted is specified (using a **SELECT** statement) along with a different line type and color.

Labels

Labels are added to the plot with **LABEL** statements. Each **LABEL** statement specifies a text string to be plotted and a pair of x and y coordinates. The text string is plotted starting at the specified coordinates. **X** and **Y** are in the units of the plot axes, in this case microns and \log_{10} (concentration), respectively. **X** and **Y** could also be given in centimeters, using the **CM** parameter. The final plot is shown in [Figure 4-2](#).

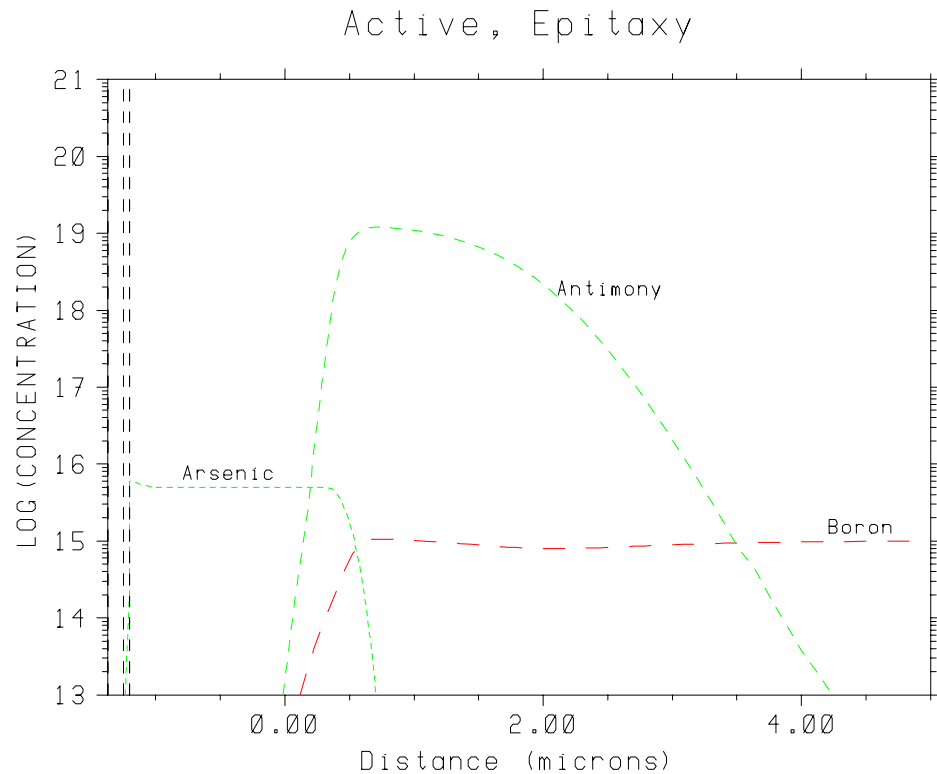


Figure 4-2 Impurity distributions in bipolar structure at end of input file *s4ex1a.inp*

Printing Layer Information

Although plots are the primary means of examining the results of a TSUPREM-4 simulation, useful information is also available from the **PRINT .1D** statement.

The **PRINT .1D** Statement

The **PRINT .1D** statement works with the **Z** expression given by a **SELECT** statement. Information is presented along a one-dimensional section specified in the same manner as in the **PLOT .1D** statement.

Three kinds of information can be printed:

1. A complete list of the values of the **Z** expression along the section.
2. The integral of the **Z** expression through each layer along the section.
3. The locations along the section where the **Z** expression has a specified value.

Using **PRINT .1D** **LAYERS**

The **PRINT .1D** statement at the end of file *s4ex1a.inp* uses the **LAYERS** parameter to request the integral of the **Z** expression (net doping) along a vertical section at $x=0$. The resulting output is shown in [Figure 4-3](#). For purposes of **PRINT .1D LAYERS**, a layer is defined as a portion of the section in which the material does not change and the sign of the **Z** expression is constant. In the case of net doping, each material is separated into layers of net n-type or net p-type doping.

```

$ Print layer information
SELECT      Z=DOPING
PRINT.1D   LAYERS
** Printing along X.VALUE=0:

  Num      Material      Top      Bottom  Thickness      Integral
  1         nitride     -1.3820  -1.2525   0.1200      1.2000e+00
  2          oxide     -1.2620  -1.2031   0.0491      5.5423e+08
  3         silicon     -1.2129   3.4935   4.7108      1.4215e+15
  4         silicon      3.4979  200.0000  196.5021     -1.9630e+13

```

Figure 4-3 Output listing from **PRINT.1D** command in file *s4ex1a.inp*

In this example, the **PRINT.1D LAYERS** statement provides a number of useful results. For each layer, the material type, the top and bottom coordinates of the layer, the thickness of the layer, and the integral of the selected quantity are printed. A number of useful values are evident, such as the oxide thickness (491 Å) and the depth of the buried layer/substrate junction (4.71 μm).



Note:

SELECT statement is required before a **PRINT.1D LAYERS** statement is processed, but you can use **SELECT Z=0** if all you want to know is the layer thicknesses or the coordinates of the material interfaces.

Completing the Active Region Simulation

The simulation is now completed with a separate execution of TSUPREM-4, using the input file *s4ex1b.inp* shown in Figure 4-4. This file follows the same outline as the first (initialize, select models, process, output), and most of the statements are of the types discussed previously. There are some differences, however.

Reading a Saved Structure

The most important difference is that instead of generating a mesh, the saved structure saved by the previous simulation is read from the file *S4EX1AS*. This is accomplished with the **IN.FILE** parameter on the **INITIALIZE** statement. The saved file includes complete mesh and solution information.

The next step specifies any needed model specifications. Because the choices of oxidation and point defect models were saved in the structure file, they do not need to be specified again. Any different models could be specified at this step.

Field Oxidation

The next step in the process is to grow the field isolation oxide. Although the active region of the device is being simulated, and no oxide is grown because of the nitride mask, the field oxidation step is included in order to simulate the dopant redistribution that occurs during the step. The field oxidation step illustrates how temperature and pressure ramping are specified.

The temperature is first ramped from 800°C to 1000°C, over a time of 20 minutes. The **TEMP** parameter specifies the starting temperature, while the **T.FINAL** parameter gives the temperature at the end of the step. The next step starts at 1000°C in a dry oxygen ambient at one atmosphere, and ramps the temperature to 1100°C and the pressure to 5 atmospheres over a period of 10 minutes. The starting pressure is given by the **PRESSURE** parameter (not needed here because the

default value of one atmosphere is used), and the final pressure is given by **P.FINAL**. Note that ramp rates could also have been specified in degrees/minute or atmospheres/minute, using the **T.RATE** and **P.RATE** parameters, respectively.

```

$ TSUPREM-4 -- Example 1, Part B
$ Bipolar active device region:  Field oxide, base, and emitter

$ Read structure
INITIALIZE IN.FILE=S4EX1AS

$ Grow the field oxide
DIFFUSION  TEMP=800   TIME=20           T.FINAL=1000
DIFFUSION  TEMP=1000  TIME=10  DRYO2    T.FINAL=1100  P.FINAL=5
DIFFUSION  TEMP=1100  TIME=50  STEAM    PRESSURE=5
DIFFUSION  TEMP=1100  TIME=10  DRYO2    PRESSURE=5    P.FINAL=1
DIFFUSION  TEMP=1100  TIME=60           T.FINAL=800

$ Remove nitride and pad oxide
ETCH       NITRIDE  ALL
ETCH       OXIDE   ALL

$ Implant the boron base
IMPLANT    BORON   DOSE=2E13  ENERGY=100

$ Implant the phosphorus emitter
IMPLANT    PHOSPHORUS  DOSE=1E15  ENERGY=50

$ Anneal to activate base and emitter regions
DIFFUSION  TEMP=1000  TIME=12  DRYO2

$ Plot results
SELECT     Z=LOG10(BORON)  TITLE="Active Region"  LABEL=LOG(CONCENTRATION)
PLOT.1D    BOTTOM=13  TOP=21  RIGHT=5  LINE.TYP=5  COLOR=2
SELECT     Z=LOG10(PHOSPHORUS)
PLOT.1D    ^AXES  ^CLEAR  LINE.TYP=4  COLOR=4
SELECT     Z=LOG10(ARSENIC)
PLOT.1D    ^AXES  ^CLEAR  LINE.TYP=2  COLOR=3
SELECT     Z=LOG10(ANTIMONY)
PLOT.1D    ^AXES  ^CLEAR  LINE.TYP=3  COLOR=3

$ Label the impurities
LABEL      X= 2.0  Y=15.1  LABEL=Boron
LABEL      X=-1.0  Y=19.5  LABEL=Phosphorus
LABEL      X= 0.3  Y=15.8  LABEL=Arsenic
LABEL      X= 2.0  Y=18.4  LABEL=Antimony

$ Print the layer information
SELECT     Z=DOPING
PRINT.1D   X.V=0  LAYERS

```

Figure 4-4 Listing of input file *s4ex1b.inp*, showing statements for simulating the field oxide, base, and emitter region processing for a bipolar transistor

The third diffusion step does a 50-minute oxidation in steam, at a pressure of 5 atmospheres. Steps four and five ramp down the pressure, then the temperature.

The remaining process steps are similar to those performed in the first part. The nitride and oxide are removed, and the base and emitter are implanted.

Final Structure

The plotting and labeling of the profiles use the statements described previously. The final plot is shown in Figure 4-5. A **PRINT .1D** statement is used to print layer information for the final structure. The **SELECT Z=DOPING** statement specifies that integrals of net doping are to be calculated and that layers are defined by the sign of the doping. From the printed output (Figure 4-6) a number of useful values can be extracted, such as the oxide thickness (255 Å), the thickness of the emitter region (0.28 μm), and the integrated doping in the base region ($8.12 \times 10^{12}/\text{cm}^3$).

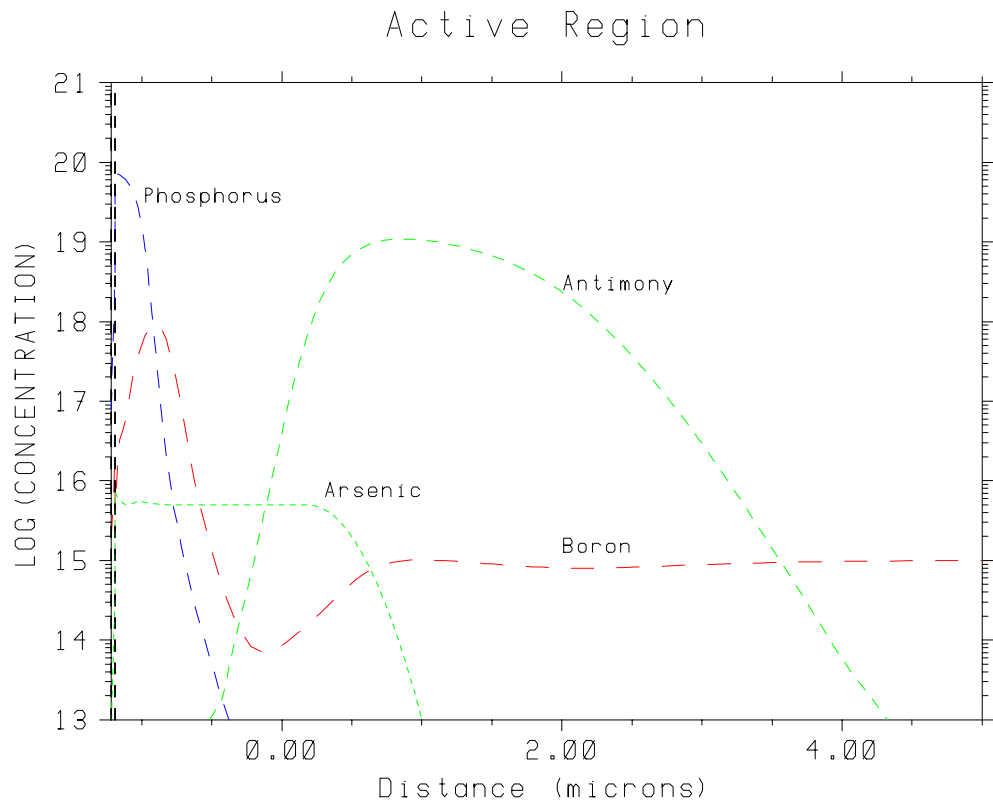


Figure 4-5 Final profiles produced by input file *s4ex1b.inp*

```

$ Print the layer information
SELECT      Z=DOPING
PRINT.1D   LAYERS
** Printing along X.VALUE=0:
    
```

| Num | Material | Top | Bottom | Thickness | Integral |
|-----|----------|---------|----------|-----------|-------------|
| 1 | oxide | -1.2281 | -1.2025 | 0.0255 | 3.7599e+12 |
| 2 | silicon | -1.2025 | -0.9202 | 0.2823 | 1.0427e+15 |
| 3 | silicon | -0.9202 | -0.5746 | 0.3456 | -8.1165e+12 |
| 4 | silicon | -0.5746 | 3.5837 | 4.1584 | 1.4212e+15 |
| 5 | silicon | 3.5837 | 200.0000 | 196.4136 | -1.9623e+13 |

Figure 4-6 Output listing from **PRINT .1D** command in file *s4ex1b.inp*

Local Oxidation

This section presents an example of a two-dimensional simulation of a local oxidation process. The purpose of this example is to illustrate the two-dimensional simulation capabilities of TSUPREM-4, and to provide some practical hints on the effective use of these capabilities.

This example simulates a narrow, locally-oxidized isolation region. The two features of interest are the oxide thickness and shape and the impurity distribution in the underlying silicon. These are examined in separate simulations. The commands for performing the simulations are contained in the simulation input files *s4ex2a.inp* and *s4ex2b.inp*.

Calculation of Oxide Shape

The first goal of this simulation is to determine the effect of using a narrow mask opening on the oxide shape. The input statements for this simulation are shown in [Figures 4-7](#) and [4-9](#). The steps are similar to those used in the previous example, but are complicated slightly by the two-dimensional nature of the simulation.

```

$ TSUPREM-4 narrow window example
$   Part 1: Oxide shape

$ Set up the grid
LINE  X      LOC=0.0   SPAC=0.15
LINE  X      LOC=1.25  SPAC=0.04
LINE  X      LOC=1.5   SPAC=0.1

LINE  Y      LOC=0     SPAC=0.03
LINE  Y      LOC=0.5   SPAC=0.1

$ No impurities, for faster oxidation simulation
INITIALIZE

$ Deposit pad oxide and define nitride mask
DEPOSITION OXIDE      THICKNES=0.03  SPACES=2
DEPOSITION NITRIDE    THICKNES=0.10  SPACES=2
ETCH       NITRIDE    RIGHT  P1.X=1.25

$ Plot the grid
SELECT      TITLE="Grid for Oxidation"
PLOT.2D     GRID  SCALE  C.GRID=2

$ Do the oxidation
METHOD      VISCOEL  DY.OXIDE=0.05  INIT=0.15
MATERIAL    MAT=OXIDE  VC=425
MATERIAL    MAT=NITRIDE VC=170
DIFFUSION   TEMP=1000  TIME=100  WETO2

$ Save the structure
SAVEFILE    OUT.FILE=S4EX2AS

```

Figure 4-7 First part of input file *s4ex2a.inp*, for determining LOCOS shape

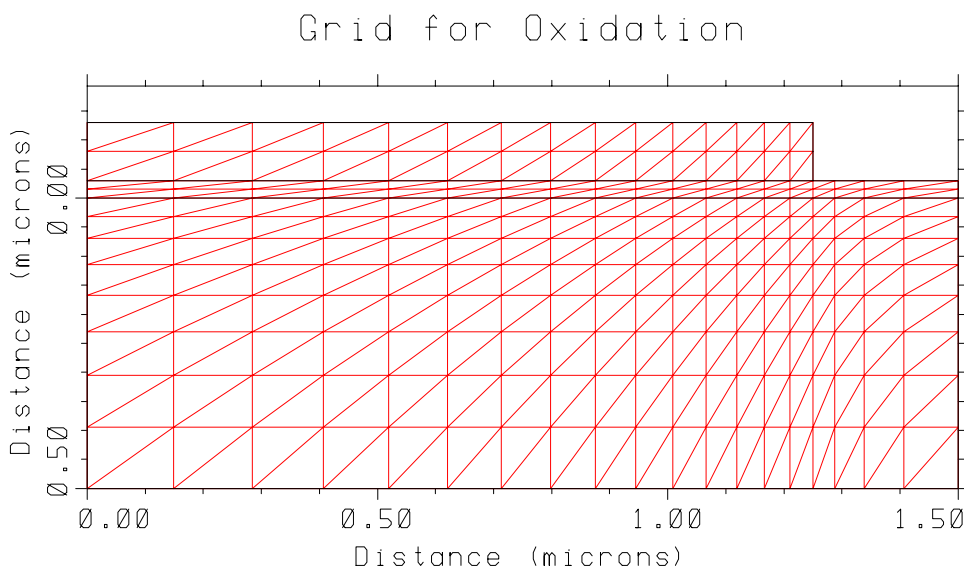


Figure 4-8 Mesh used for oxidation simulation. Produced by **PLOT . 2D GRID** statement in input file *s4ex2a.inp*

Mesh Generation

The simulation starts by defining a mesh. The object in this case is to define as coarse a mesh as possible without losing accuracy in the solution. (The accuracy can be checked later by repeating the simulation with a finer mesh.) Only half of the structure needs to be simulated, because of the symmetry of the structure. The **LINE X** statements specify the horizontal locations of vertical grid lines. The first and last **LINE X** statements specify the left and right edges of the structure; the **SPACING** parameters specify the horizontal grid spacings at these locations. Other **LINE X** statements are used to add grid lines and specify spacings at other locations in the structure. In the vertical direction, **LINE Y** statements specify the top and bottom locations (and corresponding grid spacings) for the mesh.

The grid for this example does not need to be very fine. In the horizontal direction the spacing is set to 0.05 microns at the edge of the nitride mask, where the two-dimensional effects are greatest ($x=1.25$); the spacing expands to 0.1 micron at the right edge and 0.15 micron at the left edge. In the vertical direction, the spacing is set to 0.03 at the top, expanding to 0.1 at the bottom.

The effect of the grid spacing on accuracy can be checked by running the simulation again with a finer mesh; this can be accomplished by adding a **MESH** statement with the **GRID . FAC** parameter anywhere before the **INITIALIZE** statement. There are no impurities specified in this simulation, so no adaptive gridding occurs based on implantation or diffusion of impurities. However, the grid is refined as needed based on oxidant concentration.

The **INITIALIZE** statement performs the actual mesh generation. Note that no substrate doping has been specified. This speeds up the simulation slightly by eliminating the need to solve for impurity diffusion while oxidizing.

Pad Oxide and Nitride Layers

Next the pad oxide is deposited. To improve the accuracy when calculating oxidant diffusion in the oxide, two grid spaces are placed in the deposited layer. (When checking the accuracy of the simulation, more divisions would be useful.) More grid spaces are added automatically as the oxide grows.

To mask the oxidation, a nitride layer is deposited and then patterned with an **ETCH** statement. The **RIGHT** keyword on the **ETCH** statement specifies that material is to be etched to the right of the line defined by the **P1.X**, **P1.Y**, **P2.X**, and **P2.Y** parameters. **P1.X=1.25** is specified; **P1.Y** defaults to a value above the highest point of the structure; **P2.X** defaults to the value of **P1.X**; and **P2.Y** defaults to a value below the bottom of the structure. Thus the etch occurs to the right of a vertical line through the structure at $x=1.25$ microns.

Plotting the Mesh

At this point it is wise to inspect the mesh. This is done with the **PLOT.2D** statement. The **PLOT.2D** statement is used to set up any two-dimensional plot. By itself, it can plot the material boundaries of a structure, the grid, and oxide stress and velocity vectors. It is also used to set the scale and draw axes for contour plots. In this case, the **GRID** keyword requests that the grid be plotted, and the **C.GRID** keyword specifies the color to be used.

The **SCALE** parameter causes the x or y axis to be scaled to reflect the true aspect ratio of the structure. If **SCALE** had not been specified, the plot would be stretched in the y direction to fill the plotting area. The **SELECT** statement is used only to specify a title for the plot. The resulting plot is shown in [Figure 4-8](#).

Model Selection

Next a **METHOD** statement is used with the **VISCOELA** keyword to select the oxidation model. The **VISCOELA** model is used because it allows fast simulation of two-dimensional, stress-dependent oxidation. The **DY.OXIDE** parameter sets the grid spacing to be used in the growing oxide. (The same grid is used both for the diffusion of the oxidizing species and for the calculation of oxide movement.) The **INIT.TIM**

T.TIM parameter specifies an initial time step of 0.15 minutes; this saves a small amount of calculation time. The default value is 0.002 minutes, which is appropriate for many diffusion steps but smaller than necessary for oxidation-only steps.

The effects of oxidant diffusion, interface reaction rate, and material viscosities on the stresses in the structure are included by default. Because the models for the stress effects do not include the dependence of parameter values on temperature, it is necessary to use **MATERIAL** statements to specify appropriate values for **VC** (the dependence of stress on material viscosity) at the oxidation temperature. (Suitable values are listed in the notes to Appendix A.)

The oxidation is accomplished by the **DIFFUSION** statement, which specifies a wet oxygen ambient at 1000°C for 100 minutes. The **SAVEFILE** statement saves the final structure for later analysis.

```

$ Plot the final structure, showing flow lines
SELECT      TITLE="Flow at End of Oxidation Step"
PLOT.2D     SCALE FLOW  VLENG=0.065  X.MIN=0.5  C.FLOW=4

$ Plot the final structure, showing stress vectors
SELECT      TITLE="Stresses After Oxidation"
PLOT.2D     SCALE STRESS  VLENG=0.2  X.MIN=0.5  +
            C.COMPRES=4  C.TENSIO=2  L.TENSIO=2

$ Plot filled contours of hydrostatic pressure
SELECT      Z=( -0.5 * ( SXX + SY Y ) ) TITLE="Contours of Hydrostatic Pressure"
PLOT.2D     SCALE X.MIN=0.5  X.MAX=1.8
FOREACH I ( 1 TO 5 )
    COLOR MIN.V=(( I - 0.5)*2E9) MAX.V=(( I + 0.5)*2E9) COLOR=(13 + I )
    COLOR MIN.V=(- I - 0.5)*2E9) MAX.V=(- I + 0.5)*2E9) COLOR=(25 - I )
END

$ Create a legend
LABEL X=1.52 Y=-0.2 LABEL="Compression" SIZE=0.3
LABEL X=1.60 Y=-0.15 LABEL="1-3E9" C.RECT=14 SIZE=0.3 W.RECT=0.35 H.R=0.35
LABEL X=1.60 Y=-0.1 LABEL="3-5E9" C.RECT=15 SIZE=0.3 W.RECT=0.35 H.R=0.35
LABEL X=1.60 Y=-0.05 LABEL="5-7E9" C.RECT=16 SIZE=0.3 W.RECT=0.35 H.R=0.35
LABEL X=1.60 Y= 0.0 LABEL="7-9E9" C.RECT=17 SIZE=0.3 W.RECT=0.35 H.R=0.35
LABEL X=1.52 Y= 0.05 LABEL="Tension" SIZE=0.3
LABEL X=1.60 Y= 0.1 LABEL="1-3E9" C.RECT=24 SIZE=0.3 W.RECT=0.35 H.R=0.35
LABEL X=1.60 Y= 0.15 LABEL="3-5E9" C.RECT=23 SIZE=0.3 W.RECT=0.35 H.R=0.35
LABEL X=1.60 Y= 0.20 LABEL="5-7E9" C.RECT=22 SIZE=0.3 W.RECT=0.35 H.R=0.35
LABEL X=1.60 Y= 0.25 LABEL="7-9E9" C.RECT=21 SIZE=0.3 W.RECT=0.35 H.R=0.35

$ Redraw boundaries
PLOT.2D     ^AX ^CL

$ Print location of interface
SELECT      Z=Y
PRINT.1D    SILICON /OXIDE

```

Figure 4-9 Second part of statement input file *s4ex2a.inp*, showing statements for plotting results of LOCOS process

Plotting the Results

The results are plotted using the statements shown in [Figure 4-9](#). The **PLOT . 2D** statement is used to plot the structure boundaries and material interfaces. The following parameters specify the plot format and content:

- **FLOW** adds vectors showing the speed and direction of material flow in the structure due to oxidation.
- **VLENG** specifies the maximum length of a flow vector (in microns) to be plotted.
- The **X . MIN** parameter specifies the minimum value of the *x* axis: (**X . MAX**, **Y . MIN**, and **Y . MAX** can be used to specify the other axis limits, but the default values are used in this case).
- The **SCALE** parameter is used to avoid distorting the structure as it is scaled for plotting.

The result of the **PLOT . 2D FLOW** statement is shown in Figure 4-10.

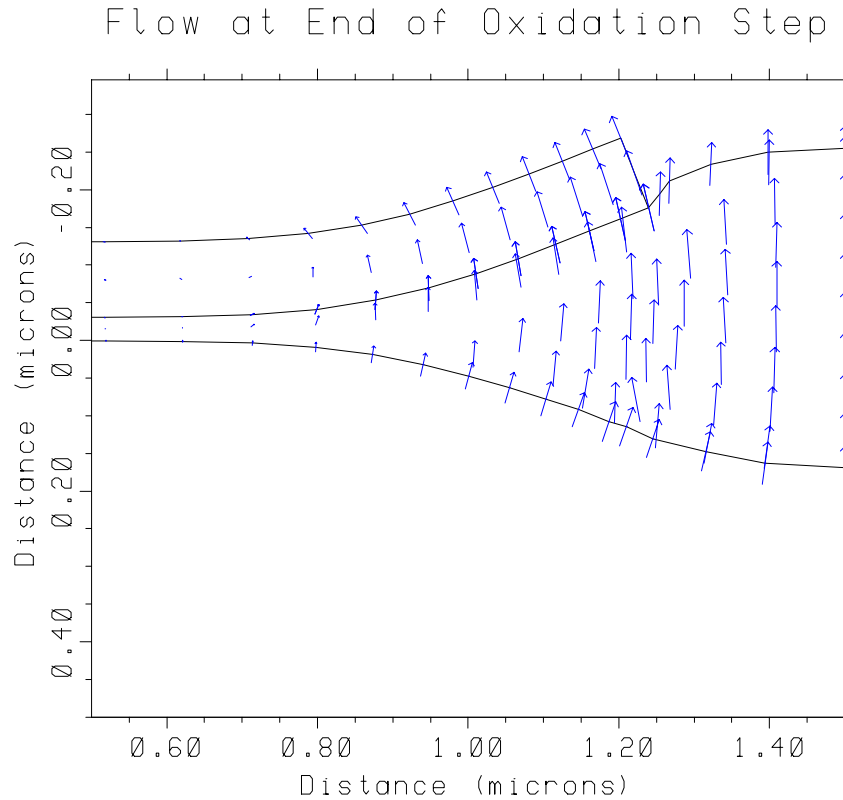


Figure 4-10 Plot produced by the **PLOT . 2D FLOW** statement in input file *s4ex2a.inp*

Plotting Stresses

Stresses in the structure can be plotted in several ways. One way is to use the **PLOT . 2D** statement with the **STRESS** parameter. The **STRESS** parameter requests that the principal components of stress be shown. The stresses are indicated by short lines whose lengths are proportional to the stress components. The line type indicates whether the material is in tension or compression. By default, both use line type one, but in this example **LINE . TEN=2** is specified so that a dashed line is used to indicate tension (the default is one). Tension and compression are also distinguished by color, using the **C . TENSIO** and **C . COMPRE** parameters. The **VLENG** parameter specifies the length of the line (in the units of the plot axes, i.e., microns) used for the maximum value of stress. Smaller stresses produce proportionally smaller lines.

The stress plot is shown in Figure 4-11. The long, solid lines indicate large compressive stresses in the direction of the lines; dashed lines indicate tension. As expected, the stresses are concentrated near the portion of the structure where the oxide growth is nonuniform and where the nitride is forced to bend.

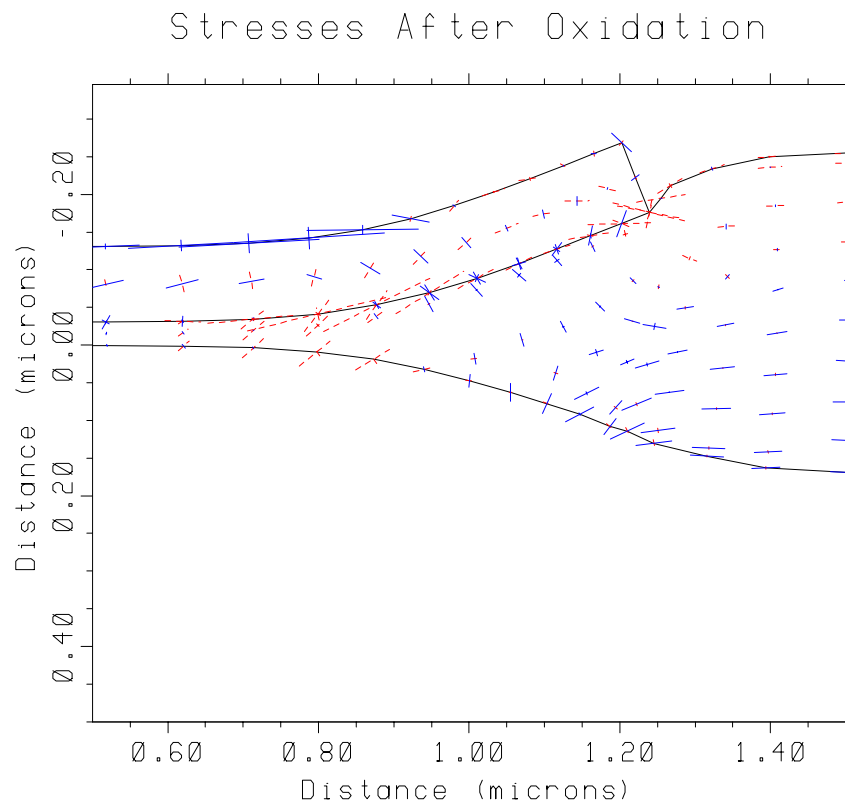


Figure 4-11 Plot produced by the **PLOT . 2D STRESS** statement in input file *s4ex2a.inp*

Contour Plots

Another way to show the stresses in the structure is with contour plots. It is difficult to show all three components of stress in a contour plot, and it is not very useful to show individual components because of their dependence on the choice of coordinate system. However, meaningful combinations of the stresses can be calculated with the **SELECT** statement and plotted with the **PLOT . 2D** and **CONTOUR** or **COLOR** statements.

This example shows how the hydrostatic pressure, defined as the negative average of the S_{xx} and S_{yy} stress components, can be plotted.

- The **SELECT** statement simply gives the mathematical expression for the function to be plotted.
- The **PLOT . 2D** statement plots the axes, structure boundaries, and material interfaces, **X . MAX** is specified greater than the right edge of the structure in order to leave room for a legend.
- A pair of **COLOR** statements inside a **FOREACH** loop are used to shade regions of differing pressures.

The **FOREACH** statement specifies a variable (I) and a range of values (1 to 5, in this case). The statements between the **FOREACH** statement and the matching **END** statement are executed once for each value of the variable. The **COLOR** statements specify a color and the range of pressures to be represented

by that color. Both the color and the minimum and maximum values are given by arithmetic expressions that depend on the variable I .

- A series of **LABEL** statements are used to produce a legend, showing the amount of tension or compression corresponding to each color.

Each **LABEL** statement specifies the location of the label (in microns, in this case), the text of the label, and a filled rectangle that is placed before the label. The size of the characters (in centimeters) is specified by the **SIZE** parameter. Width and height of the rectangle are given by **W.RECTAN** and **H.RECTAN** (abbreviated to **W.R** and **H.R**, respectively).

- Finally, a last **PLOT .2D** with the **^AXES** and **^CLEAR** parameters is done in order to redraw the structure boundaries, which may have been obscured when plotting the shaded contours. The result is shown in [Figure 4-12](#).

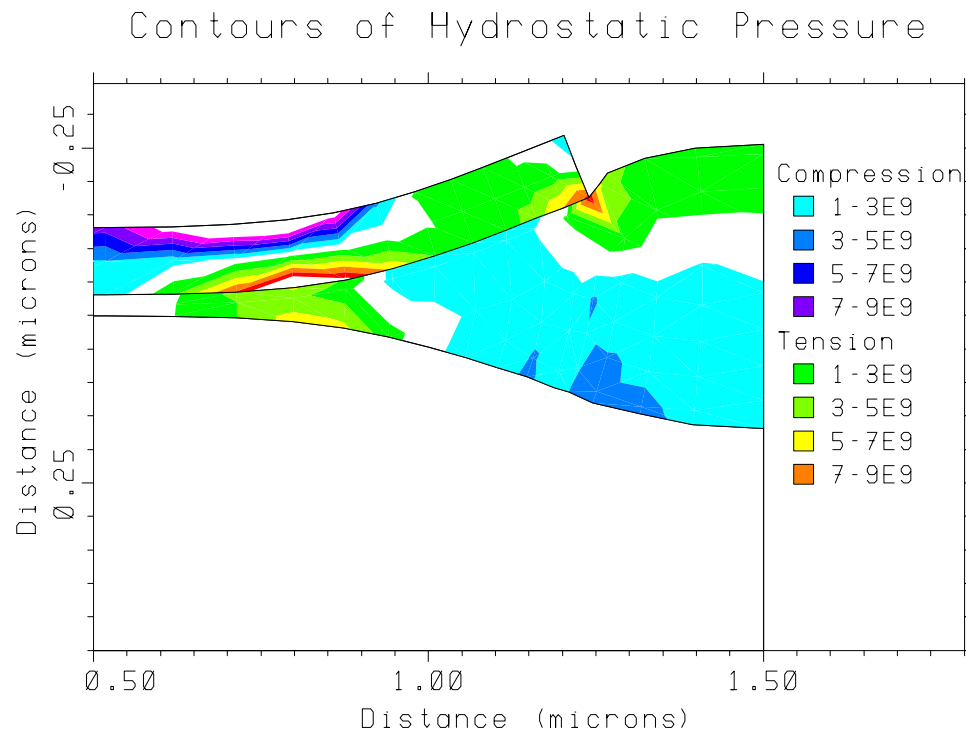


Figure 4-12 Contours of hydrostatic pressure plotted by statements in input file *s4ex2a.inp*

The last two lines of *s4ex2a.inp* show how the y coordinate of the silicon/oxide interface can be printed as a function of x , using the **PRINT .1D** statements to print solution values along the interface between two materials.

Two-Dimensional Diffusion with Point Defects

This example analyzes the diffusion of impurities during the local oxidation step. The same process is simulated as in the previous sections, except with boron in the substrate and a boron implant. The effects of point defect generation during oxidation are also included, but with a simpler oxidation model to reduce the com-

puter time requirements. The listing of the input (from file *s4ex2b.inp*) is shown in Figures 4-13 and 4-15.

Automatic Grid Generation

For convenience, automatic grid generation is used for this analysis. By specifying **WIDTH=1.5** on the **INITIALIZE** statement, the grid extends from $x=0$ to $x=1.5$, just as in the previous simulation. The background concentration of the substrate is specified with the **BORON** parameter. The grid is automatically refined as needed to maintain the accuracy of the impurity profiles.

Field Implant

The deposition of the pad oxide and nitride are the same as before. Now, however, a photoresist layer is deposited and patterned to be used as a boron implant mask. The boron is implanted through the pad oxide, then the photoresist is removed.

Oxidation

The **COMPRESS** oxidation model is chosen in order to obtain a fast two-dimensional simulation. The two-dimensional point defect model is specified by the **PD.TRANS** keyword on the **METHOD** statement. The **DIFFUSION** statement is the same as in the previous section.

The structure is saved at the end of the simulation so the results can be examined further without rerunning the simulation. In this example, the solution is saved in the file *S4EX2BS*.

```

$ TSUPREM-4 narrow window example
$   Part 2:  Impurity distribution

$ Use default grid
INITIALIZE BORON=1E15 WIDTH=1.5

$ Deposit pad oxide, LOCOS mask, and implant mask
DEPOSITION OXIDE      THICKNES=0.03  SPACES=2
DEPOSITION NITRIDE    THICKNES=0.10
DEPOSITION PHOTO      THICKNES=2

ETCH      PHOTO      RIGHT  P1.X=1.25
ETCH      NITRIDE    TRAP

$ Implant boron
IMPLANT   BORON   DOSE=2E13  ENERGY=100
ETCH      PHOTORESIST  ALL

$ Do the drive-in, with point defects
METHOD    COMPRESS  PD.TRANS
DIFFUSION TEMP=1000 TIME=100  WETO2

$ Save the structure
SAVEFILE  OUT.FILE=S4EX2BS

$ Plot the grid after diffusion
SELECT    TITLE="Grid After Impurity Diffusion"
PLOT.2D   SCALE  GRID  Y.MAX=1.2  C.GRID=2

```

Figure 4-13 First part of input file *s4ex2b.inp*, showing processing steps

Grid After Impurity Diffusion

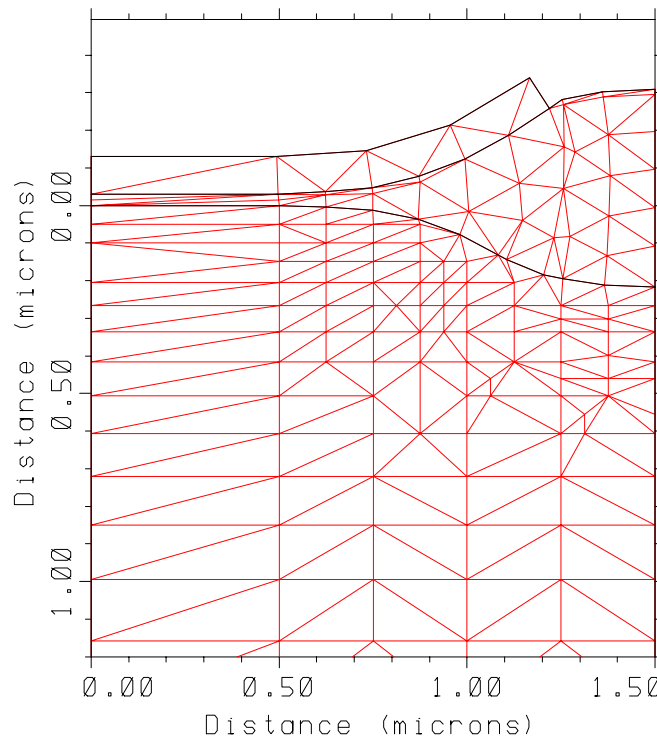


Figure 4-14 Grid plot produced by first **PLOT.2D** statement in input file *s4ex2b.inp*

Grid Plot

The grid at this point is plotted by the **PLOT.2D GRID** statement. The plot is shown in [Figure 4-14](#). The grid has been refined near the surface and around the field isolation oxide to improve the resolution of the boron profile.

The **Y.MAX** parameter is used to restrict the plot to the upper part of the structure. If **Y.MAX** had been omitted, the entire grid would have been plotted down to the bottom of the structure (at $y=200$ microns) in its 1.5:200 aspect ratio, producing a nearly useless result. If the **SCALE** parameters had been omitted, the plot would be compressed in the y direction to make it fit the available plotting area, but the shape of the structure would be distorted.

The **SELECT** statement has been used to specify a title for the plot.

Contour of Boron Concentration

[Figure 4-15](#) shows the input statements used to plot the results. The first plot shows contours of equal boron concentration. A **PLOT.2D** statement sets up the axes for the plot and draws the boundaries of the structure and the material interfaces. **Y.MAX**=1.2 (microns) is specified so that only the portion of the device

near the surface is plotted. The **SELECT** statement specifies that the (base 10) logarithm of the boron concentration is to be plotted.

```

$ Plot contours of boron
SELECT      Z=LOG10(BORON)  TITLE="Contours of Boron Concentration"
PLOT.2D     SCALE  Y.MAX=1.2
FOREACH     X (15.5 16.5 17)
  CONTOUR   VAL=X  COLOR=2
END
CONTOUR     VAL=16  LINE.TYP=2  COLOR=2

$ Plot interstitials and vacancies vs. depth
SELECT      Z=INTER/CI.STAR  TITLE="Point Defects vs. Y (X=1.5)" +
                                LABEL="Normalized Defect Concentration"
PLOT.1D     X.VALUE=1.5  BOTTOM=-1  RIGHT=80.0  COLOR=4
SELECT      Z = VACAN/CV.STAR
PLOT.1D     X.VALUE=1.5  ^AXES  ^CLEAR  LINE.TYP=2  COLOR=6

$ Label the plot
LABEL      X=5  Y=4.0  LABEL=Interstitials
LABEL      X=4  Y=0.7  LABEL=Vacancies

$ Plot interstitials and vacancies vs. width
SELECT      Z=INTER/CI.STAR  TITLE="Point Defects vs. X (Y=2um)" +
                                LABEL="Normalized Defect Concentration"
PLOT.1D     Y.VAL=2  BOTTOM=-1  COLOR=4
SELECT      Z=VACAN/CV.STAR
PLOT.1D     Y.VAL=2  ^AX  ^CL  LINE.TYP=2  COLOR=6
LABEL      X=0.2  Y=4.5  LABEL=Interstitials
LABEL      X=0.2  Y=0.5  LABEL=Vacancies

$ Prepare to plot contours of point defect concentrations
SELECT      Z=INTER/CI.STAR  TITLE="Interstitial Contours"
PLOT.2D     SCALE  Y.MAX=1.5  X.MAX=2.1

$ Plot contours of interstitials
FOREACH     I ( 0 to 7 )
  COLOR MIN.V=(( I -0.5)*1.0+1.5)  MAX.V=(( I +0.5)*1.0+1.5)  COLOR=(9 + I )
END

$ Create a legend
LABEL X=1.68 Y=0.15 LABEL="I/I*:" SIZE=.3
LABEL X=1.8 Y=0.30 LABEL="1.5" SIZE=.3 W.RECT=.35 H.R=.35 C.R=9
LABEL X=1.8 Y=0.45 LABEL="2.5" SIZE=.3 W.RECT=.35 H.R=.35 C.R=10
LABEL X=1.8 Y=0.60 LABEL="3.5" SIZE=.3 W.RECT=.35 H.R=.35 C.R=11
LABEL X=1.8 Y=0.75 LABEL="4.5" SIZE=.3 W.RECT=.35 H.R=.35 C.R=12
LABEL X=1.8 Y=0.90 LABEL="5.5" SIZE=.3 W.RECT=.35 H.R=.35 C.R=13
LABEL X=1.8 Y=1.05 LABEL="6.5" SIZE=.3 W.RECT=.35 H.R=.35 C.R=14
LABEL X=1.8 Y=1.20 LABEL="7.5" SIZE=.3 W.RECT=.35 H.R=.35 C.R=15
LABEL X=1.8 Y=1.35 LABEL="8.5" SIZE=.3 W.RECT=.35 H.R=.35 C.R=16

$ Redraw the boundaries
PLOT.2D     ^AX  ^CL

$ Label the plot
LABEL      X=1.4  Y= 0.0  LABEL=Oxide  RIGHT

$ Print boron vs. depth in field region
SELECT      Z=BORON
PRINT.1D    X.VALUE=0  X.MAX=2

```

Figure 4-15 Second part of input file *s4ex2b.inp*, showing statements for plotting the results of the diffusion simulation

The **CONTOUR** statement plots a contour at a single value of the selected quantity, so a series of **CONTOUR** statements is needed for a typical contour plot. The specification of these **CONTOUR** statements is simplified by using the **FOREACH** and **END** statements to define a loop.

Using the **FOREACH** Statement

The **FOREACH** statement is special in that it does not use the usual “parameter=value” type of syntax. Instead it requires a variable name followed by a list of values in parentheses. The variable name can have up to 8 characters and must start with a letter. The values in the list can be numbers or other syntactic items (e.g., character strings) and must be separated by spaces. When the **FOREACH** loop is executed, each of the following statements up to the matching **END** statement is executed with the variable replaced by successive values from the **FOREACH** list wherever it occurs.

In this example the variable *X* takes on the values 15.5, 16.5, and 17. The variable *X* is used as the value of the **VALUE** parameter in the **CONTOUR** statement to generate a contour at each of the values listed in the **FOREACH** statement. The value 16 is purposely omitted from the list so that the contour could be drawn with a separate **CONTOUR** statement using **LINE.TYP=2**. Thus the contour for a boron concentration of 10^{16} is drawn with a dashed line to distinguish it from the other contours. The resulting plot is shown in [Figure 4-16](#).

Contours of Boron Concentration

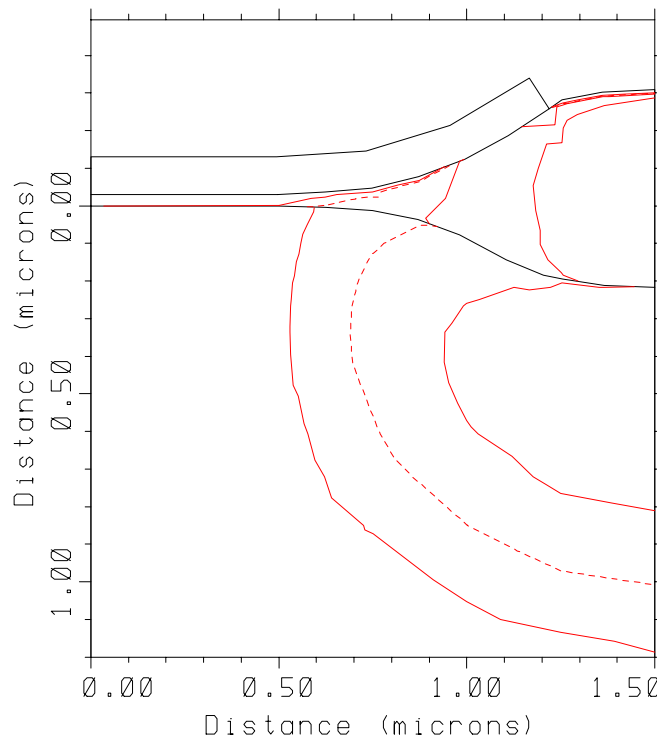


Figure 4-16 Contours of boron concentration produced by input file *s4ex2b.inp*

Vertical Distribution of Point Defects

The distribution of interstitials and vacancies is examined next. By understanding the behavior of these point defects, more accurate and efficient simulations may be set up. It is usually most convenient to normalize the interstitial and vacancy concentration to their equilibrium values, since it is the normalized quantities that are used in the equations for impurity diffusion. Thus, in the **SELECT** statements, divide the interstitial and vacancy concentrations by *CI.STAR* and *CV.STAR*, the equilibrium concentrations for interstitials and vacancies, respectively.

The first **PLOT . 1D** statement plots the distribution of interstitials along a vertical section at **X . VALUE=0**. The maximum value on the *x* axis (which corresponds to depth, in this case) is given by **RIGHT**. The minimum value on the *y* axis is set to -1 with the **BOTTOM** parameter. The normalized vacancy concentration is plotted on the same graph, using line type 2 and color 6. The **LABEL** statements help us to remember which curve is which on the plot.

The resulting plot, shown in Figure 4-17, reveals several interesting things about the point defect concentrations during oxidation:

- The point defect profiles extend to a much greater depth than the dopant profiles. Thus a deeper simulation structure is needed, although in this case a structure only 50 to 100 microns deep would have sufficed.
- The oxidation has produced a greatly enhanced interstitial concentration near the surface, and a greatly reduced vacancy concentration. The diffusivity of impurities that diffuse with interstitials (e.g. boron and phosphorus) is enhanced, while the diffusion of impurities that diffuse with vacancies is retarded (e.g., antimony).

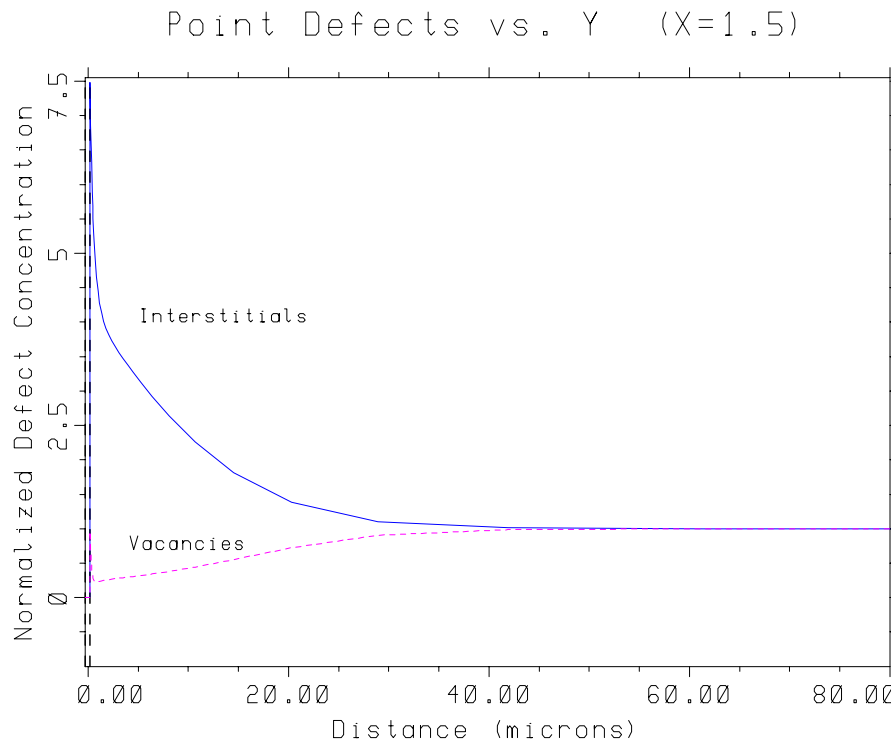


Figure 4-17 Concentration of point defects vs. depth, as plotted by input file *s4ex2b.inp*

Lateral Distribution of Point Defects

The distribution of point defects across the width of the device is examined to determine how dense the horizontal grid must be deep in the structure. The next set of **SELECT**, **PLOT . 1D**, and **LABEL** statements plot the point defect distributions across the width of the device at $y=2$ microns. The sequence of statements is the same as for the previous plot, except that **Y . VALUE=2** has been specified on the **PLOT . 1D** statements instead of **X . VALUE=0**. The results are shown in Figure 4-18.

Although visually uninteresting, this plot reveals a very important property of the point defect distributions—because point defects diffuse so rapidly, the point defect profiles deep in the structure are essentially one-dimensional. Thus it is possible to eliminate the vertical grid lines deep in the structure in order to reduce the simulation time.

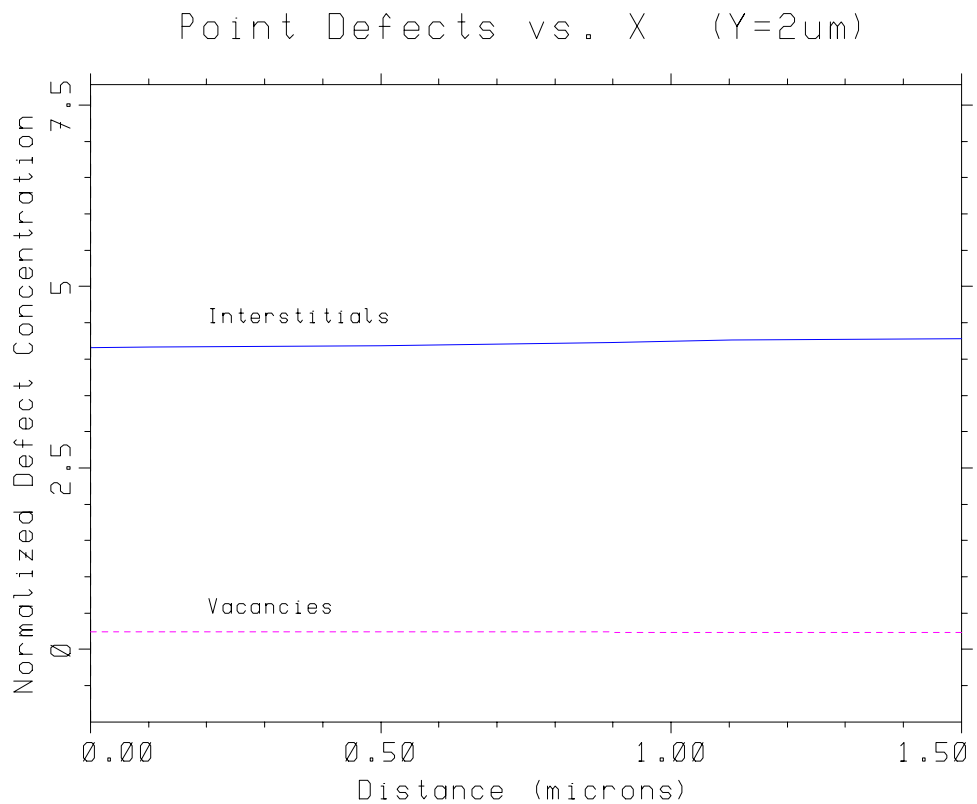


Figure 4-18 Concentration of point defects vs. width, as plotted by input file *s4ex2b.inp*

Shaded Contours of Interstitial Concentration

Finally, shaded contours of interstitial concentrations are plotted. The procedure is the same as for plotting the boron contours. The axes and material boundaries are plotted with a **PLOT . 2D** statement. In this case, **Y . MAX=2** microns is used to plot a deeper section of the device. The **SELECT** statement is used to specify the normalized interstitial concentration as the plot quantity and to give a title for the plot. Again, the general expression capability of the **SELECT** statement is used to scale the interstitial concentration by the equilibrium value (*CI.STAR*). A series of shaded contours are plotted, using the **FOREACH**, **COLOR**, and **END** statements.

LABEL statements with shaded rectangles are used to create a legend for the plot. A final **PLOT . 2D** with **^AXES** and **^CLEAR** is used to replot the boundaries of the structure. The final plot is shown in [Figure 4-19](#).

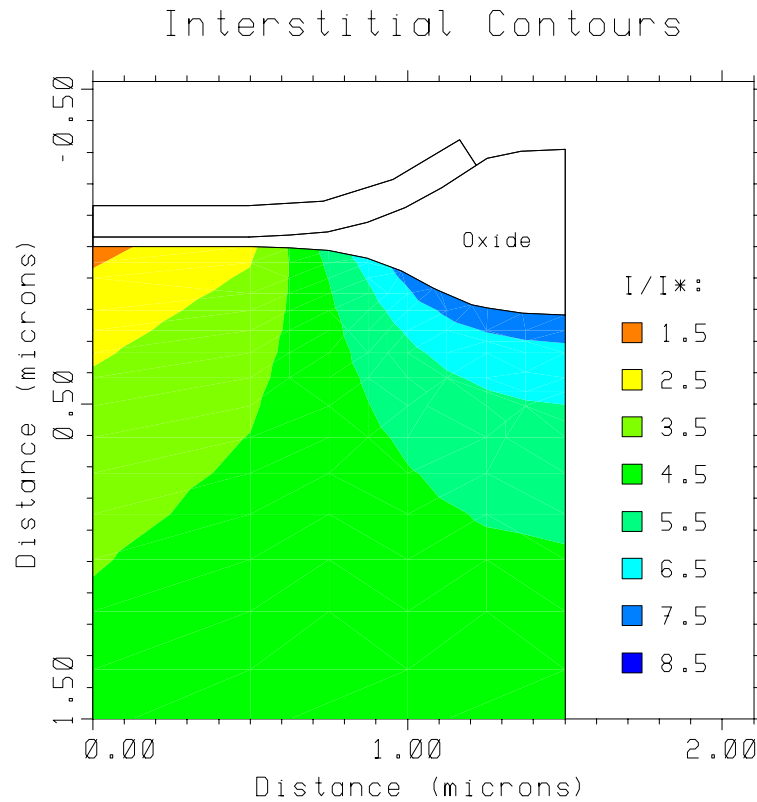


Figure 4-19 Contours of interstitial concentration, as plotted by input file *s4ex2b.inp*

Local Oxidation Summation

This section has shown how the shape of an isolation structure and the corresponding impurity profiles can be determined by separate simulations. In many cases this is the most efficient way to obtain an accurate simulation, because it allows the grids and solution methods to be optimized for each aspect of the problem. Adjustments to the process can be analyzed more quickly; often it is only necessary to repeat one of the simulations. As a final check, the two simulations could be combined, using a somewhat denser mesh. This solution would be more time-consuming, but a single simulation should suffice to verify the results obtained previously.

Point Defect Models

This example shows the differences in results using the three point defect models:

- The **PD.FERMI** model is the simplest point defect model; it is the fastest of the three, because it assumes that the point defect concentrations are at equilibrium and thus do not need to be calculated explicitly.

The **PD.FERMI** model does not simulate the effects of nonequilibrium point defect concentrations on impurities, nor the effects of impurities on point defect concentrations (except for the dependence on Fermi level).

- The **PD.TRANS** model simulates the generation, diffusion, and recombination of point defects in two dimensions; it is slower, but is necessary for simulation of oxidation-enhanced and transient-enhanced diffusion.

The **PD.TRANS** model simulates the effects of nonequilibrium point defect concentrations on impurity diffusion, but does not simulate the effects of impurity diffusion on the point defect profiles.

- The **PD.FULL** model simulates both the effects of nonequilibrium point defect profiles on impurity diffusion and the impact of impurity diffusion on the point defect distributions.

The **PD.FULL** model is required for accurate simulation of high-concentration effects (e.g., phosphorus kink and tail and emitter push) and is often the best model for simulation of transient diffusion enhancement caused by implantation damage.

```

$ TSUPREM-4 -- Example 3
$
$ Simulate the diffusion of impurities and point defects using the various
$ defect models.

MESH          GRID.FAC=0.5

INITIALIZE <100>  BORON=1E10

$ Implant phosphorus and boron
IMPLANT  PHOSPHORUS DOSE=2.0E15  ENERGY=50
IMPLANT  BORON      DOSE=1.0E13  ENERGY=120

$ Specify the point defect model (no point defects)
METHOD   PD.FERMI

$ Save the structure
SAVEFILE OUT.FILE=S4EX3S

$ Perform the diffusion using the Fermi point defect model
DIFFUSION TEMP=900  TIME=20  DRYO2

$ Plot 1-D profiles of concentration of boron and antimony (Fermi)
SELECT   Z=LOG10(PHOSPHORUS) TITLE="Comparison of Point Defect Models" +
        LABEL=log10(concentration)
PLOT.1D  RIGHT=1.2  TOP=21  BOTTOM=14  LINE.TYP=3  COLOR=3  SYMBOL=5
SELECT   Z=LOG10(BORON)
PLOT.1D  ^AXES  ^CLEAR  LINE.TYP=3  COLOR=2  SYMBOL=4

```

Figure 4-20 First part of input file *s4ex3.inp*, showing processing and plotting using the **PD.FERMI** point defect model

The input file *s4ex3.inp*, shown in Figures 4-20 and 4-21, demonstrates some of the differences among the point defect models. It also illustrates a procedure for simulating alternative processing sequences. In this case, alternative models are being examined, but the procedure can also be used to simulate run splits.

```

$ Read in the structure
LOADFILE  IN.FILE=S4EX3S

$ Perform the diffusion again using the two-dimensional point defect model
METHOD    PD.TRANS
DIFFUSION TEMP=900  TIME=20  DRYO2

$ Add profiles of phosphorus and boron (from the PD.TRANS model) to first
plot
SELECT    Z=LOG10(PHOSPHORUS)
PLOT.1D   ^AXES ^CLEAR LINE.TYP=2  COLOR=3  SYMBOL=3
SELECT    Z=LOG10(BORON)
PLOT.1D   ^AXES ^CLEAR LINE.TYP=2  COLOR=2  SYMBOL=6

$ Read in the structure
LOADFILE  IN.FILE=S4EX3S

$ Perform the diffusion using the full point defect model
METHOD    PD.FULL
DIFFUSION TEMP=900  TIME=20  DRYO2

$ Add results to previous plot
SELECT    Z=LOG10(PHOSPHORUS)  TITLE="Comparison of Point Defect Models" +
                                LABEL=log10(concentration)
PLOT.1D   ^AXES ^CLEAR LINE.TYP=1  COLOR=3  SYMBOL=1
SELECT    Z=LOG10(BORON)
PLOT.1D   ^AXES ^CLEAR LINE.TYP=1  COLOR=2  SYMBOL=2

$ Label the line types
LABEL     X=0.8  Y=20.5  LABEL="Phosphorus (PD.FERMI)" +
                                LINE.TYP=3  C.LINE=3  SYMBOL=5  C.SYMB=3
LABEL     X=0.8  Y=20.1  LABEL="Phosphorus (PD.TRANS)" +
                                LINE.TYP=2  C.LINE=3  SYMBOL=3  C.SYMB=3
LABEL     X=0.8  Y=19.7  LABEL="Phosphorus (PD.FULL)" +
                                LINE.TYP=1  C.LINE=3  SYMBOL=1  C.SYMB=3
LABEL     X=0.8  Y=19.3  LABEL="Boron (PD.FERMI)" +
                                LINE.TYP=3  C.LINE=2  SYMBOL=4  C.SYMB=2
LABEL     X=0.8  Y=18.9  LABEL="Boron (PD.TRANS)" +
                                LINE.TYP=2  C.LINE=2  SYMBOL=6  C.SYMB=2
LABEL     X=0.8  Y=18.5  LABEL="Boron (PD.FULL)" +
                                LINE.TYP=1  C.LINE=2  SYMBOL=2  C.SYMB=2

$ Print junction locations
SELECT    Z=DOPING
PRINT.1D  SPOT=0  LAYERS

$ Plot 1-D profile of interstitials
SELECT    Z=INTER/CI.STAR  TITLE="Point Defect Profiles" +
                                LABEL="Normalized Concentration"
PLOT.1D   RIGHT=20.0  COLOR=4

$ Add 1-D profile of vacancies to second graph
SELECT    Z=VACAN/CV.STAR
PLOT.1D   ^AXES ^CLEAR LINE.TYP=2  COLOR=6

$ Label the plot
LABEL     X=5.5  Y=10  LABEL="Interstitials"
LABEL     X=1  Y=1.0  LABEL="Vacancies"

```

Figure 4-21 Second part of input file *s4ex3.inp*, using the full two-dimensional point defect model

Creating the Test Structure

To create the test structure use the following directions.

Automatic Grid Generation

Because no **LINE** statements are specified in this example, a grid is generated automatically. In the x direction, two vertical lines are generated, at $x=0$ and $x=1$ micron. In the y direction, a more complicated grid is generated. The details of this grid are specified by a **MESH** statement in the *s4init* file (see [Chapter 2, Automatic Grid Generation on page 2-5](#), [Explicit Specification of Grid Structure on page 2-3](#), and [Appendix A](#)). The default automatic grid extends to a depth of 200 microns to accommodate the deep diffusion of interstitials and vacancies.

The **GRID.FAC** parameter on the **MESH** statement provides an easy way to increase or decrease the grid spacings throughout a simulation. In this example, all grid spacings are multiplied by a factor of 0.5, doubling the grid density. The **GRID.FAC** parameter makes it easy to determine how the accuracy of the simulation depends on the grid spacing: simply decrease the value of **GRID.FAC** until the changes in the simulated result become insignificant.

Outline of Example

This example compares the three point defect models for two impurities, phosphorus and boron. A large implanted dose of phosphorus is used to demonstrate high-concentration effects. A smaller and deeper boron implant is used to monitor coupling between impurities caused by point defects. A dry oxidizing ambient is used to drive in the impurities. The oxidation produces point defects that enhance the impurity diffusion. The oxide layer also prevents impurities from escaping from the surface of the structure. After performing the implants, the structure is saved. By reloading the saved structure the simulation can be resumed from this point as many times as desired.

Oxidation and Plotting of Impurity Profiles

- Simulation procedure
- **PD.FERMI** and **PD.TRANS** models
- **PD.FULL** model
- Printing junction depth
- Doping and layer information

Simulation Procedure

The oxidation (20 minutes at 900° at a dry oxygen ambient) using the **PD.FERMI** model is first simulated. Because the **PD.FERMI** model is used, the effects of oxidation on the impurity diffusion are not simulated. The results of diffusion with the **PD.FERMI** model are plotted with a pair of **SELECT/PLOT.1D** sequences. A single plot is used for the phosphorus and boron concentrations produced by both models, using various different line types, symbols, and colors to distinguish the different profiles. Next, the saved structure is restored, using the **LOADFILE** statement, **PD.TRANS** model is selected. The oxidation step is repeated and the resulting profiles obtained using the **PD.TRANS** model are added to the plot. The

procedure is repeated, loading the saved structure and simulating the diffusion, this time with the **PD.FULL** model. Finally, **LABEL** statements are added to create a legend, showing the significance of the different line types and symbols.

PD.FERMI and PD.TRANS Models

The final plot is shown in Figure 4-22. The lines with the longer and shorter dashes show the profiles obtained with the **PD.FERMI** and **PD.TRANS** models, respectively. The phosphorus profiles have flat tops and steep tails (the gentler tail beyond $y=0.4$ microns is due to channelling during the implant), and the boron profiles are symmetrical. The effect of oxidation-enhanced diffusion is significant, increasing the junction location from about $y=0.26$ microns when the **PD.FERMI** model is used to about $y=0.36$ microns with the **PD.TRANS** model.

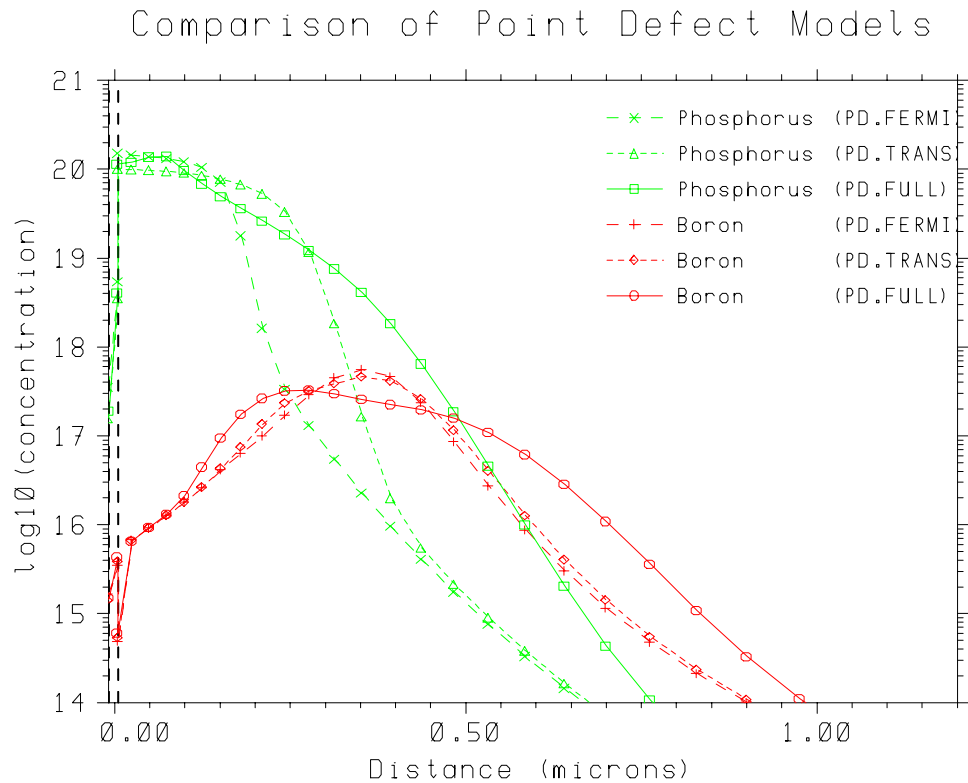


Figure 4-22 Profiles with **PD.FERMI** and **PD.FULL** models, from *s4ex3.inp*

PD.FULL Model

The solid lines show the results obtained with the **PD.FULL** model. With this model, the phosphorus profile is no longer flat on top and has developed an extended tail, while the boron profile has lost its symmetry. Both impurities have diffused significantly farther than they did with the **PD.TRANS** model, with the junction location increasing to about $y=0.51$ microns. All of these effects are caused by the two-way interactions between impurities and point defects: Near the peak of the phosphorus profile, interstitials interact with substitutional phosphorus atoms to form interstitial-phosphorus pairs. (Actually, they may result in phosphorus atoms in interstitial sites, but the result is the same.) The pairs diffuse into the substrate where they break up, leaving a substitutional phosphorus atom and an interstitial. The excess interstitials introduced into the silicon substrate by this

process then contributes to the enhanced diffusion observed in both the boron profile and the phosphorus tail. The removal of interstitials from the region of the phosphorus peak reduces the effective diffusivity in this area, creating a kink between the region of retarded diffusivity near the peak and the region of enhanced diffusion around the tail of the profile.

Printing Junction Depth

Plots such as the one in Figure 4-22 are often used to determine the approximate junction locations in simulated structures. In many cases, however, a more accurate value is needed than that which can be obtained from the plot. A convenient way get a more accurate value is with the **PRINT .1D** statement. To obtain the junction depth, specify *DOPING* on the **SELECT** statement and then use **PRINT .1D** with the **SPOT** or **LAYERS** parameters. In this case, both are used, the results being shown in Figure 4-23. The **SPOT** parameter requests that the locations at which the selected expression has the specified value be printed. This case prints the points at which the net doping is zero, i.e., the metallurgical junctions. From Figure 4-23, observe that the junction is at $y=0.50$ in the structure.

```

$ Print junction locations
SELECT      Z=DOPING
PRINT.1D    SPOT=0  LAYERS
** Printing along X.VALUE=0:

      Value is 0 at 0.498930 microns.

  Num      Material      Top      Bottom  Thickness      Integral
  ---      -
  1         oxide      -0.0078   0.0045   0.0123      2.5782e+12
  2         silicon    0.0045   0.4989   0.4944      1.9716e+15
  3         silicon    0.4989  200.0000  199.5011     -8.1086e+11
    
```

Figure 4-23 Output produced by **PRINT .1D** statement in input file *s4ex3.inp*

Doping and Layer Information

The **LAYERS** parameter prints more information about the doping in the structure. The **LAYERS** output (included in Figure 4-23) gives the top and bottom coordinates of the oxide, the n-type surface region, and the p-type substrate. In addition, it calculates the thickness of each layer and the integral of the selected expression (doping, in this case) over each layer.



Note:

The definition of a layer depends on the selected expression—the boundaries between layers are taken to be material interfaces or the points where the selected quantity is zero.

To use **PRINT .1D LAYERS** to get junction depths, you must specify **SELECT Z=DOPING**.

Point Defect Profiles

The last step is to examine the interstitial and vacancy concentrations in the structure. Although the point defect profiles do not have a direct effect on device performance, they do aid understanding how the final doping profiles were produced.

The point defect profiles, as plotted by the last two **SELECT/PLOT .1D** pairs, are shown in Figure 4-24. Note that the interstitial and vacancy concentrations have been normalized by *CI.STAR* and *CV.STAR*, the equilibrium interstitial and vacancy concentrations, respectively. Figure 4-24 shows that the interstitial profile has a large peak about 0.4 microns below the surface. This peak is produced by the interaction with the phosphorus profile. The smaller peak at the surface is caused by injection of interstitials by the oxidation process.

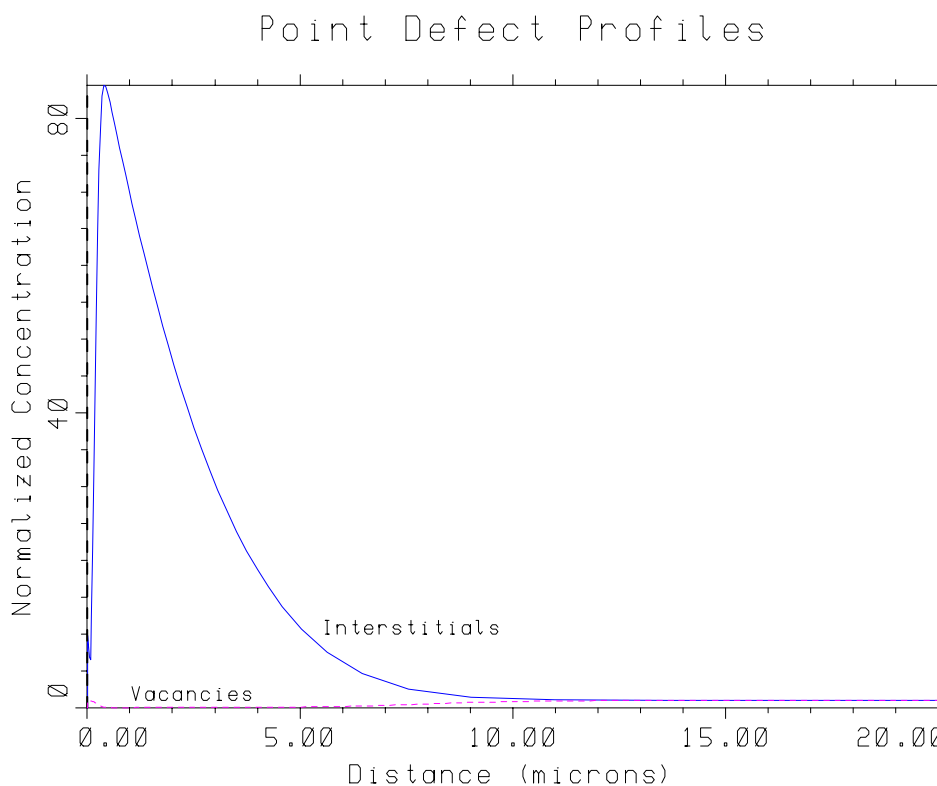


Figure 4-24 Point defect profiles plotted by *s4ex3.inp*

Commentary

Because of the extra accuracy afforded by the **PD.FULL** model, one may be tempted to use it at all times. This would be a good idea, except that the **PD.FULL** model is slower than the **PD.TRANS** model and there are many cases in which the extra accuracy obtained with the **PD.FULL** model is negligible. Similar arguments can be made in comparing the **PD.TRANS** and **PD.FERMI** models.

Choosing a Point Defect Model

The simplest way to determine which point defect model is needed in a particular simulation is by trial and error. Although it may appear easier to use **PD.FULL**, it is usually faster to set up a simulation using the **PD.FERMI** model and then switch to **PD.TRANS** or **PD.FULL** only after correct simulation of the structure and approximately correct simulation of the impurity profiles have been verified.

There are several cases in where it can be determined in advance which point defect model is most appropriate. Because point defects are generated only during oxidation or by implantation (with the **DAMAGE** parameter), there is no need to use **PD . TRANS** or **PD . FULL** before the first oxidation or implantation (with **DAMAGE**) step. During inert diffusions, any point defects present in the structure recombine and approach their equilibrium values. After a sufficiently long inert diffusion, the **PD . FERMI** model can be used without loss of accuracy.

Finally, it should be noted that the point defects affect the diffusivity of impurities. If the impurity concentrations are small, or if the gradients of the concentrations are small, there is little diffusion, and no need to use one of the more expensive point defect models. The **PD . FULL** should be considered whenever concentrations (particularly of phosphorus) are very high, or whenever it is necessary to model implant damage effects.

Advanced Examples

Overview

This chapter presents several examples that illustrate the application of TSUPREM-4 to real-world problems. Most of these examples show full, two-dimensional simulations of complete processes. Included are:

- Simulations of several specialized processes such as poly-buffered LOCOS
- A power DMOS process
- A silicon-on-insulator process

In addition, the examples demonstrate a number of techniques that are useful for a wide range of applications.



Note:

Because some of the examples in this chapter include simulations of complete processes, they require more computer time than the examples of [Chapter 4](#). The initial grids in these examples have been carefully chosen to give reasonable accuracy while minimizing computer time requirements. These are the grids that would be used for the bulk of a process design or analysis project. Simpler grids would be used for initial checkout of the simulation input file, while finer grids would be used for a final check. Similarly, the oxidation and point defect models have been chosen for reasonable accuracy. In some cases, slower but more accurate models could be used for a final check of the simulation.

NMOS LDD Process

This application presents a complete simulation of an n-channel MOS transistor with a lightly-doped drain. It illustrates how mask information extracted from a layout by Taurus Layout — IC Layout Interface can be used by TSUPREM-4.

Because this application is rather lengthy, it has been broken into four parts:

- Input file *s4ex4a.inp* simulates the growth of the field oxide and the consequent boron diffusion; the results are stored in the structure file *S4EX4AS*.
- The source/drain processing is simulated (input file *s4ex4b.inp*), and the results are stored in structure file *S4EX4BS*.
- The input file *s4ex4c.inp* reflects the half-structure about the left edge to form the complete NMOS device, saving the structure in file *S4EX4CS* and plotting contours of boron and arsenic.
- The input file *s4ex4d.inp* is used to extract electrical characteristics of the final structure.

The input statements in file *s4ex4a.inp*, shown in [Figures 5-1](#) and [5-3](#), simulate the initial portion of the lightly-doped drain NMOS process up through the gate region enhancement implant.

Creating the Initial Structure

This example uses an automatically generated, two-dimensional grid with adaptive gridding. The grid in the *x* direction is derived from mask information extracted from a layout file by Taurus Layout. The width of the grid is set equal to the width of the cut line specified in Taurus Layout. The grid is made finer near mask edges and coarser far from mask edges. Vertical grid lines are eliminated deep in the structure, to save simulation time. The grid in the *y* direction is generated automatically using the default parameters; these default parameters are set on **MESH** statements in the *s4init* file, but can be changed as needed. See [Automatic Grid Generation on page 2-5](#), [Explicit Specification of Grid Structure on page 2-3](#), and [Appendix A](#).

To automatically generate a two-dimensional grid, a **MASK** statement is needed to read the mask information produced by Taurus Layout and an **INITIALIZE** statement is used to do the grid generation. No **LINE** statements are required, and **ELIMINATE** statements are optional.


```

$ TSUPREM4 NMOS transistor simulation
$ Part a: Through field oxidation

$ Define the grid
MESH          GRID.FAC=1.5
$METHOD       ERR.FAC=2.0

$ Read the mask definition file
MASK          IN.FILE=s4ex4m.t11  PRINT  GRID="Field,Poly"

$ Initialize the structure
INITIALIZE <100>  BORON=5E15

$ Initial oxidation
DIFFUSION     TIME=30  TEMP=1000  DRY  HCL=5

$ Nitride deposition and field region mask
DEPOSIT       NITRIDE          THICKNESS=0.07  SPACES=4
DEPOSIT       PHOTORESIST      POSITIVE  THICKNESS=1
EXPOSE        MASK=Field
DEVELOP

ETCH          NITRIDE          TRAP
ETCH          OXIDE            TRAP  UNDERCUT=0.1
ETCH          SILICON          TRAP  THICKNES=0.25  UNDERCUT=0.1

$ Boron field implant
IMPLANT       BORON  DOSE=5E12  ENERGY=50  TILT=7  ROTATION=30
ETCH          PHOTORESIST  ALL

$ Field oxidation
METHOD        PD.TRANS  COMPRESS
DIFFUSION     TIME=20   TEMP=800   T.FINAL=1000
DIFFUSION     TIME=180  TEMP=1000  WETO2
DIFFUSION     TIME=20   TEMP=1000  T.FINAL=800
ETCH          NITRIDE   ALL

$ Unmasked enhancement implant
IMPLANT       BORON  DOSE=1E12  ENERGY=40  TILT=7  ROTATION=30

$ Save structure
SAVEFILE      OUT.FILE=S4EX4AS

```

Figure 5-1 First part of input file *s4ex4a.inp*: Setting up the grid for simulating an NMOS process

Setting the Grid Density

To save execution time, the recommended procedure when developing a new TSUPREM-4 simulation is to start with a very coarse grid. After an input file has been entered and is running correctly (no syntax errors, all etches in the correct places, etc.), a finer grid is used for the bulk of the simulation work. Lastly, an even finer grid is used to get final answers or to verify that additional grid points (with the associated increased execution time) do not significantly improve the accuracy of the answers.

The **GRID.FAC** parameter on the **MESH** statement makes it easy to adjust the grid density during the various stages of a simulation project. In this example, **GRID.FAC** has been set to 1.5, so that all grid spacings are 1.5 times their specified value. This gives a rather coarse grid for efficient simulations. When the

results are satisfactory with this grid, **GRID.FAC** can be decreased to 1.0 or 0.5 to check the results.

Adaptive Gridding

The **ERR.FAC** parameter on the **METHOD** statement is used in a similar way to control the accuracy criteria used for adaptive gridding. A value of 2.0 could be used to reduce the accuracy for a faster simulation, but for this example the default value of 1.0 is satisfactory. When the simulation is running correctly, this can be reduced to 0.7 or 0.5 to check the results.

Masking Information

The **MASK** statement reads mask descriptions from the file *s4ex4m.tl1*. This file, produced by **Taurus Layout**, describes the mask levels present over a cross-section of the device layout. The **PRINT** parameter specifies that the mask information should be printed after it is read. The result is shown in [Figure 5-2](#). Four mask levels, named *Field*, *Poly*, *Contact*, and *Metal* are defined. These names are used to refer to the masks as they are needed later in the processes. The **GRID** parameter specifies that only the *Field* and *Poly* layers in the mask file are considered when calculating the horizontal grid spacing; the *Contact* and *Metal* are ignored.



Note:

There are no references to mask coordinates in the TSUPREM-4 input file. The mask file contains all the layout information required by the simulation. The same input file can be used to simulate other structures simply by specifying a different mask file.

```

$ Read the mask definition file
MASK          IN.FILE=s4ex4m.tl1  PRINT

Comments from mask data file "s4ex4m.tl1":
/ Mask definition file s4ex4m.tl1, for use with s4ex4[abc].
(End of comments from mask data file)

The following masks are currently defined (locations in microns):
Name:  Field      min X:   0.0000  max X:   5.0000
      opaque between 0.0000  and 3.9000
Name:  Poly       min X:   0.0000  max X:   5.0000
      opaque between 0.0000  and 0.6500
Name:  Contact    min X:   0.0000  max X:   5.0000
      opaque between 0.0000  and 1.9500
      opaque between 3.2500  and 5.0000
Name:  Metal      min X:   0.0000  max X:   5.0000
      opaque between 1.3000  and 5.0000

```

Figure 5-2 Listing of mask information read from file *s4ex4m.tl1*

Field Isolation Simulation

1. The first step is to grow a pad oxide. The default **VERTICAL** model is sufficient for this planar oxidation step; the **PD.FERMI** model for defects (the default) is acceptable because there is no significant doping in the structure yet.

2. The next step is to define the active and field isolation regions of the structure. A typical photolithography sequence is used:
 - a. Deposit a layer to be patterned (nitride, in this case).
 - b. Deposit a a layer of photoresist (positive resist, in this example).
 - c. Expose the resist, using the appropriate mask (the *Field* mask).
 - d. Use the **DEVELOP** statement to remove the exposed photoresist.
 - e. Etch the underlying layers, using the **TRAP** etch model. The remaining photoresist serves as a mask for the etch. In this case, the nitride is etched first, then the oxide, then the silicon. The oxide and silicon etches specify undercutting of the mask layer by 0.1 and 0.25 microns, respectively.
 - f. In this example, the photoresist is also used to mask the boron field implant.
 - g. Remove the remaining photoresist.

**Note:**

Don't forget to remove the remaining photoresist at the end of the photolithography sequence.

3. The next step is the field oxidation. The **PD . TRANS** model is used, resulting in the modeling of interstitials and vacancies, and hence oxidation-enhanced diffusion; the **COMPRESS** oxidation model is also selected at this point, because all further oxidations will be nonplanar. (If the details of the bird's beak shape were important, the **VISCOEL** model with stress dependence could be selected at this point). The field oxidation is done in three steps:
 - a. For the first 20 minutes, the temperature is ramped up from 800°C to 1000°C at the rate of 10°C/minute in an inert ambient.
 - b. The second step is a wet oxidation for 180 minutes at a constant temperature of 1000°C.
 - c. Finally, the temperature is ramped down from 1000°C to 800°C at the rate of 10°C/minute in an inert ambient.
4. The final process steps specified in the first input file are the removal of the nitride oxidation mask and the implantation of boron for adjusting the n-channel threshold voltage.

The structure to this point is saved in the structure file *S4EX4AS* for use in continued simulation of the source and drain regions.

Displaying the Plot

Figure 5-3 shows the input statements for displaying the results thus far. The first **PLOT . 2D** statement plots the grid at this point (**Figure 5-4**). As planned, the grid is fairly coarse; some refinement by adaptive gridding is evident near the surface and around the field oxide. The structure at this point is shown in **Figure 5-5**. Contour lines of boron concentration are plotted starting at a concentration of 10^{15} and extending up to a concentration of 10^{20} atoms/cm³ in half-decade steps.

```

$ Plot the initial NMOS structure
SELECT      Z=LOG10(BORON)  TITLE="LDD Process - NMOS Isolation Region"
PLOT.2D     SCALE GRID C.GRID=2 Y.MAX=2.0
PLOT.2D     SCALE  Y.MAX=2.0

$ Color fill the regions
COLOR      SILICON  COLOR=7
COLOR      OXIDE    COLOR=5

$ Plot contours of boron
FOREACH     X (15 TO 20 STEP 0.5)
  CONTOUR   VALUE=X  LINE=5  COLOR=2
END

$ Replot boundaries
PLOT.2D    ^AX ^CL

$ Print doping information under field oxide
SELECT      Z=DOPING
PRINT.1D   X.VALUE=4.5  X.MAX=3

```

Figure 5-3 Second part of input file *s4ex4a.inp*, for simulating an NMOS process

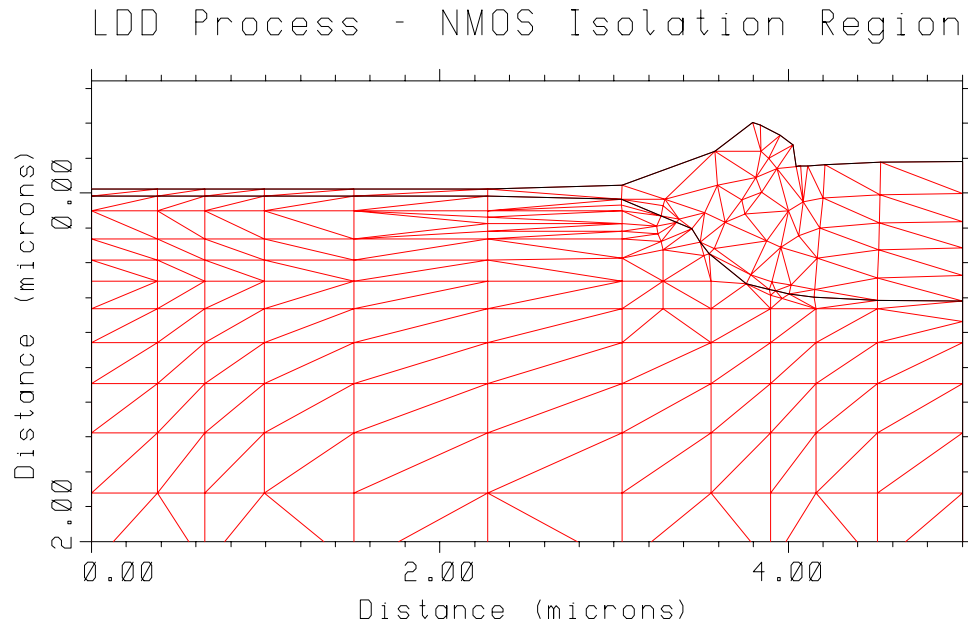


Figure 5-4 Grid after formation of isolation region, plotted by *s4ex4a.inp*

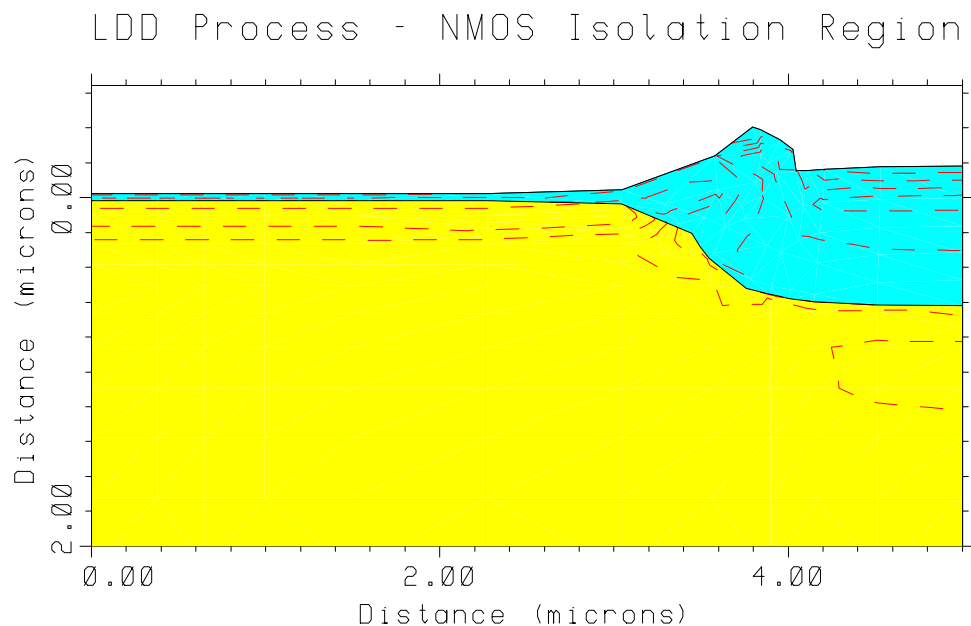


Figure 5-5 Structure with contours of boron concentration, after formation of isolation region, as plotted by file *s4ex4a.inp*

Active Region Simulation

The input statements in the file *s4ex4b.inp* complete the processing of the NMOS transistor. The listing of this input file is presented in Figures 5-6 and 5-8. The starting structure for this stage of the simulation is read from the file *S4EX4AS* generated by the simulation input file *s4ex4a.inp*. The grid spacing and error control factors are not saved in the structure file, so they must be reset to their desired values with the **MESH** and **METHOD** statements, respectively. Similarly, the mask information from Taurus Layout must be reread with the **MASK** statement.

The first process steps in this input file define and oxidize a polysilicon gate. The *Poly* mask is used in a typical photolithography sequence (i.e., **DEPOSITION**, **EXPOSE**, **DEVELOP**, and **ETCH**). The **COMPRESS** model is used to simulate the oxidation of the polysilicon gate and the **PD.TRANS** model is used for diffusion. These models are not explicitly specified in input file *s4ex4b.inp*; rather, they are set in *s4ex4a.inp*, saved (automatically) in the structure file *S4EX4AS*, and then read in by the **INITIALIZE** statement in *s4ex4b.inp*.

Modeling Polysilicon

The statement **MATERIAL MAT=POLY ^POLYCRYS** disables the advanced models for grain growth and impurity diffusion in polysilicon (to reduce the computation time). For studying the details of impurity diffusion and activation in the gate, this statement would be omitted. Also, a finer grid would be used in the poly, either by specifying more grid spaces in the **DEPOSITION** statement or by reducing the relative error for adaptive gridding in polysilicon (i.e., by specifying a smaller value for **REL.ADAP** for impurities in poly).

```

$ TSUPREM4 NMOS transistor simulation
$ Part b: Through source/drain metallization

$ Set grid spacing and accuracy parameters
MESH      GRID.FAC=1.5
$METHOD   ERR.FAC=2.0

$ Read structure from initial simulation
INITIAL   IN.FILE=S4EX4AS

$ Read the mask definition file
MASK      IN.FILE=s4ex4m.tll

$ Define polysilicon gate
MATERIAL  MAT=POLY ^POLYCRYS
DEPOSIT   POLYSILICON THICK=0.4 SPACES=2
DEPOSIT   PHOTORESIST THICK=1.0
EXPOSE    MASK=Poly
DEVELOP
ETCH      POLYSILICON TRAP THICK=0.7 ANGLE=79
ETCH     PHOTORESIST ALL

$ Oxidize the polysilicon gate
DIFFUSION TIME=30 TEMP=1000 DRYO2

$ LDD implant at a 7-degree tilt
IMPLANT  ARSENIC DOSE=5E13 ENERGY=50 TILT=7.0 ROTATION=30 IMPL.TAB=ARSENIC

$ Plot structure
SELECT   Z=LOG10(BORON) TITLE="LDD Process - After LDD Implant"
PLOT.2D  SCALE Y.MAX=2.0

$ Add color fill
COLOR    SILICON COLOR=7
COLOR    OXIDE   COLOR=5
COLOR    POLY    COLOR=3

$ Plot contours
FOREACH  X (15 TO 18 STEP 0.5)
  CONTOUR VALUE=X LINE=5 COLOR=2
END
SELECT   Z=LOG10(ARSENIC)
FOREACH  X (16 TO 20)
  CONTOUR VALUE=X LINE=2 COLOR=4
END

$ Replot boundaries
PLOT.2D  ^AX ^CL

```

Figure 5-6 First part of input file *s4ex4b.inp*, showing polysilicon gate formation

LDD Implant

Next, the lightly doped source/drain region extension is implanted. Because the implantation is through an oxide, **IMPL.TAB=arsenic** is used instead of the default dual-Pearson implant moments (the *dual.ars* tables) with channelling tails. The mesh nodes needed to resolve the implanted profile are added automatically through adaptive gridding. A two-dimensional plot of the structure, showing con-

tours of boron and arsenic concentration after the LDD implant, is shown in [Figure 5-7](#).

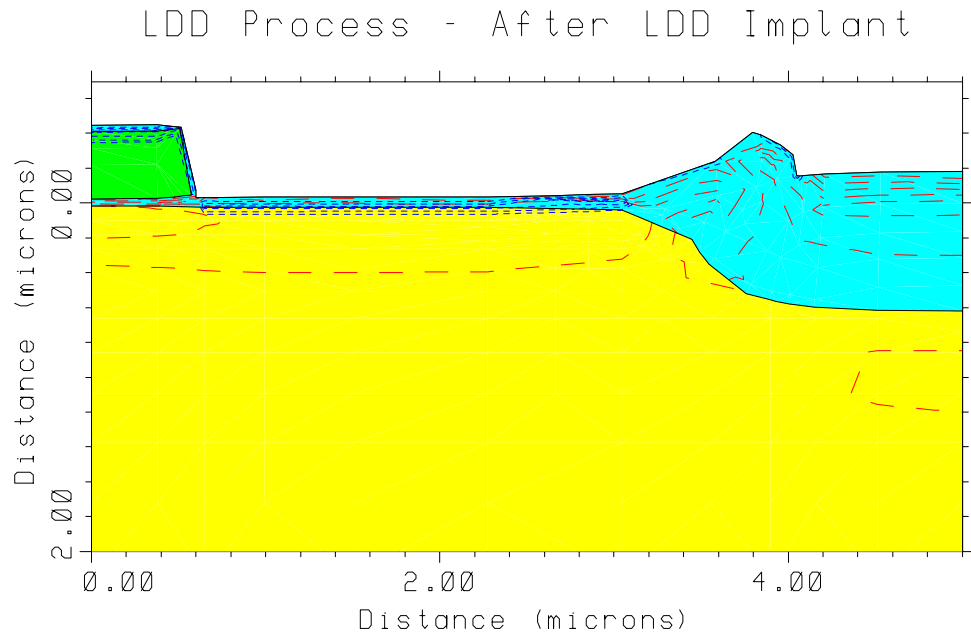


Figure 5-7 NMOS structure after LDD implant, as plotted by file *s4ex4b.inp*

Oxide Spacer and Source/Drain Implant

The remainder of the simulation input appears in [Figure 5-8](#). The next two steps define an oxide sidewall spacer. The **DEPOSITION** statement adds a conformal layer of oxide 0.4 microns thick, while the **ETCH TRAP** statement removes all oxide within a vertical distance of 0.45 microns from the surface. As a result, the oxide is removed where the surface is planar, but a spacer oxide remains on the sidewalls of the poly gate. This is followed by the implantation of arsenic to form the heavily doped source/drain regions. For this implant into bare silicon, the default implant tables, which include channelling are used. A 15-minute anneal at 950°C is used to activate the arsenic implants.

Source/Drain Contacts

The last two masks are used to locate the source/drain contacts and to pattern the aluminum. A layer of BPSG is used as an insulator between the aluminum and the structure beneath it.

```

$ Define the oxide sidewall spacer
DEPOSIT  OXIDE  THICK=0.4
ETCH     OXIDE  THICK=0.45  TRAP

$ Heavy S/D implant at a 7-degree tilt
IMPLANT  DOSE=1E15  ENERGY=200  ARSENIC  TILT=7.0  ROTATION=30

$ Anneal to activate the arsenic
DIFFUSION TIME=15  TEMP=950

$ Deposit BPSG and cut source/drain contact holes
DEPOSIT  OXIDE  THICKNES=0.7
DEPOSIT  PHOTORESIST  POSITIVE  THICKNESS=1.0
EXPOSE   MASK=Contact
DEVELOP
ETCH     OXIDE  THICKNESS=1.0  TRAP  ANGLE=75
ETCH     PHOTORESIST  ALL

$ Define the metallization
DEPOSIT  ALUMINUM  THICKNESS=1.0
DEPOSIT  PHOTORESIST  POSITIVE  THICKNESS=1.0
EXPOSE   MASK=Metal
DEVELOP
ETCH     ALUMINUM  TRAP  THICKNESS=1.5  ANGLE=75
ETCH     PHOTORESIST  ALL

$ Save the final structure
SAVEFILE  OUT.FILE=S4EX4BS

$ Plot the half NMOS structure
SELECT   Z=LOG10(BORON)  TITLE="LDD Process - Half of NMOS Structure"
PLOT.2D  SCALE  Y.MAX=2.0  GRID C.GRID=2
PLOT.2D  SCALE  Y.MAX=2.0

$ Color fill
COLOR    SILICON  COLOR=7
COLOR    OXIDE    COLOR=5
COLOR    POLY     COLOR=3
COLOR    ALUM     COLOR=2

$ Plot contours
FOREACH  X (15 TO 18 STEP 0.5)
  CONTOUR  VALUE=X  LINE=5  COLOR=2
END
SELECT   Z=LOG10(ARSENIC)
FOREACH  X (15 TO 20)
  CONTOUR  VALUE=X  LINE=2  COLOR=4
END

$ Replot boundaries
PLOT.2D  ^AX  ^CL

$ Print doping through drain
SELECT   Z=DOPING
PRINT.1D  LAYERS  X.VALUE=2

```

Figure 5-8 Second part of input file *s4ex4b.inp*, showing source/drain processing and metallization

Plots The final grid is plotted in Figure 5-9. Adaptive gridding has produced a fine grid where it is needed: in the source/drain region and particularly in the lightly doped extension. A two-dimensional contour plot of the boron and arsenic concentrations (Figure 5-10) shows that the shallow extension of the source/drain region is well defined. The results of the drain region simulation are saved with the **SAVEFILE** statement in the structure file *S4EX4BS*. This file is used as the basis for forming the complete NMOS device.

LDD Process - Half of NMOS Structure

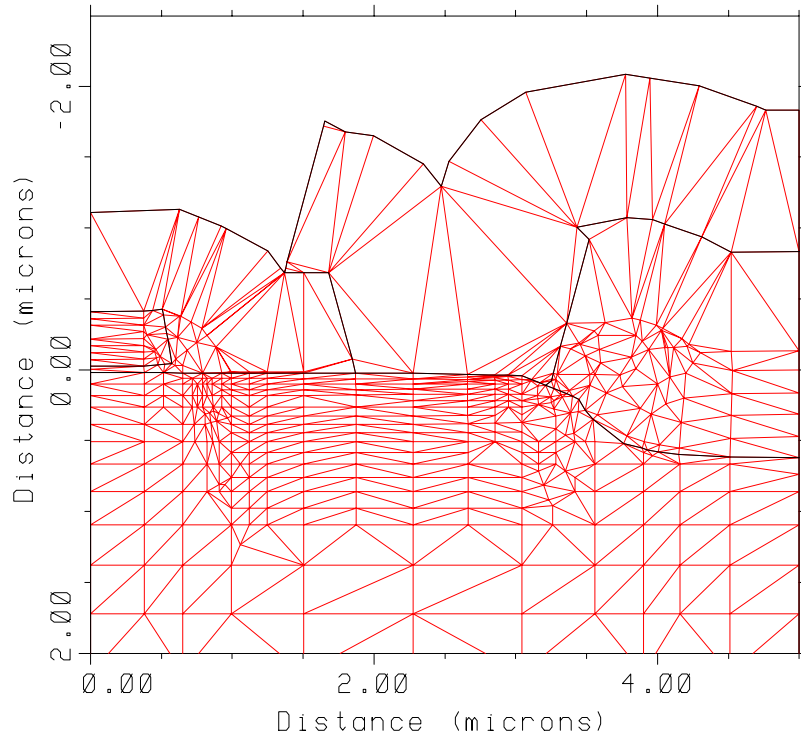
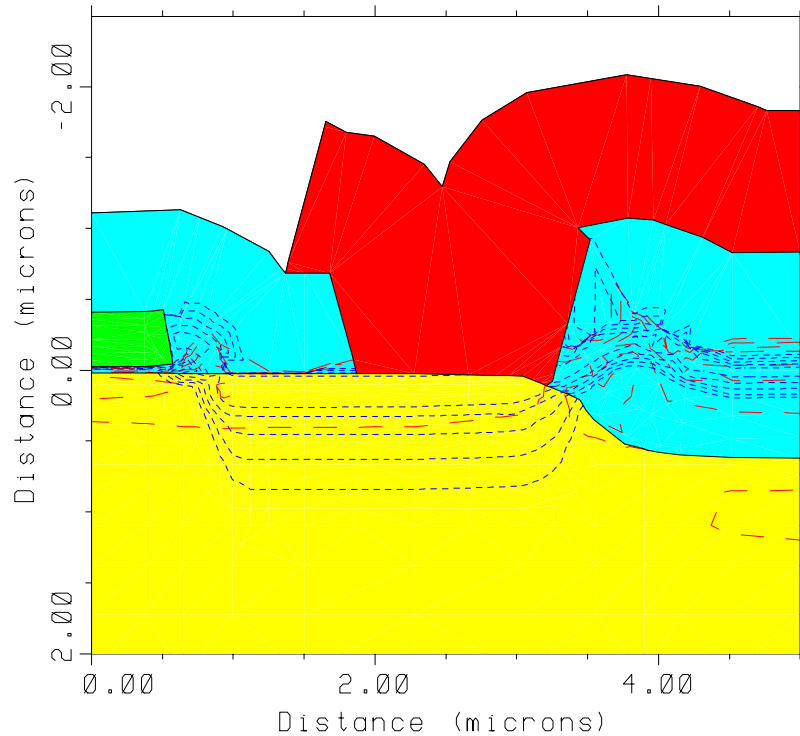


Figure 5-9 Final grid for LDD NMOS example, produced by input files *s4ex4a.inp* and *s4ex4b.inp*

LDD Process - Half of NMOS Structure

Figure 5-10 Final NMOS structure, as plotted by file *s4ex4b.inp*

Formation of the Complete NMOS Transistor

The complete NMOS transistor is formed by reading in the right half of the structure from the previous simulation and reflecting it about its left edge. The final structure is then plotted, and the various material regions are shaded and labeled. Contours of boron (long dashes) and arsenic (medium dashes) are plotted. The completed structure is saved in file *S4EX4CS*. The input statements for doing this are shown in [Figure 5-11](#); the resulting plot is shown in [Figure 5-12](#).

```

$ TSUPREM4 NMOS transistor simulation
$ Part c: Formation of complete structure

$ Read right half of structure
INITIAL IN.FILE=S4EX4BS

$ Reflect about the left edge to form the complete structure
STRUCTURE REFLECT LEFT

$ Plot the complete NMOS structure
SELECT Z=LOG10(BORON) TITLE="Example 4 - Complete NMOS Structure"
PLOT.2D SCALE Y.MAX=2.0 Y.MIN=-3.0

$ Color fill
COLOR SILICON COLOR=7
LABEL X=-4.1 Y=-2.5 LABEL="Silicon" SIZE=.3 C.RECT=7 W.RECT=.4 H.R=.4
COLOR POLYSILI COLOR=3
LABEL X=-1.8 Y=-2.5 LABEL="Polysilicon" SIZE=.3 C.RECT=3 W.RECT=.4 H.R=.4
COLOR OXIDE COLOR=5
LABEL X=1.2 Y=-2.5 LABEL="Oxide" SIZE=.3 C.RECT=5 W.RECT=.4 H.R=.4
COLOR ALUMINUM COLOR=2
LABEL X=3.2 Y=-2.5 LABEL="Aluminum" SIZE=.3 C.RECT=2 W.RECT=.4 H.R=.4

$ Plot contours
FOREACH X (15 16 17 18)
  CONTOUR VAL=X LINE=5 COLOR=2
END
SELECT Z=LOG10(ARSENIC)
FOREACH X (15 16 17 18 19 20)
  CONTOUR VAL=X LINE=3 COLOR=4
END

$ Replot boundaries
PLOT.2D ^AX ^CL

SAVEFILE OUT.FILE=S4EX4CS
    
```

Figure 5-11 Input file *s4ex4c.inp*, for plotting the final LDD NMOS structure
 Example 4 - Complete NMOS Structure

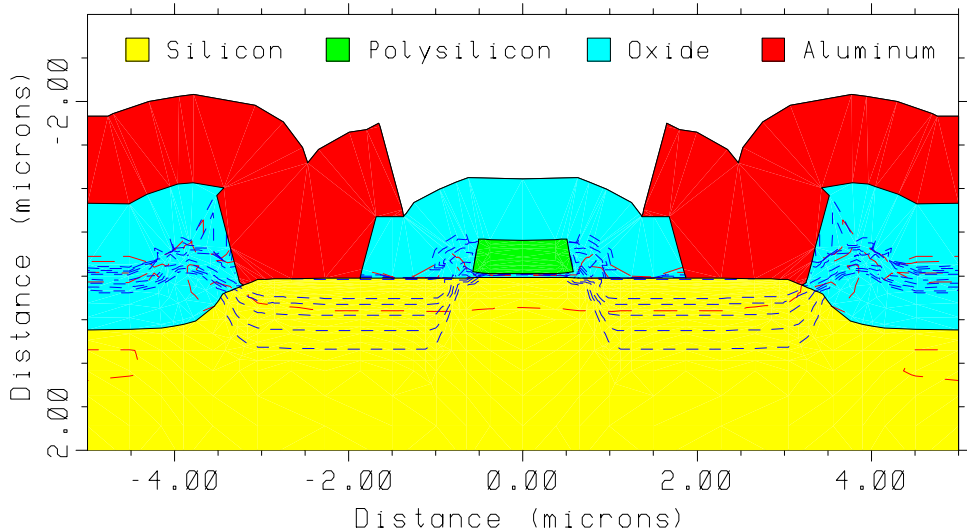


Figure 5-12 Complete NMOS structure, plotted by input file *s4ex4c.inp*

Electrical Extraction

This example shows the extraction of the following electrical information:

- Threshold voltage
- Gate capacitance
- Source/Drain junction capacitance

The statements for performing the electrical extraction are in the input file *s4ex4d.inp*.

Electrical extraction is specified by the **ELECTRICAL** statement. Results are calculated along a vertical slice through the device specified by the value of the **X** parameter. The type of extraction is specified by the **THRESHOLD**, **MOSCAP**, **JCAP**, and **RESISTAN** parameters.

Threshold Voltage

The **THRESHOLD** parameter is used to extract the threshold voltage. The **NMOS** parameter specifies that the type of MOS transistor is NMOS. The position **X** is located at the center of the gate. The gate voltage **V** is stepped from 0 to 2 volts in 0.1 volt increments. The other regions (i.e., the source, drain, and bulk) are grounded. The surface state density **QSS** defaults to $1 \times 10^{10}/\text{cm}^2$.

Because TSUPREM-4 solves the one-dimensional Poisson's equation, the extracted quantity is not drain current, but sheet conductance of the channel. However, the drain current can be approximately calculated from the sheet conductance, assuming that the channel is long and wide so that small-geometry effects can be neglected. This example supposes that NMOS with channel length of 1.2 microns and width of 25 microns is measured and that the drain is biased to 0.1 volt to minimize lowering of the energy barrier in the channel region. The scaling factor is the drain voltage multiplied by W/L. **ASSIGN** statements are used to set the values of length, width, and drain voltage and to calculate the scale factor.

The shift in the threshold voltage due to the body effect is examined by changing the back bias. The value of the body-effect parameter is approximately equal to the difference between the threshold voltages at zero and 2.5 volts back bias. This example shows approximately 0.9 volts threshold voltage shift and approximately $1.0 \text{ V}^{1/2}$ body effect.

The input statements for extracting and plotting the threshold characteristics are shown in Figure 5-13; the results are shown in Figure 5-16.

```

$ TSUPREM-4 - Electrical Extraction

$ Read structure from Example 4
INITIAL      IN.FILE=S4EX4CS

$ Part A: Threshold voltage
$ Extract the gate bias vs. the sheet conductance in channel region
$ -- VBS=0V
ELECTRIC     X=0.0  THRESHOLD  NMOS  V="0 2 0.1"  OUT.FILE=S4EX4DS1

$ -- VBS=-2.5V
ELECTRIC     X=0.0  THRESHOLD  NMOS  V="0 3 0.05"  VB=-2.5 +
              OUT.FILE=S4EX4DS2

$ Plot the Vgs vs Ids
$ -- Define the scale to convert the sheet conductance to the current
ASSIGN       NAME=Lch   N.VAL=1.2
ASSIGN       NAME=Wch   N.VAL=25.0
ASSIGN       NAME=Vds   N.VAL=0.1
ASSIGN       NAME=Scale N.VAL=(@Vds*@Wch/@Lch)
$ -- Plot
SELECT       TITLE="Vgs vs. Ids"
VIEWPORT     X.MAX=0.5
PLOT.1D      IN.FILE=S4EX4DS1  Y.SCALE=@Scale +
              Y.LABEL="I(Drain) (Amps)"  X.LABEL="V(Gate) (Volts)" +
              TOP=1E-4  BOT=0  RIGHT=3.5  COLOR=2
PLOT.1D      IN.FILE=S4EX4DS2  Y.SCALE=@Scale ^CL ^AX  COLOR=3  LINE=2
LABEL        LABEL="Vbs=0"  X=1.9  Y=9E-5  RIGHT
LABEL        LABEL="Vbs=-2.5"  X=3.35  Y=7.6E-5  RIGHT

```

Figure 5-13 First part of input file *s4ex4d.inp*, showing the threshold voltage extraction

MOS Capacitance

Many process monitoring parameters—the grown oxide thickness, the surface state density, the flat band voltage, and the mobile charge in oxide, for example—are extracted from MOS C-V measurements during manufacturing. While transient device simulations are required for rigorous analysis of gate capacitance as a function of frequency, TSUPREM-4 can give reasonably accurate simulations of gate capacitance if a sufficiently high frequency (above 100kHz) is assumed.

The parameter **MOSCAP** specifies that the MOS capacitance is extracted. Depending on the input frequency compared with the lifetime of carriers in the channel, there are three types of C-V plot.

1. In most cases, the input signal is composed of a slow bias signal and fast AC signal. This is the normal C-V plot, which is specified by **HIGH** (the default) in TSUPREM-4.
2. The second assumes a slow bias signal and a low-frequency AC input signal; that is specified by the **LOW** parameter.
3. Finally, if a fast bias sweep is performed with a high-frequency AC signal so that the inversion charge does not have time to accumulate, the depletion region expands deeper into the substrate. The **DEEP** parameter specifies this.

In this example, gate bias V is increased from -5 volts to 5 volts in 0.2 volt steps. The capacitance is calculated from dQ/dV . The perturbed AC bias, dV is calculated from the DC increment multiplied by a constant $E.RVCAP$, which is defined as the ratio of AC amplitude to DC increment in the **METHOD** statement (default value of 0.2 or 20%). The perturbed AC bias in this example is 0.04V(= 0.2V \times 0.2). For example, the capacitance at 1 volt is extracted from the charge variation between 0.96 volts and 1.04 volts. The input statements for extracting the MOS capacitance are shown in Figure 5-14; the results are shown in Figure 5-16.

```

$ Part B: C-V plot for MOS capacitance
$ Extract the capacitance

$ -- High Frequency
ELECTRIC X=0.0 MOSCAP NMOS V="-5 5 0.2" OUT.F=S4EX4DS3
$ -- Low Frequency
ELECTRIC X=0.0 MOSCAP NMOS V="-5 5 0.2" LOW OUT.F=S4EX4DS4
$ -- Deep depletion
ELECTRIC X=0.0 MOSCAP NMOS V="-5 5 0.2" DEEP

$ Plot the C-V curve
SELECT TITLE="MOS C-V"
VIEWPORT X.MIN=0.5 Y.MIN=0.51
PLOT.1D ELECTRIC COLOR=2 TOP=1E-7 BOT=0 LEFT=-6 RIGHT=6 +
X.OFF=1.5 ^CL
PLOT.1D IN.FILE=S4EX4DS3 ^CL ^AX COLOR=3 LINE=2
PLOT.1D IN.FILE=S4EX4DS4 ^CL ^AX COLOR=4 LINE=3
LABEL LABEL="Low" X=3 Y=8.3E-8
LABEL LABEL="High" X=3 Y=3.7E-8
LABEL LABEL="Deep" X=3 Y=0.7E-8

```

Figure 5-14 Second part of input file *s4ex4d.inp*, showing the MOS capacitance extraction

Source/Drain Junction Capacitance

The junction capacitance between source (or drain) and bulk is one of important parameters to determine the delay characteristics of the MOS transistor. This capacitance is composed of two kinds of capacitance: areal and peripheral capacitance. In device characterization, these two components are separated and characterized by universal parameters independent of the shape of device. Usually, the areal capacitance is dominant. TSUPREM-4 extracts the areal junction capacitance (in Farads/cm²) when the **JCAP** parameter is specified.

The **JUNCTION** parameter selects the junction to be analyzed. Junctions are numbered from the bottom of the structure to the top, with the deepest junction being junction number one. For example, if **X** specifies the emitter region of a bipolar transistor, there might be three junctions: E-B (emitter-base, junction number 3), B-C (base-collector, junction number 2), and C-S (collector-substrate, junction number 1). In this example, there is only one junction, identified as number 1.

The input bias V must be all positive or all negative including zero bias. TSUPREM-4 chooses the biased region by considering the polarity of the input voltage so that the junction is reverse biased. All regions except for the biased region are grounded. The perturbed AC bias is applied using the same method as in the MOS capacitance.

Input statements for extracting the junction capacitance are shown in [Figure 5-15](#).

```

$ Part C: Junction Capacitance
$ Extract the S/D (area) junction capacitance
ELECTRIC X=2.0 JCAP JUNCTION=1 V="0 5 0.5"
SELECT TITLE="S/D Junction C"
VIEWPORT X.MIN=0.5 Y.MAX=0.49
PLOT.1D ELECTRIC COLOR=2 +
TOP=2.5E-8 BOT=0 LEFT=-1 RIGHT=6 X.OFF=1.5 ^CL
    
```

Figure 5-15 Third part of input file *s4ex4d.inp*, showing the junction capacitance extraction

Plotting Results of Electrical Extraction

The results of the electrical extractions are shown in [Figure 5-16](#). Note how the results of electrical extraction can be saved in an output file for later plotting.

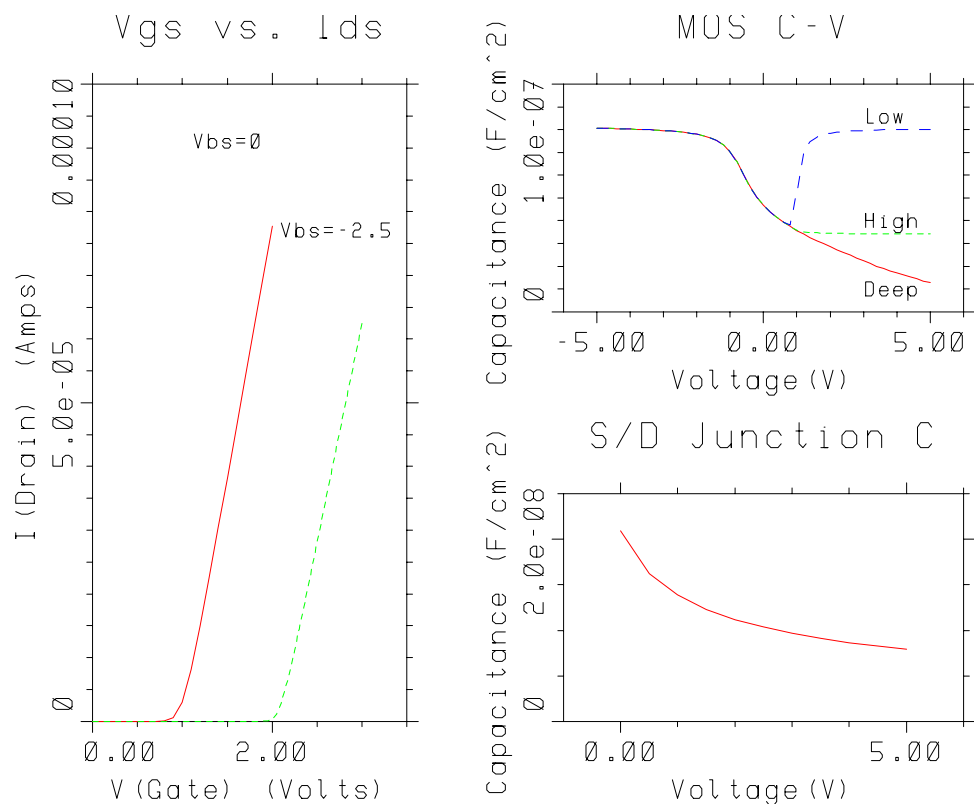


Figure 5-16 Electrical characteristics, plotted by input file *s4ex4d.inp*

Trench Implant Simulation

This example shows two methods for performing a tilted ion implantation into a trench. First, TSUPREM-4's analytical model is used to implant ions at a 15-degree tilt. Next, TSUPREM-4's Monte Carlo method is used to perform the same implantation. The results of the two methods are compared and contrasted.

```

$ TSUPREM-4 Example 5 - Implant Trench Application
$
$ Simulate ion implantation into a trench using the analytical approach
$ followed by the Monte Carlo approach.

$ Place the finest grid around the trench
LINE X LOCATION=0.0 SPACING=1.0
LINE X LOCATION=0.25 SPACING=0.01
LINE X LOCATION=0.32 SPACING=0.01
LINE X LOCATION=0.5 SPACING=0.2

LINE Y LOCATION=0.0 SPACING=0.01
LINE Y LOCATION=0.05 SPACING=0.01
LINE Y LOCATION=0.92 SPACING=0.04
LINE Y LOCATION=1.0 SPACING=0.01
LINE Y LOCATION=1.05 SPACING=0.01
LINE Y LOCATION=1.2 SPACING=1.0

ELIMINATE ROWS X.MAX=0.2 Y.MIN=0.15 Y.MAX=1.0
ELIMINATE ROWS X.MAX=0.25 Y.MIN=1.0 Y.MAX=1.1
ELIMINATE COLUMNS X.MIN=0.23 X.MAX=0.35 Y.MIN=1.1
ELIMINATE COLUMNS X.MIN=0.25 X.MAX=0.34 Y.MIN=1.1

$ Initialize the structure
INITIALIZE BORON=2E15

$ Etch the left half of the trench
ETCH START X=0.3 Y=0.0
ETCH CONTINUE X=0.3 Y=0.5
ETCH CONTINUE X=0.32 Y=0.9
ETCH CONTINUE X=0.34 Y=0.95
ETCH CONTINUE X=0.4 Y=1.0
ETCH CONTINUE X=2.0 Y=1.0
ETCH DONE X=2.0 Y=0.0

$ Form the complete trench structure
STRUCTURE REFLECT

$ Save the structure
SAVEFILE OUT.FILE=S4EX5S

$ Plot the grid

```

Figure 5-17 First part of input file *s4ex5.inp*, showing grid setup

Structure Generation

The input statements for this simulation are shown in [Figures 5-17](#), [5-19](#), and [5-23](#). The initial mesh setup uses the symmetry of the trench structure, as suggested by SEM photographs of trenches generated by reactive ion etching. The

initial grid, established by the **LINE** and **ELIMINATE** statements, is for the left half of the trench only. Particular attention has been given to producing a fine grid where the side and bottom walls of the trench are formed in order to resolve the implanted profiles. The **ELIMINATE** statements are used to reduce the number of grid points in areas where the implant is not expected to penetrate. The sequence of **ETCH** statements specifies geometrically the shape of the left half of the trench structure. The resulting half-structure is reflected to generate the full structure, using the **REFLECT** parameter on the **STRUCTURE** statement. The full structure is saved with the **SAVEFILE** statement.

The grid for the full structure is then plotted (Figure 5-18). Note that a fine mesh has been placed around the trench, without wasting nodes in the lower corners.

Grid for Trench Application

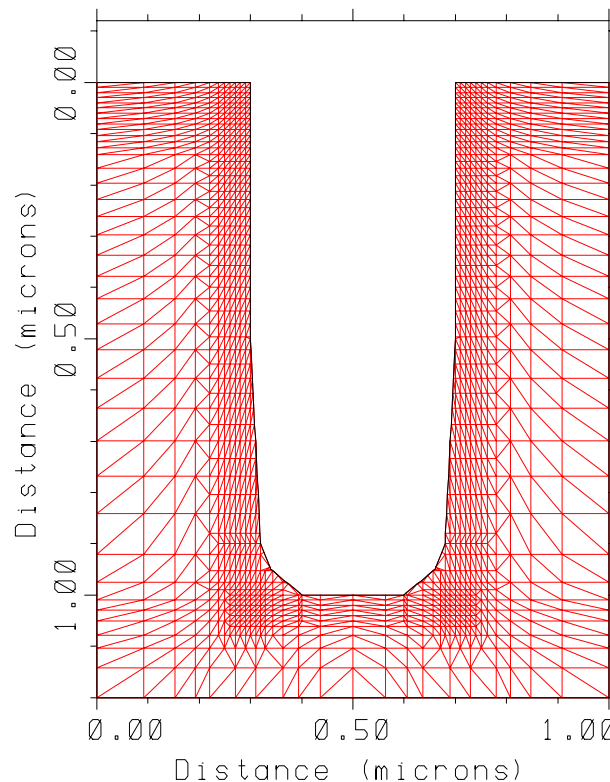


Figure 5-18 Grid for trench implant example

Analytic Implant

The boron implant is now performed at an energy of 5 keV. In the absence of the **MONTECAR** parameter, TSUPREM-4 uses the analytic method. The **TILT** parameter is used to specify that the angle of the incident ion beam is 15 degrees counter-clockwise from the vertical (i.e., the beam enters from the left). The

implant is followed by a short diffusion to activate the boron and anneal implant damage.

```

$ Analytic implant at a 15-degree tilt to dope the trench sidewalls
IMPLANT  BORON  ENERGY=5  DOSE=1E14  TILT=15

$ A short anneal
DIFFUSION  TIME=5  TEMP=900

$ 2D contour plot of boron contours
SELECT  Z=LOG10(BORON)  TITLE="Contours of Boron (Analytic)"
PLOT.2D  SCALE  X.MAX=1.5
FOREACH  X  (16 TO 21 STEP 1)
  COLOR  MIN.V=X  MAX.V=( X + 1)  COLOR=( X - 2)
END
PLOT.2D  ^AXES  ^CLEAR
LABEL  X=1.05  Y=0.25  LABEL="Log10(Boron)"  SIZE=0.3
LABEL  X=1.15  Y=0.33  LABEL="16-17"  SIZE=0.3  C.RECT=14  W.R=0.4  H.R=0.4
LABEL  X=1.15  Y=0.41  LABEL="17-18"  SIZE=0.3  C.RECT=15  W.R=0.4  H.R=0.4
LABEL  X=1.15  Y=0.49  LABEL="18-19"  SIZE=0.3  C.RECT=16  W.R=0.4  H.R=0.4
LABEL  X=1.15  Y=0.57  LABEL="19-20"  SIZE=0.3  C.RECT=17  W.R=0.4  H.R=0.4
LABEL  LABEL="Tilt angle=15 degrees"  X=0.02  Y=1.18

$ 1D plots of boron

$ Vertical profiles
SELECT  Z=LOG10(BORON)  TITLE="Vertical Profiles (Analytic)"
PLOT.1D  X.VALUE=0.1  TOP=21  BOTTOM=15  RIGHT=1.2  COLOR=2
LABEL  X=0.06  Y=19  LABEL="Surface"
LABEL  X=0.06  Y=18.8  LABEL="(x=0.1)"

PLOT.1D  X.VALUE=0.5  ^AXES  ^CLEAR  COLOR=2
LABEL  X=1.05  Y=18.5  LABEL="Trench"
LABEL  X=1.05  Y=18.3  LABEL="(x=0.5)"

$ Horizontal profiles
DEPOSIT  OXIDE  THICK=0.002
SELECT  Z=LOG10(BORON)  TITLE="Sidewall Profiles at y=0.2 (Analytic)"
PLOT.1D  Y.VALUE=0.2  LEFT=0.1  RIGHT=0.9  BOTTOM=15  TOP=21  COLOR=2

$ Print profile through bottom of trench
PRINT.1D  X.VALUE=0.5

```

Figure 5-19 Second part of input file *s4ex5.inp*, showing tilted implantation using analytic implant model

Plotting the Results of the Analytic Method

Figure 5-20 shows shaded contours of boron concentration produced by the **COLOR** statement and associated **SELECT**, **PLOT . 2D**, **FOREACH**, **END**, and **LABEL** statements. Observe that dopant is present only on the right side of the trench where the ions are directly incident. This is because the analytic method does not account for reflected ions. The profile at the bottom right corner of the trench (where boron is directly incident) is approximately the same as that on the surface of the silicon.

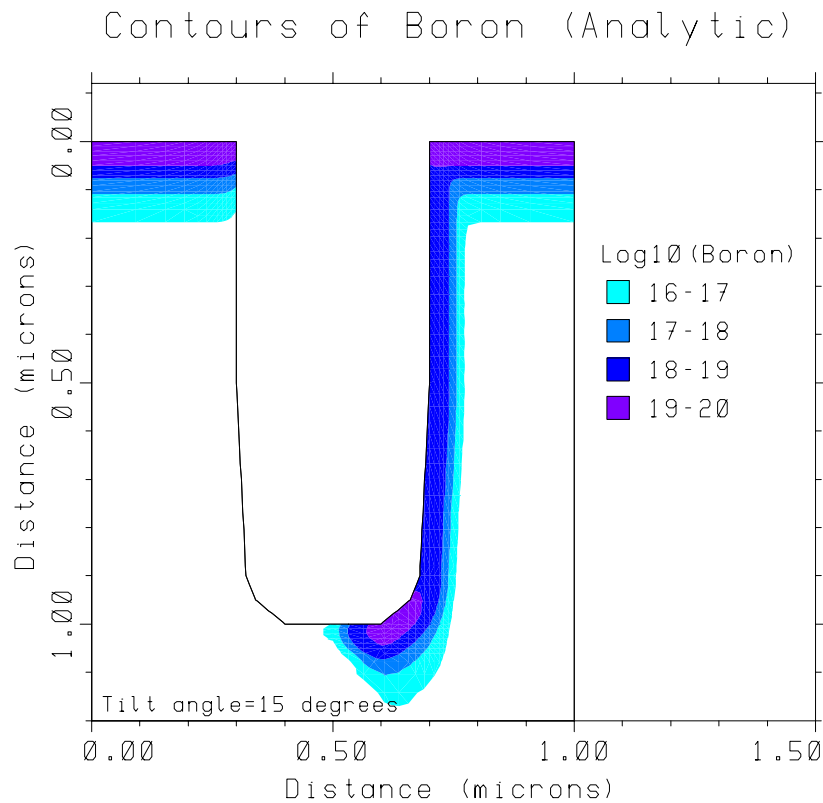


Figure 5-20 Contours of boron after analytic implant

Figure 5-21 shows one-dimensional boron profiles vertically through the middle of the trench bottom and at the surface of the silicon. The peak boron concentration is much greater at the silicon surface than at the bottom of the trench due to shadowing of the bottom by the sidewalls.

Figure 5-22 shows the boron concentration on the left and right sides of the trench at a depth of $y=0.2$ microns. The flat profile on the left hand side, at a value nearly that of the background concentration, confirms the lack of reflected ions when the analytic method is used. The corresponding plot for the Monte Carlo implantation model is very different.

(A thin layer of oxide is deposited to ensure that the structure boundaries are correctly plotted by the **PLOT. 1D** statement. Without the added oxide, the section at $y=0.2$ would pass through silicon, ambient, and silicon as x varies from 0 to 1.0 microns. Due to a limitation in the current version of the program, the silicon/ambient and ambient/silicon interfaces are only recognized at the edges of the structure, but not in its interior. By adding the thin layer of oxide, silicon/oxide and oxide/silicon boundaries are added. These boundaries are plotted correctly in Figures 5-20 and 5-26.)

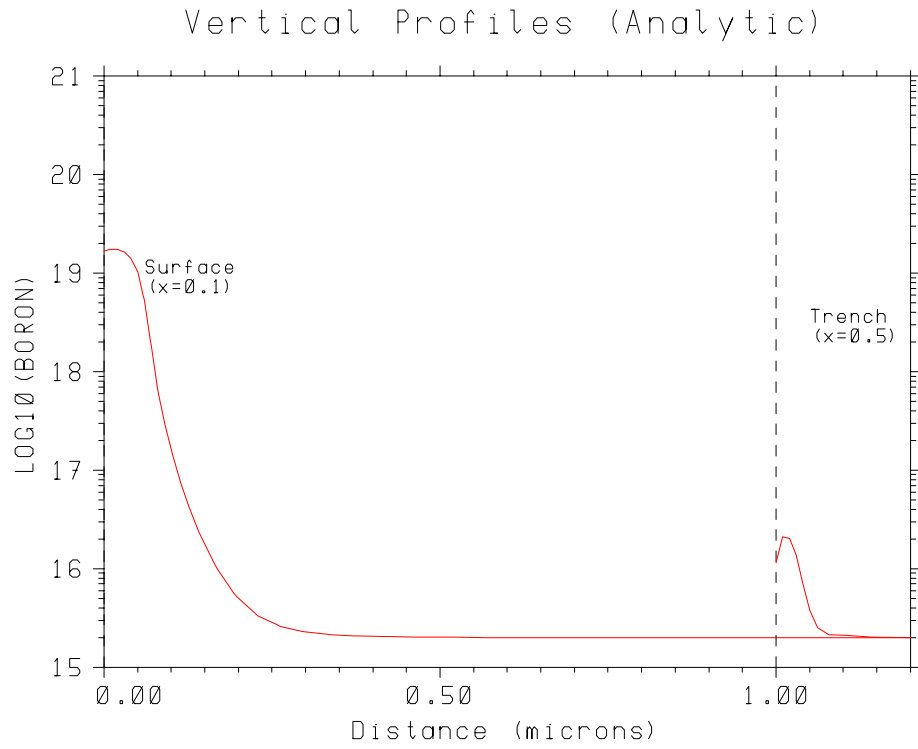


Figure 5-21 Vertical profiles produced by analytic implant

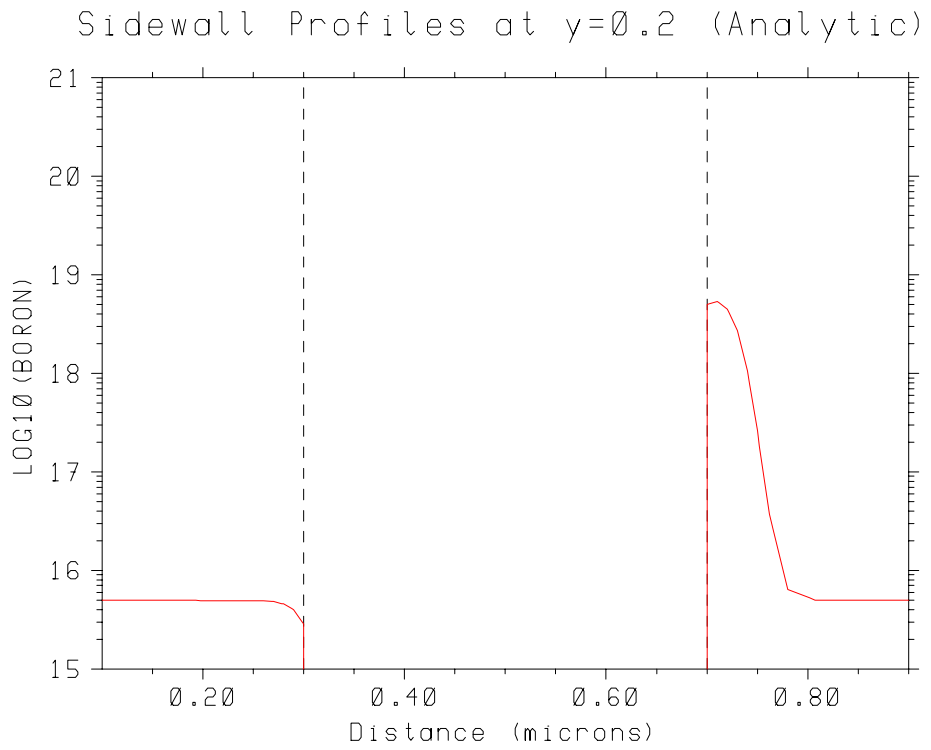


Figure 5-22 Sidewall profiles produced by analytic implant

Monte Carlo Implant

The second half of the simulation demonstrates the use of the Monte Carlo implantation method.

Overview

Unlike the analytic approach, the Monte Carlo algorithm accurately models the effect of reflected ions in a trench structure. The Monte Carlo calculation follows the trajectories of individual ions through each material present in the structure; ion trajectories through a vacuum, including those reflected off trench sidewalls, are calculated in exactly the same way as through other materials. The velocity of a reflected ion remains constant until it enters a material other than vacuum or leaves the top or bottom of the simulated structure. For ions that leave either side of the simulation space, the appropriate boundary condition (periodic, reflecting, or vacuum) is invoked.

Using the Monte Carlo Model

The first step in the Monte Carlo portion of the simulation is to read in the trench structure, just prior to the analytic implant. The boron implant is performed once again, but this time the **MONTECAR** parameter on the **IMPLANT** statement directs TSUPREM-4 to use the Monte Carlo implant model. The **ENERGY**, **DOSE**, and **TILT** parameters are the same as those used during the analytic implant.

In addition to the **MONTECAR** parameter, two new parameters are now specified:

- **N . ION**—The **N . ION** parameter specifies the number of ion trajectories to calculate. The simulation time is directly proportional to the number of ion trajectories calculated. Decreasing this number, however, increases the statistical noise in the output of the Monte Carlo calculation.
- **^CRYSTAL**—The specification of **^CRYSTAL** on the **IMPLANT** command disables the crystal model for ion penetration in silicon. This speeds the calculation significantly by eliminating the calculation of channeling and damage effects.

A short post-implant anneal is performed as before to activate the implant. The diffusion also acts to smooth the results of the Monte Carlo calculation.

Plotting the Results of the Monte Carlo Method

The three figures output after the Monte Carlo implant ([Figures 5-24](#) through [5-26](#)) are the counterparts of the three analytic plots ([Figures 5-20](#) through [5-22](#)).

Boron Contours

[Figure 5-24](#) shows the contours of boron concentration produced by the Monte Carlo method. Note that dopant is present in substantial quantities on the left side of the trench (unlike the analytic results). The plot illustrates that dopant is present in greater quantity on the right side of the trench where the ions are directly incident. The boron in the left side of the trench is due entirely to reflection of ions. As before, the profile at the bottom right corner of the trench is approximately the same as that on the surface of the silicon.

```

$ Repeat the implantation using the Monte Carlo method

$ Read in the structure
LOADFILE IN.FILE=S4EX5S

$ Monte Carlo implant at a 15-degree tilt to dope the trench sidewalls
IMPLANT BORON ENERGY=5 DOSE=1E14 MONTECAR TILT=15 N.ION=10000
^CRYSTAL

$ A short anneal
DIFFUSION TIME=5 TEMP=900

$ 2D contour plot of boron contours
SELECT Z=LOG10(BORON) TITLE="Contours of Boron (Monte Carlo)"
PLOT.2D SCALE X.MAX=1.5
FOREACH X (16 TO 21 STEP 1)
  COLOR MIN.V=X MAX.V=( X + 1) COLOR=( X - 2)
END
PLOT.2D ^AXES ^CLEAR
LABEL X=1.05 Y=0.25 LABEL="Log10(Boron)" SIZE=0.3
LABEL X=1.15 Y=0.33 LABEL="16-17" SIZE=0.3 C.RECT=14 W.R=0.4 H.R=0.4
LABEL X=1.15 Y=0.41 LABEL="17-18" SIZE=0.3 C.RECT=15 W.R=0.4 H.R=0.4
LABEL X=1.15 Y=0.49 LABEL="18-19" SIZE=0.3 C.RECT=16 W.R=0.4 H.R=0.4
LABEL X=1.15 Y=0.57 LABEL="19-20" SIZE=0.3 C.RECT=17 W.R=0.4 H.R=0.4
LABEL LABEL="Tilt angle=15 degrees" X=0.02 Y=1.18

$ 1D plots of boron

$ Vertical profiles
SELECT Z=LOG10(BORON) TITLE="Vertical Profiles (Monte Carlo)"
PLOT.1D X.VALUE=0.1 TOP=21 BOTTOM=15 RIGHT=1.2 COLOR=2
LABEL X=0.06 Y=19 LABEL="Surface"
LABEL X=0.06 Y=18.8 LABEL="(x=0.1)"

PLOT.1D X.VALUE=0.5 ^AXES ^CLEAR COLOR=2
LABEL X=1.05 Y=18.5 LABEL="Trench"
LABEL X=1.05 Y=18.3 LABEL="(x=0.5)"

$ Horizontal profiles
DEPOSIT OXIDE THICK=0.002
SELECT Z=LOG10(BORON) TITLE="Sidewall Profiles at y=0.2 (Monte Carlo)"
PLOT.1D Y.VALUE=0.2 LEFT=0.1 RIGHT=0.9 BOTTOM=15 TOP=21 COLOR=2

$ Print profile through bottom of trench
PRINT.1D X.VALUE=0.5

```

Figure 5-23 Third part of file *s4ex5.inp*, using the Monte Carlo implantation model

Vertical Profiles

Figure 5-25, shows one-dimensional boron profiles vertically through the middle of the trench bottom and at the surface of the silicon for the Monte Carlo implant. The peak boron concentration is approximately a factor of five greater at the silicon surface than at the bottom of the trench due to shadowing of the bottom by the sidewalls. Observe that the difference between the peaks is much less for the Monte Carlo approach than the analytic method. This is due to the reflected ion calculations included in the Monte Carlo algorithm. The depth of the implant is less at the bottom of the trench than at the silicon surface, due to the loss of energy when an ion is reflected.

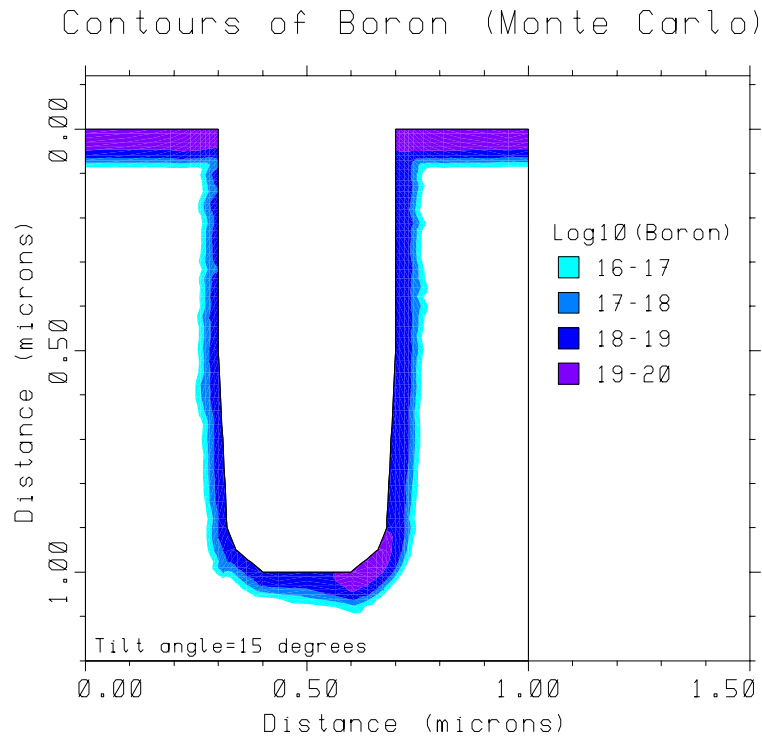


Figure 5-24 Contours of boron after Monte Carlo implant

Comparing the vertical profiles produced by the analytic and Monte Carlo models (Figures 5-21 and 5-25, respectively), shows that the boron penetrates more deeply with the analytic model than with the Monte Carlo approach. This is due to the specification of **^CRYSTAL** during the Monte Carlo implantation. This causes the silicon to be treated as amorphous, which speeds the calculation significantly but prevents calculation of channeling in the crystal structure.

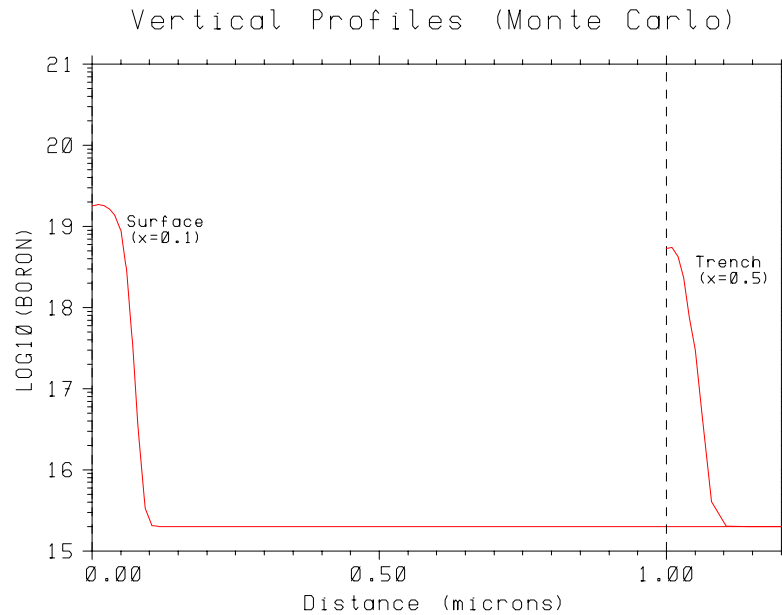


Figure 5-25 Vertical profiles after Monte Carlo implant

In contrast, the analytic method automatically includes the channeling calculations, thus resulting in deeper penetration. If the **^CRYSTAL** were omitted, channeling calculations would be included in the Monte Carlo implant, giving a depth of penetration very similar to the analytic results.

Sidewall Profiles

The last Monte Carlo plot (Figure 5-26) shows the boron profiles on the left and right sidewalls of the trench. The profile on the left side of the trench is due entirely to reflected ions and is not as deep as the profile on the right, due to the reduced energy of reflected ions. The peak concentration of the profile on the left side is also reduced compared to that on the right. This result incorporates effects of ions reflecting from various points in the trench as well as multiple reflections.

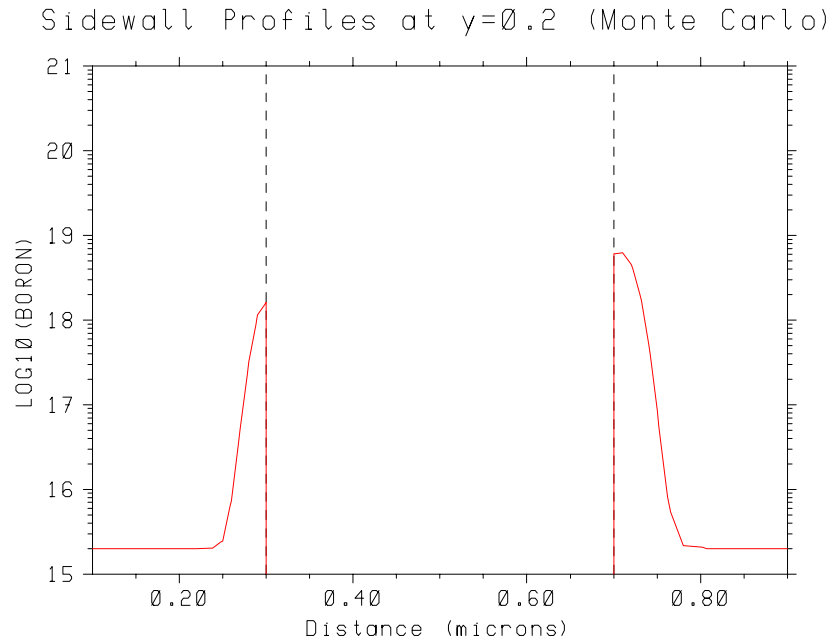


Figure 5-26 Sidewall profiles after Monte Carlo implant

Summary

The Monte Carlo model can be very useful in simulating implantation into complex structures, due to its ability to model the effects of reflected ions. It is also useful in cases where analytical range statistics are not available, and in some multilayer implant simulations. The analytic model is more suitable, however, for simulating the implant steps that are encountered in most practical situations.

Poly-Buffered LOCOS

This example illustrates the use of TSUPREM-4 to simulate a poly-buffered LOCOS process. This process reduces the typical “bird’s beak” formation and stresses on the silicon substrate through use of a layer of polycrystalline silicon between the pad oxide and the nitride mask.

Structure Generation

The input statements for generating the structure and performing the field isolation processing are shown in Figure 5-27. A relatively coarse initial grid is used. In the horizontal direction, it is finest at the mask edge, at $x=1$ micron. The pad oxide and poly buffer layers are deposited with two grid spaces in each. Between the poly layer and the nitride mask a thin (20Å) layer of oxide is deposited. A thick nitride mask layer is used. The nitride mask is etched to the right of $x=1.0$ to expose the region to be oxidized. This example oxidizes through the polysilicon layer. The grid prior to oxidation is shown in Figure 5-28.

```

$ TSUPREM-4: Poly-Buffered LOCOS Application

$ Set up the grid
LINE X LOCATION=0 SPACING=0.1
LINE X LOCATION=1.0 SPACING=0.06
LINE X LOCATION=1.5 SPACING=0.1

LINE Y LOCATION=0 SPACING=0.03
LINE Y LOCATION=0.5 SPACING=0.1

INITIALIZE

$ Deposit pad oxide and define nitride mask
DEPOSITION OXIDE THICK=0.02 SPACES=2
DEPOSITION POLY THICK=0.04 SPACES=2
DEPOSITION OXIDE THICK=0.002
DEPOSITION NITRIDE THICK=0.2 SPACES=2

ETCH NITRIDE RIGHT P1.X=1.0

$ Plot initial grid
SELECT TITLE="Initial Grid"
PLOT.2D SCALE GRID C.GRID=2

$ Use viscoelastic model for oxidation
METHOD VISCOEL
MATERIAL MAT=OXIDE VC=750
MATERIAL MAT=NITRIDE VC=250
DIFFUSION TEMP=1050 WETO2 TIME=85

$ Save structure
SAVEFILE OUT.FILE=S4EX6S

```

Figure 5-27 First part of input file *s4ex6.inp*: Poly-buffered LOCOS process

Using the VISCOEL Model

The **METHOD** statement selects the viscoelastic oxidation model. The stress dependence of the physical parameters of the model are included in the calculation by default. This combination gives results that are much more accurate than those obtained with the **COMPRESS** model, but takes much less simulation time than the **VISCOUS** model with stress dependence.

The stress dependence of the oxide and nitride viscosities is given by the **VC** parameter on the **MATERIAL** statement. Because these values depend on temperature, it is usually necessary to specify appropriate values for each oxidation step. The two **MATERIAL** statements in this example set **VC** for nitride and oxide to values given in the literature for a 1050°C oxidation. (Appendix A lists the default values for **VC** and suggests values for use at various temperatures.)

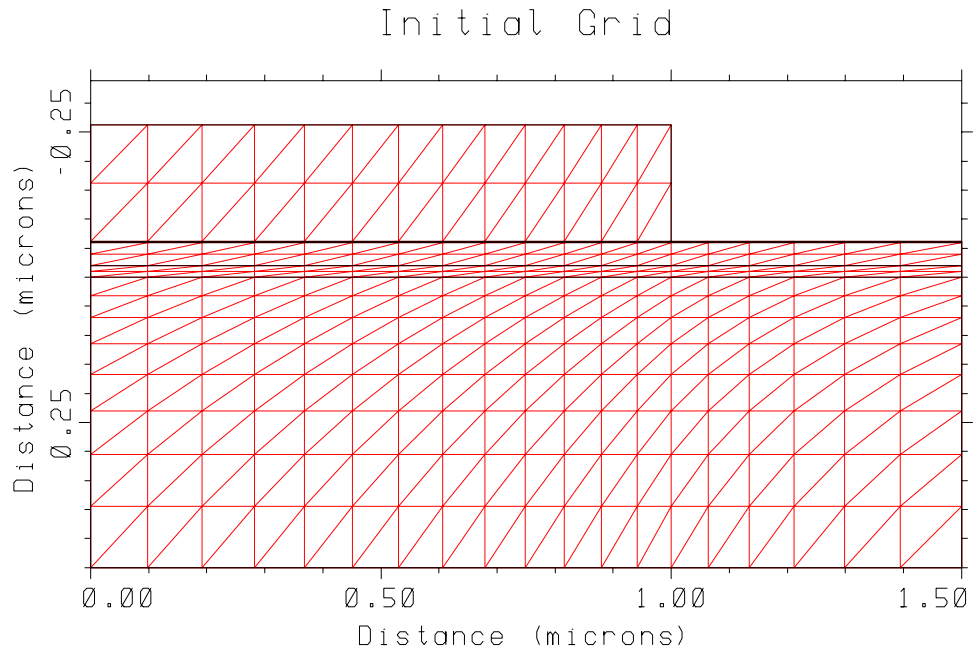


Figure 5-28 Grid for poly-buffered LOCOS application

Plotting the Results

[Figure 5-29](#) shows the input statements used to plot the results. Of particular interest are the stresses in the structure, as indicated by the hydrostatic pressure. The statement **SELECT Z = (-0.5 * (SXX + SYY))** calculates the hydrostatic pressure from the stress components. The minus sign is used so that positive values of **Z** correspond to compression. A **FOREACH** loop is used to plot contours of compressive and tensile stress at values from 3×10^9 to 3×10^{10} , in steps of 3×10^9 .

The final plot is shown in [Figure 5-30](#). The very high pressure gradients at the end of the poly layer (near $x=0, y=-0.05$) result from the volume expansion produced by the oxidation of the poly layer. The high pressure in this region confirms that stress effects are important in this simulation, and that the stress-dependent oxidation models must be used to obtain an accurate simulation of this structure.

```

$ Plot the structure with oxide stress
SELECT      TITLE="Contours of Pressure"
PLOT.2D     SCALE
COLOR       SILICON  COLOR=7
COLOR       OXIDE    COLOR=5
COLOR       NITRIDE  COLOR=4
COLOR       POLY     COLOR=3
SELECT      Z=(-0.5*(SXX+SYY))
FOREACH     I (1 TO 10)
  CONTOUR   VALUE=( I *3E9)  LINE=2  COLOR=6
  CONTOUR   VALUE=(- I *3E9) LINE=3  COLOR=6
END
PLOT.2D     ^AX ^CL

LABEL       LABEL="Poly"      X=0.1  Y=-0.03
LABEL       LABEL="Oxide"     X=1.2  Y=-0.05
LABEL       LABEL="Nitride"   X=0.1  Y=-0.15
LABEL       LABEL="Substrate" X=0.1  Y=0.20

$ Print location of Si/SiO2 interface
SELECT      Z=Y
PRINT.1D    SILICON /OXIDE
    
```

Figure 5-29 Second part of *s4ex6.inp*: Plotting final poly-buffered LOCOS structure

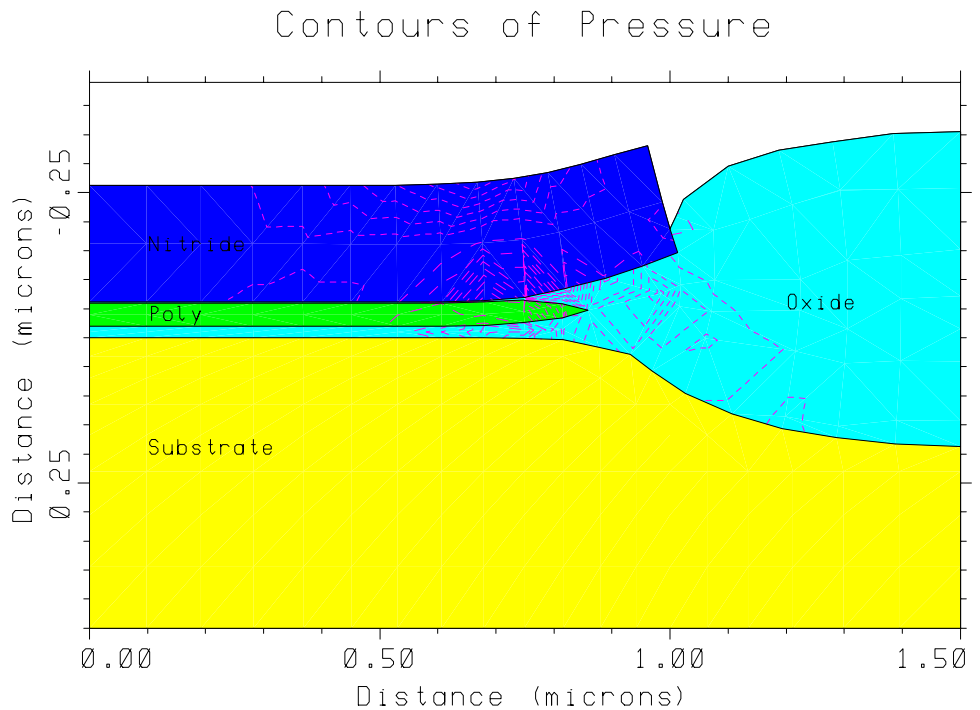


Figure 5-30 Contours of hydrostatic pressure in final poly-buffered LOCOS structure

CMOS Process

This example simulates the fabrication of NMOS transistors of two different channel lengths as part of a CMOS process. The goal is to produce an output file containing complete structure, mesh, and doping information that can be read into the Medici device simulator. The considerations needed to produce such a file are illustrated.

The two channel lengths are simulated with a single input file, using a **FOREACH** loop around the entire simulation, from mesh generation to saving of the final structure. The use of the **FOREACH** loop simplifies the input and ensures that any changes in the process sequence are applied to both devices equally. This use of the **FOREACH** loop illustrates several useful techniques for parameterizing TSUPREM-4 input files. The listing of the first input file for this example (*s4ex7a.inp*) is shown in Figures 5-31 through 5-33. Comment statements have been used to number the steps in the simulation.

```

$ TSUPREM-4 N-channel MOS application
$
$ 1. Identify the graphics driver
$ Default from DEFPDEV, TERM, or S4PCAP "default" entry used

$ 2. Beginning of the main loop
FOREACH   LD ( 3 5 )

$ 3. Specify the mesh
MESH      GRID.FAC=1.5
MESH      DY.SURF=0.01  LY.SURF=0.04  LY.ACTIV=2.0

$ 4. Initialize
INITIALIZE <100>  BORON=1E15  WIDTH=( 0.7 + ( LD / 10.0 ) ) DX=0.1

$ 5. Plot the initial mesh
SELECT    TITLE="Mesh for Delta=0.0@{LD}"
PLOT.2D   SCALE GRID Y.MAX=3.0 C.GRID=2

$ 6. Initial oxide pad
DEPOSIT   OXIDE THICKNESS=0.03

$ 7. P-well implant
IMPLANT   BORON DOSE=1E12 ENERGY=35

$ 8. Use a point defect models that simulates OED
METHOD    PD.TRANS

$ 9. P-well drive
DIFFUSE   TEMP=1100 TIME=120 DRYO2 PRESS=0.02

$ 10. P-well doping profile
SELECT    Z=LOG10(BORON) TITLE="Channel Doping (Delta=0.0@{LD})"
PLOT.1D   X.VALUE=0 RIGHT=3.0 BOTTOM=15 TOP=19 LINE.TYP=2 COLOR=2
LABEL     X=1.8 Y=18.5 LABEL="After p-well drive" LINE.TYP=2 C.LINE=2

```

Figure 5-31 First part of input file *s4ex7a.inp*, to set up grid for simulating a CMOS process

```

$ 11. Pad nitride
DEPOSIT    NITRIDE    THICKNESS=0.1

$ 12. Field implant and oxidation
IMPLANT    BORON    DOSE=5E13    ENERGY=80
DIFFUSE    TEMP=1000    TIME=360    WETO2

$ 13. Etch to remove the pad
ETCH       NITRIDE    ALL

$ 14. Vt adjust implant
IMPLANT    BORON    ENERGY=100    DOSE=1E12

$ 15. P-well doping profile
SELECT     Z=LOG10(BORON)
PLOT.1D    X.VALUE=0    ^AXES    ^CLEAR    COLOR=2
LABEL      X=1.8    Y=18.2    LABEL="After Vt implant"    LINE.TYP=1    C.LINE=2

$ 16. Print oxide and silicon thicknesses
SELECT     Z=1
PRINT.1D   X.VALUE=0.0    LAYERS

$ 17. Etch oxide
ETCH       OXIDE    TRAP    THICK=0.05

$ 18. Gate oxidation
DIFFUSE    TEMP=950    TIME=30    DRYO2

$ 19. Poly deposition
DEPOSIT    POLYSILICON    THICKNESS=0.3    DIVISIONS=4

$ 20. Poly and oxide etch between x = 0.0 and 0.5 microns
ETCH       POLY    LEFT    P1.X=0.5
ETCH       OXIDE    TRAP    THICK=0.04

$ 21. Deposit a thin layer of oxide
DEPOSIT    OXIDE    THICKNESS=0.02

$ 22. LDD implant
IMPLANT    PHOS    ENERGY=50    DOSE=5E13    IMPL.TAB=PHOSPHORUS

$ 23. LTO
DEPOSIT    OXIDE    THICK=0.2

$ 24. Establish a sidewall spacer
ETCH       OXIDE    TRAP    THICK=0.22

$ 25. Source/drain implant
IMPLANT    ARSENIC    ENERGY=100    DOSE=2E15

$ 26. Oxide etch
ETCH       OXIDE    LEFT    P1.X=0.5

$ 27. Use an oxidation model that understands polysilicon
METHOD     COMPRESS

$ 28. Source/drain reoxidation (including the polysilicon gate)
DIFFUSE    TEMP=900    TIME=30    DRYO2

```

Figure 5-32 Second part of input file *s4ex7a.inp*, showing statements for simulating a CMOS process

```

$ 29. BPSG -- etch to open windows for aluminum contact
DEPOSIT    OXIDE    THICK=0.3
ETCH       OXIDE    LEFT   P1.X=0.3

$ 30. Metallization -- etch to create a source contact
DEPOSIT    ALUMINUM THICK=0.5 SPACES=3
DEPOSIT    PHOTORESIST THICK=1.0
ETCH       PHOTORESIST RIGHT P1.X=0.6
ETCH       ALUMINUM TRAP ANGLE=85 THICK=0.8
ETCH       PHOTORESIST ALL

$ 31. Reflect to form the complete structure; then save it
SAVEFILE   OUT.FILE=S4EX7AS@LD
STRUCTURE  REFLECT  RIGHT
SAVEFILE   OUT.FILE=S4EX7AP@LD  MEDICI

$ 32. End of loop
END

```

Figure 5-33 Third part of input file *s4ex7a.inp*, for simulating a CMOS process

Main Loop

The main loop begins at step 2 and ends at step 32. The name *LD* is used as a looping parameter, taking on the values 3 and 5. These values represent an increment in channel length to be added to each half of the symmetrical structure. *LD* is referenced in step 4, where it results in the width of the simulated structure being set to $0.7 + 3/10 = 1.0$ microns for $LD = 3$ and $0.7 + 5/10 = 1.2$ microns for $LD = 5$. Because the device is symmetrical only the left half of the structure needs to be included in the simulation; the structure is reflected about its right edge to form the full structure in step 31.

The entire structure has a width of $1.0 \times 2 = 2$ microns for the first pass through the loop ($LD = 3$) and a width of $1.2 \times 2 = 2.4$ microns for the second pass ($LD = 5$). The final channel lengths are approximately 0.8 and 1.2 microns.

Mesh Generation

In generating the mesh for this example, keep in mind that the same mesh is used for the Medici simulation as for the TSUPREM-4 simulation. In particular, it must taken into account that in the vertical direction, the field-dependent mobility models are calibrated for a grid spacing of about 100 Angstroms.

Automatic grid generation with adaptive grid is used. A **MESH** statement is used to modify the default parameters: The spacing at the surface (**DY.SURF**) is set to 0.01 for Medici, and the locations of the bottoms of the surface and active regions (**LY.SURF** and **LY.ACTIV**, respectively) are reduced to 0.04 and 2 microns (from the defaults of 0.1 and 4 microns) to reduce the simulation time. Grid spacings are increased by 50% for this simulation, using **GRID.FAC**=1.5, but this should be changed to 1.0 to get a 100 Angstrom spacing at the surface before doing the final Medici simulations. The default mesh extends to a depth of 200

microns. The structure can be truncated (with **STRUCTURE TRUNCATE**) before saving it for Medici, or Medici can truncate the saved structure as it is read in.



Note:

A separate mesh in Medici could be generated and the doping profiles read without the mesh from the TSUPREM-4 output file. In that case, the generation of the mesh would be guided more by process simulation considerations and less by the needs of the device simulation.

It is often a good idea to check the initial mesh structure (as in step 5) before starting a TSUPREM-4 simulation. The **PLOT . 2D** statement in step 5 specifies **Y . MAX** = 3 microns, since the area of interest does not extend much below the deepest junction depth. Given the times and temperatures of the processing steps to be used, do not expect the junction depth to exceed 1 micron.

The plot of the mesh produced the first time through the loop ($LD = 3$) is shown in Figure 5-34. The grid appears coarse, but is refined as needed by adaptive gridding.

Mesh for Delta=0.3

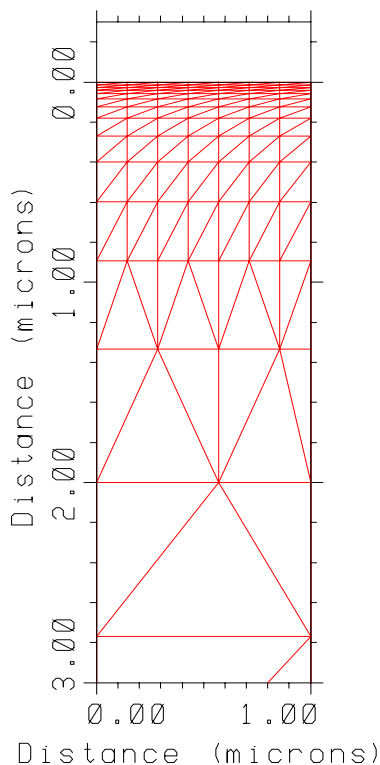


Figure 5-34 Initial grid for the 0.8 micron NMOS transistor, produced by *s4ex7a.inp*

The **FOREACH** parameter LD is used in step 5 in the title for the mesh plot. The name LD is enclosed in braces to separate it from the rest of the character string and preceded by a “@” character to force substitution of the value defined on the

FOREACH statement. The resulting titles are “Mesh for Delta=0.3” and “Mesh for Delta=0.5” for the two passes through the main loop.

CMOS Processing

The simulation of the CMOS process is straightforward. A variety of models are used to simulate the various processes.

Models

Because the process is planar up to the patterning of the polysilicon gate, the **VERTICAL** oxidation model can be used without any loss of accuracy. Once the poly gate is patterned, a switch is made to the **COMPRESS** model, both because of the nonplanarity of the structure and the desire to simulate the oxidation of the polysilicon gate. In order to accurately simulate oxidation-enhanced diffusion effects, the **PD.TRANS** model for point defects is chosen.



Note:

There might be significant OED effects produced by the field oxidation; to accurately simulate these, the structure would need to be expanded to include the field region.

The advanced models for grain growth and dopant diffusion in polysilicon are used by default. For a faster simulation but with less accuracy in the gate region, advanced poly models could be disabled before depositing the polysilicon layer.

Channel Doping Plot

The channel doping profile is plotted at two points in the process—after the p-well drive and after the threshold adjust implant. The result is shown in [Figure 5-35](#). Again, *LD* has been used in the title of the plot.

Lightly Doped Drain Structure

The remaining steps in the loop are fairly standard, but the method for producing the lightly-doped drain structure should be pointed out. The light source/drain extension implant is performed in step 22; because the implantation is through an oxide, **IMPL.TAB=phosphorus** is used to access implant data tables that do not include channelling effects. This is followed with a conformal deposition of 0.2 microns of oxide. Step 24 uses the **TRAP** parameter on the **ETCH** statement to remove all oxide within a vertical distance of 0.22 microns from the surface. This produces a spacer on the sidewall of the poly gate that serves as an implant mask for the heavy source/drain implant of step 25.

Contacts

An important final step for structures to be used in Medici is the deposition and patterning of aluminum contacts. Interfaces between aluminum and silicon are converted to contacts when the structure is passed to Medici. In addition, the Medici user can specify whether polysilicon regions are to be treated as semiconductor regions or contacts, and whether a contact should be added along the back side of the structure.

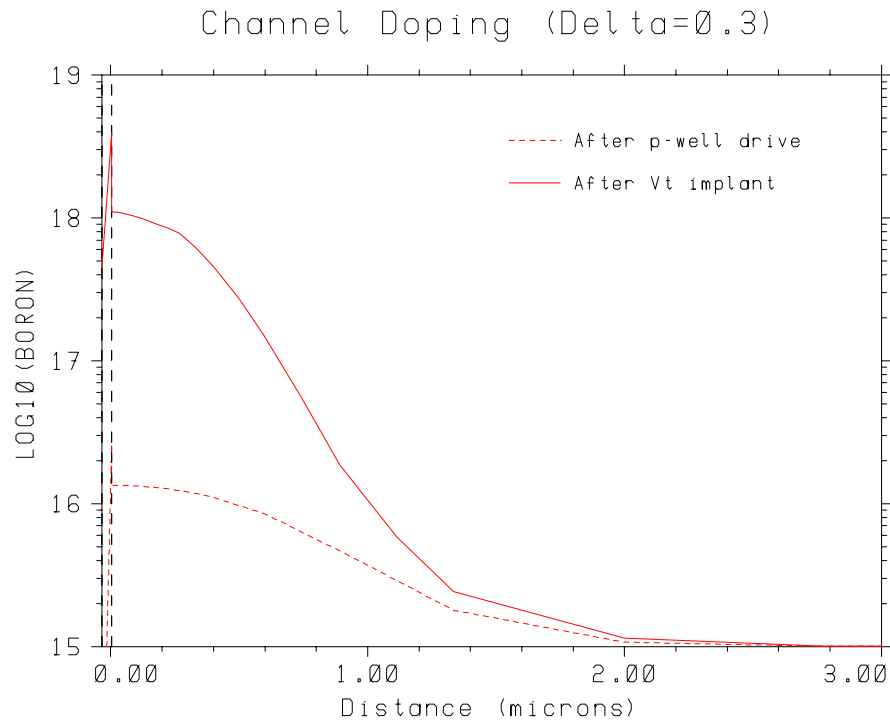


Figure 5-35 Channel doping profile for NMOS transistor

Saving the Structure

Once processing is complete, the finished structure is saved. Each structure is saved twice, once in TSUPREM-4's file format and once in the Medici format. For the Medici file, the structure must first be reflected to produce a full transistor. The structure is saved in TSUPREM-4 format before reflection. Note that the **FOREACH** parameter *LD* has been used once again to produce the output file names. This time the "@" character has been used to force substitution of the parameter, but the braces are not needed.

End of Main Loop

The **END** statement signals the end of the main **FOREACH** loop. After the entire sequence of input statements has been processed for *LD*=3, it is repeated for *LD*=5.

Plotting the Results

Once the complete structure has been saved, it can be read back in using the **INITIALIZE** statement and the results plotted without rerunning the simulation. Figure 5-36 shows an input file (*s4ex7b.inp*) that reads file *S4EX7AS3*, produced by the first **SAVEFILE** statement of step 31 on the first pass through the main loop.

```

$ TSUPREM-4 N-channel MOS application -- Part B
$ Plot the results for delta = 0.3

$ Read the structure
INITIALIZE IN.FILE=S4EX7AS3
STRUCTURE REFLECT RIGHT

$ Prepare to plot contours of boron, phosphorus, and arsenic
SELECT TITLE="N-Channel (Delta=0.3)"
PLOT.2D SCALE Y.MAX=1.5
COLOR SILICON COLOR=7
COLOR OXIDE COLOR=5
COLOR POLY COLOR=3
COLOR ALUMI COLOR=2

SELECT Z=LOG10(Boron)
FOREACH VAL (14 TO 21 STEP 1)
  CONTOUR VALUE=VAL LINE=5 COLOR=2
END

SELECT Z=LOG10(Phosphorus)
FOREACH VAL (16 TO 21 STEP 1)
  CONTOUR VALUE=VAL LINE=4 COLOR=3
END

SELECT Z=LOG10(Arsenic)
FOREACH VAL (16 TO 21 STEP 1)
  CONTOUR VALUE=VAL LINE=2 COLOR=4
END

PLOT.2D ^AX ^CL

$ Add labels
LABEL X=0.01 Y=-0.8 LABEL="Aluminum"
LABEL X=1.99 Y=-0.8 LABEL="Aluminum" RIGHT
LABEL X=1.0 Y=-0.45 LABEL="BPSG" CENTER
LABEL X=1.0 Y=-0.2 LABEL="Poly" CENTER
LABEL X=0.05 Y=0.35 LABEL="Source"
LABEL X=1.95 Y=0.35 LABEL="Drain" RIGHT

$ Plot the grid
SELECT TITLE="Final Mesh (Delta=0.3)"
PLOT.2D GRID SCALE Y.MAX=3 C.GRID=2

$ Plot arsenic profile in poly
SELECT Z=Log10(Active(Arsenic)) TITLE="Arsenic Concentration in Gate"
PLOT.1D X.V=1 LEFT=-0.4 RIGHT=0 BOT=16 TOP=21 COLOR=4
SELECT Z=Log10(Arsenic)
PLOT.1D X.V=1 ^AX ^CL COLOR=4 LINE=2
LABEL X=-0.16 RIGHT Y=19 SIZE=0.28 COLOR=4 LABEL="Active"
LABEL X=-0.15 LEFT Y=19.7 SIZE=0.28 COLOR=4 LABEL="Total"

```

Figure 5-36 Input file *s4ex7b.inp*, for plotting results

0.8 Micron Device

In order to plot results for the full device, the structure is first reflected about its right edge. The structure is then plotted, with contours of boron, arsenic, and phosphorus concentration. The result is shown in Figure 5-37. The phosphorus LDD regions can be clearly seen, extending from the heavily-doped arsenic source/drain regions to the edge of the poly gate.

N-Channel (Delta=0.3) Final Mesh (Delta=0.3)

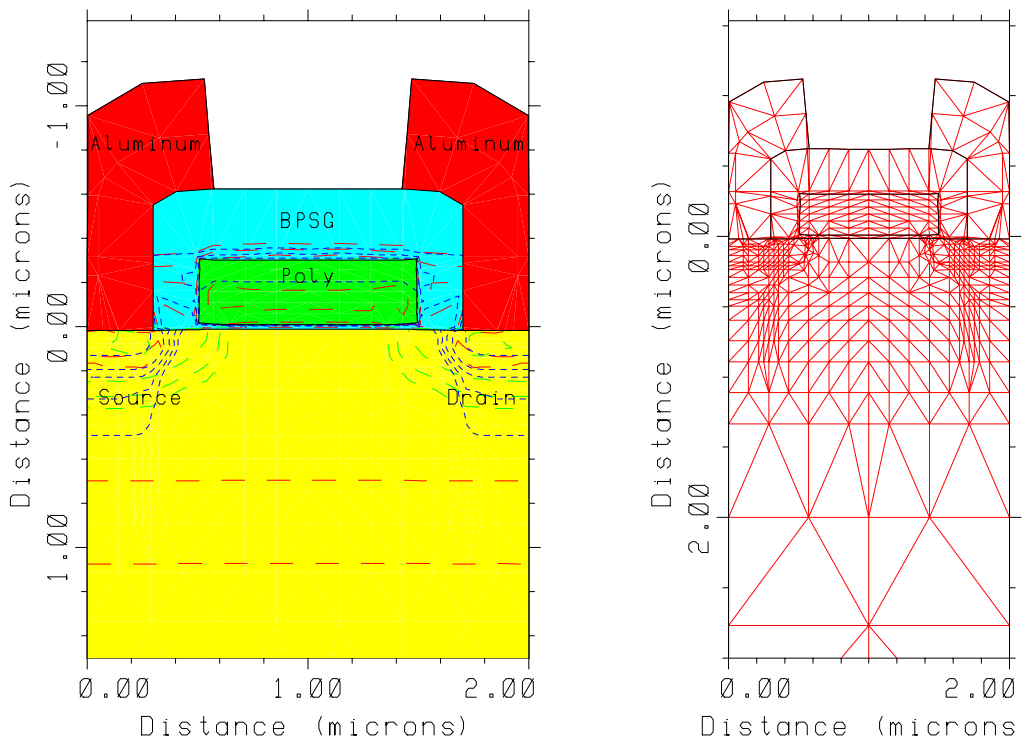


Figure 5-37 Final 0.8 micron structure, NMOS structure, plotted by *s4ex7b.inp* (left) and final mesh for 0.8 micron NMOS structure (right)

Labels have been added to facilitate interpretation of the plot. The labels near the center of the plot have been automatically centered (with the **CENTER** parameter on the **LABEL** statement), and the labels on the right have been right justified (with the **RIGHT** parameter).

Final Mesh

The final mesh is shown in Figure 5-37 (right). Observe the fine grid in the channel, as desired, and a fine grid around the source and drain junctions. Medici can refine the grid or use its own grid, if needed, provided that the doping profiles are accurately represented by the TSUPREM-4 grid.

Arsenic Profiles in Gate

Figure 5-38 shows the arsenic profiles through the poly gate. The active concentration includes only the electrically active arsenic in the interiors of the polycrystalline grains. The total concentration also includes any clustered arsenic in the grain interiors and the arsenic atoms that occupy sites on the grain boundaries.

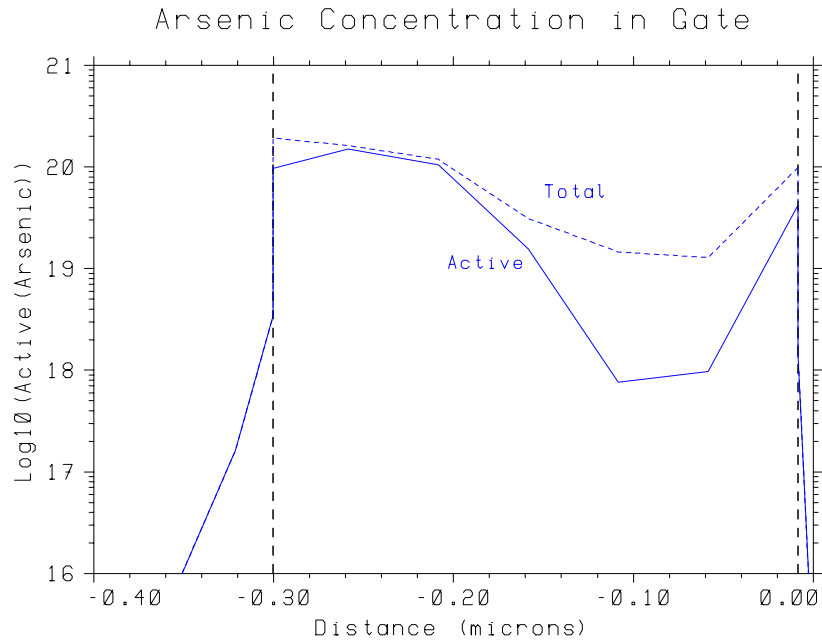


Figure 5-38 Profiles of active and total arsenic concentration through the poly gate

1.2 Micron Device

A third input file (*s4ex7c.inp*, not shown) can be used to plot the results saved by the second pass through the main loop, in the same way that *s4ex7b.inp* plotted the results of the first pass. Figure 5-39 shows the structure plot of the 1.2 micron NMOS transistor, as plotted by input file *s4ex7c.inp*.

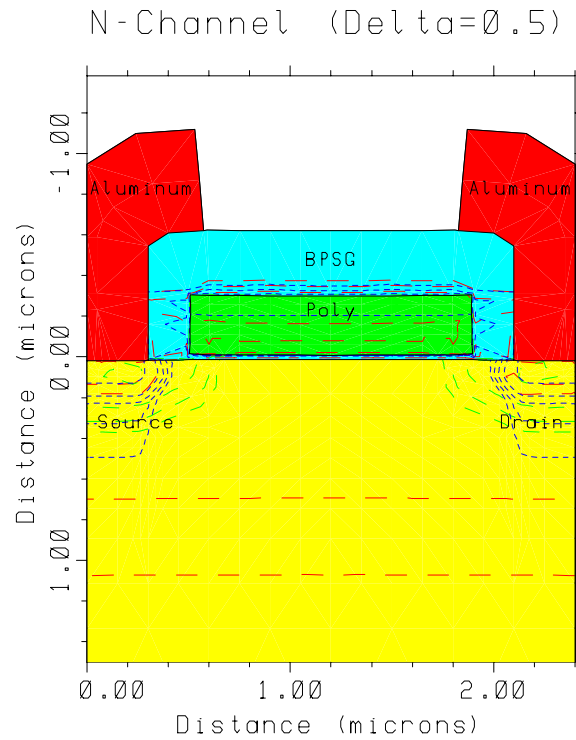


Figure 5-39 Final 1.2 micron NMOS structure, plotted by input file *s4ex7c.inp*

DMOS Power Transistor

This section illustrates the fabrication of a DMOS power transistor structure. The DMOS transistor is formed by diffusing boron into an n-type substrate, under a polysilicon gate. A self-aligned source region is formed by implanting phosphorus, using the poly gate as a mask. Contact to the drain is made at the back side of the wafer. The source metallization also provides contact to the p-type substrate of the n-channel MOS transistor. The TSUPREM-4 statements for simulating this DMOS power transistor are shown in Figures 5-40, 5-42, and 5-45. This particular input file features frequent saving of the structure. This is useful for analyzing the results at intermediate points in the processing, and for recovering from errors that may occur during simulation.

```

$ TSUPREM-4 DMOS Application

$ 1. Set grid spacing and error tolerance
MESH      LY.SURF=0.2  DY.SURF=0.1  LY.ACTIV=5.0  DX.MAX=0.75
METHOD    ERR.FAC=1.5

$ 2. Select models
METHOD    COMPRESS

$ 3. Initialize structure
INITIALIZE WIDTH=10.0  PHOSPHORUS=1E15

$ 4. Plot initial mesh
SELECT    TITLE="Initial Mesh"
PLOT.2D   SCALE  GRID C.GRID=2  Y.MAX=5

$ 5. P-well formation
DEPOSIT   OXIDE  THICK=0.70  SPACES=2
ETCH      OXIDE  RIGHT  P1.X=8.0

DIFFUSE   TEMP=1000  TIME=50  DRYO2

IMPLANT   BORON  ENERGY=50  DOSE=5E15
DIFFUSE   TEMP=950  TIME=120  STEAM
DIFFUSE   TEMP=1100  TIME=50

$ 6. Save the structure
SAVEFILE  OUT.FILE=S4EX8S4

$ 7. Boron contour plots
SELECT    Z=LOG10(BORON)  TITLE="After P-Well Diffusion"
PLOT.2D   SCALE  Y.MAX=5
COLOR     SILICON  COLOR=7
COLOR     OXIDE    COLOR=5
FOREACH   VAL (16 TO 21 STEP 1)
  CONTOUR  VALUE=VAL  LINE=5  COLOR=2
END
PLOT.2D   ^AX  ^CL

$ 8. Gate oxidation
ETCH      OXIDE  LEFT  P1.X=8.1
DIFFUSE   TIME=180  TEMP=1000  DRYO2

```

Figure 5-40 Mesh generation for DMOS power transistor, from input file *s4ex8.inp*

Mesh Generation

This simulation uses automatic grid generation and adaptive gridding. The automatic grid generation parameters have been adjusted to accommodate this large-geometry power device. The depth and spacing of the surface region (**LY.SURF** and **DY.SURF**, respectively) have been doubled and the depth of the active region (**DY.ACTIV**) has been increased from 4 to 5 microns.

A maximum horizontal grid spacing of **DX.MAX**=0.75 is used for this large-geometry device. The error factor for adaptive gridding has been set to 1.5 to reduce the number of nodes and speed up the simulation. A plot of the initial mesh is shown in [Figure 5-41](#).

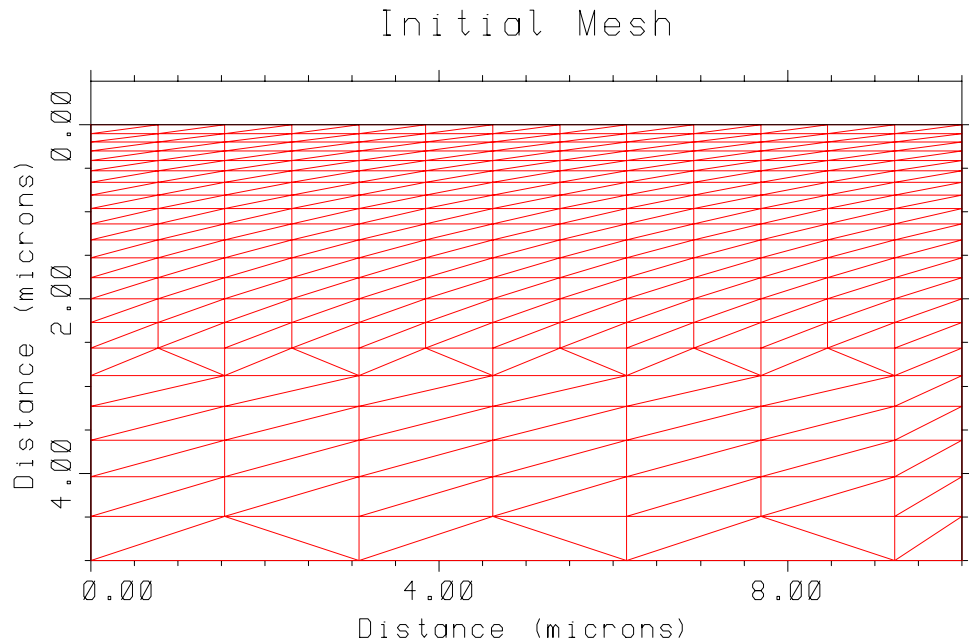


Figure 5-41 Initial grid for simulating DMOS power transistor

Processing the DMOS Power Transistor

Processing of the DMOS structure starts with the deposition of 7000Å of oxide. An opening in this oxide layer is then made and a thin oxide is grown on the exposed silicon. Boron is then implanted through the opening to form the p⁺ contact for the body of the transistor. After implantation, the structure is oxidized, and the boron is driven further with an inert diffusion step. The **COMPRESS** oxidation model is used throughout the simulation to simulate oxidation of nonplanar structures.

```

$ 9. Save the structure
SAVEFILE OUT.FILE=S4EX8S5

$ 10. Print oxide thickness
SELECT Z=1
PRINT.1D LAYERS X.VALUE=0

$ 11. Poly gate formation
DEPOSIT POLY THICK=0.5 PHOS=1E20 SPACES=2
ETCH POLY RIGHT Pl.X=4

$ 12. Body formation
ETCH OXIDE START X=4.0 Y=-10
ETCH OXIDE CONT X=4.0 Y=10
ETCH OXIDE CONT X=7.9 Y=10
ETCH OXIDE END X=7.9 Y=-10

DIFFUSE TIME=40 TEMP=1000 DRYO2

IMPLANT BORON DOSE=5E14 ENERGY=70
DIFFUSE TIME=60 TEMP=950 DRYO2
DIFFUSE TIME=200 TEMP=1100

$ 13. Save the structure
SAVEFILE OUT.FILE=S4EX8S6
    
```

Figure 5-42 Second part of file *s4ex8.inp*: Processing of DMOS power transistor, through body diffusion

The structure after the oxidation and p-well drive is shown in [Figure 5-43](#). Note that some oxide growth has occurred underneath the thick oxide, but not as much as occurred where the original oxide was removed.

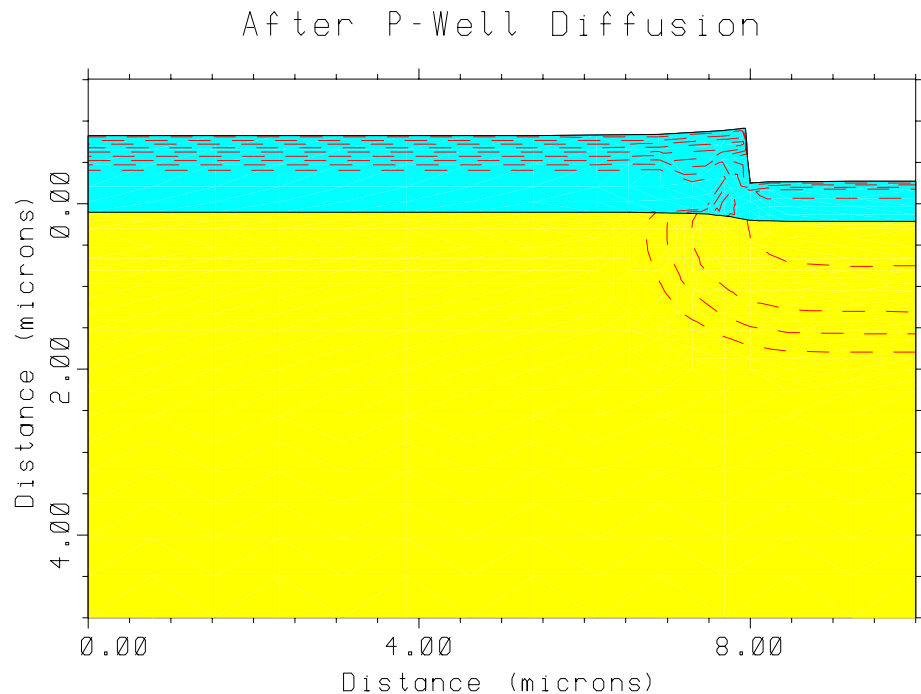


Figure 5-43 Structure with contours of boron concentration, after first p-well diffusion

Gate Processing

Next, the thick oxide is etched from the left portion of the structure, and a gate oxide is grown in this region. Again, one of the numerical oxidation models is required to simulate oxidation with nonuniform initial oxide thickness (**COMPRESS** is still being used). The thickness of the grown gate oxide is printed with the **PRINT .1D LAYERS** statement of step 10.

Next, the polysilicon gate is deposited and patterned, and the oxide is cleared from the source region. Boron is then implanted through the source opening and driven under the poly gate. The structure at this point is plotted in [Figure 5-44](#). Observe that the boron has indeed diffused under the gate, where it forms the substrate of the n-channel transistor.

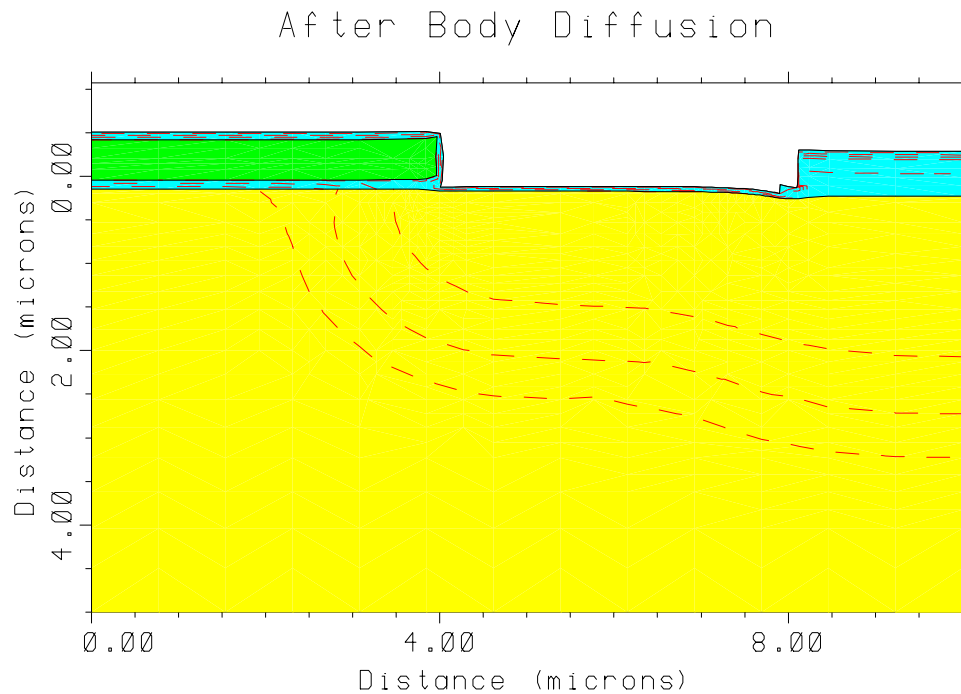


Figure 5-44 DMOS power transistor after p-type body diffusion

Source Processing

The opening for the source implant is now cleared and a thin oxide is deposited. Phosphorus is then implanted and annealed to form the source of the transistor. Because the implantation is through an oxide, the *phosphorus* implant table which does not include channelling, is used. BPSG is deposited to form an insulating layer over the gate. Finally, a contact hole is opened for the source and p-type body contact, and aluminum is deposited. The final structure is shown in [Figure 5-46](#) (right). The silicon region has been labeled according to the doping type, and the other regions according to the type of material.


```

$ 14. Boron contour plots
SELECT      Z=LOG10(BORON)  TITLE="After Body Diffusion"
PLOT.2D     SCALE  Y.MAX=5
COLOR      SILICON  COLOR=7
COLOR      OXIDE    COLOR=5
COLOR      POLY     COLOR=3
FOREACH     VAL (16 TO 21 STEP 1)
  CONTOUR   VALUE=VAL  LINE.TYP=5  COLOR=2
END
PLOT.2D     ^AX  ^CL

$ 15. Source/drain formation
ETCH       OXIDE  START  X=4.1  Y=-10
ETCH       OXIDE  CONT   X=4.1  Y=10
ETCH       OXIDE  CONT   X=7.9  Y=10
ETCH       OXIDE  END    X=7.9  Y=-10

DEPOSIT    OXIDE  THICK=0.03

IMPLANT    PHOS  DOSE=1E15  ENERGY=50  IMPL.TAB=PHOSPHORUS
DIFFUSE    TEMP=950  TIME=60  DRYO2

$ 16. BPSG and metallization
DEPOSIT    OXIDE  THICK=1.00
DIFFUSE    TEMP=950  TIME=60
ETCH       OXIDE  RIGHT  P1.X=6

DEPOSIT    ALUMINUM  THICK=1.5

$ 17. Save the structure
SAVEFILE   OUT.FILE=S4EX8S7

$ 18. Plot final mesh
SELECT     TITLE="Final Mesh"
PLOT.2D    SCALE  GRID  C.GRID=2  Y.MAX=5

$ 19. Boron and phosphorus contour plots
SELECT     Z=LOG10(BORON)  TITLE="Final structure"
PLOT.2D    SCALE  Y.MAX=5
COLOR     SILICON  COLOR=7
COLOR     OXIDE    COLOR=5
COLOR     POLY     COLOR=3
COLOR     ALUMINUM  COLOR=2
FOREACH    VAL (16 TO 21 STEP 1)
  CONTOUR  VALUE=VAL  LINE.TYP=5  COLOR=2
END

SELECT     Z=LOG10(PHOS)
FOREACH    VAL (16 TO 21 STEP 1)
  CONTOUR  VALUE=VAL  LINE.TYP=4  COLOR=4
END

PLOT.2D    ^AX  ^CL

LABEL     X=9.8  Y=-0.5  LABEL="Aluminum"  RIGHT
LABEL     X=1    Y=-0.9  LABEL="BPSG"
LABEL     X=1    Y=-0.1  LABEL="Poly"
LABEL     X=0.5  Y=1.5   LABEL="n-"
LABEL     X=9.8  Y=1.0   LABEL="p+"  RIGHT
LABEL     X=3.1  Y=0.4   LABEL="p-"
LABEL     X=5.2  Y=0.4   LABEL="n+"
    
```

Figure 5-45 Third part of *s4ex8.inp*: Final processing and plotting

Figure 5-46 (left) shows the final grid for the structure. Observe that the mesh has been refined (automatically, by adaptive gridding) where needed to resolve the impurity profiles. Note that the grid refinement is primarily vertical in the center of the source region where the profiles are essentially one-dimensional, but occurs in both directions at the corners of the source diffusion, where two-dimensional effects are important.

The final structure is shown in Figure 5-46 (right).

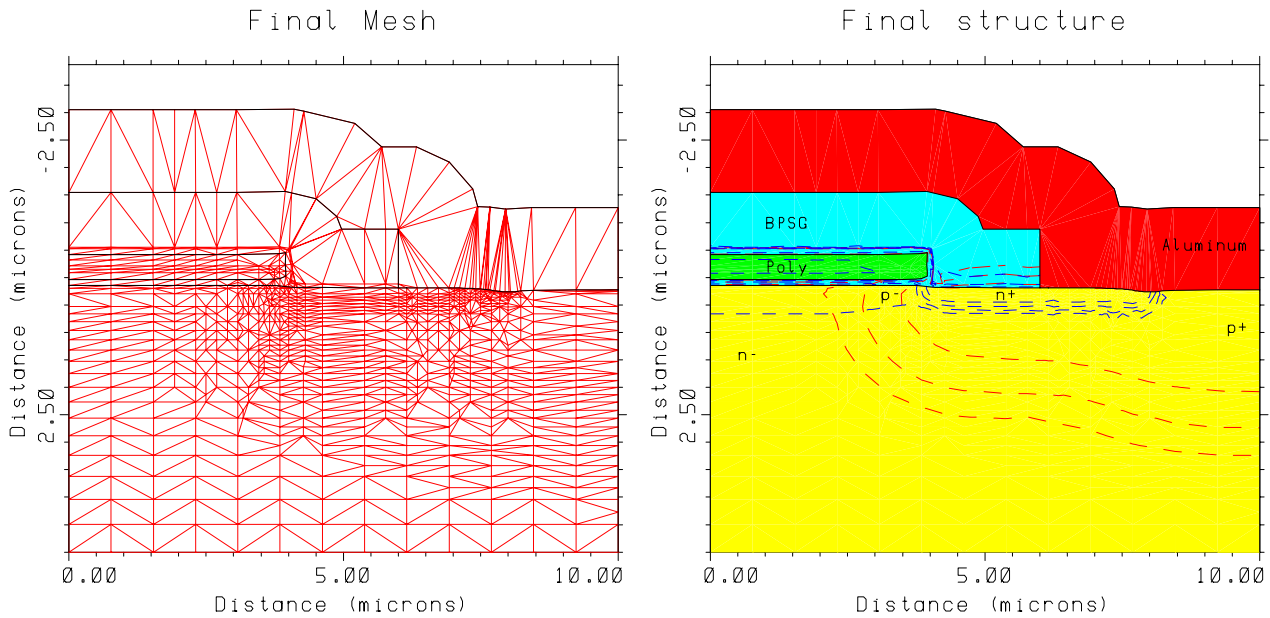


Figure 5-46 Final mesh for DMOS simulation (left), showing the result of adaptive gridding, and Final DMOS power transistor structure (right), produced by input file *s4ex8.inp*

Summary

The process shown here does not necessarily produce an ideal DMOS structure, nor is the simulation as accurate as it might be. Of particular importance are the conditions of the final drive of the p-type body diffusion in step 22 and the possibility that impurities may diffuse from the polysilicon gate through the gate oxide and into the channel. Observe in the final plot that there has apparently been some diffusion of both boron and phosphorus from the poly into the channel. Another concern is that oxidation-enhanced diffusion effects may be important in this example. This can be checked by repeating the simulation using the **PD.TRANS** point defect model.

SOI MOSFET

This example shows how TSUPREM-4 can be used to simulate a silicon-on-insulator (SOI) process. It illustrates several features of the program that have not been used in the other examples, including the use of the **REGION** statement and the technique for depositing layers with nonuniform grid spacings. The input statements from file *s4ex9.inp* are shown in Figures 5-47 and 5-49.

```

$ TSUPREM-4 SOI Structure (0.2um epi) Application

$ Specify x mesh
LINE X LOCATION=0 SPACING=0.25 TAG=LEFT
LINE X LOCATION=0.5 SPACING=0.025
LINE X LOCATION=0.6 SPACING=0.025
LINE X LOCATION=1.05 SPACING=0.1 TAG=RIGHT

$ Specify y mesh
LINE Y LOCATION=0 SPACING=0.02 TAG=OXTOP
LINE Y LOCATION=0.5 SPACING=0.15 TAG=OXBOTTOM
LINE Y LOCATION=2.0 SPACING=1.0 TAG=SIBOTTOM

ELIMINATE COLUMNS X.MIN=0.45 X.MAX=0.7 Y.MIN=0.05
ELIMINATE COLUMNS Y.MIN=0.5

$ Define isolation oxide and silicon substrate
REGION OXIDE XLO=LEFT XHI=RIGHT YLO=OXTOP YHI=OXBOTTOM
REGION SILICON XLO=LEFT XHI=RIGHT YLO=OXBOTTOM YHI=SIBOTTOM
INITIALIZE <100> BORON=1E15

$ Deposit epi with nonuniform vertical grid spacing
DEPOSIT SILICON BORON=1E17 THICKNESS=0.1 SPACES=5 DY=0.01 YDY=0.1
DEPOSIT SILICON BORON=1E17 THICKNESS=0.1 SPACES=5 DY=0.01

$ Plot initial mesh
SELECT Z=1 TITLE="Initial Mesh"
PLOT.2D GRID Y.MAX=1 SCALE C.GRID=2
PRINT.1D X.VALUE=0.0 LAYERS

```

Figure 5-47 Mesh generation for SOI MOSFET, from input file *s4ex9.inp*

Mesh Generation

The structure to be simulated consists of a silicon wafer with a half micron of oxide and 0.2 microns of silicon on top. The active device is fabricated in the top silicon layer. The actual structure is formed by implantation of oxygen and recrystallization of the silicon layer on top, a process that cannot be simulated by TSUPREM-4. Instead, an initial structure is defined consisting of the silicon substrate and the oxide layer, on which the top silicon layer is deposited. Note that due to the symmetry of the final structure, only the left half needs to be simulated.

The silicon substrate and oxide layers are defined with **LINE** statements, as in previous examples. In this case, the **TAG** parameter must be added to grid lines on structure boundaries and material interfaces. Thus, in the *y* direction the tag *OXTOP* is used to mark the top of the oxide, *OXBOTTOM* to mark the oxide/silicon interface, and *SIBOTTOM* to mark the bottom of the structure. **REGION** state-

Depositing a Layer with Nonuniform Grid Spacing

ments are added to identify the silicon and oxide regions. The **REGION** statements use the tags to specify the extent of the regions. Thus, for example, it is specified that the oxide region extends from the grid line tagged *OXTOP* to the line tagged *OXBOTTOM*, and from *LEFT* to *RIGHT* in the horizontal direction.

The top silicon layer is added using **DEPOSITION** statements. To accurately simulate the final structure with the Medici device simulator, a fine grid spacing (about 100Å) is needed near each silicon/oxide interface. To reduce simulation time, a coarser grid is specified in the middle of the layer. Thus, the ideal mesh would have a spacing that varies smoothly from fine to coarse to fine.

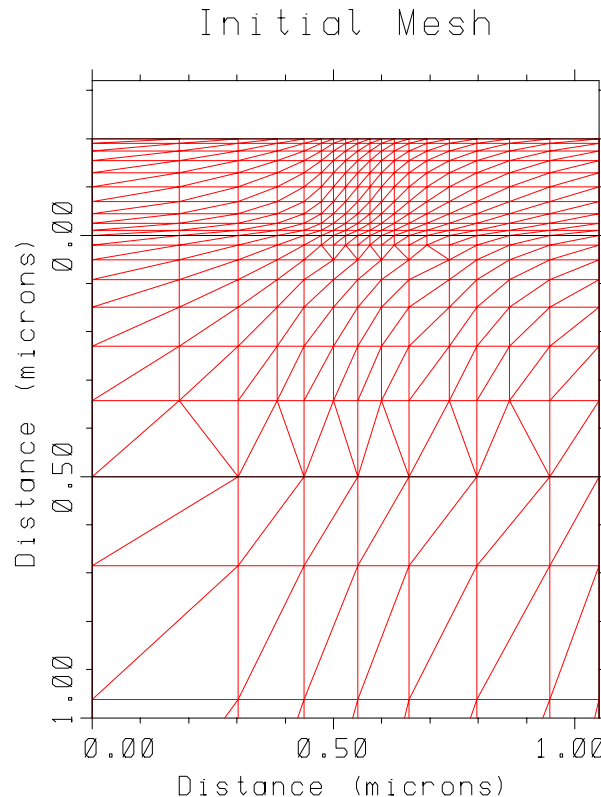


Figure 5-48 Initial grid for simulating SOI MOSFET

A mesh of this type (shown in [Figure 5-48](#)) can be generated using the **DY** and **YDY** parameters on the **DEPOSITION** statement. The **DY** parameter specifies the desired grid spacing at some depth in the deposited layer specified by the **YDY** parameter. The grid spacing above and below this depth is increased or decreased to produce the specified number of spaces. To specify the grid spacing at two points in the layer (i.e., at each interface), two **DEPOSITION** statements must be used. Each specifies a 1000Å layer of silicon divided into five grid spaces, as specified by the **SPACES** parameter. In each case the **DY** parameter is used to specify the desired grid spacing of 0.01 microns. In the first case the 0.01-micron grid spacing occurs 0.1 microns below the top of the layer (i.e., at the bottom of the layer), while in the second case the default value of **YDY**=0 is used, so the fine grid spacing occurs at the top of the layer. Note that the same mesh could have been generated by adding **LINE** and **REGION** statements to define the top silicon layer. The difficulty is that only a single doping concentration can be specified for

the entire structure on the **INITIALIZE** statement, and it was necessary to specify a doping of $10^{17}/\text{cm}^3$ in the top layer, but only $10^{15}/\text{cm}^3$ in the lower layers.

```

$ Use vertical oxidation model to grow gate oxide
DIFFUSE    TIME=55 TEMP=950 DRYO2

SELECT     Z=1
PRINT.1D   X.VALUE=0.0 LAYERS

$ Deposit, dope, and pattern poly
DEPOSIT    POLY THICK=0.3 SPACES=2
METHOD     COMPRESS
DIFFUSE    TIME=30 TEMP=800 PHOS=1E20
ETCH      POLY LEFT P1.X=0.5

$ Source/drain implant and anneal
IMPLANT    ARSENIC DOSE=4E15 ENERGY=45
DIFFUSE    TIME=70 TEMP=950

$ BPSG and contact holes
DEPOSIT    OXIDE THICK=0.3
DEPOSIT    PHOTORESIST THICK=1.0
ETCH      PHOTORESIST LEFT P1.X=0.3
ETCH      OXIDE TRAP THICK=0.7 ANGLE=85
ETCH      PHOTORESIST ALL

$ Metallization
DEPOSIT    ALUMINUM THICK=0.4
DEPOSIT    PHOTORESIST THICK=1.0
ETCH      PHOTORESIST RIGHT P1.X=0.6
ETCH      ALUMINUM TRAP THICK=0.7 ANGLE=85
ETCH      PHOTORESIST ALL

$ Reflect structure, then save complete MOSFET
SAVEFILE   OUT.FILE=S4EX9S1
STRUCTURE  REFLECT RIGHT
SAVEFILE   OUT.FILE=S4EX9P2 MEDICI

$ Plot grid & profiles for complete structure
SELECT     Z=DOPING TITLE="SOI MOSFET"
PLOT.2D    SCALE Y.MAX=1
COLOR     SILICON COLOR=7
COLOR     OXIDE COLOR=5
COLOR     POLY COLOR=3
COLOR     ALUMINUM COLOR=2
FOREACH    X ( 16 TO 21 STEP 1 )
  CONTOUR  VALUE=( 10^ X ) COLOR=4 LINE.TYP=2
  CONTOUR  VALUE=(-(10^ X )) COLOR=2 LINE.TYP=5
END
PLOT.2D    ^AX ^CL

PLOT.2D    GRID SCALE Y.MAX=1 C.GRID=2

$ Plot 1D concentration profiles
SELECT     Z=LOG10(ACTIVE(BORON)) TITLE="Doping Profiles" +
          LABEL="log(Active Concentration)"
PLOT.1D    X.VALUE=1 BOTTOM=15 TOP=21 LEFT=-0.3 RIGHT=0 LINE.TYP=5 +
          COLOR=2
LABEL      LABEL="Boron (x=1)" X=-0.1 Y=17

SELECT     Z=LOG10(ACTIVE(ARSENIC))
PLOT.1D    X.VALUE=0 ^AXES ^CLEAR LINE.TYP=2 COLOR=3
LABEL      LABEL="Arsenic (x=0)" X=-0.1 Y=20.3
    
```

Figure 5-49 Processing of SOI MOSFET, from input file *s4ex9.inp*

Process Simulation

The processing of the SOI MOSFET is straightforward. Note that the analytical oxidation models cannot be used in this example, even though the surface is planar and unoxidized, because of the existence of the lower oxide/silicon interfaces.

The **VERTICAL** model can be used, but only because there is no path for oxygen to diffuse to the lower oxide layer, and thus no growth at the lower interfaces. If there were a path for oxygen to diffuse to the lower oxide, the **COMPRESS** or **VISCOEL** model would be needed, and if it were necessary to model lifting of the upper silicon layer, the **VISCOUS** model with **SKIP.SIL** set false would be needed.

Source/drain contacts have been opened and aluminum has been deposited and patterned in order to form contacts for Medici. The structure is reflected along the right edge to form a full device, and saved in Medici format in file *S4EX9P2*. The full, final structure is shown in [Figure 5-50](#) on the left; the final grid is plotted in on the right.

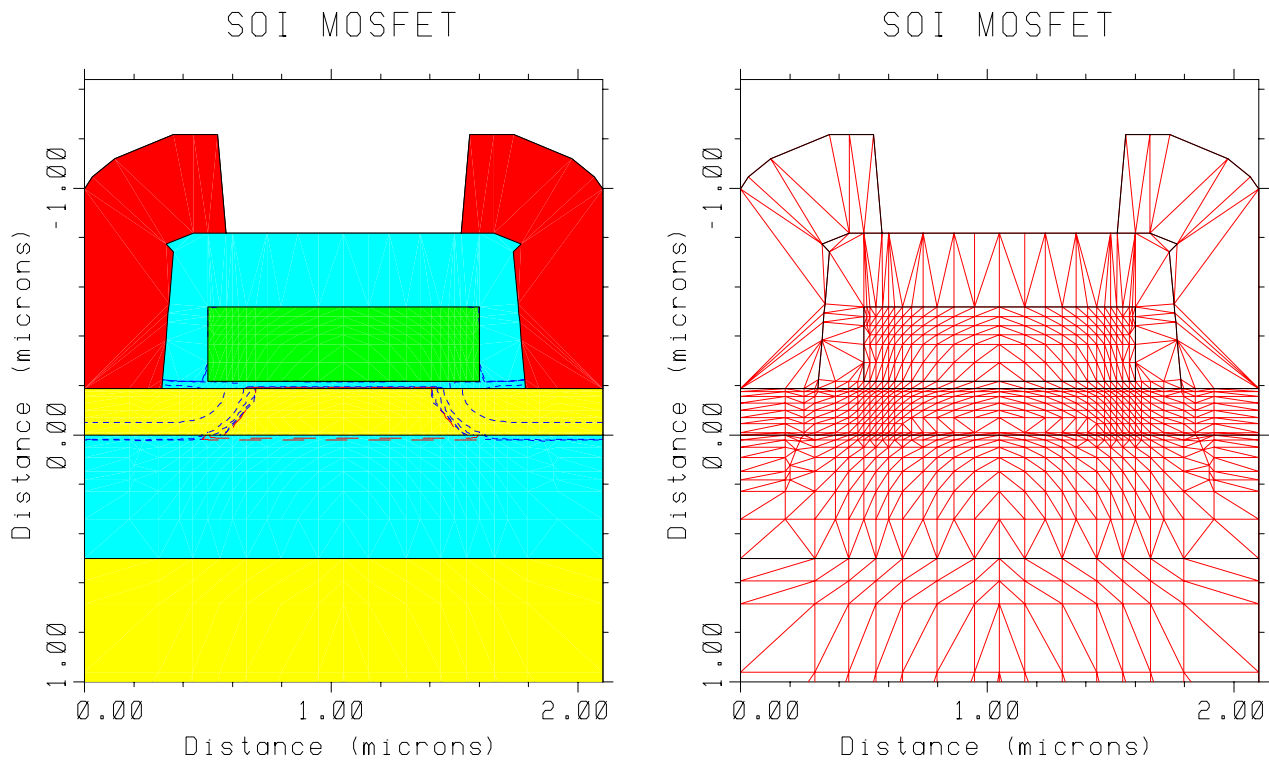


Figure 5-50 Final structure, showing contours of net doping for SOI MOSFET (left) and final grid for SOI MOSFET (right)

[Figure 5-51](#) shows the doping profiles in the final structure. The arsenic profiles in the source/drain ($x=0$) and the boron profiles in the channel ($x=1$) are shown.

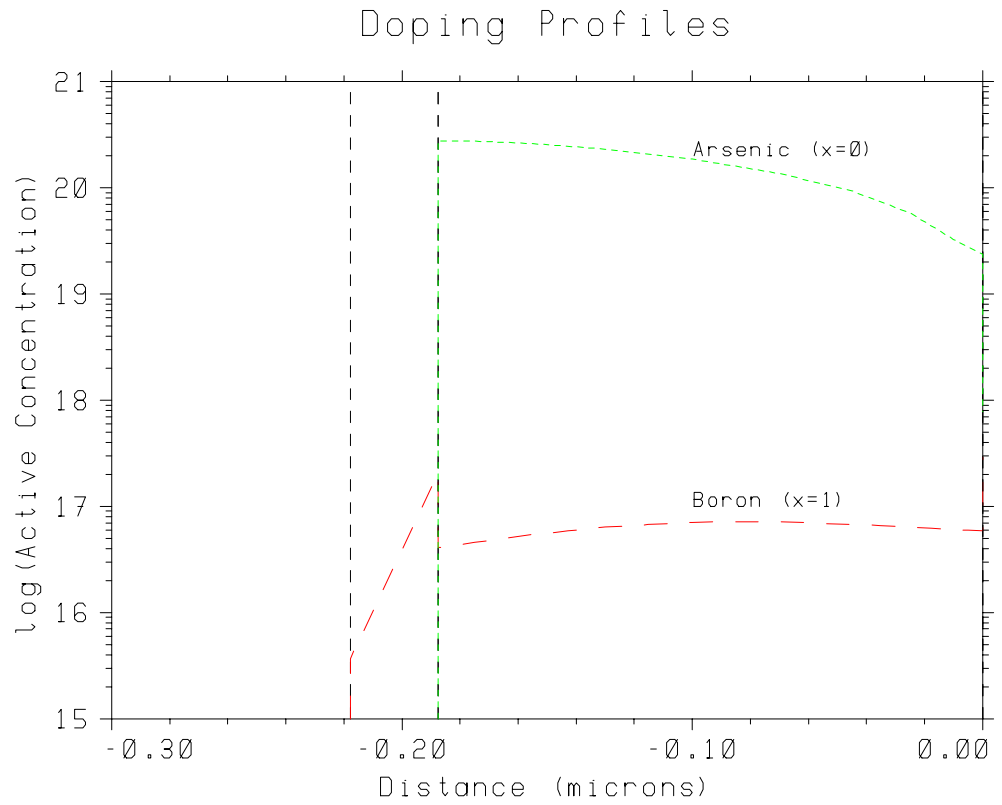


Figure 5-51 Channel and source/drain doping profiles for SOI MOSFET

MOSFET with Self-Aligned Silicides

This example illustrates the simulation of silicide growth using TSUPREM-4. The input files for this example are shown in [Figures 5-52, 5-53, and 5-54](#). The statements are separated into two files—*s4ex10.inp* contains the processing statements for performing the simulation, while *s4ex10p.inp* contains the statements for plotting the structure.

Preparation for Silicidation

[Figure 5-52](#) shows the statements used to initialize the structure, grow the gate oxide, deposit and pattern the poly gate, form the lightly-doped source/drain extension and the heavily-doped source/drain, and clear the oxide from the poly gate and the source/drain in preparation for silicidation. Automatic grid generation is used in both the vertical and horizontal directions, but because the structure is very small it is appropriate to reduce the depth of the active region and the grid space is provided automatically by adaptive gridding.

```

$ TSUPREM-4 Silicidation Example

MESH      GRID.FAC=0.9

MESH      DY.SURF=0.03  LY.ACTIV=0.5  DY.ACTIV=0.1
INITIALIZE BORON=1.0E15  WIDTH=0.6  DX=0.1

$ Grow gate oxide
DIFFUSION TIME=25  TEMP=850  HCL=4.5  DRYO2

$ Deposit and pattern gate
DEPOSIT  POLYSILI  THICKNES=0.2  SPACES=5  DY=0.02
ETCH     POLYSILI  RIGHT P1.X=0.125
ETCH     OXIDE     RIGHT P1.x=0.125

$ Oxidize
METHOD   COMPRESS
DIFFUSION TIME=25  TEMP=850  HCL=4.5  DRYO2

$ LDD implant
IMPLANT  ARSENIC  DOSE=5E13  ENERGY=30  IMPL.TAB=ARSENIC

$ Form sidewall spacer
DEPOSIT  OXIDE  THICKNESS=0.126  SPACES=2
ETCH     OXIDE  TRAPEZOI THICKNESS=0.175

$ Heavy source/drain implant and anneal
IMPLANT  ARSENIC  DOSE=1E15  ENERGY=60  IMPL.TAB=ARSENIC
METHOD   PD.TRANS
DIFFUSION TIME=10  TEMP=900  DRYO2

$ Isotropic etch to clear gate and source/drain
ETCH     OXIDE  TRAP THICK=0.015  ANGLE=45

```

Figure 5-52 First part of input file *s4ex10.inp*: NMOS transistor processing

Silicidation

Figure 5-53 shows the statements for growing the self-aligned titanium silicide layer. First, titanium is deposited over the entire structure. Where titanium comes into contact with silicon or polysilicon, a thin (2 nm, by default) layer of TiSi_2 is inserted automatically by the program. The structure after titanium deposition is shown in Figure 5-55. The next step is to anneal for a little over a minute at 650 degrees. Because it is expected that impurity diffusion is negligible, the **PD.FERMI** model is selected to reduce the simulation time.

During silicide growth, silicon atoms enter the silicide layer and diffuse to the TiSi_2/Ti interface where they react to form more TiSi_2 . Silicon and titanium are consumed at their respective interfaces with the silicide, and silicide is formed at the TiSi_2/Ti interface. The deformation of the various layers due to the consumption of silicon and titanium and the production of titanium silicide is calculated using the **COMPRESS** model for viscous flow

```

$ Deposit titanium
DEPOSIT MAT=TITANIUM THICK=0.025 SPACES=2

$ Plot structure before silicidation
SELECT TITLE="Before Silicidation"
SOURCE s4ex10p.inp

$ Grow the silicide
METHOD PD.FERMI
DIFFUSION TIME=1.287 TEMP=650

$ Plot structure after silicidation
SELECT TITLE="After Silicidation"
SOURCE s4ex10p.inp

$ Final structure
ETCH MAT=TITANIUM ALL
STRUCTURE REFLECT LEFT

$ Plot structure after silicidation
SELECT TITLE="After Titanium Etch"
SOURCE s4ex10p.inp

$ Print layer information
SELECT Z=DOPING
PRINT.1D X.V=0.6 LAYERS
PRINT.1D X.V=0.0 LAYERS

```

Figure 5-53 Second part of input file *s4ex10.inp*: Silicide growth

```

$ Plot sequence for silicidation example
$
$ To use:
$   select title="Title"
$   source s4ex10p.inp
$
PLOT.2D  Scale Y.MAX=0.3
COLOR    MAT.POLY    COLOR=3
SELECT   Z=LOG10(ARSENIC)
FOREACH  X ( 15 TO 20 )
    CONTOUR VAL=@{X} COLOR=3 LINE=3
END
COLOR    MAT=TITANIUM COLOR=2
COLOR    MAT=TiSi2    COLOR=6
COLOR    MAT=OXIDE    COLOR=5
LABEL    X=0.36 Y=-0.23 SIZE=0.3 RECT C.RECT=2 LABEL="Titanium"
LABEL    X=0.36 Y=-0.18 Size=0.3 Rect C.RECT=6 LABEL="TiSi2"
LABEL    X=0.36 Y=-0.13 SIZE=0.3 RECT C.RECT=3 LABEL="Polysilicon"
LABEL    X=0.36 Y=-0.08 Size=0.3 Rect C.RECT=5 LABEL="Oxide"
LABEL    X=0.55 Y=0.25 Size=0.3 Line=3 C.LINE=3 LABEL="Arsenic" RIGHT

```

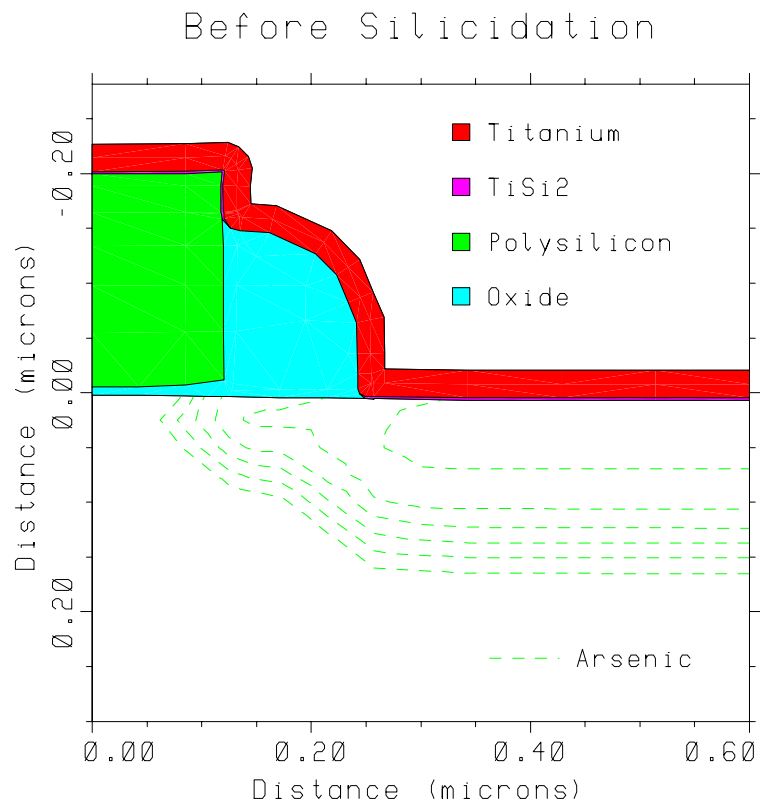
Figure 5-54 Input file *se4ex10p.inp*: Plotting results

Figure 5-55 Structure immediately before silicide growth step

The structure after the silicide growth step is shown in [Figure 5-56](#). Observe that most of the titanium has been consumed. After removing the remaining titanium the full structure is shown in [Figure 5-57](#).

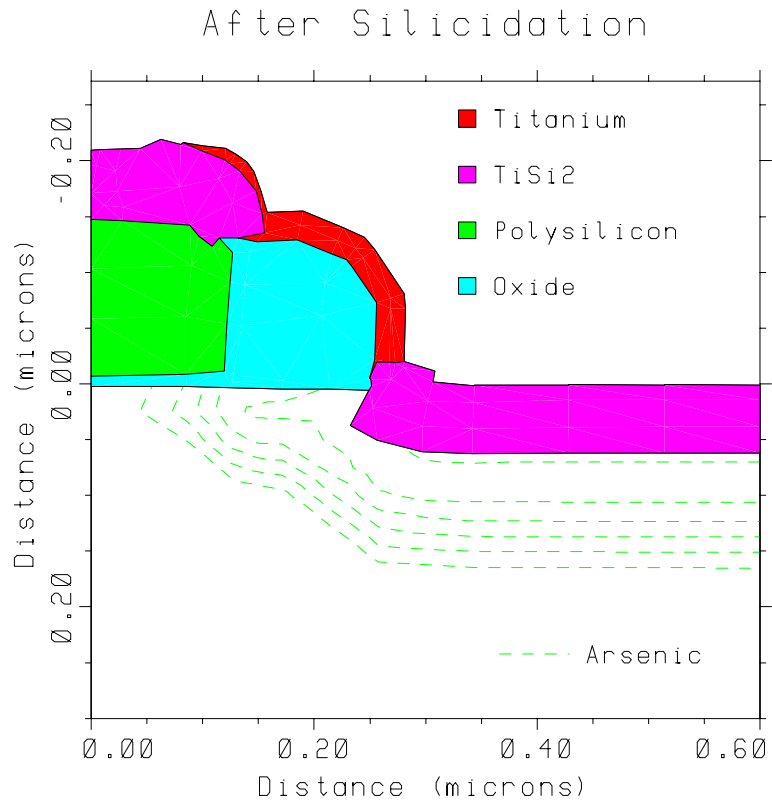


Figure 5-56 Structure after silicide growth step

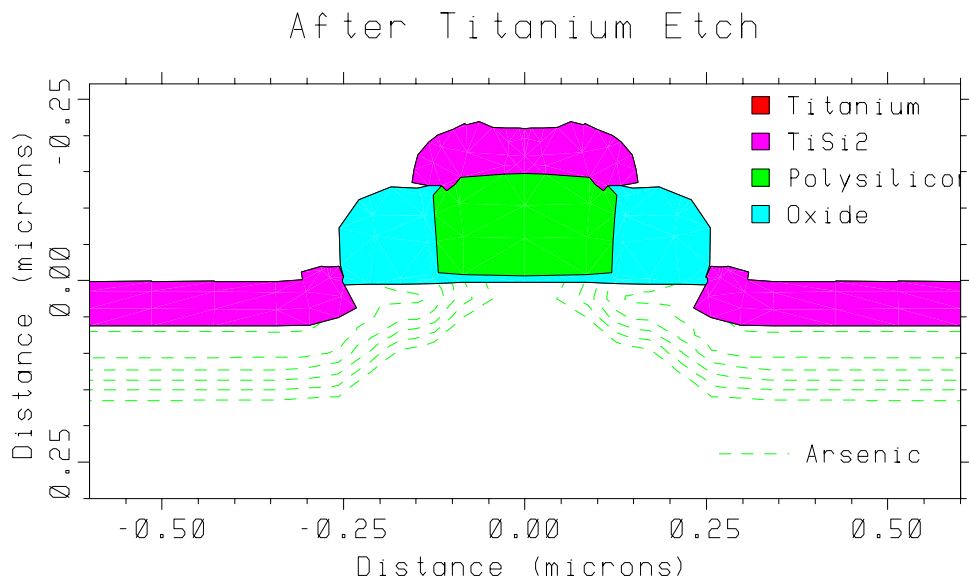


Figure 5-57 Final structure, after removal of remaining titanium

Polysilicon Emitter Study

This example shows a simulation of the emitter region of a double-polysilicon bipolar transistor. The effect of the arsenic out-diffusion from the emitter polysilicon on the emitter and base widths is shown. The simulations can be used to examine the dependence of the device performance on the width of the emitter stripe. The advanced polysilicon grain-growth and diffusion models are used to simulate the distribution of arsenic in the emitter.

Process Simulation

Figure 5-58 shows the input file *s4ex11a.inp* for simulating the emitter region of the bipolar transistor. The emitter width is defined by the name **EMITWID**. The input file *s4ex11b.inp* for simulating the 2-micron emitter is identical except for the values of **IDENT** (which is set to "B") and the value of **EMITWID** which is set to 2.0.

The critical dimensions of the structure are specified by defining the parameters **EMITSTA**, **EMITWID**, and **SPACERWID**. The grid spacings are set to give a reasonable compromise between speed and accuracy. Adaptive gridding is used to increase the grid density where the arsenic diffuses to form the emitter; because of the shallowness of the emitter, the minimum grid spacing for adaptive gridding on arsenic is decreased to 0.01 micron. The accuracy of the results can be verified by reducing the value of **GRID.FAC** for later simulations.

Models

Because oxidation-enhanced and transient-enhanced diffusion are not expected to be significant in this structure, the **PD.FERMI** diffusion model can be used with a shallow simulation structure. Because of the high arsenic concentration in the emitter, however, it would be wise to check the results using the **PD.FULL** model and a deep simulation structure.

Processing

The emitter structure is formed by a series of deposition and etch steps, culminating in the deposition of the poly layer in the trench formed by the emitter opening. The temperature of the poly deposition is 620°C; this temperature determines the initial grain size. The emitter is then implanted with arsenic at a 7-degree tilt. Because of the tilt, the right sidewall of the emitter receives more of the implant dose than the left sidewall, but much less than the bottom of the emitter opening. The arsenic is then diffused throughout the poly and into the single-crystal silicon with a high-temperature RTA step.

```

$ Parameters
%DEFINE IDENT      A
%DEFINE EMITSTA 0.5
%DEFINE EMITWID 1.0
%DEFINE SPACERWID 0.2

$ Half the structure
MESH  GRID.FAC=1.5

LINE  X LOC=0.0 SPACING=0.2
LINE  X LOC=@{EMITSTA}+@{SPACERWID}/2 SPACING=0.05
LINE  X LOC=@{EMITSTA}+@{EMITWID}/2 SPACING=0.15
LINE  Y LOC=0.0 SPACING=0.025
LINE  Y LOC=0.1 SPACING=0.025
LINE  Y LOC=1.0 SPACING=0.1
ELIMINATE COLUMNS Y.MIN=0.2
ELIMINATE COLUMNS Y.MIN=0.2
INITIAL <100> BORON=1E14

METHOD  MAT=SILICON IMP=ARSENIC MIN.SPAC=0.01

$ Emitter Cut
DEPOSIT OXIDE THICK=0.6 DY=0.05
ETCH    OXIDE RIGHT P1.X=@{EMITSTA}

$ Dielectric Screening Layers
DEPOSIT OXIDE THICK=0.04 SPACES=2
DEPOSIT NITRIDE THICK=0.05 SPACES=2

$ Spacer
ETCH NITRIDE RIGHT P1.X=@{EMITSTA}+@{SPACERWID}

$ Oxide dip
ETCH OXIDE  TRAPEZ  THICK=0.06 UNDERCUT=0.06

$ Build Full Structure
STRUCTUR REFLECT RIGHT

$ N-poly
DEPOSIT POLYSILI THICK=0.22 ARC.SPAC=0.05 DY=0.02 TEMPERAT=620

$ N+ implant
IMPLANT DOSE=1.0E+16 ENERGY=50 ARSENIC TILT=7.0

SAVE OUT.FILE=S4EX11@{IDENT}S1

$ Emitter drive-in
DIFFUSIO TEMPERAT=1125. TIME=0.5 NITROGEN

SAVE OUT.FILE=S4EX11@{IDENT}S2

```

Figure 5-58 Listing of input file *s4ex11a.inp* for simulating the bipolar emitter structure

Plotting the Results

The results are now plotted for the following structures:

- After Implant
- Doping and Grain Size
- Doping vs. Stripe Width

After Implant

Figure 5-59 shows the layers of the structure and the arsenic concentration in the poly immediately after the implant. This figure and the following figures were plotted using the input file *s4ex11c.inp*, which illustrates techniques for producing multiple plots on a single screen and for annotating shaded contour plots. The first part of *s4ex11c.inp* is shown in Figure 5-60.

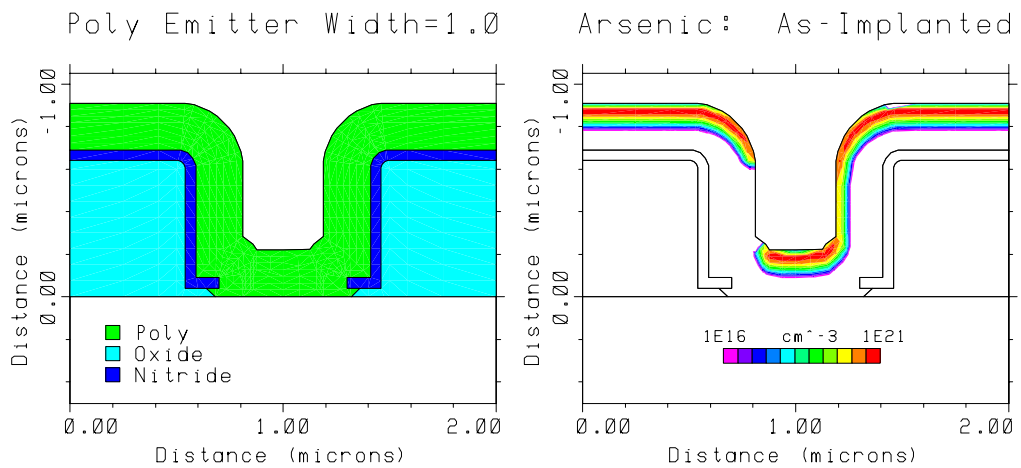


Figure 5-59 Bipolar emitter structure and as-implanted arsenic profiles, as plotted using *s4ex11c.inp*

```

$ Parameters for plot size and placement
%define YMAX 0.5
%define YLEN 5.8
%define XOFF1 1.5
%define XOFF2 10.8
%define LSIZ 0.3
%define TSIZ 0.35

$ As-implanted structure
initial in.file=S4EX11AS1

$ Plot structure
select title="Poly Emitter Width=1.0"
plot.2d scale y.max=@{YMAX} x.off=@XOFF1 y.len=@YLEN +
  t.siz=@TSIZ x.siz=@LSIZ y.siz=@LSIZ
color poly color=3
color oxide color=5
color nitride color=4
label label="Poly" c.rect=3 size=0.25 x=0.3 y=0.2
label label="Oxide" c.rect=5 size=0.25 x=0.3 y=0.3
label label="Nitride" c.rect=4 size=0.25 x=0.3 y=0.4
plot.2d ^ax ^cl

$ Plot as-implanted arsenic
select z=log10(arsenic) title="Arsenic: As-Implanted"
plot.2d scale y.max=@{YMAX} ^cl x.off=@XOFF2 y.len=@YLEN +
  t.siz=@TSIZ x.siz=@LSIZ y.siz=@LSIZ
%define TOP 1E21
%define BOTT 1E16
%define INC (log10(@{TOP})-log10(@{BOTT}))/10
foreach I (0 to 10)
  color min.value=(log10(@{BOTT}) + @{INC}* ( I )) +
    max.value=(log10(@{BOTT}) + @{INC}* ( I +1)) +
    color=(18- I )
  label rectangle c.recta=(18- I ) c.siz=0.25 +
    x=(@{XOFF2}+2.6+0.25*( I )) y=2.5 cm
end
label label="@{BOTT}" c.siz=0.2 x=@{XOFF2}+2.1 y=2.75 cm
label label="cm^-3 @{TOP}" c.siz=0.2 x=@{XOFF2}+3.5 y=2.75 cm
plot.2d ^ax ^cl

```

Figure 5-60 First part of *s4ex11c.inp*, for plotting the structure and contours of as-implanted arsenic concentration

Doping and Grain Size

Figure 5-61 shows the total arsenic concentration and the poly grain size after the RTA step. The arsenic concentration is largest at the left and right edges of the structure, where the initial doping is high and the arsenic is confined to the poly layer by the underlying nitride layer. The concentration is lowest in the sidewalls of the emitter opening, where the as-implanted doping is low. The concentration at the center of the emitter opening is lower than that at the left and right edges of the structure because of diffusion into the sidewalls of the opening and into the single-crystal silicon. The rate of grain growth during high-temperature processing

depends strongly on the doping concentration; thus the final contours of grain size are similar to the final contours of total arsenic concentration.

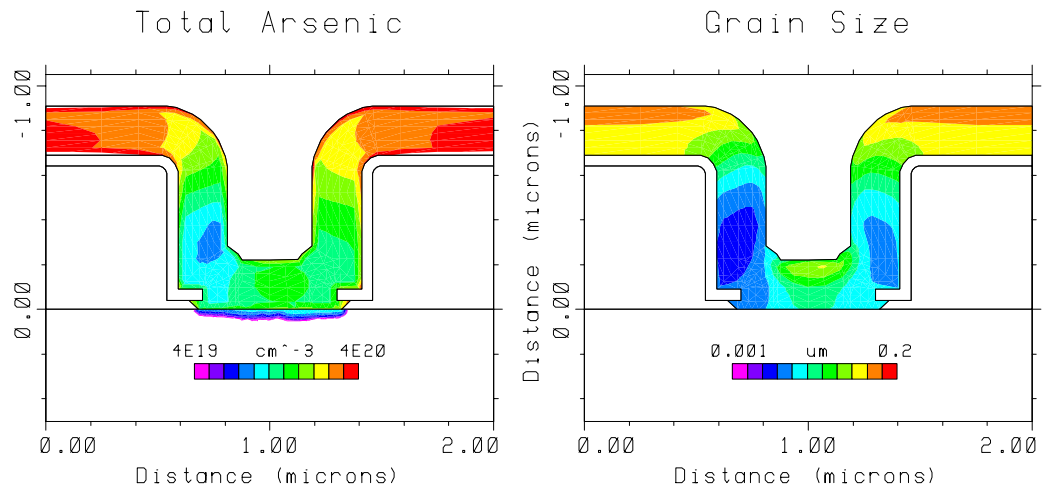


Figure 5-61 Contours of total arsenic concentration and poly grain size after RTA

Doping vs. Stripe Width

The depletion of the arsenic in the center of the emitter opening is a strong function of the width of the emitter stripe. Figure 5-62 compares the net doping for a 2-micron emitter stripe with that for a 1-micron stripe. For the 2-micron stripe, the doping in the poly across the width of the emitter opening is much higher. It can also be observed that more arsenic has diffused from the poly into the single-crystal silicon; the 2-micron emitter stripe produces a wider emitter and a narrower base than the 1-micron emitter stripe.

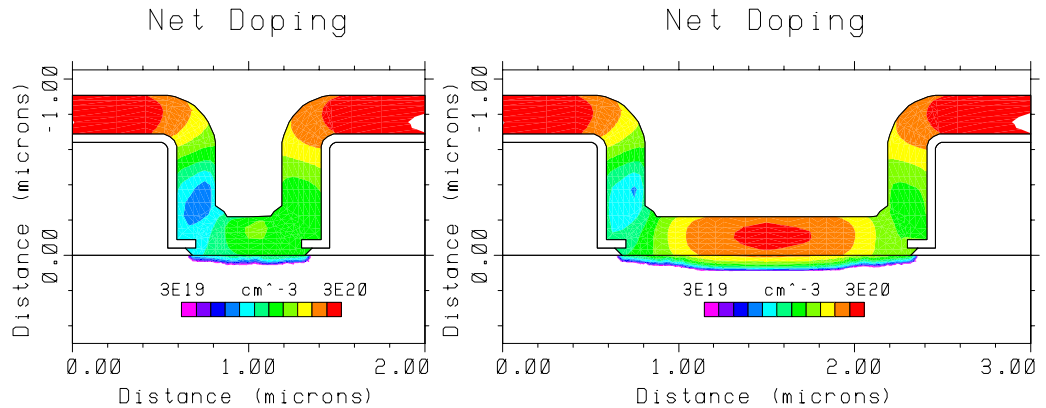


Figure 5-62 Contours of net doping for 1-micron and 2-micron emitter stripes

User-Specified Equation Interface- USEIT

The User-Specified Equation Interface (USEIT) is an optionally available capability for TSUPREM-4 that helps you quickly develop and evaluate your own process models, without requiring the program to be recompiled and linked. USEIT supports *Avant!*'s Physical Model and Equation Interface (PMEI) standard.

The following sections include:

- [“6.1 Physical Model and Equation Interface” on page 6-3](#) describes this universal language.
- [“6.2 Using USEIT” on page 6-19](#) details the use of USEIT.
- [“6.3 Examples” on page 6-33](#) details USEIT examples.

Overview

You can use USEIT to specify model equations that have the general form:

$$\frac{\partial X}{\partial t} = \textit{Previous_RHS} + \mathbf{ADDTOEXP} + \mathbf{INTERFAC} \text{ at specified boundaries}$$

where X is the quantity to be solved.

TSUPREM-4 assembles the matrices to solve equations by assuming that the time derivative term of X is placed at the left-hand-side (LHS) of the equation and all the other terms at the right-hand-side (RHS). To describe this equation using USEIT, any new terms that appear on the RHS of the equation are specified by the **ADDTOEXP** (alias **ADD.RHS**) parameter and the **INTERFAC** parameter on the **EQUATION** statement. The **INTERFAC** parameter is used to define the flux at interfaces. The term *Previous_RHS* in the above expression represents terms that are already present on the RHS of the equation being modified. The initial condi-

tion of the variable X is specified with the **INITIAL** parameter in the **EQUATION** statement. The **INTERMED** statement specifies the intermediate quantities associated with equations.

Since USEIT is designed as a part of TSUPREM-4, you can easily use the physical quantities such as **CI.STAR**(= I^*) and **ETA**(n/n_i) already calculated in the built-in routines. Similarly, you have the choice of using the built-in functions for calculating active and mobile concentrations or defining your own functions with the **ACTIVE** and **MOBILE** parameters in the **IMPURITY** statement.

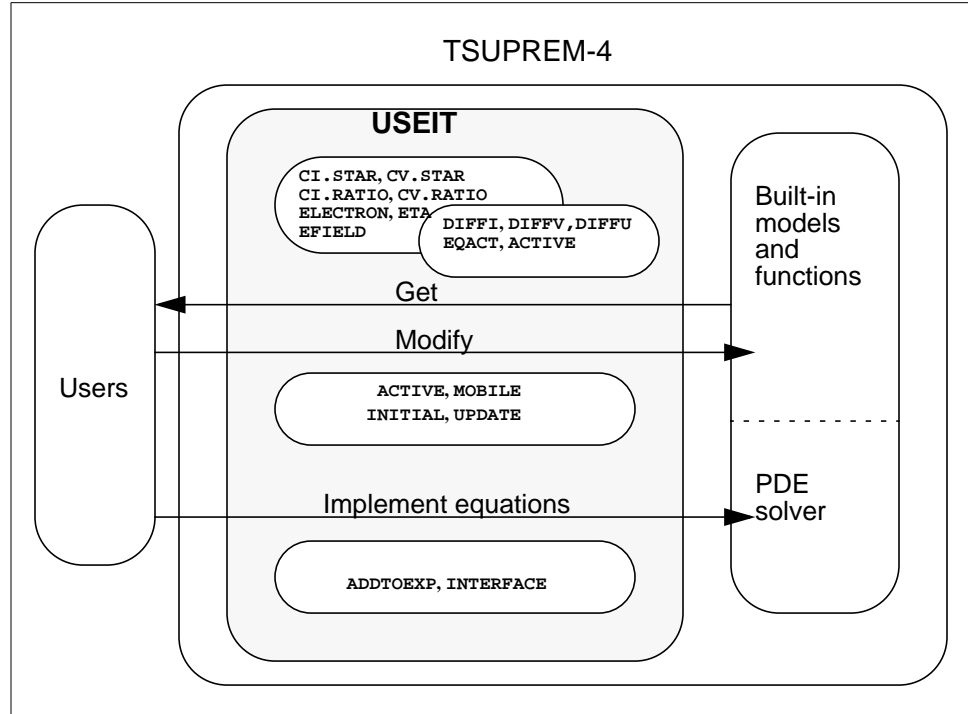


Figure 6-1 USEIT working diagram

6.1 Physical Model and Equation Interface

This section describes the Physical Model and Equation Interface (PMEI) language. PMEI is intended to be an universal language for scripting models and equations for semiconductor process and device simulation. It is *Avant!*'s hope that this language will be adopted by all TCAD simulation tools which support custom model and equation definition.

The standard PMEI language uses a hierarchical format to enforce encapsulation of data and allow model implementation in an object oriented paradigm. Hierarchy also provides a context for grouping an equation with all of its parts—such as initial condition, interface/boundary condition(s), multiple equation expressions for different materials, and etc. Though the language standard is hierarchical, provisions have been made to support tools which do not allow for a hierarchical format, see the section “[Flat Structure Syntax](#)” on page 6-5.

Another issue that can differentiate simulation tools is whether the equations are interpreted at run time, or whether the equations must be constructed and executed in two separate steps e.g., source code generation and compilation. Provisions have been made in the language to support both structures. Consequently, some commands are unnecessary for the interpreted mode, and some are unnecessary for the predefined mode. For example, **Parameters**, PMEI objects which have a default value that can be changed region-wise at run time, are unnecessary for interpreted mode tools.

Some simulation tools already have a set of available equations, functions, and models built-in and there is a need to access and/or modify these equations with PMEI. Accessing the models occurs inside the model and/or equation expressions by the name of the internal model. To modify internal equations there are two allowed methods, one is the addition of terms into existing equations, and the other is replacing internal equations with PMEI defined ones. The machinery of selecting which equations will be solved up to the specific simulation tool will depend on the internal structure. The PMEI language only specifies rules for defining the equations and models.



Note:

The term “model” used here is meant to imply an expression which denotes a physical phenomena such as a mobility model in device simulation. However, the command “Model” in the PMEI language has been expanded to include essentially any expression. Models are used to break up complicated equations and indicate a quantity that can be saved in an output file.

Syntax

In order to help orient you to the basic syntax and operational issues, a few generic examples are provided. The standard syntax for the PMEI is as follows

(for tools with a flat structure i.e. TSUPREM-4, see “Flat Structure Syntax” on page 6-5):

```

Command(
  Parameter1=Value1,
  Parameter2=Value2,
  ...
  SubCommand(
    Parameter1=Value1,
    Parameter2=Value2,
    ...
  ){object1,object2,...}
){object1,object2,...}

```

- Command parameters and sub-commands are in a comma separated list enclosed by round parentheses.
- Command objects, if they exist, are in a comma separated list enclosed in curly braces following the parameter list.
- White space (spaces, tabs and line ends) is ignored.
- Command parameters sometimes take a string value in which non-alphanumeric characters can be used.
- If a string contains a non-alphanumeric character, it should be enclosed in double quotes.

Next, is the description of the basic form for defining a PMEI equation. You want to simulate equations of the form:

$$\text{BulkEquation}(\text{variable } X; \text{material } M) = \text{BasicExpression} + \text{AdditionalExpression} = 0$$

or,

$$\text{InterfaceEquation}(\text{variable } IX; \text{material } M1, \text{material } M2) = \text{BasicExpression} + \text{AdditionalExpression}$$

The PMEI command to define a new equation is **Equation**. This statement is used to specify the start of an **Equation** block in which properties of the equation such as boundary conditions, initial conditions, and the equation expression are defined. **Equation** is used for both bulk and interface equations.

Equation(*description of equations*)

The equation *BulkEquation(variable X; material M)* in material *M* for a solution variable *X* consists of *BasicExpression* and *AdditionalExpression*. The solution variable *X* is specified by the **VariableName** parameter. The material *M* to which the equation is applied is specified by the **Material** parameter. The **Expression** parameter describes the *BasicExpression*.

```

Equation( VariableName=X, Material=M,
          Expression{ expression },
          ..... )

```

An **Expression** is a string enclosed in double quotes and composed of named objects (variables, **Models**, **Parameters**, and etc.), operators, functions, and numbers.

When another expression needs to be added after the *BasicExpression* has been defined, use the **AddToExpression** parameter to describe the *AdditionalExpression*.

```
Equation( VariableName=X, Material=M,
          Expression{ expression },
          ..... )
Equation( VariableName=X, Material=M,
          AddToExpression{ expression },
          ..... )
```

This capability need not be supported for predefined mode simulation tools. The initial values of the variable *X* can be specified with the **InitialCondition** parameter.

```
Equation( VariableName=X, Material=M,
          Expression{ expression },
          InitialCondition(initial_condition),
          ..... )
```

The boolean parameter **Boundary** specifies that the **Expression** and **AddToExpression** parameters describes the equation *InterfaceEquation(variable IX; material M1, material M2)* at interface layers between materials *M1* and *M2* for an interface solution variable *IX*. The **Material** (alias **Material1**) and **Material2** (alias **/Material2**) parameters specify *M1* and *M2*, respectively.

```
Equation( VariableName=IX,
          Material=M1, Material2=M2,
          Boundary=true,
          Expression{ expression },
          ..... )
```

Flat Structure Syntax

The flat structure only changes the syntax. The semantics (command and parameter names) are the same as the hierarchical structure. For the flat structure, the format is as follows:

```
Command Parameter1=value1, Parameter2=value2, ...
```

Commands are separated by new lines and a line continuation character '+' is used to continue commands beyond a line end. A number of commands are associated with an equation and require a context be supplied so that it is clear which equation the command belongs to. For these cases (**InitialCondition**, **InterfaceCondition**, **BoundaryCondition**, **ImplantModel**) the

VariableName parameter must be set to indicate which equation the command belongs to. In the case of the **Expression**, the hierarchical structure allows for a material or material type specific expression. For the flat format, this is not allowed and the **Expression** should just be a command parameter of the command in which it appears.

PMEI Commands

This section details the commands, parameters, and expressions of the PMEI language starting with the highest level command, **Equation**. Each heading in this section is the name of a command and each table following the heading lists all of the parameters and commands which can appear inside the command. The commands or parameters which are shaded represent optional or non-necessary commands which, although useful, do not need to be supported by simulation tools to comply with the language.

For simulation tools which support a hierarchical structure, the hierarchy implies a limited scope for each named object. In general, the scope in which a quantity is defined, is the scope for which it is valid. The major exception to this rule is that **Models** which are defined at the first level under the **Equation** have a global scope and can be used by other equations. The depth of the **Model** hierarchy is unlimited, but there is rarely a need to go more than a couple of levels down. For simulation tools without a hierarchy, all named objects have a global scope and objects will not appear inside of other objects.

In the tables below, parameters of a command are indicated with an '=' sign and the type of parameter is indicated by the right hand side. Parameters which don't have a '=' are objects in the hierarchical format and are listed to indicate which objects can appear inside of a particular object. A '/' in the parameter name column indicates a difference required for hierarchical/non-hierarchical structure and a '|' indicates an "or" condition.

Equation

Use the **Equation** command to define a new PMEI equation.

| Parameter-Object | Definition | Default |
|------------------------------------|---|---------|
| Name=string | The name of the Equation . | |
| VariableName=string | Solution variable for the Equation . | |
| Material Material1=string | The Material should be a name from the Materials database. Use this Parameter as often as necessary to define all material types for which the equation is valid. For interface equations, pairs of materials must be specified. For 1 pair of materials, Material1 , and Material2 can be set. For multiple pairs, set the Material parameter to a comma-separated pair of materials and repeat as often as necessary. | All |

| Parameter-Object | Definition | Default |
|--|---|---------|
| Material2 (alias /Material2) =string | If a Material2 is specified (and the interface flag is set) then this Equation is an interface equation and the variable has units of the interface. | All |
| Boundary=bool | A boolean flag to explicitly set the Equation to the interface type. | False |
| Model | New Models can be defined inside an equation or inside a Model and belong to the scope in which they are defined. | |
| Parameter | New Parameters can be defined inside an Equation or inside a Model and belong to the scope in which they are defined. | |
| Array | New Arrays can be defined inside an Equation or inside a Model and belong to the scope in which they are defined. | |
| Loop | New Arrays can be initialized with a Loop inside an Equation or inside a Model . | |
| Expression/ Expression= AddToExpression/ AddToExpression= | Expression is used to write the entire expression—assumed to be equal to 0 whereas, AddToExpression is used to add additional terms to an existing Equation . For Boundary=true (i.e. for an interface equation) The functions OnMat1(<expression>) and OnMat2(<expression>) can be used in the Expression to return <expression> on the surrounding materials. | |
| Scaling=float | The internal values of the variable are scaled by this parameter, i.e. internal_value=external_value/scaling . This parameter is often used to prevent extremely large or extremely small numbers. Note: Models using scaled variables will be calculated using the scaled (internal) value. | 1.0 |
| DopantCharge= float | Set to +1 if the variable is a donor and to -1 if the variable is an acceptor. Note: Multiply charged and fractionally charged variables are allowed. If unset, the variable is not considered a dopant. | 0 |
| ErrorLimit=float | Sets the convergence criteria. The ErrorLimit has the same units as the Equation variable. | 1.0 |

Model (alias INTERMEDIATE)

Models are the fundamental building blocks of the equation. They allow the equation writer to break up complicated expressions into simpler terms which are calculated once per node and reused if necessary. **Models** also perform the function of a switch allowing the creation of several computational schemes any of which can be selected at run time. The simulation tool should allow **Models** to be saved as fields in the output. You can use the alias, **INTERMEDIATE**, instead of **Model** to avoid confusion with other uses of the keyword **Model**.

| Parameter-Object | Definition | Default |
|--|---|---------|
| Name=string | The name of the Model . | |
| Expression / Expression= string | Expression is used to define the Model expression. In general, the Expression can contain Parameter , Model , and variable names as well as operators. The list of available operators is given below. | |

| | | |
|------------------------|---|--|
| Model | New Models can be defined inside an Equation or inside a Model and belong to the scope in which they are defined. | |
| Parameter | New Parameters can be defined inside an Equation or inside a Model and belong to the scope in which they are defined. | |
| Array | New Arrays can be defined inside an Equation or inside a Model and belong to the scope in which they are defined. | |
| Loop | New Arrays can be initialized with a Loop inside an Equation or Model . | |
| DataType=string | DataType will automatically be set to either Element or Node depending on the type of expression. It can be overridden by specifying one of following: NodeModel , NodeParameter , ElementModel , VectorElementModel , EdgeModel , or VectorEdgeModel . These types refer to the type of data (vector or scalar), the position in the element where the data is stored (Node , Edge , or Element), and whether the quantity is calculated, or initialized to a field and fixed during a solve (Model or Parameter , respectively). | |

Expression

Expression defines the actual expression of an **Equation** or **Model**. **Expressions** are composed of operators, **Models**, **Parameters**, variables, etc. and may have special predefined references available depending on the context. Predefined references are listed in the appropriate command description table below. For simulation tools with a hierarchical structure, multiple **Expressions** can be used for different material or material types. If multiple expressions appear, the last expression must be a default expression with no **Material** or **MaterialType** parameter given. For simulation tools with a flat structure, the **Expression** should appear as a parameter of the command (**Equation**, **Model**, **Loop**, and etc.).

| Parameter-Object | Definition | Default |
|--|---|---------------|
| MaterialType=string | A different Expression can be specified for each material type (Semiconductor , Insulator , or Conductor). A default Expression must always be given (one without a Material or MaterialType). | All Materials |
| Material=string | A different Expression can be specified for each material. The Material should be a name from the Materials data base. A default Expression must always be given (one without a Material or MaterialType). | All Materials |
| Object for Expression | Expressions take only one object, a string. This defines how a value is to be calculated. Expressions contain previously defined objects, operators (both arithmetic and differential), and functions (see " Functions and Operators " on page 6-14). Objects include Models , Parameters , Intermediates , Variables , Sums , and Conditions . These objects are referenced by their names as specified in the respective Name parameter. | |
| Predefined references of Expression | All Expressions have access to the predefined references below. Additional predefined keywords are available in specific contexts, see the Expression statements in the following commands. | |

| | | |
|--------------------|---|--|
| Temperature | The temperature of the simulation domain in Kelvin (not to be used with the Lattice Temperature model). | |
| kT | The temperature of the simulation domain in eV (not to be used with the Lattice Temperature model). | |

Parameter

A **Parameter** is a number, boolean, or string constant which can have different values in different materials. This command is required for simulation tools in which the equation definition and execution happen in two steps. The **Parameter** command provides a hook for changing values at run time. The method to specify the values is left up to the simulation tool with the requirement that both “bulk” parameters associated with a material and “interface” parameters associated with a pair of materials be available.

| Parameter-Object | Definition | Default |
|----------------------|---|---------|
| Name=string | The name of the Parameter . | |
| Boundary=bool | Parameters defined at the Equation level can be specified as an interface equation by setting this flag to true. Specification of the value for different pairs materials happens at run time | False |
| Default=float | Set the default value of the parameter. The default value is used for this parameter unless it is overridden at run-time. | |

Array

An array of numbers, parameters, model or variable can be created with the use of the **Array** command. **Array** elements are accessed in Expressions with square bracket syntax. For example, if ‘a’ is an **Array**, the i’th element (starting with i=0) is accessed with a[i].

| Parameter-Object | Definition | Default |
|---|---|---------|
| Name=string | The name of the Array . | |
| Type=type | The type can be one of numeric or general. Only the numeric Array type can have it’s value changed. | |
| Size = "integer,integer,..." | This parameter sets the size of the Array for each dimension. The format is a series of integers designating the Array dimension in the corresponding axes. | |
| MinIndex = "integer,integer,..." | Set the starting index for each dimension. If this parameter is set, the number of integers listed must be the same as for the Size parameter. | 0 |

| | | |
|---------------------------|---|---|
| DefaultValue=float | Default initialization for all Array members. | 0 |
| Object/Value= | For general Arrays , the value cannot be changed after the Array is created and is always equal to the Model /variable/ number which is entered in this parameter. For numeric Arrays , the Array can be initialized to a starting value and then changed later in a Loop . | |

Syntax:

```
Array(Name=ArrayName)
    {"element0, element1, element2"}
```

Flat Structure Syntax:

```
Array Name=ArrayName, +
    Value="element0, element1, element2"
```

Loop

The **Loop** command is supplied to allow for initialization of **Arrays**. Nesting of **Loops** is allowed as deep as necessary. A **Loop** can appear anywhere that an **Array** can appear. For the flat structure, an **End** command is used to end the loop.

| Parameter-Object | Definition | Default |
|---------------------------------------|---|---------|
| Index=string | The name of the Index . | |
| Steps=integer | The number of steps over which the Loop will be traversed. | |
| Start=integer | The starting value of the Index . | 0 |
| Name=string | The indexed Array name should go here (inside double quotes). i.e. Name=" Array [i]". Only one Name and one Expression should appear in a set of nested Loops —in the inner-most Loop . | |
| Expression / Expression=string | The Expression which corresponds to the indexed Array of Name . Also, the Index name(s) can appear in the Expression as well. | |
| Object of Loop | The object is used to define a nested Loop . Only a Loop can appear in the object. | |

Syntax:

```
Loop(Index=i, Steps=<number>, Name="ArrayName[i]",
    Expression{"<a function of i, models, etc.>"})
```

Flat Structure Syntax:

```

Loop Index=i, Steps=<number>
  Array Name="ArrayName[i]" +
  Expression="<a function of i, models, etc.>"
End
    
```

InitialCondition

The **InitialCondition** command is used to specify a special initial condition for the variable. The initial value of the solution variable of the equation can depend on other solution variables or constant nodal quantity (such as net doping during device simulation). The selector can be used in the **Expression** to allow a multiple initializations of the equation. The choice of initialization is made at run time. For simulation tools with a flat structure, the **InitialCondition** should appear as a parameter of the **Equation** command or as a separate command with the **VariableName** specified.

| Parameter-Object | Definition | Default |
|--|---|---------|
| Object of InitialCondition / InitialCondition= | The object is a string similar to the Expression statement and is used to initialize the variable, see the command " Expression " on page 6-8. | |

BoundaryCondition

The **BoundaryCondition** command is used to specify a boundary condition for the equation. The **BoundaryCondition** command may be used multiple times to specify different conditions for different boundary types. The boundary of an equation is defined as a boundary where the equation is solved on one side but not on the other and includes the outer most boundaries of the simulation. **InterfaceCondition** on the other hand, defines what happens to the variable at an interface between two materials within the solution domain. Use either the type parameter to describe the type of **BoundaryCondition** or use the material pair, but not both. For simulation tools with a flat structure, the **Bound-**

aryCondition should appear as a parameter of the **Equation** command or as a separate command with the **VariableName** specified.

| Parameter-Object | Definition | Default |
|--|--|---------|
| Type=string | Valid boundary condition types are: REFLECTIVE , TOP_AMBIENT , COND_COND , INSUL_INSUL , SEMI_SEMI , INSUL_SEMI , INSUL_COND , COND_SEMI , BACK_SURFACE , EXPOSED_SURFACE , THERMAL_CONTACT , and CONTACT . You can use the Type parameter multiple times to specify the same condition for different boundary types. If a type is specified the material parameters should not be set. | |
| Material1=string MaterialType1=string | Specify a material or a material type for Material1 . If either of these parameters are set then the Type parameter <i>should not</i> be set, and Material2 or MaterialType2 <i>should</i> be set. | |
| Material2(alias /Material2)=string MaterialType2=string | Specify a material or a material type for Material2 . If either of these parameters are set then the Type parameter <i>should not</i> be set, and Material2 or MaterialType2 <i>should</i> be set. | |
| Object of BoundaryCondition / BoundaryCondition= | There is only one object for BoundaryCondition which is a string containing an expression (see " Expression " on page 6-8). The expression to return is the equation expression on a boundary node of Material1 . Several predefined keywords may be needed to accomplish this; OnMat1 (<expression>), OnMat2 (<expression>), and FluxOfEquation . OnMat1 and OnMat2 are functions which calculate the expression on Material1 and Material2 , respectively. Any object not inside of OnMat1 or OnMat2 should be an "boundary=true" object. FluxOfEquation is a shorthand reference to the full equation expression. Because the desired boundary condition is often the equation expression plus an additional term, FluxOfEquation simplifies the BoundaryCondition expression. Also, parameters of the equation (either Interface or Bulk parameters of Material1) may be called in the expression of BoundaryCondition . | |

InterfaceCondition

The **InterfaceCondition** is the command used, if necessary, to specify a flux of the solution variable across an interface where the equation is solved on both sides. If no **InterfaceCondition** is defined the solution variable is considered continuous across the boundary. You can use the **InterfaceCondition** command multiple times to specify different conditions at different interface types. For simulation tools with a flat structure, the **InterfaceCondition** should appear as a parameter of the **Equation** command or as a separate command with the **VariableName** specified.

| Parameter-Object | Definition | Default |
|---|--|---------|
| Material1=string MaterialType1=string | Specify a material or a material type for Material1 . | |
| Material2 (alias /Material2)=string MaterialType2=string | Specify a material or a material type for Material2 . | |
| Symmetric (alias Transport) = bool | If symmetric is true, the flux specified by InterfaceCondition will be applied to both bulk regions specified in (Material1 MaterialType1) and (Material2 MaterialType2). | |
| Object of InterfaceCondition / InterfaceCondition= | There is only one object for InterfaceCondition which is a string containing an Expression , see " Expression " on page 6-8. The expression represents the flux of the variable to Material1 from both the interface (if any) and from Material2 . If the variable is continuous across the interface, no interface condition is required. The Expression is similar to the BoundaryCondition and will be added to any applicable BoundaryCondition . Any objects which are not inside of OnMat1 or OnMat2 must be "boundary=true" objects. All other objects should be inside OnMat1(<expression>) or OnMat2(<expression>) . Also, Parameters of the equation (either Interface or bulk parameters of Material1) may be called in the expression of InterfaceCondition . | |

ImplantModel

The **ImplantModel** is an optional command used to define and/or redefine what should happen when an implant of this variable is performed. For typical dopants, this model is not necessary. For point defects, this model can be used to create specialized initial conditions based on the ion and Frenkel pair fields. In order for the **Equation** (and its corresponding **ImplantModel**) to be called for each implant, the variable must be named "interstitial" or "vacancy" and the **Equation** must substitute the internal interstitial and vacancy equations. For simulation tools with a flat structure, the **ImplantModel** should appear as a parameter of the **Equation** command or as a separate command with the **VariableName** specified.

| Parameter-Object | Definition | Default |
|---|--|---------|
| Object of ImplantModel / ImplantModel= | The object is a string similar to the object of the Expression command and is used to set the variable after an implant. In addition to the standard objects which can appear in an Expression . You can use the keywords Ion and Frenkel . They denote the newly implanted ion concentration field and the newly created frenkel pair implant damage field, respectively. | |

Selector

The **Selector** is an optional command used to create a **Model** selection capability or “model switch” in the present context. The **Selector** name can be used in the **Expression** of the current context (**Model**, **Equation**, and **InitialCondition**), and the **Model** which is actually used for the computation shall be set at run-time.

| Parameter-Object | Definition | Default |
|-----------------------|--|---------|
| Name=String | The name of the Selector . | |
| Default=String | The default Model . This Model should be known by the PMEI, either defined earlier in the same file, an internal file, or defined in an Equation database file. | |

Functions and Operators

This section describes the functions and operators you can be use inside an **Expression**. For operations on numbers, all standard c library operators and math functions should be available. In addition, standard arithmetic operators as well as differential operators should be available for **Models**. A number of standard mathematical functions for **Models** should be available as well (see list below for details).

| | |
|-----------------------|---|
| -=, *=, /=, += | Unary Operators |
| +, -, *, / | |
| %, ^, pow(base,power) | To the power of |
| | Vector Product |
| ~ , Div() | If the object/operator on is an elemental quantity (such as a gradient of a nodal quantity or a vector), the divergence is applied. |
| ! ,Grad() | Gradient |
| ddt | Time derivative |
| ddx, ddy, ddz | Directed derivatives |
| makevector(x,y,z) | Construct a vector out of components |
| upwindgradient(s1,s2) | Upwind gradient according to the Scharfetter-Gummel discretization. s1 variable, s2 potential |
| log | Natural logarithm |
| log10 | Logarithm base 10 |

| | |
|------------------------------|------------------------|
| exp | exponentiation |
| asinh | inverse hyperbolic sin |
| sqrt | square root |
| min(expression1,expression2) | minimum |
| max(expression1,expression2) | maximum |
| abs | absolute vale |

PMEI Functions

Some simulation tools already have a set of available equations, functions, and models built-in and there is a need to access and/or modify these equations with PMEI. The following functions are available in PMEI.

Sum

The **Sum** function allows calculations to be performed on all elements of **Arrays** in a looping fashion, sums must be used inside of an **Expression** like a function.

| Parameter-Object | Definition | Default |
|---|--|---------|
| $\text{Sum}(\text{IndexName}, \text{start}, \text{end})$ $\{ \text{Expression} \} /$ $\text{Sum}(\text{IndexName}, \text{start}, \text{end},$ $\text{expression})$ | IndexName is a string for the name of the Index (which can be used in the Expression). Start is an integer which indicates the starting value of the index and end indicates the last value, inclusively, of the Index . Expression is a normal expression and nested Sums can appear here too. | |

Condition

The **Condition** function allows for if-else if-else functionality found in tradition programming languages. The condition function must appear inside an **Expression**.

| Parameter-Object | Definition | Default |
|---|--|---------|
| Condition (BooleanExpression , TrueExpression , FalseExpression) | The Condition function returns TrueExpression if BooleanExpression is true else it returns FalseExpression . The BooleanExpression can contain a boolean parameter or an Expression containing one of the boolean operators, (<, <=, >, >=, ==, !=, !) in addition to the normal references available in Expression . | |

Exist

The **Exist** function allows region-by-region boolean function which indicates the availability of the named variable or model. The **Exist** function can appear inside the **Condition** function's **BooleanExpression**.

| Parameter-Object | Definition | Default |
|--------------------------------|---|---------|
| Exist (string) | Return true or false if the named object is available in a region | |

OnMat1 and OnMat2

These functions are used for interface and boundary contexts (interface equations, boundary conditions, and interface conditions). They return the value of the expression on the corresponding materials.

| Parameter-Object | Definition | Default |
|---|---|---------|
| OnMat1 (<expression> OnMat2 (<expression>) | Return <expression> on Material1 or Material2 , respectively. | |

Proposed Language Additions

The following functions are being considered for permanent addition to the PMEI.

Parameterized Functions

Parameterized functions would be provided to allow you to define and use complicated functions which have a set of numerical parameters which are dependent on

the function parameter. This would allow for example, a dopant function to be defined once and used in all the dopant equations.

Syntax:

Function[Parameter](Function input list)

Physical Functions The following functions are examples of internal functions available in TSUPREM-4 and could be called using the parameterized function syntax.

| Function | Arguments | Definition |
|--------------------|---|---|
| DIFFI | arg1 impurity | Diffusivity of the impurity <i>arg1</i> with interstitials, <i>i.e.</i> $\sum_i D_m^{(i)} \left(\frac{n}{n_i}\right)^{-i}$, where $D_m^{(i)}$ values are material dependent parameters. Example: DIFFI (BORON) Note: Not allowed for the built-in point defects <i>interst</i> and <i>vacancy</i> |
| DIFFV | arg1 impurity | Diffusivity of the impurity <i>arg1</i> with vacancies, <i>i.e.</i> $\sum_i D_n^{(i)} \left(\frac{n}{n_i}\right)^{-i}$, where $D_n^{(i)}$ values are material dependent parameters. Example: DIFFV (BORON) Note: Not allowed for the built-in point defects <i>interst</i> and <i>vacancy</i> |
| DIFFUSIVITY | arg1 impurity | Diffusivity of the impurity <i>arg1</i> . If <i>arg1</i> is neither <i>interst</i> and nor <i>vacancy</i> , $\frac{I}{I^*} \sum_i D_m^{(i)} \left(\frac{n}{n_i}\right)^{-i} + \frac{V}{V^*} \sum_i D_n^{(i)} \left(\frac{n}{n_i}\right)^{-i}$, otherwise, $\frac{\sum_i D_I^{(i)} \phi_I^{(i)} \left(\frac{n}{n_i}\right)^{-i}}{\sum_i \phi_I^{(i)} \left(\frac{n}{n_i}\right)^{-i}}$ and $\frac{\sum_i D_V^{(i)} \phi_V^{(i)} \left(\frac{n}{n_i}\right)^{-i}}{\sum_i \phi_V^{(i)} \left(\frac{n}{n_i}\right)^{-i}}$ for <i>interst</i> and Example: DIFFU (BORON) |
| EQACTIVE | arg1 impurity arg2 number or nodal value | Equilibrium active concentration of <i>arg2</i> expression. Calculated with the solid solubility and equilibrium clustering coefficients of the impurity <i>arg1</i> . Example: EQACT (BORON, BORON-BCL) |
| ACTIVE | arg1 impurity | Active concentration of the impurity <i>arg1</i> Example: ACTIVE (BORON) Note: May not include the functions DIFFI , DIFFV , and DIFFUSIVITY . |

6.2 Using USEIT

This section provides details on using USEIT.

- Additional details of the **EQUATION** statement and its advantages.
- Additional details of the **INTERMEDIATE** statement and its advantages.
- Details of some applications of user-specified models.
- Hints on defining active or mobile concentrations in USEIT.

This section also contains useful tables detailing:

- Built-in Keywords
- Operators
- Mathematical Functions
- Physical Functions
- Interface Functions
- Miscellaneous Functions

EQUATION

The **EQUATION** statement is used to specify user-specified equations. The **EQUATION** statement is available only with the USEIT (User-Specified Equation Interface for TSUPREM-4) Option. For additional details see “[EQUATION](#)” on [page 3-204](#).

New Solution Variable

When a new solution variable is first specified in the **EQUATION** statement, it is automatically defined. A warning message is displayed explaining that the **VARIABLE** does not currently exist and is being defined. Alternatively, the solution variable can be defined in the **IMPURITY** statement with the **NEW** parameter. The following statement defines the new impurity *BCL*.

```
IMPURITY IMP=BCL NEW
```

Solving Method

The **METHOD** statement specifies the numerical methods used to solve the equations. The factorization method of block matrices, the absolute, and the relative error tolerances for solutions should be specified. For example:

```
METHOD VARIABLE=RDL NONE ABS.ERR=0.1
```

where:

- *RDL* is a solution variable for dislocation-loop radius.
- The **VARIABLE** parameter is an alias for the **IMPURITY** parameter.

Since the *RDL* is supposed to be immobile, the factorization method is defined to **NONE**. The absolute error is set to 0.1. The unit of the absolute error is the same as the unit of the solution variable. The absolute error value is set to 1×10^9 by default, unless it is specified. Since the **REL.ERR** parameter is not specified, the relative error is set to 0.01, by default.

Initialization

For solutions (for example, cluster) not introduced by external sources, such as implantation, predeposition, or epitaxy; the solution variable must be initialized to solve the partial differential equations. The **INITIAL** parameter in the **EQUATION** statement specifies the initial values of a solution variable in the specified **MATERIAL**. Initialization is performed *when* the following conditions are satisfied:

- The solution variable has not yet been initialized before thermal processes (**DIFFUSION** or **EPITAXY**), or if the solution variable was initialized, implantation was performed prior to thermal process.

- All other solution variables associated with the **INITIAL** expression are available so that the **INITIAL** expression can be evaluated. Or the solution variable is produced by reactions in the **REACTION** statement or the **INTERFACE** parameter in the **EQUATION** statement.

```
EQUATION VARIABLE=RDL MAT=Si +
  INIT="(1e20<inters+damage&&inters+damage<=1.15e22)?+
    300:(EXIST(RDL)?RDL:0.0)"
```

The example shows the initialization of the dislocation-loop radius. For additional details see the [“Dislocation-Loop Model”](#) on page 6-36.

```
IMPLANT ARSENIC DOSE=1E15 ENERGY=30 ;$1
SELECT Z=RDL; PRINT.1D X.V=0 ;$2
DIFFUSION TEMP=800 TIME=0 ;$3
SELECT Z=RDL; PRINT.1D X.V=0 ;$4
```

In the second line, the *RDL* solution has not yet been initialized, so a warning message is displayed explaining that *RDL* does not currently exist. The *RDL* is initialized at the third line and then the spatial distribution of *RDL* is printed out at the fourth line.



Note:

The intermediates in the INITIAL expression store the values at the previous thermal step. Thus, if the initialization includes the intermediates associated with the built-in point-defects, the initialization may be incorrect; because the changes of the point-defect concentrations due to implants have not been reflected in the intermediates. For example:

```
INTERMED NAME=Itotal VALUE=damage+interst
EQUATION VARIABLE=RDL MAT=Si +
  INIT="(1e20<Itotal&&Itotal<=1.15e22)? +
  LRAD:(EXIST(RDL)?RDL:0.0)"
```

The damage is zero at the previous thermal step so that the Itotal stores the only interstitial concentration at the previous thermal step. Thus, the above initialization is incorrect. You can obtain the correct initialization by avoiding the use of INTERMED values in the INITIAL expression.

Adding Expressions to Equations

The terms specified by the **ADDTOEXP** parameter in the **EQUATION** statement are added to the current right-hand-side (RHS) term of equations.

```
EQUATION VARIABLE=X MAT=Si ADDTOEXP=DIV(dif*GRAD(X))
```

The above statement represents the following equation for the unknown X:

$$\frac{\partial X}{\partial t} = \text{Previous_RHS} + \nabla \cdot (\text{diff} \nabla X)$$

where *Previous_RHS* means the previously specified expressions or the built-in terms in the right-hand-side (RHS).

An additional specification of **ADDTOEXP** for the same solution variable in the same material adds the new expression to the previously specified terms; it does *not* replace the present expression with the new expression. For example:

```
EQUATION VARIABLE=X MAT=Si ADDTOEXP=expr1
EQUATION VARIABLE=X MAT=Si ADDTOEXP=expr2
```

The final expression is the sum of expressions *expr1* and *expr2*. You can replace the above statements with:

```
EQUATION VARIABLE=X MAT=Si ADDTOEXP=expr1+expr2
```

Flux at Interfaces

The parameter **INTERFAC** in the **EQUATION** statement specifies the transport flux from material **/MAT2** to material **MAT1** at interfaces between the materials. The specified flux is added at the boundary nodes on the material **MAT1**. The parameter **TRANSPOR** specifies that the flux should also be subtracted at the boundary nodes on material **/MAT2** so that the dose of the solution is conserved. The **TRANSPOR** parameter is turned *on* by default.

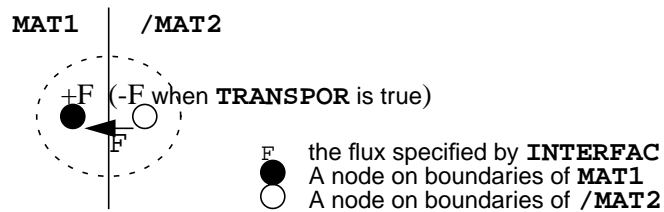


Figure 6-2 Flux at interfaces

```
$ B segregation at Si/Ox
INTERMED NAME=m VALUE=1126*exp(-0.91/kt)
EQUATION VARIABLE=BORON MAT1=Si /MAT2=Ox +
INTERFAC=0.1*(ONMAT2(BORON)-ONMAT1(BORON)/m)
```

The traditional segregation flux of boron between silicon and oxide is described. No specification of **TRANSPOR** implies that the flux must be subtracted on the boundary nodes in oxides.

The additional specification for the same solution variable on the same interface adds the new flux to the previously specified terms; it does *not* replace the present flux with the new flux. For example:

```
EQUATION VARIABLE=X MAT1=Si /MAT2=Ox INTERFAC=expr1
EQUATION VARIABLE=X MAT1=Si /MAT2=Ox INTERFAC=expr2
```

The final flux is the sum of expressions *expr1* and *expr2*. You can replace the above statements with:

```
EQUATION VAR=X MAT1=Si /MAT2=Ox INTERFAC=expr1+expr2
```

Diffusion Along Boundaries

If the **ADDTOEXP** expression is specified with the **BOUNDARY** parameter in the **EQUATION** statement, the **ADDTOEXP** expression is applied only on the boundaries of **MAT1**, adjacent to **/MAT2**.

```
$ Diffusion on the silicon surface along the Si/Ox.
EQUATION VARIABLE=IBORON MAT1=Ox /MAT2=Si BOUNDARY +
      ADDTOEXP=DIV(dif*GRAD( IBORON ) )
```

The expression in the **ADDTOEXP** parameter shows the diffusion along the boundaries of oxide adjacent to silicon.

Update Solution

Use the parameter **UPDATE** in the **EQUATION** statement for the special case of updating the solution during solving equations. For example:

```
$ RDL : Dislocation loop radius (angstrom unit)
EQUATION VARIABLE=RDL MAT=Si UPDATE=RDL>100?RDL:0.0
```

This example shows how to discard dislocation-loops with a radius of less than 100 angstroms. When solving the equation, the values of *RDL* less than 100 angstroms are set to zero.

INTERMEDIATE

The **INTERMEDIATE** statement (alias **MODEL** or **ARRAY**) specifies the intermediate quantities associated with equations. For additional details see “[INTERMEDIATE](#)” on page 3-32.

Advantages of Intermediates

The advantages of using intermediates quantities are:

- Simplifying long and complicated equations.
- Avoiding the duplicated calculation of quantities used in several equations. Once the intermediates are evaluated, the calculated values are shared by all associated equations until the intermediates need to be reset.
- Avoiding duplicated calculation of quantities depending *only* on temperature. If the intermediates depend *only* on temperature, they are evaluated *only* when the temperature changes.
- Use intermediates in the same manner as the built-in solution values in the **SELECT** statement.



Note:

Intermediates are stored as symbolic expressions that are evaluated later (for example, when solving the model equations).

Application of User-Specified Models

The **MODEL** parameter controls the application of specified equations. By turning *on/off* the **ENABLE** parameter with the **MODEL** parameter in the **METHOD** statement, the equations grouped by the **MODEL** parameter in the **IMPURITY** and **REACTION** statements can be enabled or disabled. The models not specified by the **MODEL** parameter are always turned *on*.

```
EQUATION MODEL=BIC VARIABLE=BORON MAT=Si ADDTOEXP=rhs1
EQUATION MODEL=BIC VARIABLE=INTER MAT=Si ADDTOEXP=rhs2
$ process1
METHOD MODEL=BIC ^ENABLE
$ process2
METHOD MODEL=BIC ENABLE
$ process3
```

At *process1*, the equations in the model *BIC* are applied because it's turned *on* by default. Turning *off* the **ENABLE** parameter prior to *process2* disables the *BIC* model. Simulate *process3* with the *BIC* model by turning *on* the **ENABLE** parameter again.

The additional specification for the same solution variable in the same material or on the same interface with the same model name adds the new expression to the previously specified terms; it does *not* replace the present expression with the new expression. For example:

```
EQUATION MODEL=mdl VARIABLE=X MATERIAL=m EXPR=expr1
EQUATION MODEL=mdl VARIABLE=X MATERIAL=m EXPR=expr2
```

The final expression is the sum of expressions *expr1* and *expr2*. You can replace the above statements with:

```
EQUATION MODEL=mdl VAR=X MAT=m EXPR=expr1+expr2
```

Specify the **INITIAL**, **ACTIVE**, **MOBILE**, and **UPDATE** parameters only once for the specified impurity and material in one model. The duplicated specification of these parameters for the same model replaces the previous definitions with the new ones. Even though it is possible for each different model to have its own **INITIAL**, **ACTIVE**, **MOBILE**, and **UPDATE** expressions, only one model among them should be enabled. Otherwise, it causes incorrect results.

Modification of Built-in Functions

Define the modification of the active or mobile concentrations in USEIT using the following hints.

ACTIVE

Usually, active concentrations are defined with the solid solubilities and the equilibrium clustering coefficients specified in the **IMPURITY** statement. When active concentrations need to be calculated differently, the **ACTIVE** parameter in the **IMPURITY** statement provides the new method to calculate active concentrations. The physical function **ACTIVE()** returns the values of active concentration calculated by the method that is specified in the **ACTIVE** parameter, while the **ACTIVE** parameter defines the calculation method itself for active concentration.

```
IMPURITY IMP=BCL NEW
INTERMED NAME=Ceq EXPRESS=EQACTIVE(boron, boron)
INTERMED NAME=ni +
          EXPRESS=3.87e16*exp(-0.605/kt+1.5*log(temp))
IMPURITY IMP=boron MAT=Si +
          ACTIVE=MAX(boron-BCL, MIN(Ceq, ni))
```

The *Ceq* and *ni* are defined as the equilibrium active boron concentration and intrinsic carrier concentration, respectively.

MOBILE

In most cases, mobile concentrations are the same as active concentrations. However, when the mobile concentration is different from the active concentration, use the **MOBILE** parameter in the **IMPURITY** statement. For example, the diffusion retardation due to ion-pairing is described with the reduction of the mobile concentration.

$$C_m = \left(1 - \frac{N_p}{N_a}\right) C_a \text{ for acceptors, } C_m = \left(1 - \frac{N_p}{N_d}\right) C_a \text{ for donors}$$

$$\text{where } N_p = \frac{1}{2} \left((N_d + N_a + \Omega) - \sqrt{(N_d + N_a + \Omega)^2 - 4N_d N_a} \right)$$

The above equations are described by using USEIT as follows:

```

INTERMED NAME=ni +
          EXPR=3.87e16*exp(-0.605/kt+1.5*log(temp))
INTERMED NAME=Nd EXPR=ACTIVE(arsenic)+ACTIVE(phospho)
INTERMED NAME=Na EXPR=ACTIVE(boron)
INTERMED NAME=Nt EXPR=Nd+Na+6.0*ni
INTERMED NAME=Np EXPR=0.5*(Nt-sqrt(Nt*Nt-4*Nd*Na))
INTERMED NAME=Fpa EXPR=1-Np/Na
INTERMED NAME=Fpd EXPR=1-Np/Nd
IMPURITY MODEL=IONPAIR IMP=boro MAT=Si +
          MOBILE=Fpa*ACTIVE(boron)
IMPURITY MODEL=IONPAIR IMP=phos MAT=Si +
          MOBILE=Fpd*ACTIVE(phos)
IMPURITY MODEL=IONPAIR IMP=arse MAT=Si +
          MOBILE=Fpd*ACTIVE(arse)

```

Built-in Keywords

The following keywords are available.

| Name | Type | Value | Unit |
|-------------------|---------------|---|------------------|
| KB | constant | Boltzmann constant divided by electron charge, $.61708 \times 10^{-7}$ | Volts/°K |
| QE | constant | Electron charge, $.602 \times 10^{-18}$ | Coulomb |
| TEMP | parameter | Kelvin temperature | °K |
| KT | parameter | Thermal voltage, KB × TEMP | Volts |
| CI . STAR | node solution | Equilibrium interstitial concentration | cm ⁻³ |
| CV . STAR | node solution | Equilibrium vacancy concentration | cm ⁻³ |
| CI . RATIO | node solution | Ratio of interstitial concentrations to equilibrium | None |
| CV . RATIO | node solution | Ratio of vacancy concentrations to equilibrium | None |
| ELECTRON | node solution | Electron concentration Note: Calculated by assuming local charge neutrality | cm ⁻³ |
| ETA | node solution | Electron concentration ratio to intrinsic carrier concentration | None |
| EFIELD | edge solution | Electric field divided by thermal voltage | cm ⁻¹ |

Operators

The operators in higher tables have higher precedence than the operators in lower tables. *expr* means an expression.

| Operator | Usage | Definition |
|--------------|----------------------------|---------------------------------------|
| ^ | <i>expr</i> ^ <i>expr</i> | Power Example: CI . RATIO^2 |
| + | + <i>expr</i> | Unary positive |
| - | - <i>expr</i> | Unary negative |
| * | <i>expr</i> * <i>expr</i> | Multiply |
| / | <i>expr</i> / <i>expr</i> | Divide |
| + | <i>expr</i> + <i>expr</i> | Add |
| - | <i>expr</i> - <i>expr</i> | Subtract |
| > | <i>expr</i> > <i>expr</i> | Greater than |
| >= | <i>expr</i> >= <i>expr</i> | Greater than or equal to |
| < | <i>expr</i> < <i>expr</i> | Less than |
| <= | <i>expr</i> <= <i>expr</i> | Less than or equal to |
| == | <i>expr</i> == <i>expr</i> | Equal |

| Operator | Usage | Definition |
|----------|----------------------|--|
| != | $expr \neq expr$ | Not equal |
| && | $expr \&\& expr$ | And |
| | $expr \ \ expr$ | Or |
| ?: | $expr ? expr : expr$ | $condition_expression ? true_expression : false_expression$ Example: BORON>PHOS?BORON: PHOS Note: Same as the CONDITION function. See “Miscellaneous Functions” on page 6-31. |

Mathematical Functions

The argument types, *number*, *node value*, and *edge value* in the argument list mean that the evaluation of the argument expressions results in number, node solution, and edge solution, respectively.

| Function | Argument List | Return Value |
|----------|---|--|
| LOG | $arg1$ number or node value | Natural logarithmic value Example: LOG (10) Example: LOG (INTERST+DAMAGE) |
| LOG10 | $arg1$ number or node value | 10-based logarithmic value |
| EXP | $arg1$ number or node value | Exponential value |
| ERF | $arg1$ number or node value | Error function value |
| ERFC | $arg1$ number or node value | Complementary error function value |
| ABS | $arg1$ number or node value | Absolute value |
| SQRT | $arg1$ number or node value | Square root value |
| MIN | $arg1$ number or node value : $argN$ number or node value Note: N must be larger than 1 | Smallest value Example: MIN (1 , 2 , 3) Example: MIN (BORON+PHOS , 1E21) |
| MAX | $arg1$ number or node value : $argN$ number or node value Note: N must be larger than 1 | Largest value Example: MAX (1 , 2 , 3) Example: MAX (BORON , PHOS+ARSENIC) |
| SUM | $arg1$ index $arg2$ number $arg3$ number $arg4$ number or node value | Summed value $\sum_{i=1}^{10} i^2$ is described as $SUM(i, 1, 10, i^2)$ $\sum_{i=-2}^2 \sum_{j=0}^5 k_{ij}$ is described as $SUM(i, -2, 2, SUM(j, 0, 5, k[i][j]))$ |

| Function | Argument List | Return Value |
|-------------------|--|---|
| SOLVE | <i>arg1</i> variable name <i>arg2</i> expression <i>arg3</i> initial value | Extract the <i>arg1</i> 's value which makes the expression equal to zero by Newton-Raphson iteration Example: Extract <i>x</i> near 1.0 from the equation $x^2 - 3x + 1 = 0$ SOLVE (<i>x</i> , $x^2 - 3*x + 1$, 1.0) Example: Extract C_a satisfying $C_{as} = C_a + \left(1 \times 10^{-17} \exp\left(\frac{0.4}{kT}\right) C_a\right)^{4.0}$ INTERM NAME=CT EXPR=1E-17*exp(0.4/kT) SOLVE (<i>Ca</i> , $Ca + (CT*Ca)^{4.0} - Cas$, <i>Cas</i>) |
| GRADIENT | <i>arg1</i> number or node value | Gradient vector Example: $\vec{\nabla} C$ is described as GRAD (<i>C</i>) Note: Only allowed for the parameters ADDTOEXP and INTERFAC in the EQUATION statement. |
| DIVERGENCE | <i>arg1</i> edge value | Divergence $\vec{\nabla} \cdot \left(\vec{\nabla} C + C \frac{q\vec{E}}{kT} \right)$ is described as $DIV(GRAD(C) + C * EFIELD)$ Note: Only allowed for the parameters ADDTOEXP and INTERFAC in the EQUATION statement. |
| DDT | <i>arg1</i> number or node value | Time derivative value Example: $\frac{\partial}{\partial t}(C + I)$ is described as DDT (<i>C+I</i>) Note: Only allowed for the parameters ADDTOEXP and INTERFAC in the EQUATION statement. |

Physical Functions

The following physical functions are examples of functions available in TSUPREM-4.

| Function | Argument List | Return Value |
|-----------------|---|---|
| EQACTIVE | <i>arg1</i> impurity name <i>arg2</i> number or node value | Equilibrium active concentration of <i>arg2</i> expression. Calculated with the solid solubility and equilibrium clustering coefficients of the impurity <i>arg1</i> . Example: EQACT (BORON , BORON-BCL) |
| ACTIVE | <i>arg1</i> impurity name | Active concentration of the impurity <i>arg1</i> Example: ACTIVE (BORON) Note: May not include the functions DIFFI , DIFFV and DIFFUSIVITY . |

| Function | Argument List | Return Value |
|-------------------|---------------------------|---|
| DIFFI | <i>arg1</i> impurity name | Diffusivity of the impurity <i>arg1</i> with interstitials, <i>i.e.</i> $\sum_i D_m^{(i)} \left(\frac{n}{n_i}\right)^{-i}$, where $D_m^{(i)}$ values are defined by the DIX.0 , DIX.E , DIP.0 , DIP.E , DIM.0 , DIM.E , DIMM.0 , DIMM.E , DIC.0 , and DIC.E parameters in the IMPURITY statement Example: DIFFI (BORON) Note: Not allowed for the built-in point defects <i>interst</i> and <i>vacancy</i> |
| DIFFV | <i>arg1</i> impurity name | Diffusivity of the impurity <i>arg1</i> with vacancies, <i>i.e.</i> $\sum_i D_n^{(i)} \left(\frac{n}{n_i}\right)^{-i}$, where $D_n^{(i)}$ values are defined by the DVX.0 , DVX.E , DVP.0 , DVP.E , DVM.0 , DVM.E , DVMM.0 , DVMM.E , DVC.0 , and DVC.E parameters in the IMPURITY statement Example: DIFFV (BORON) Note: Not allowed for the built-in point defects <i>interst</i> and <i>vacancy</i> . |
| DIFFUSIVIT | <i>arg1</i> impurity name | Diffusivity of the impurity <i>arg1</i> . If <i>arg1</i> is neither <i>interst</i> and nor <i>vacancy</i> , $\frac{I}{I^*} \sum_i D_m^{(i)} \left(\frac{n}{n_i}\right)^{-i} + \frac{V}{V^*} \sum_i D_n^{(i)} \left(\frac{n}{n_i}\right)^{-i}$, otherwise, $\frac{\sum_i D_I^{(i)} \phi_I^{(i)} \left(\frac{n}{n_i}\right)^{-i}}{\sum_i \phi_I^{(i)} \left(\frac{n}{n_i}\right)^{-i}}$ and $\frac{\sum_i D_V^{(i)} \phi_V^{(i)} \left(\frac{n}{n_i}\right)^{-i}}{\sum_i \phi_V^{(i)} \left(\frac{n}{n_i}\right)^{-i}}$ for <i>interst</i> and <i>vacancy</i> respectively. D.0 , D.E , DC.0 , DC.E , and C.STATE for D_I and D_V are specified on the INTERSTITIAL and VACANCY statements, respec- tively. Example: DIFFU (BORON) |

Interface Functions

The following interface functions are examples of functions available in TSUPREM-4.

| Function | Argument List | Return Value |
|---------------|----------------------------------|--|
| ONMAT1 | <i>arg1</i> number or node value | Values for the given expression <i>arg1</i> on the boundary of the MATERIAL (alias MAT1) material adjacent to the /MATERIA (alias /MAT2) material in the EQUATION statement. Example: ONMAT1 (PHOS / m) Note: Only allowed in the INTERFAC parameter in the EQUATION statement. |
| ONMAT2 | <i>arg1</i> number or node value | Values for the given expression <i>arg1</i> on the boundary of the /MATERIA (alias /MAT2) material adjacent to the MATERIAL (alias MAT1) material in the EQUATION statement. Example: ONMAT2 (PHOS / m) Note: Only allowed in the INTERFAC parameter in the EQUATION statement. |

Miscellaneous Functions

The following functions are examples of functions available in TSUPREM-4.

| Function | Argument List | Return Value |
|------------------|--|---|
| CONDITION | <i>arg1</i> logic expression <i>arg2</i> number or node value <i>arg3</i> number or node value | <i>arg2</i> if <i>arg1</i> is true, otherwise <i>arg3</i> Example: CONDITION (PHOS > 1E21 , 1E21 , PHOS) Note: Same as <i>arg1</i> ? <i>arg2</i> : <i>arg3</i> . (PHOS > 1E21 ? 1E21 : PHOS) |
| EXIST | <i>arg1</i> solution variable name | true(1) if <i>arg1</i> exists in the structure, otherwise false (0). Example: EXIST (BORON) |

6.3 Examples

The following USEIT models are detailed in this examples section:

- Interstitial clustering model
- Transient clustering model
- Dislocation-loop model
- Diffusion in **PD**. **TRANS**
- Interface trap model
- Boron-Interstitial clustering model with small clusters
- Diffusion enhancement in thinner oxides
- Trapped nitrogen dependent surface recombination rate
- 5-stream diffusion model
- SiGe model

Interstitial Clustering Model

In the interstitial clustering model:

- Name the new impurity, *ICL*.
- Specify the **NONE** parameter in the **METHOD** statement since the new impurity *ICL* is supposed to be immobile.
- Set the reaction rate k_{fi} of the initial clustering term to zero.
- Set k_{fc}, k_r as intermediates because they depend *only* on temperature.
- Set *G* as an intermediate because its quantity is used in both *ICL* and interstitial equations.
- Set the **CL.IFC**, **CL.ISFC**, and **CL.CR** values to 1.0.
- Set the **CL.CF** value to 0.9398.
- Set the initial background concentration to $10^9/\text{cm}^3$.
- The function call, **EXIST(ICL)**, checks to see if the variable *ICL* already exists.
- The **INITIAL** condition implies that the initialization takes place only when the *ICL* solution does not exist.

The change of interstitial concentration due to recombination with clusters must be taken into account by subtracting the *G* term from the interstitial equation.

$$\frac{\partial C}{\partial t} = k_{fi} \cdot \frac{I^{\text{CL.IFI}}}{I^{\text{CL.ISFI}}} + k_{fc} \cdot \frac{I^{\text{CL.IFC}}}{I^{\text{CL.ISFC}}} (C + I)^{\text{CL.CF}} - k_r \cdot C^{\text{CL.CR}}$$

```

$----- Start of implementation -----
IMPURITY IMP=ICL NEW
METHOD IMP=ICL NONE
INTERMED NAME=kfc EXPRESS=5.207E14*exp(-3.774/kt)
INTERMED NAME=kr EXPRESS=9.431E13*exp(-3.017/kt)
INTERMED NAME=G +
EXPRESS=kfc*CI.RATIO*(ICL+INTER)^0.9398-kr*ICL
EQUATION MODEL=ICL VARIABLE=ICL MAT=Si +
INIT=EXIST(ICL)?ICL:1E9 ADDTOEXP=G
EQUATION MODEL=ICL VARIABLE=INTERS MAT=Si ADDTOEXP=-G
$----- End of implementation -----

METHOD PD.FULL
IF (1) ;$ use USEIT model
METHOD MODEL=ICL ENABLE ;$ turn-on USEIT model
INTERST MAT=Si ^CL.MODEL ;$ turn-off built-in model
ELSE ;$ use built-in model
METHOD MODEL=ICL ^ENABLE;$ turn-off USEIT model
INTERST MAT=Si CL.MODEL ;$ turn-on built-in model
IF.END

```

Transient Clustering Model

In the transient clustering model:

- Even though C_a appears as a solution variable in the model equation, solving the equation for the clustered boron concentration $BCL (=C-C_a)$ is easier because the clustered borons are immobile.
- Define the intrinsic carrier concentration n_i and the time constant τ_a for activation.
- The **EQACTIVE(boron, boron)** returns the equilibrium active concentration of total boron profile.
- Since the initial active concentration is assumed to be a level comparable to n_i , define the initial clustered concentration to the value **MAX(boron-ni, 0.0)** in the **INITIAL** parameter.
- If BCL already exists, it is initialized to the difference between the total boron concentration and the active concentration C_a .
- The expression of the intermediate C_a ensures that the active concentration stays above n_i .

$$\frac{\partial(C - C_a)}{\partial t} = \frac{C_a - C_a^*}{\tau_a},$$

$$C_a \geq \min(C_a^*, n_i) \text{ for initialized active concentration}$$

```

$----- Start of implementation -----
IMPURITY IMP=BCL NEW
METHOD IMP=BCL NONE
INTERMED NAME=ni +
          EXPRESS=3.87E16*exp(-0.605/kt+1.5*log(temp))
INTERMED NAME=ta EXPRESS=8.0E-16*exp(4.2/kt)
INTERMED NAME=Ceq EXPRESS=EQACT(boron, boron)
INTERMED NAME=Ca EXPRESS=MAX(boron-BCL, MIN(Ceq, ni))
IMPURITY MODEL=TCL IMP=boron MAT=Si ACTIVE=Ca
EQUATION MODEL=TCL VARIABLE=BCL MAT=Si +
          INIT=EXIST(BCL)?boron-Ca:MAX(boron-ni, 0.0) +
          ADDTOEXP=(boron-BCL-Ceq)/ta
$----- End of implementation -----

METHOD PD.FULL
IF (1) ;$turn on USEIT model
METHOD MODEL=TCL ENABLE ;$ turn-on USEIT model
METHOD ACT.EQUI ;$ turn-off built-in model
ELSE ;$turn on built-in model
METHOD MODEL=TCL ^ENABLE;$ turn-off USEIT model
METHOD ACT.TRAN ;$ turn-on built-in model
IF.END
    
```

Dislocation-Loop Model

In the dislocation-loop model:

- Specify the initial volume density and radius of dislocation-loops by the *LDENS* and *LRAD* intermediates, respectively.
Even though the default values are assumed to be $1 \times 10^{15}/\text{cm}^3$ and 300\AA for density and radius, respectively, their values can be changed for different implant conditions.
- Calculate the radius *RDL* in angstroms.
- Define the new variable *RDL* first in the **EQUATION** statement so that the following warning message will be shown:

*** Warning: rdl does not exist. It will be newly defined.

- The expression in the **INITIAL** parameter represents that the 300\AA dislocation-loops form initially in the place where the total interstitial concentration is in the range between $1 \times 10^{20}/\text{cm}^3$ and $1.15 \times 10^{22}/\text{cm}^3$ while the current dislocation-loops elsewhere are kept.
- Since it is assumed that the Frenkel-pair point-defect *damage* and the previous interstitials *interst* are distinguished, the total interstitial concentration is the sum of both. *LP1* and *LP2* are $\exp(\gamma\Omega/(bkT))$, $\mu b\Omega/(4\pi kT(1-\nu))$, respectively.
- *ileq* represents the equilibrium interstitial concentration around the dislocation-loop.
- Loops of radius 100\AA or less dissolve quickly during diffusion so that such loops are discarded, which is specified by the **UPDATE** parameter.
Thus, the time derivative of the loop radius *RDL* is taken into account only in the region $RDL > 100\text{\AA}$.
- In the **METHOD** statement, the **VARIABLE** parameter is an alias for the **IMPURITY** parameter.
- Since the *RDL* is supposed to be immobile, the factorization method is defined to **NONE**.
- Set the absolute error to 0.1. The unit of the absolute error is the same as one of the solution variable. The absolute error value is set to 1×10^9 by default, unless it is specified.
- Since the **REL. ERR** parameter is not specified, the relative error is set to 0.01 by default.

$$\frac{dr}{dt} = \frac{\pi}{N_0} K_L D_I (I - I_{loop}^*)$$

$$R_I = 2\pi^2 K_L D_I \rho_v r (I - I_{loop}^*) \text{ for the rate of absorption of interstitials by dislocation loops}$$

$$\text{where } I_{loop}^* = I^* \exp\left(\frac{\gamma\Omega}{bkT}\right) \exp\left(\frac{\mu b\Omega}{4\pi r kT(1-\nu)} \ln\left(\frac{8r}{b}\right)\right)$$

```

$----- Start of implementation -----
INTERMED NAME=LDEN  EXPR=1E15
INTERMED NAME=LRAD  EXPR=300
EQUATION MODEL=DLL VAR=RDL MAT=Si +
  INIT="(1e20<inter+damage&&inter+damage<=1.15e22)? +
  LRAD:(EXIST(RDL)?RDL:0.0)"
INTERMED NAME=LP1   EXPR=exp(2.783E-2/kt)
INTERMED NAME=LP2   EXPR=2.877/kt
INTERMED NAME=KL    EXPR=29.8047E8*exp(-0.4/kt)
INTERMED NAME=ileq  +
  EXPR=CI.STAR*LP1*exp(LP2/RDL*(log(2.55*RDL)))
INTERMED NAME=KD    EXPR=KL*DIFFU(inters)*(inters-ileq)
EQUATION MODEL=DLL VARIABLE=inters MAT=silicon +
  ADDTOEXP=-1.97e-15*LDEN*RDL*KD
EQUATION MODEL=DLL VARIABLE=RDL MAT=silicon +
  ADDTOEXP=RDL>100?2.00E-15*KD:0.0 +
  UPDATE=RDL>100?RDL:0.0
METHOD VARIABLE=RDL NONE ABS.ERR=0.1
$----- End of implementation -----

METHOD PD.FULL
IF (1) ;$ use USEIT model
METHOD MODEL=DLL ENABLE
INTERMED NAME=LRAD EXPR=350
INTERMED NAME=LDEN EXPR=2E15
IMPLANT ARSENIC DOSE=1E15 ENERGY=30
ELSE ;$ use built-in model
METHOD MODEL=DLL ^ENABLE
IMPLANT ARSENIC DOSE=1E15 ENERGY=30 +
  L.DEN=2E15 L.RAD=350E-8
IF.END
    
```

Diffusion in PD.TRANS

In the diffusion in **PD.TRANS**:

- Use the **DIVERGENCE()** and **GRADIENT()** functions for divergence and gradient calculations, respectively.
- The **DIFFUSIVITY(boron)** is the same as **CI.RATIO*DIFFI(boron)+CV.RATIO*DIFFV(boron)**.
- Turning *off* the **DIFFUSE** parameter in the **IMPURITY** statement disables the built-in diffusion equations for the specified impurity in the specified material.

$$\frac{\partial C}{\partial t} = \vec{\nabla} \cdot \left(\left(\frac{I}{I^*} \sum_i D_m^{(i)} \left(\frac{n}{n_i} \right)^i + \frac{V}{V^*} \sum_i D_n^{(i)} \left(\frac{n}{n_i} \right)^i \right) \left(\vec{\nabla} C_m + C_m \frac{q\vec{E}}{kT} \right) \right)$$

```

$----- Start of implementation -----
INTERMED NAME=Db EXPRESS=DIFFUS(boron)
INTERMED NAME=Cm EXPRESS=ACTIVE(boron)
EQUATION MODEL=PDTRANS VARIABLE=boron MAT=silicon +
      ADDTOEXP=DIV(Db*(GRAD(Cm)+Cm*EFIELD))
$----- End of implementation -----

METHOD PD.TRANS
IF (1) ;$ use USEIT model
  $ turn-off the built-in diffusion equation
  IMPURITY IMP=boron MAT=Si ^DIFFUSE
  METHOD MODEL=PDTRANS ENABLE
ELSE
  METHOD MODEL=PDTRANS ^ENABLE
IF.END

```

Interface Trap Model

In the interface trap model:

- The **BOUNDARY** parameter in the **IMPURITY** statement specifies that the new impurity exists only on boundaries.
- For phosphorus, set the coefficient values for h_{Si} , h_{ox} , κ_{Si} , κ_{ox} and σ_{max} to $0.715 \exp(-1.75\text{eV}/kT)$, $10h_{Si}$, $5882 \exp(0.37\text{eV}/kT)$, $\kappa_{Si}/30.0$, and 6.8×10^{14} , respectively.
- Set the diffusivity D_o of trapped phosphorus along interfaces to $1 \times 10^5 \exp(-4.44\text{eV}/kT)$.
- Since the interfaces mediate phosphorus between oxide and silicon, the parameter **TRANSPOR** must be turned *off*.
- The function **ONMAT1**() returns the values on the boundaries of **MAT1** adjacent to **/MAT2**.
- The function **ONMAT2**() returns the values on the boundaries of **/MAT2** adjacent to **MAT1**.
- The **BOUNDARY** parameter in the **EQUATION** statement specifies that the diffusion term in the description of the **ADDTOEXP** parameter considers the diffusion of trapped phosphorus along the boundaries of **MAT1** adjacent to **/MAT2**.

$$\frac{\partial \sigma}{\partial t} = -\frac{\partial}{\partial l} \left(D_{\sigma} \frac{\partial \sigma}{\partial l} \right) + F_{ox} + F_{Si} \quad \text{where} \quad F_i = h_i \left(C_i \left(1 - \frac{\sigma}{\sigma_{max}} \right) - \kappa_i \sigma \right)$$

It is assumed that the detrapping flux from interface to silicon is independent of the surface concentration in silicon and that the competitive occupation for the trap sites is negligible.

```

$----- Start of implementation -----
IMPURITY IMP=IPHO NEW BOUNDARY
METHOD IMP=IPHO NONE ABS.ERR=1.0e5
INTERMED NAME=dip EXPR=1e5*exp(-4.44/kt)
INTERMED NAME=hsi EXPR=0.715*exp(-1.75/kt)
INTERMED NAME=hox EXPR=10*hsi
INTERMED NAME=ksi EXPR=5882*exp(0.37/kt)
INTERMED NAME=kox EXPR=ksi/30
INTERMED NAME=qmax EXPR=6.8E14
EQUATION MODEL=DLOSS VARIABLE=PHOS MAT1=Si /MAT2=Ox +
INTERFAC="-hsi*(ONMAT1(PHOS)*(1-ONMAT2(IPHO)/qmax)+
-ksi*ONMAT2(IPHO))" ^TRANSPOR
EQUATION MODEL=DLOSS VARIABLE=PHOS MAT1=Ox /MAT2=Si +
INTERFAC="-hox*(ONMAT1(PHOS)*(1-ONMAT1(IPHO)/qmax)+
-kox*ONMAT1(IPHO))" ^TRANSPOR
EQUATION MODEL=DLOSS VARIABLE=IPHO MAT1=Ox /MAT2=Si +
BOUNDARY ADDTOEXP=DIV(dip*GRAD(IPHO)) +
INTERFAC="hsi*(ONMAT2(PHOS)*(1-ONMAT1(IPHO)/qmax)+
-ksi*ONMAT1(IPHO)) +
+hox*(ONMAT1(PHOS)*(1-ONMAT1(IPHO)/qmax)+
-kox*ONMAT1(IPHO))" ^TRANSPOR
$----- End of implementation -----

IF (1) ;$turn on USEIT model
METHOD MODEL=DLOSS ENABLE
METHOD ^ITRAP
ELSE ;$turn on built-in model
METHOD MODEL=DLOSS ^ENABLE
METHOD ITRAP
IMPURITY IMP=I_PHO MAT=Ox /MAT=Si CM.SEC +
^/SEG.SS SEG.EQ3 +
DIX.0=1E5 DIX.E=4.44 +
TRANS.0=7.15 TRANS.E=1.75 +
/TRANS.0=0.715 /TRANS.E=1.75 +
SEG.0=196 SEG.E=-0.37 +
/SEG.0=5882 /SEG.E=-0.37

IF.END

```


Boron-Interstitial Clustering Model with Small Clusters

In the boron-interstitial clustering model with small clusters:

- The *cieq* and *cveq* intermediates are the intrinsic equilibrium concentrations of interstitials and vacancies, respectively.
- The intermediate *phiv* specifies the charge state fraction of point-defects so that it depends on the charge states.
- After initializing *phiv* array elements with zero values, the values are assigned by the **LOOP** statement.
- The index string *i* in the **LOOP** statement is expanded by '@' character as the quantity defined by the **ASSIGN** statement.
- The small clusters of vacancies are ignored so that the K_{V_i} , V_c , and F_{I_c} are set to zero.
- The relative reaction rate K_{cV} between small interstitial clusters and vacancies depends on each charge state so it can be represented by a two-dimensional array intermediate.
- It is assumed that all K_I values for all charge states but the neutral state are zeros. All K_{cV} values are initialized to a constant value of 1. This implies that the Fermi-level dependences of K_I and K_{cV} are ignored. Thus, the K_I and K_{cV} values do not need to be specified with arrays.

K_I and K_{cV} values, however, are specified with arrays in order to show how to use arrays.

- The active concentration should be adjusted to account for dopant-defect clusters.
- It is hypothesized that B_3I is the dominant boron-interstitial cluster at longer diffusion times. Thus, the parameters **DDCF.D.N**, **DDCF.N.N**, **DDCF.I.N**, **DDCR.N.N**, and **DDCR.I.N** have been specified to 3.0, -2.0, 2.0, 0.0, and 1.0 respectively.
- Ideally, **IFRACM** should be equal to $(\text{DDCF.I.N} - \text{DDCR.I.N}) / \text{DDCF.D.N}$, but it is given the value 1.356 which is calibrated together with K_{dd} and τ_{dd} .
- Since the matrix assembly in TSUPREM-4 assumes the time derivative term of a solution variable to the left-hand-side (LHS) of equation, the term $\frac{\partial}{\partial t}(I_c + I_{dds})$ is subtracted from the right-hand-side (RHS) term of interstitial equation.

$$\frac{\partial I}{\partial t} + \frac{\partial(I_c + I_{dds})}{\partial t} = RHS - \mathbf{IFRACM} \cdot \frac{\partial C_{dd}}{\partial t} - K_b(F_{cV} + F_{Ic}) \left(\frac{I}{I^*} \frac{V}{V^*} - 1 \right)$$

$$\frac{\partial V}{\partial t} + \frac{\partial V_c}{\partial t} = RHS - K_b(F_{cV} + F_{Ic}) \left(\frac{I}{I^*} \frac{V}{V^*} - 1 \right)$$

$$\frac{\partial C_{dd}}{\partial t} = \frac{1}{\tau_{dd}} \left(K_{dd} n_i \left(\frac{C_a}{n_i} \right)^{\mathbf{DDCF.D.N}} \eta^{\mathbf{DDCF.N.N}} \left(\frac{I}{I^*} \right)^{\mathbf{DDCF.I.N}} - C_{dd} \eta^{\mathbf{DDCR.N.N}} \left(\frac{I}{I^*} \right)^{\mathbf{DDCR.I.N}} \right)$$

$$\text{where } I_c = \mathbf{ECLUST.N} \sum_i (K_{Ii} \eta^{-i}) \left(\frac{I}{I^*} \right)^{\mathbf{ECLUST.N}},$$

$$V_c = \mathbf{ECLUST.N} \sum_i (K_{Vi} \eta^{-i}) \left(\frac{V}{V^*} \right)^{\mathbf{ECLUST.N}},$$

$$I_{dds} = \mathbf{IFRACS} (C - C_{dd} - C_{a2}),$$

$$C_{a2} = \frac{C - C_{dd}}{1 + \mathbf{DDCS.0} \cdot \exp\left(-\frac{\mathbf{DDCS.E}}{kT}\right) \left(\frac{I}{I_i^*} \right)^{\mathbf{IFRACS}}},$$

$$F_{cV} = \left(\frac{I}{I^*} \right)^{\mathbf{ECLUST.N}-1} V_i^* \sum_{i,j} (K_{cV,ij} K_{Ii} \phi_{Vj} \eta^{-i-j}),$$

$$F_{Ic} = \left(\frac{V}{V^*} \right)^{\mathbf{ECLUST.N}-1} I_i^* \sum_{i,j} (K_{Ic,ij} K_{Vj} \phi_{Ii} \eta^{-i-j})$$

```

$----- Start of implementation -----
IMPURITY IMP=BIC NEW
METHOD IMP=BIC NONE ABS.ERR=1.0e5
INTERMED NAME=ni +
    EXPR=3.87e16*exp(-0.605/kt+1.5*log(temp))
INTERMED NAME=cieq EXPR=1.25E29*exp(-3.26/kt)
INTERMED NAME=cveq EXPR=1.25E29*exp(-3.26/kt)
INTERMED NAME=phiv MININDEX=-2 SIZE=5 +
    EXPRESS="32.47*exp(-0.62/kt), 5.68*exp(-0.145/kt), +
        1.0, +
        5.58*exp(-0.455/kt), 0.0"
INTERMED NAME=tphiv EXPR="SUM(i,-2,2,phiv[i])"
INTERMED NAME=phiv MININDEX=-2 SIZE=5 DEFAULT=0.0
LOOP STEPS=5 INDEX=i START=-2
    INTERMED NAME=phiv[@i] EXPR=phiv[@i]/tphiv
L.END
INTERMED NAME=ki MININDEX=-2 SIZE=5 DEFAULT=0.0
INTERMED NAME=ki[0] EXPRESS=6.69E-2*exp(1.096/kt)
INTERMED NAME=kcV +
    MININDEX="-2,-2" SIZE="5,5" DEFAULT=1
INTERMED NAME=FcV +
    EXPR="CI.RATIO*cveq*SUM(i,-2,2,SUM(j,-2,2, +
        kcV[i][j]*ki[i]*phiv[j]*ETA^(-i-j)))"
INTERMED NAME=ddcs EXPR=8.73796*exp(-2.769/kt)
INTERMED NAME=kddf EXPR=4.3356E-5*exp(0.224/kt)
INTERMED NAME=tdd EXPR=EXP(LOG(4.18529E-31)+7.765/kt)
INTERMED NAME=Ca1 EXPR=BORON-BIC
INTERMED NAME=Ca2 EXPR=Ca1/(1+ddcs*(inter/cieq)^2)
INTERMED NAME=Ca EXPR=EQACTIVE(BORON,Ca2)
INTERMED NAME=Idds EXPR=2*(Ca1-Ca2)
INTERMED NAME=Ic +
    EXPR=2*SUM(i,-2,2,ki[i]/eta^i)*CI.RATIO*CI.RATIO
INTERMED NAME=Rb +
    EXPR=1E-21*exp(1/kt)*FcV*(CI.RATIO*CV.RATIO-1)
INTERMED NAME=dBIC +
    EXPR="(kddf*ni*(Ca/ni)^3*(CI.RATIO/ETA)^2 +
        -BIC*CI.RATIO)/tdd"
IMPURITY MODEL=BIC IMP=BORON MAT=Si ACTIVE=Ca
EQUATION MODEL=BIC VARIABLE=INTER MAT=Si +
    ADDTOEXP=-DDT(Ic+Idds)-1.356*dBIC-Rb
EQUATION MODEL=BIC VARIABLE=VACAN MAT=Si ADDTOEXP=-Rb
EQUATION MODEL=BIC VARIABLE=BIC MAT=Si ADDTOEXP=dBIC +
    INIT=EXIST(BIC)?BIC:1E5
$----- End of implementation -----

```

```

IF (1) ;$turn on USEIT model
METHOD  MODEL=BIC ENABLE ;$ turn-on USEIT model
METHOD  ACT.EQUI          ;$ turn-off built-in model
ELSE    ;$turn on built-in model
METHOD  MODEL=BIC !ENABLE;$ turn-off USEIT model
METHOD  ACT.FULL          ;$ turn-on built-in model
INTERST MAT=SI C.STATE=0 +
        ECLUST.0=0.0669 ECLUST.E=-1.096 ECLUST.N=2
INTERST MAT=SI C.ALL KCV.0=1 KCV.E=0
VACANCY MAT=SI C.STATE=0 +
        ECLUST.0=0 ECLUST.E=3.26 ECLUST.N=2
VACANCY MAT=SI C.ALL KIC.0=1 KIC.E=0
IMPURITY IMP=BORON MAT=SI
        IFRACS=2 IFRACM=1.356
        DDC.F.0=4.3356E-5 DDC.T.0=4.18529E-31
        DDC.F.E=-0.2240 DDC.T.E=-7.765
        DDCF.D.N=3.0 DDCF.N.N=-2.0 DDCF.I.N=2.0
        DDCR.N.N=0 DDCR.I.N=1.0
        DDCS.0=8.73796 DDCS.E=2.769
IF.END

```

Diffusion Enhancement in Thinner Oxides

In the diffusion enhancement in thinner oxides:

- Since TSUPREM-4 assumes that interstitials exist only in silicon material, the silicon atom transported by segregation from silicon into oxide is specified as the new impurity *SOX*.
- The *SOOX* is a SiO molecule produced by the reaction between *SOX* and oxygen O₂ in oxides.
- Solving the equation of mobile, *SOX* uses the nearest neighbor to fill terms produced in factorization of the block matrices so that the **PART** method is specified.
- The *SOOX* is assumed to be immobile so that the **NONE** method is used for factorization.
- Since the interstitials in silicon are transported to oxides such as *SOX* impurities, the **TRANSPOR** parameters for interstitials and *SOX* should be turned *off*.
- The surfaces of structures (i.e., ambient/material) have the exposed boundary condition so the **TRANSPOR** should be turned *off* for the surface recombination of *SOX*.
- The **DI.FAC** parameter is specified to the enhancement factor of boron diffusivity in oxides.

$$\frac{\partial C_{Si}}{\partial t} = -\vec{\nabla} \cdot (-D_{Si} \vec{\nabla} C_{Si}) - k_{Si} C_{Si} \sqrt{C_{O_2}}$$

$$\frac{\partial C_{SiO}}{\partial t} = k_{Si} C_{Si} \sqrt{C_{O_2}} - k_{SiO} C_{SiO} \sqrt{C_{O_2}}$$

where $F_{Si} = h_{Si}(C_{Si} - mI)$ at silicon/oxide interfaces,

$$F_{Si} = k_i C_{Si} \text{ at oxide surfaces}$$

Diffusion enhancement is
$$D = D_o \left(1 + f_{SiO} \frac{C_{SiO}}{C_{SiO_2}} \right)$$

```

$----- Start of implementation -----
IMPURITY IMP=SOX      NEW
IMPURITY IMP=SOOX     NEW
METHOD    IMP=SOX     PART ABS.ERR=1E9
METHOD    IMP=SOOX     NONE ABS.ERR=1E9
INTERMED  NAME=Dsi    EXPR=13.0*exp(-4.5/kt)
INTERMED  NAME=ksi    EXPR=7.676E6*exp(-4.0/kt)
INTERMED  NAME=ksio   EXPR=ksi
INTERMED  NAME=hsi    EXPR=0.01
INTERMED  NAME=m      EXPR=2.9E-5*exp(2.19/kt)
INTERMED  NAME=ki     EXPR=8.67E7*exp(-4.0/kt)
INTERMED  NAME=CO     EXPR=SQRT(O2)
INTERMED  NAME=R      EXPR=ksi*SOX*CO
EQUATION  MODEL=DENH  VARIABLE=SOX  MAT=Ox  +
          ADDTOEXP=DIV(Dsi*GRAD(SOX))-R  +
          INIT=EXIST(SOX)?SOX:1E5
EQUATION  MODEL=DENH  VARIABLE=SOOX  MAT=Ox  +
          ADDTOEXP=R-ksio*SOOX*CO      +
          INIT=EXIST(SOOX)?SOOX:1E5
EQUATION  MODEL=DENH  VARIABLE=INTERS MAT1=Si /MAT2=Ox  +
          INTERF=hsi*(ONMAT2(SOX)-m*ONMAT1(INTERS)) ^TRANSPOR
EQUATION  MODEL=DENH  VARIABLE=SOX   MAT1=Ox /MAT2=Si  +
          INTERF=hsi*(m*ONMAT2(INTERS)-ONMAT1(SOX)) ^TRANSPOR
EQUATION  MODEL=DENH  VARIABLE=SOX   MAT1=Ox /MAT2=AMB  +
          INTERFAC=-ki*ONMAT1(SOX) ^TRANSPOR
IMPURITY  MODEL=DENH  IMP=BORON  MAT=Ox  +
          DI.FAC=1+4.55E-22*exp(0.28/kt)*SOOX
$----- End of implementation -----

IF (1) ;$turn on USEIT model
METHOD  MODEL=DENH  ENABLE
METHOD  MODEL=DEOX ^ENABLE
ELSE   ;$turn on built-in model
METHOD  MODEL=DENH ^ENABLE
METHOD  MODEL=DEOX ENABLE
IF.END

```

Surface Recombination Rate Depending On Nitrogen

In the surface recombination model of interstitials for nitridation effect:

- Specify the interface at which the recombination happens with the **MAT** and **/MAT** parameters.
- The surface recombination is reduced by $K_s(\sigma_N/N_{max})(I-I^*)$.
- Use the **INTERFACE** parameter in order to describe the reduction of the surface recombination rate.
- Since nitrogen atoms are trapped on the boundary nodes in oxides, call the function **ONMAT2 ()** to the trapped nitrogen solution, *i_nitrogen*.
- Turn *off* the **TRANSPOR** parameter in order to avoid transporting interstitials into oxides.

$$R = K_s \left(1 - \frac{\sigma_N}{N \cdot \text{MAX} \cdot 0 \exp\left(-\frac{N \cdot \text{MAX} \cdot E}{kT}\right)} \right) (I - I^*)$$

```

$----- Start of implementation -----
INTERMED NAME=Ks   EXPRESS=1.4E-6*EXP(1.75/kt)
INTERMED NAME=Nmax EXPRESS=15/14*2E14
EQUATION MODEL=INREC VARIABLE=interst MAT=silicon /MAT=oxide +
          INTERFAC=Ks*ONMAT2(i_nitrogen)/Nmax*(interst-CI.STAR) +
          ^TRANSPOR
$----- End of implementation -----

IF (1) ;$ use USEIT model
METHOD  MODEL=INREC ENABLE
ELSE   ;$turn on built-in model
METHOD  MODEL=INREC ^ENABLE
INTERST SILICON /OXIDE N.MAX.0=15/14*2E14 N.MAX.E=0.0
METHOD  PD.NREC
IF.END

```

5-Stream Diffusion Model

The 3-stream diffusion model (**PD.FULL**) in TSUPREM-4 has been proven to be accurate enough to simulate most silicon process technologies. In the **PD.FULL** model, the quasi-static assumption under local equilibrium eliminates the equations for dopant-defect pairs. However, at very low thermal temperature, this assumption may not be applied any more, and the whole 5-stream diffusion system must be introduced. This example shows how to implement the 5-stream diffusion model with USEIT.

Bulk Equations

$$\frac{\partial C}{\partial t} = -(\vec{\nabla} \cdot \vec{J}_m + \vec{\nabla} \cdot \vec{J}_n)$$

$$\frac{\partial M}{\partial t} = -\vec{\nabla} \cdot \vec{J}_m + K_m \left(S \frac{I}{I^*} - \frac{M}{\alpha_m} \right) + K_{mv} \left(S - \frac{M V}{\alpha_m V^*} \right)$$

$$\frac{\partial N}{\partial t} = -\vec{\nabla} \cdot \vec{J}_n + K_n \left(S \frac{V}{V^*} - \frac{N}{\alpha_n} \right) + K_{ni} \left(S - \frac{N I}{\alpha_n I^*} \right)$$

$$\frac{\partial I}{\partial t} = -\vec{\nabla} \cdot \vec{J}_I - R_b - K_m \left(S \frac{I}{I^*} - \frac{M}{\alpha_m} \right) + K_{ni} \left(S - \frac{N I}{\alpha_n I^*} \right)$$

$$\frac{\partial V}{\partial t} = -\vec{\nabla} \cdot \vec{J}_V - R_b - K_n \left(S \frac{V}{V^*} - \frac{N}{\alpha_n} \right) + K_{mv} \left(S - \frac{M V}{\alpha_m V^*} \right)$$

where I and V denote an interstitial and a vacancy respectively, and S, M and N denote a substitutional dopant, a dopant-interstitial pair and a dopant-vacancy pair, respectively.

$$\vec{J}_m = -D_m \left(\vec{\nabla} \left(\frac{M}{\alpha_m} \right) + z_s \left(\frac{M}{\alpha_m} \right) \left(\frac{\vec{\nabla} \eta}{\eta} \right) \right)$$

$$\vec{J}_n = -D_n \left(\vec{\nabla} \left(\frac{N}{\alpha_n} \right) + z_s \left(\frac{N}{\alpha_n} \right) \left(\frac{\vec{\nabla} \eta}{\eta} \right) \right)$$

$$\vec{J}_I = -D_I I^* \vec{\nabla} \left(\frac{I}{I^*} \right)$$

$$\vec{J}_V = -D_V V^* \vec{\nabla} \left(\frac{V}{V^*} \right)$$

$$\text{where } D_m \equiv D_M \sum_k a_m \phi_{mk} \eta^{-k}, \quad D_n \equiv D_N \sum_k a_n \phi_{nk} \eta^{-k}, \quad \alpha_m \equiv a_m F_m = \frac{D_m}{D_M}$$

$$\text{and } \alpha_n \equiv a_n F_n = \frac{D_n}{D_N}.$$

The reaction rates K_m , K_n , K_{mv} , and K_{ni} are given by:

$$K_m \equiv \sum_k (g_{mk} \phi_{Ik} \eta^{-k})$$

$$K_n \equiv \sum_k (g_{nk} \phi_{Vk} \eta^{-k})$$

$$K_{mv} \equiv \sum_k (r_{mvk} a_m \phi_{mk} \phi_{V(-k)})$$

$$K_{ni} \equiv \sum_k (r_{nik} a_n \phi_{nk} \phi_{I(-k)})$$

The point-defect recombination R_b is given by:

$$R_b = -\sum_j \sum_k (r_{IV,jk} \phi_{Ij} \phi_{Vk} \eta^{-(j+k)}) \left(1 - \frac{I}{I^*} \frac{V}{V^*}\right)$$

Boundary Conditions

Since the dopant-defect pairs are mobile, their boundary conditions (recombination or transporting flux) must be specified. In the other materials but silicon, the point-defects, and also dopant-defect pairs, are ignored so that the annihilation of dopant-defect pairs at interfaces needs to be specified. The recombinations are given by:

$$F_m = h \left(\frac{M}{m} - M^* \right)$$

$$F_n = h \left(\frac{N}{m} - N^* \right)$$

where $M^* = \frac{\alpha_m C_a}{1 + \alpha_m + \alpha_n}$ and $N^* = \frac{\alpha_n C_a}{1 + \alpha_m + \alpha_n}$

The h and m are given by the transport and segregation coefficients of an impurity, respectively.

Implementation

Since USEIT allows the additional terms to be added to built-in equations, one might assume that the kick-out and Frank-Turnbull reaction terms are simply added to the built-in point-defect equations instead of using the built-in reaction terms in **PD.FULL**. However, this is not possible because the built-in initialization procedure of free point-defects at the first post-implant annealing process is performed by assuming that the dopant-defect pairs are already formed in equilibrium. Thus, the whole equation system for point-defects must be implemented.

Point defect parameter

The parameters for the diffusivities, equilibrium concentrations, charge fractions, surface recombination rates, and bulk recombination rate, as well as interstitial generation for oxidation, are specified as intermediates. Note that the charge fraction intermediates are the arrays starting with -2(double negative), the size of which is 5, and they count up to double positive. When you want to see the effect

due to a different parameter set, the specified values can be replaced with the new ones.

```

intermed name=DIO      expr=3.65e-4
intermed name=DIE      expr=1.58
intermed name=DV0      expr=3.65e-4
intermed name=DVE      expr=1.58
intermed name=IiS0     expr=1.25e29
intermed name=IiSE     expr=3.26
intermed name=ViS0     expr=1.25e29
intermed name=ViSE     expr=3.26
intermed name=PhiIO    size=5 minindex=-2 +
                      expr=" 0.0, 5.68, 1.0, 5.68, 0.0"
intermed name=PhiIE    size=5 minindex=-2 +
                      expr=" 0.0, 0.50, 0.0, 0.26, 0.0"
intermed name=PhiV0    size=5 minindex=-2 +
                      expr="32.47, 5.68, 1.0, 5.68, 0.0"
intermed name=PhiVE    size=5 minindex=-2 +
                      expr=" 0.62, 0.145,0.0, 0.455, 0.0"
intermed name=KSI0     expr=1.4E-6
intermed name=KSIE     expr=-1.75
intermed name=KSV0     expr=4.0E-11
intermed name=KSVE     expr=-1.75
intermed name=KB0      expr=1E-21
intermed name=KBE      expr=-1.0
$ I small clusters
intermed name=kIO      size=5 minindex=-2 default=0.0
intermed name=kIE      size=5 minindex=-2 default=0.0
intermed name=kIO[0]   expr=0.0668848
intermed name=kIE[0]   expr=-1.096
intermed name=kcV      minindex="-2,-2" length="5,5" default=1.0
$ OED
intermed name=VNORM0   expr=1.0
intermed name=VNORME   expr=0.0
intermed name=THETA0   expr=0.01
intermed name=THETAE   expr=0.0
intermed name=GPOW0    expr=-0.7
intermed name=GPOWE    expr=0.0

```

Dopant parameter

The diffusivities for dopant-defect pairs, the reaction rates for pairing with point-defects, the diffusivities in silicon, and the segregation flux are defined. The specified values here are for boron.

```

intermed name=DIPAIRO expr=3.65e-4
intermed name=DIPAIRE expr=1.58
intermed name=DVPAIRO expr=3.65e-4
intermed name=DVPAIRE expr=1.58
intermed name=RIS expr=10
intermed name=RVS expr=10
intermed name=RIPV expr=10
intermed name=RVPI expr=10
intermed name=EIS expr=0.0
intermed name=EVS expr=0.0
intermed name=EIPV expr=0.0
intermed name=EVPI expr=0.0
intermed name=FIB expr=0.95
intermed name=DX0 expr=0.0368
intermed name=DXE expr=3.46
intermed name=DP0 expr=0.716
intermed name=DPE expr=3.46
intermed name=TRANS0 expr=1.67E-7
intermed name=TRANSE expr=0.0
intermed name=SEG0 expr=1.126E3
intermed name=SEGE expr=0.91
    
```

**Thermal parameter
for point defects**

```

$ point-defect diffusivity
intermed name=DI expr=DI0*exp(-DIE/kt)
intermed name=DV expr=DV0*exp(-DVE/kt)
$ intrinsic point-defect concentration in equilibrium
intermed name=IiStar expr=IiS0*exp(-IiSE/kt)
intermed name=ViStar expr=ViS0*exp(-ViSE/kt)
$ point-defect charge fraction
intermed name=sphi expr=sum(i,-2,2,PhiI0[i]*exp(-PhiIE[i]/kt))
intermed name=sphv expr=sum(i,-2,2,phiV0[i]*exp(-PhiVE[i]/kt))
intermed name=phiI size=5 minindex=-2 default=0
intermed name=phiV size=5 minindex=-2 default=0
loop steps=5 index=i start=-2
  intermed name=phiI[@i] expr=phiI0[@i]*exp(-PhiIE[@i]/kt)/sphi
  intermed name=phiV[@i] expr=phiV0[@i]*exp(-PhiVE[@i]/kt)/sphv
l.end
$ equilibrium I and V
intermed name=FI expr=sum(i,-2,2,phiI[i]*ETA^(-i))
intermed name=FV expr=sum(i,-2,2,phiV[i]*ETA^(-i))
intermed name=IStar expr=IiStar*FI
intermed name=VStar expr=ViStar*FV
$ bulk recombination
intermed name=kbsi expr=KB0*exp(-KBE/kt)
intermed name=FIV expr=IiStar*ViStar
$ surface recombination
intermed name=ksurfi expr=KSI0*exp(-KSIE/kt)
intermed name=ksurfv expr=KSV0*exp(-KSVE/kt)
    
```

OED - Interstitial generation due to oxidation

Note that *x.vel* and *y.vel* are the reserved keywords for x- and y-directional boundary velocities respectively.

```
$ surface interstitial generation during oxidation
intermed name=vnorm   expr=VNORM0*exp(-VNORME/kt)
intermed name=gpow    expr=GPOW0*exp(-GPOWE/kt)
intermed name=theta   expr=THETA0*exp(-THETA0E/kt)
intermed name=vox     expr=sqrt(x.vel^2+y.vel^2)
intermed name=GSurf   expr=5E22*theta*vox*(vox/vnorm)^gpow
```

Thermal parameter for the dopant

```
$ dopant-defect pair diffusivity
intermed name=DiPairb expr=DIPAIR0*exp(-DIPAIRE/kt)
intermed name=DvPairb expr=DVPAIR0*exp(-DVPAIRE/kt)

$ measured diffusivity
intermed name=Db      size=2 minindex=0 +
                      expr="DX0*exp(-DXE/kt),DP0*exp(-DPE/kt)"
intermed name=Dib     size=5 minindex=-2 +
                      expr="0,0,FIB*Db[0],FIB*Db[1],0"
intermed name=Dvb     size=5 minindex=-2 +
                      expr="0,0,(1-FIB)*Db[0],(1-FIB)*Db[1],0"
intermed name=Dmb     expr=Dib[0]+Dib[1]/ETA
intermed name=Dnb     expr=Dvb[0]+Dvb[1]/ETA
intermed name=Amb     expr=Dmb/DiPairb
intermed name=Anb     expr=Dnb/DvPairb

$ Kick-out and Frank-Turnbull reaction
intermed name=gmb     expr=12.57*RIS*1e-8*exp(-EIS/kt)*DI*IiStar
intermed name=gnb     expr=12.57*RVS*1e-8*exp(-EVS/kt)*DV*ViStar
intermed name=rmvb    +
                      expr=12.57*RIPV*1e-8*exp(-EIPV/kt)*(DiPairb+DV)*ViStar
intermed name=rnib    +
                      expr=12.57*RVPI*1e-8*exp(-EVPI/kt)*(DvPairb+DI)*IiStar
intermed name=Kmb     expr=gmb*FI
intermed name=Knb     expr=gnb*FV
intermed name=Kmvb    +
                      expr=rmvb*sum(i,-2,2,Dib[i]/DiPairb*phiV[-i])
intermed name=Knib    +
                      expr=rnib*sum(i,-2,2,Dvb[i]/DvPairb*phiI[-i])

$ transport and segregation coefficients
intermed name=Trans   expr=TRANS0*exp(-TRANSE/kt)
intermed name=Seg     expr=SEG0*exp(-SEGE/kt)
```

Point defect solutions

```
impurity imp=Isi new
impurity imp=Vsi new
method   imp=Isi lu full ^symm time.ste rel.err=0.005 abs.err=1e5
method   imp=Vsi lu full ^symm time.ste rel.err=0.005 abs.err=1e5
```

The *Isi* and *Vsi* are the free interstitial and vacancy respectively.

Pair solutions

```
impurity imp=Mb new
impurity imp=Nb new
method   imp=Mb lu full ^symm time.ste rel.err=0.01 abs.err=1.0e5
method   imp=Nb lu full ^symm time.ste rel.err=0.01 abs.err=1.0e5
```

Substitutional boron

```
intermed name=Subb expr=active(boron)-Mb-Nb
```

Intermediates depending on point defects

```
$ I/I* and V/V*
intermed name=ciratio expr=Isi/Istar
intermed name=cvratio expr=Vsi/Vstar

$ Rb
intermed name=RIVB expr=kbsi*FIV*(ciratio*cvratio-1)

intermed name=IPb expr=(Kmb*ciratio+Kmbv)/(Kmb+Kmbv*cvratio)
intermed name=VPb expr=(Knbcvratio+Knib)/(Knbcvratio+Knib)
intermed name=Asumb expr=1+Amb*Ipb+Anb*Vpb
intermed name=AmStar expr=Amb/(1+Amb+Anb)
intermed name=AnStar expr=Anb/(1+Amb+Anb)
intermed name=CMovrM expr=Amb*Ipb/Asumb
intermed name=CNovrN expr=Anb*Vpb/Asumb
```

Kick-out and Frank-Turnbull reactions

```
$ generation/recombination
intermed name=MovrAb expr=Mb/Amb
intermed name=NovrAb expr=Nb/Anb
intermed name=GRmb expr=Kmb*(Subb*ciratio-MovrAb)
intermed name=GRmbv expr=Kmbv*(Subb-MovrAb*cvratio)
intermed name=GRnb expr=Knbcvratio*(Subb*cvratio-NovrAb)
intermed name=GRnib expr=Knib*(Subb-NovrAb*ciratio)
```

Boron diffusion equation

The **EQUATION** statements for implementation of the new boron equation replace the built-in boron diffusion equation by turning off the **DIFFUSE** parameter in the **IMPURITY** statement.

```
impurity imp=boron mat=silicon ^diffuse
equation model=fivestream variable=boron mat=silicon +
addtoexp=div(Dmb*(grad(MovrAb)+MovrAb*efield))
equation model=fivestream variable=boron mat=silicon +
addtoexp=div(Dnb*(grad(NovrAb)+NovrAb*efield))
```

Pair diffusion equations

```
equation model=fivestream variable=Mb mat=silicon +
  initial="exist(Mb)?Mb:1E5" +
  addtoexp=div(Dmb*(grad(MovrAb)+MovrAb*efield))+GRmb+GRmvb
equation model=fivestream variable=Nb mat=silicon +
  initial="exist(Nb)?Nb:1E5" +
  addtoexp=div(Dnb*(grad(NovrAb)+NovrAb*efield))+GRnb+GRnib
```

For the boundary condition, the dopant-defect pairs are annihilated at interfaces by turning off **TRANSPORT** since the dopant-defect pairs exist only in silicon.

```
$ equilibrium M* and N*
intermed name=MbStar   expr=AmStar*eqact(boron,boron)
intermed name=NbStar   expr=AnStar*eqact(boron,boron)
$ boundary equations
equation model=fivestream variable=Mb mat1=silicon /mat2=oxide +
  ^transpor interfac=-Trans*(onmat1(Mb/Seg-MbStar))
equation model=fivestream variable=Nb mat1=silicon /mat2=oxide +
  ^transpor interfac=-Trans*(onmat1(Nb/Seg-NbStar))
```

Point defect equations

The *interstitial* and *vacancy* variables mean the free interstitial and vacancy concentrations respectively. In the 3-stream diffusion models, **PD.TRANS** or **PD.FULL**, the dopant-defect pairs are instantly determined before the first post-implant anneal so that the total point-defect concentrations are conserved by subtracting the dopant-defect pairs and the interstitial small clusters from the total concentrations. When the *interstitial* or *vacancy* solution variables are used, they are not the total concentrations but the free concentrations. However, the *interstitial* or *vacancy* variables used in the **INITIAL** parameter mean the total concentrations.

```
equation model=fivestream variable=Isi mat=silicon +
  initial="damage>1.15e22? IStar: +
    (exist(Isi)?interstitial+Isi:interstitial)" +
  addtoexp=div(DI*IStar*grad(ciratio))-RIVB-GRmb+GRnib
equation model=fivestream variable=Vsi mat=silicon +
  initial="damage>1.15e22?VStar: +
    (exist(Vsi)?vacancy+Vsi:vacancy)" +
  addtoexp=div(DV*VStar*grad(cvratio))-RIVB-GRnb+GRmvb
```

The surface recombination is described as follows:

```
equation model=fivestream variable=Isi mat=silicon /mat=oxide +
  interfac=-ksurfi*onmat1(Isi-IStar) ^transpor
equation model=fivestream variable=Vsi mat=silicon /mat=oxide +
  interfac=-ksurfv*onmat1(Vsi-VStar) ^transpor
```

Since the initialization uses the built-in calculation of the point defects for implantation, *Isi* and *Vsi* in the structure must be added to the generated built-in point-defects due to implantation. After initializing and adding *Isi* and *Vsi*, the built-in point defect solutions are eliminated by setting **PD.FERMI**. For example,

```

method model=fivestream enable
$ turn on PD.TRANS in order to use the built-in initialization
method pd.trans
implant boron dose=1e14 energy=20

$ initialize Isi and Vsi
diffuse temp=900 time=0

$ remove the built-in point defect solutions
method pd.fermi

diffuse temp=900 time=10/60

$ reinitialize Isi and Vsi for the second implant
method pd.trans
implant boron dose=1e15 energy=10
diffuse temp=1000 time=0
method pd.fermi

diffuse temp=1000 time=5/60

```

Coupling with the small clusters

To see the certain effect of the 5-stream diffusion model, the non-equilibrium status for the formation of dopant-defect pairs must be dominant in the diffusion process. The diffusion process is usually performed at very low temperature and/or for very short time. The small clusters must be taken into account for such a process condition. In TSUPREM-4, the small clusters, I_2 and BI_2 , are assumed to form competitively with the boron-interstitial pair formation by assuming that I_2 , BI_2 and M_b (boron-interstitial pair) are in equilibrium. Thus, the initialization of the small clusters with the zero dopant-defect pair concentrations is not physical. Unless the transient equations are implemented for I_2 and BI_2 formation in non-equilibrium, it's not possible to account for the small clusters in the 5-stream diffusion model. It will be shown how the initialized profiles of I_2 , BI_2 , I (free interstitial), and M_b in the built-in codes are different from those in the 5-stream model using USEIT.

Interstitial small clusters - I_2

```

$ reaction for I small clusters
loop steps=5 index=i start=-2
  intermed name=kI[@i] expr=kI0[@i]*exp(-(kIE[@i])/kt)
l.end
$ I small clusters
intermed name=FcV +
  expr=ciratio*ViStar*sum(i,-2,2, +
    sum(j,-2,2,kcV[i][j]*kI[i]*phiV[j]*ETA^(-i-j)))
intermed name=Ic expr=2*ciratio^2*sum(i,-2,2,kI[i]/ETA^i)
intermed name=rc expr=RIVB/IiStar/ViStar*FcV
equation model=fivestream var=Isi mat=si addtoexp=-ddt(Ic)-rc
equation model=fivestream var=Vsi mat=si addtoexp=-rc

```

BI₂ and B₃I

```

$ BIC
impurity imp=BIC new
method   imp=BIC none abs.err=1e5
intermed name=Ca1  expr=boron-BIC
intermed name=Ca2  expr=Ca1/(1+ddcs*(Isi/IiStar)^2)
intermed name=Ca   expr=eqactive(boron,Ca1)
intermed name=dBIC +
           expr=(kddf*ni*(Ca/ni)^3*(ciratio/ETA)^2-BIC*ciratio)/tdd
intermed name=Idds expr=2*(Ca1-Ca2)
intermed name=Rdd  expr=1.356*dBIC
impurity model=BIC imp=boron  mat=si active=eqact(boron,Ca2)
equation model=BIC imp=BIC    mat=si addtoexp=dBIC +
           initial=exist(BIC)?BIC:1e5
equation model=BIC imp=Isi    mat=si addtoexp=-ddt(Idds)-Rdd

```

Modified initialization of interstitial

The initialization for the 5-stream diffusion model can be achieved by calling the function **SOLVE** for Newton-Raphson iteration.

```

equation model=fivestream variable=Isi mat=silicon +
initial=damage>1.15e22?IStar:solve(Isi,Isi+Idds+Ic-inters,inters)

```

Comparison of the initialization

[Figure 6-3](#) and [Figure 6-4](#) show the initialization results with the built-in **PD.FULL** codes and the USEIT 5-stream diffusion model, respectively. For the 3-stream diffusion model, the boron-interstitial pair concentration is calculated in equilibrium with the interstitial small clusters and the boron-interstitial small clusters. Due to the boron-interstitial pair formation by the local equilibrium assumption, the number of free interstitials is smaller than the 5-stream diffusion model, in which no initial dopant-defect pair exists. Solving the transient equations of a dopant-defect pair formation with the equilibrium assumption in the small cluster formation is not correct unless the small clusters form much faster than the dopant-defect pairing, which has not yet been proven. It might be a more reasonable approach to assume the formation of dopant-defect pairs in equilibrium, as well as I₂ and BI₂; in other words, to use the **PD.FULL** model rather than the 5-stream diffusion model unless the transient formation of the small clusters is implemented.

1

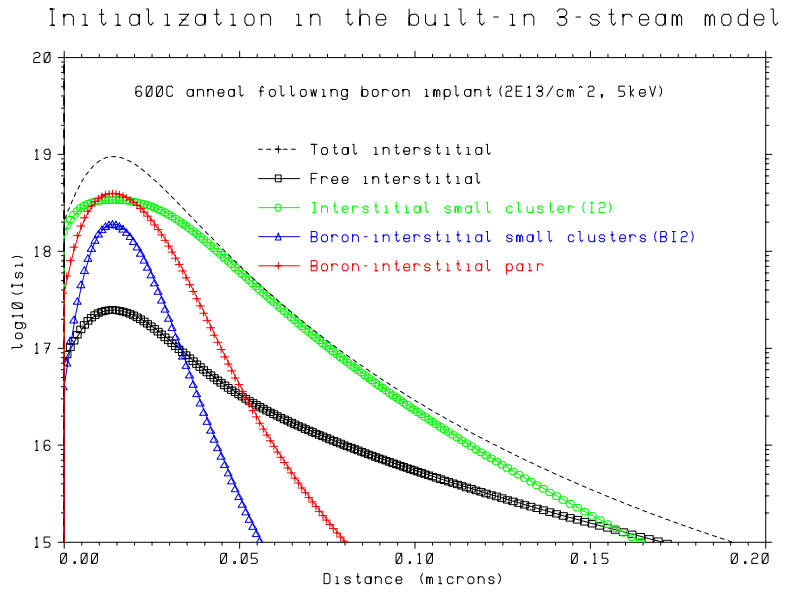


Figure 6-3 Initialization in the built-in 3-stream diffusion model

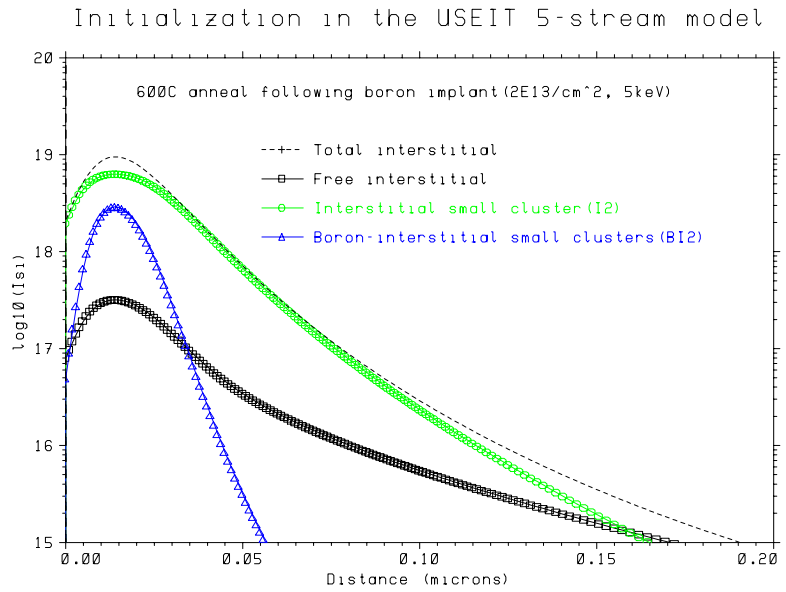


Figure 6-4 Initialization in the USEIT 5-stream diffusion model

SiGe Model

The model is based on Lever, Bonar and Willoughby's work ("Boron diffusion across silicon - silicon germanium boundaries", *J. Appl. Phys.* **83**, 1988 (1998)). In SiGe, the isovalent atoms like Ge can be as traps for boron atoms so that the number of mobile boron atoms is reduced. Trapping a boron atom forms an immobile complex Ge-B.

$$\frac{C_{GeB}}{C_a} = \frac{C_{as} - C_a}{C_a} = \frac{C_{Ge}}{C_0}$$

where C_0 is the characteristic concentration, $3.2 \times 10^{21} / \text{cm}^3$, and C_{Ge} is the germanium concentration. C_{as} is the active concentration when ignoring the boron trapping. The intrinsic carrier concentration increases with the Ge content because of the energy band-gap reduction.

$$n_i = n_{i, Si} \exp\left(\frac{\chi \delta E_g}{2kT}\right)$$

where χ is the fraction of Ge content, and δE_g is the experimental constant energy, 0.7eV. Assuming the equilibrium point defect concentrations, from the above equations, the boron diffusion flux can be calculated by:

$$\vec{J} = -D_B \left(\vec{\nabla} C_a + C_a \frac{\vec{\nabla} N_T}{Q} + C_a \frac{2n}{Q} \cdot \vec{\nabla} \ln(n_i) \right)$$

where N_T and Q mean the net doping concentration, $N_a - N_d$, and the total carrier concentration, $n + p$, respectively. The boron diffusivity D_B is given by:

$$D_B = \left(D_x + D_P \cdot \frac{p}{n_{i, Si}} \right) \cdot \exp\left(-\frac{\epsilon s \chi}{kT}\right)$$

where ϵ and s mean the energy per unit strain and the characteristic strain in the plane of the SiGe layer, respectively. Cowern *et.al.*'s work shows that the diffusion of B in compressively strained SiGe decreases due to an increase in the activation energy of B diffusion ("Diffusion in Strained SiGe", *Phys. Rev. Lett.* **72**, 2585 (1994)). They extracted 40eV/strain for ϵ and 0.042 for s . However, in this example, ϵ is set to zero so that the strain-dependence of the activation energy is ignored. Note that the intrinsic carrier concentration in the diffusivity model is acting just as the convenient normalization factor so that $n_{i, Si}$ must be used instead of n_i .

```

impurity imp=Germanium new

intermed name=GeMax      express=5e22
intermed name=StrainMax  express=0.042
intermed name=GeFrac     express=Germanium/GeMax
intermed name=StrainSiGe express=GeFrac*StrainMax

intermed name=EaStrain   express=0 ;$ instead of 40
intermed name=dEa        express=EaStrain*StrainSiGe
intermed name=DX0        expr=0.0368
intermed name=DBE        expr=3.46+dEa
intermed name=DP0        expr=0.716
intermed name=Db         size=2 minindex=0 +
                          expr="DX0*exp(-DBE/kt),DP0*exp(-DBE/kt)"

intermed name=dEg        express=0.7
intermed name=Ni         expr=3.87e16*exp(-0.605/kt+1.5*log(temp))
intermed name=NiSiGe     express=Ni*exp(GeFrac*dEg/(2*kt))
intermed name=C0         express=3.2e21
intermed name=fTrap      express=1/(1.0+Germanium/C0)

intermed name=Na         express=active(boron)
intermed name=Nd         express=active(arsenic)+active(phosphorus)
intermed name=Nt         express=fTrap*(Na-Nd)
intermed name=Q          express=sqrt(Nt^2+4*NiSiGe^2)
intermed name=nelectron  express=0.5*(Q-Nt)
intermed name=nhole      express=0.5*(Q+Nt)
intermed name=DbSiGe     express=Db[0]+Db[1]*nhole/Ni
intermed name=Bmobile    express=fTrap*active(boron)

equation model=sige variable=boron mat=si +
      addtoexp="div(DbSiGe*(grad(Bmobile)+ +
      Bmobile/Q*(grad(Nt)+2*nelectron*grad(log(NiSiGe)))))"
impurity imp=boron mat=si ^diffuse

```


Appendix A: Default Coefficients

Default Coefficient Values

This appendix contains the values of the various simulation coefficients used in TSUPREM-4, along with the references from which the values were obtained. In some cases there are also notes regarding the approximations that were made when the necessary values had not been experimentally determined, or when there was wide scatter in the available data. These notes regarding the approximations involved in the coefficient determination should help you know how much latitude is available when modifying these coefficients to match fabrication results. The units for the coefficients are given in [Chapter 3](#); where a choice of units is available, the default units (i.e., microns and minutes) are used, except as noted.

| Tables | Page |
|--|----------------------|
| Impurity Parameters | A-2 |
| Impurity Diffusion Coefficients | A-2 |
| Segregation and Transport Coefficients | A-4 |
| Polysilicon Grain Segregation | A-6 |
| Clustering and Solid Solubility | A-7 |
| Point Defect Parameters | A-8 |
| Oxidation | A-10 |
| Diffusion Enhancement in Oxides | A-15 |
| Silicidation | A-15 |
| Analytical Implant | A-16 |
| Monte Carlo Implant | A-17 |
| Electrical Parameters | A-18 |
| Material Coefficients | A-21 |
| Numerical Methods | A-24 |
| Automatic Grid Generation Parameters | A-25 |
| Adaptive Grid Parameters | A-25 |

The notes referenced in this section are described in “Default Coefficient Notes” on Page A-32. They appear in the form “(1),” in which the digit is the number of the note in the list of notes.

The references listed in this section are described in “Default Coefficient References” on page A-27. They appear in the form “[1],” in which the digit is the number of the reference in the list of references.

Impurity Parameters

Table A-1 Impurity Parameters (Silicon Dopants)

| | Boron | Phosphorus | Arsenic | Indium | Antimony |
|----------------------|----------|------------|----------|-----------|----------|
| TIF.NAME | B | P | As | In | Sb |
| Type | ACCEPTOR | DONOR | DONOR | ACCEPTOR | DONOR |
| Implant table | chboron | dual.pho | dual.ars | tr.indium | antimony |
| AT.NUM | 5.0 | 15.0 | 33.0 | 49.0 | 51.0 |
| AT.WT | 10.8 | 30.97 | 74.91 | 115 | 121.76 |

Table A-2 Impurity Parameters (Other)

| | Aluminum | Fluorine | Nitrogen | Silicon |
|-----------------|----------|----------|----------|---------|
| TIF.NAME | Al | F | N | Si |
| AT.NUM | 13.0 | 9.0 | 7.0 | |
| AT.WT | 26.98 | 18.9984 | 14.01 | |

Impurity Diffusion Coefficients

Diffusivities are in $\mu\text{m}^2/\text{min}$ except as noted.

Table A-3 Impurity Diffusion Coefficients

| | Boron | Phosphorus | Arsenic | Indium | Antimony | Fluorine | Nitrogen |
|--------------------|------------|------------|---------|-------------------------|----------|----------------------------|--------------------------|
| Silicon (1) | | | | | | | |
| DIP.0 | 4.10e9 [1] | 0 [1] | 0 [2] | 0 (2) | 0 [1] | 0 | 0 |
| DIP.E | 3.46 | 0 | 0 | 0 | 0 | 0 | 0 |
| DIX.0 | 2.11e8 | 2.31e10 | 1.37e7 | 1.8cm ² /sec | 6.420e7 | 4.23e3cm ² /sec | 0.05cm ² /sec |
| DIX.E | 3.46 | 3.66 | 3.44 | 3.6 | 3.65 | 3.5 | 3.65 |
| DIM.0 | 0 | 2.664e10 | 3.72e10 | 0 | 4.50e9 | 0 | 0 |
| DIM.E | 0 | 4.0 | 4.15 | 0 | 4.08 | 0 | 0 |
| DIMM.0 | 0 | 2.652e11 | 0 | 0 | 0 | 0 | 0 |
| DIMM.E | 0 | 4.37 | 0 | 0 | 0 | 0 | 0 |
| DVP.0 | 2.16e8 | 0 | 0 | 0 | 0 | 0 | 0 |
| DVP.E | 3.46 | 0 | 0 | 0 | 0 | 0 | 0 |
| DVX.0 | 1.11e7 | 0 | 5.47e7 | 0 | 1.220e9 | 0 | 0 |
| DVX.E | 3.46 | 3.66 | 3.44 | 0 | 3.65 | 0 | 0 |
| DVM.0 | 0 | 0 | 1.49e11 | 0 | 8.55e10 | 0 | 0 |
| DVM.E | 0 | 4.0 | 4.15 | 0 | 4.08 | 0 | 0 |
| DVMM.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Table A-3 Impurity Diffusion Coefficients

| | Boron | Phosphorus | Arsenic | Indium | Antimony | Fluorine | Nitrogen |
|---|---------|------------|---------|--------|----------|----------|----------|
| DVMM.E | 0 | 4.37 | 0 | 0 | 0 | 0 | 0 |
| Oxide [8] (3) | | | | | | | |
| DIX.0 | 1.896e6 | 4.338e10 | 1.05e10 | 18. | 7.86e25 | 2.858 | 0.05 |
| DIX.E | 3.53 | 4.44 | 4.89 | 3.6 | 8.75 | 3.5 | 3.65 |
| Polysilicon (grain interior) (diffusivities in cm²/sec) (4) | | | | | | | |
| DIP.0 | 0.72 | 0 | 0 | 0 | 0 | 0 | 0 |
| DIP.E | 3.46 | 0 | 0 | 0 | 0 | 0 | 0 |
| DIX.0 | 0.037 | 3.85 | 1.14e-2 | 1.8 | 0.214 | 1.70e5 | 0.05 |
| DIX.E | 3.46 | 3.66 | 3.44 | 3.6 | 3.65 | 3.5 | 3.65 |
| DIM.0 | 0 | 4.44 | 31.0 | 0 | 15.0 | 0 | 0 |
| DIM.E | 0 | 4.0 | 4.15 | 0 | 4.08 | 0 | 0 |
| DIMM.0 | 0 | 44.2 | 7.17e3 | 0 | 0 | 0 | 0 |
| DIMM.E | 0 | 4.37 | 5.21 | 0 | 0 | 0 | 0 |
| Polysilicon (grain boundaries) (diffusivities in cm²/sec) (5) | | | | | | | |
| DIX.0 | 420. | 6.0e3 | 110.0 | 0 | 2.14e3 | 0 | 0 |
| DIX.E | 3.46 | 3.66 | 3.53 | 0 | 3.65 | 0 | 0 |
| DIM.0 | 0 | 0 | 10.0 | 0 | 0 | 0 | 0 |
| DIM.E | 0 | 0 | 3.53 | 0 | 0 | 0 | 0 |
| TiSi₂ (diffusivities in cm²/sec) | | | | | | | |
| DIX.0 | 6.0e-7 | 4.6e-6 | 4.0e-7 | 0 | 7.0e-8 | 0 | 0 |
| DIX.E | 2.0 | 2.0 | 1.8 | 0 | 1.8 | 0 | 0 |
| WSi₂ (diffusivities in cm²/sec) (6) [9]-[12] | | | | | | | |
| DIX.0 | 1.0e-5 | 4.20e-2 | 2.60e-2 | 0 | 2.60e-2 | 0 | 0 |
| DIX.E | 1.17 | 2.14 | 2.11 | 0 | 2.11 | 0 | 0 |
| Ambient (7) | | | | | | | |
| DIX.0 | 1e13 | 1e13 | 5e13 | 180. | 5e13 | 1e13 | 5.0 |
| DIX.E | 3.5 | 3.5 | 3.95 | 3.6 | 3.95 | 3.5 | 3.65 |

Table A-4 Other Impurities and Materials

| | |
|--------|---|
| DIP.0 | 0 |
| DIP.E | 0 |
| DIX.0 | 0 |
| DIX.E | 0 |
| DIM.0 | 0 |
| DIM.E | 0 |
| DIMM.0 | 0 |
| DIMM.E | 0 |

Table A-5 Pair Kinetics Parameters (all impurities; CM. SEC units) (8)

| | |
|----------|---------|
| DIPAIR.0 | 3.65e-4 |
| DIPAIR.E | 1.58 |
| DVPAIR.0 | 3.65e-4 |
| DVPAIR.E | 1.58 |

Table A-5 Pair Kinetics Parameters (all impurities; CM . SEC units) (8)

| | |
|--------|----|
| R.I.S | 10 |
| E.I.S | 0 |
| R.V.S | 10 |
| E.V.S | 0 |
| R.IP.V | 10 |
| E.IP.V | 0 |
| R.VP.I | 10 |
| E.VP.I | 0 |

Segregation and Transport Coefficients

Table A-6 Segregation Coefficients

| | Boron | Phosphorus | Arsenic | Indium | Antimony | Fluorine | Nitrogen |
|--|--------------|---------------|-----------|-------------|-----------|----------|----------|
| Silicon/oxide | | | | | | | |
| SEG.O | 1.126e3 [13] | 30.0 [14] (9) | 30.0 [14] | 1.3e-13[15] | 30.0 [14] | 5.62E-8 | 0.14 |
| SEG.E | 0.91 | 0.00 | 0.0 | -2.55 | 0.0 | 0.0 | 0.0 |
| Polysilicon/oxide (10) | | | | | | | |
| SEG.O | 1.126e3 | 30.00 | 30.0 | 1.3e-13 | 30.0 | 5.62E-8 | 0.14 |
| SEG.E | 0.91 | 0.0 | 0.0 | -2.55 | 0.0 | 0.0 | 0.0 |
| Silicon/TiSi₂ | | | | | | | |
| SEG.O | 0.3 | 1.0 | 0.8 | 1.0 | 0.8 | 1.0 | 1.0 |
| SEG.E | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Polysilicon/TiSi₂ | | | | | | | |
| SEG.O | 0.3 | 1.0 | 0.8 | 1.0 | 0.8 | 1.0 | 1.0 |
| SEG.E | 0 | 0.0 | 0 | 0 | 0 | 0 | 0 |
| Silicon/WSi₂ [9]-[12] | | | | | | | |
| SEG.O | 1.0 | 5.0 | 10.0 | 1.0 | 10.0 | 1.0 | 1.0 |
| SEG.E | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Polysilicon/WSi₂ | | | | | | | |
| SEG.O | 1.0 | 5.0 | 10.0 | 1.0 | 10.0 | 1.0 | 1.0 |
| SEG.E | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Other Impurities and Pairs of Materials | | | | | | | |
| SEG.O | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| SEG.E | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Table A-7 Interface Transport Coefficients

| | Boron | Phosphorus | Arsenic | Indium(11) | Antimony |
|----------------------------------|---------|------------|---------|-------------|----------|
| Silicon/ambient [16] (12) | | | | | |
| TRANS.O | 1.674e7 | 9.0e5 | 9.0e5 | 1.8e8cm/sec | 1.5e3 |
| TRANS.E | 2.481 | 1.99 | 1.99 | 3.6 | 1.04 |
| Oxide/ambient (13) | | | | | |
| TRANS.O | 1.647e7 | 9.0e5 | 9.0e5 | 18e8cm/sec | 1.5e3 |
| TRANS.E | 2.481 | 1.99 | 1.99 | 3.6 | 1.04 |
| Polysilicon/ambient (13) | | | | | |

Table A-7 Interface Transport Coefficients

| | Boron | Phosphorus | Arsenic | Indium(11) | Antimony |
|--|--------------|-------------------|----------------|-------------------|-----------------|
| TRANS . 0 | 1.674e7 | 9.0e5 | 9.0e5 | 1.8e8cm/sec | 1.5e3 |
| TRANS . E | 2.481 | 1.99 | 1.99 | 3.6 | 1.04 |
| Polysilicon/silicon (14) | | | | | |
| TRANS . 0 | 100. | 100. | 100. | 1.8e8cm/sec | 100. |
| TRANS . E | 0 | 0 | 0 | 3.6 | 0 |
| Silicon/oxide and Polysilicon/oxide | | | | | |
| TRANS . 0 | 0.1 | 0.1 | 0.1 | 1.8e8cm/sec | 0.1 |
| TRANS . E | 0 | 0 | 0 | 3.6 | 0 |
| All materials with Nitride and Oxynitride | | | | | |
| TRANS . 0 | 0 | 0 | 0 | 0 | 0 |
| TRANS . E | 0 | 0 | 0 | 3.6 | 0 |
| Silicon/TiSi₂ | | | | | |
| TRANS . 0 | 1.0e6 | 1.0e6 | 1.0e6 | 0 | 1.0e6 |
| TRANS . E | 2.0 | 2.0 | 2.0 | 0 | 2.0 |
| Polysilicon/TiSi₂ | | | | | |
| TRANS . 0 | 1.0e6 | 1.0e6 | 1.0e6 | 0 | 1.0e6 |
| TRANS . E | 2.0 | 2.0 | 2.0 | 0 | 2.0 |
| Silicon/WSi₂ [9]-[12] | | | | | |
| TRANS . 0 | 2.0e-2 | 3.0e-3 | 3.0e-3 | 0 | 3.0e-3 |
| TRANS . E | 0 | 0 | 0 | 0 | 0 |
| Polysilicon/WSi₂ | | | | | |
| TRANS . 0 | 2.0e-2 | 3.0e-3 | 3.0e-3 | 0 | 3.0e-3 |
| TRANS . E | 0 | 0 | 0 | 0 | 0 |

Table A-8 Interface Trap Impurities (CM.SEC units)

| | Boron(15) | Phosphorus(16) | Arsenic(17) | Nitrogen |
|---|--------------------------------------|----------------|-------------|----------|
| Oxide/Silicon, and Oxide/Polysilicon | | | | |
| TRANS.0 | 0.166 | 7.15 | 0.231 | 4.65E-10 |
| TRANS.E | 0.486 | 1.75 | 0.766 | 0.0 |
| SEG.0 | /SEG.0 ÷ SEG.0 (Table A-5) | | | 0.0 |
| SEG.E | /SEG.E - SEG.E (Table A-5) | | | 0.0 |
| RATIO.0 | 0 | 0 | 0 | 0.0 |
| RATIO.E | 0 | 0 | 0 | 0.0 |
| /TRANS.0 | 0.0166 | 0.715 | 0.0231[18] | 0.0 |
| /TRANS.E | 0.486 | 1.75 | 0.766[18] | 0.0 |
| /SEG.0 | /RATIO.0 ÷ Q.MAX.0 × C _{ss} | | 5.96E3 | 0.0 |
| /SEG.E | /RATIO.E - Q.MAX.E | | -0.285 | 0.0 |
| /RATIO.0 | 0.178 | 4.00E-3 | 6.29E-5 | 0.0 |
| /RATIO.E | -0.086 | -0.37 | -0.738 | 0.0 |
| Q.INI.0 | 0 | 0 | 0 | 0.0 |
| Q.INI.E | 0 | 0 | 0 | 0.0 |
| Q.MAX.0 | 2.0E14[18] | 6.8E14[17] | 2.0E14[18] | 2.0E14 |
| Q.MAX.E | 0 | 0 | 0 | 0.0 |
| TWO.PHAS | false | false | false | true |

Polysilicon Grain Segregation

Table A-9 Parameters for Polysilicon Grain Interior/Boundary Segregation (18)

| | Boron | Phosphorus | Arsenic | Indium | Antimony |
|---|--------|------------|---------|--------|----------|
| Q.SITES | 2.5e15 | 2.5e15 | 2.5e15 | 0 | 2.5e15 |
| CG.MAX | 5.0e22 | 5.0e22 | 5.0e22 | 0 | 5.0e22 |
| GSEG.0 | 12.0 | 1.25 | 0.6 | 0 | 1.0 |
| GSEG.E | 0 | 0.443 | 0.414 | 0 | 0 |
| GSEG.INI | 1.0 | 1.0 | 1.0 | 0 | 1.0 |
| VELIF.0 | 1.0e7 | 1.0e7 | 1.0e7 | 0 | 1.0e7 |
| VELIF.E | 3.0 | 3.0 | 3.0 | 0 | 3.0 |
| Polysilicon diffusivity enhancement factors (19) | | | | | |
| FGB | 2.64 | 95.1 | 110. | 100. | 41.8 |

Clustering and Solid Solubility

Table A-10 Coefficients for Clustering in Silicon (20)

| | Arsenic |
|-------|-----------------|
| CTN.O | 1.03103e-17 [1] |
| CTN.E | -0.4 (21) |
| CTN.F | 4.0 |

Table A-11 Solid Solubility in Silicon

| Temperature (°C) | Boron | Phosphorus | Antimony |
|------------------|--------------|--------------|--------------|
| 650 | 1.70e19 [19] | 1.20e20 [20] | 1.70e19 [20] |
| 700 | 1.70e19 | 1.20e20 | 1.70e19 |
| 800 | 4.40e19 | 2.90e20 | 2.30e19 |
| 900 | 9.50e19 | 6.00e20 | 3.10e19 |
| 1000 | 1.70e20 | 1.00e21 | 4.00e19 |
| 1100 | 2.20e20 | 1.20e21 | 4.90e19 |
| 1200 | 2.20e20 | 1.25e21 | 5.90e19 |
| 1300 | 1.40e20 | 1.10e21 | 6.80e19 |
| 1350 | 1.40e20 | 1.10e21 | 6.80e19 |

Table A-12 Transient activation parameters (Extended Defects AAM)

| | Boron | Phosphorus | Arsenic | Antimony |
|----------|---------|------------|---------|----------|
| T.ACT.O | 8.0e-16 | 8.0e-16 | 8.0e-16 | 8.0e-16 |
| T.ACT.E | -4.2 | -4.2 | -4.2 | -4.2 |
| ACT.MIN | 1.0 | 2.0 | 1.0 | 1.0 |
| CL.INI.A | true | true | true | true |

Table A-13 Dopant-defect clustering model (ACT.FULL)

| | All Impurities |
|----------|----------------|
| DDC.T.O | 0 |
| DDC.T.E | 0 |
| DDC.F.O | 0 |
| DDC.F.E | 0 |
| DDCF.N.N | 0 |
| DDCF.I.N | 0 |
| DDCF.D.N | 0 |
| DDCR.N.N | 0 |
| DDCR.I.N | 0 |
| IFRACM | 0 |
| DDCS.O | 0 |
| DDCS.E | 0 |
| DDCS.N | 0 (22) |
| IFRACS | 0 |

Point Defect Parameters

Table A-14 Point Defect Parameters in Silicon (CM.SEC units) [21]

| | Vacancy | Interstitial |
|----------|---------|--------------|
| D.0 | 3.65e-4 | 3.65e-4 |
| D.E | 1.58 | 1.58 |
| DC.0 | 1.0 | 1.0 |
| DC.E | 0 | 0 |
| KB.0 | (23) | 1.0e-21 |
| KB.E | | -1.0 (24) |
| KIV.0 | | (25) |
| KIV.E | | 0 |
| KIV.NORM | | true |
| CEQUIL.0 | 1.25e29 | 1.25e29 |
| CEQUIL.E | 3.26 | 3.26 |
| VMOLE | 0.0 | 5.0e22 |
| DNEG.0 | 32.47 | 0 |
| DNEG.E | 0.62 | 0 |
| NEG.0 | 5.68 | 5.68 |
| NEG.E | 0.145 | 0.50 |
| NEU.0 | 1.0 | 1.0 |
| NEU.E | 0 | 0 |
| POS.0 | 5.68 | 5.68 |
| POS.E | 0.455 | 0.26 |
| DPOS.0 | 0 | 0 |
| DPOS.E | 0 | 0 |

Table A-15 Vacancy Parameters at Interfaces With Silicon (CM.SEC units)

| | Oxide [21] | Nitride | Oxynitride | Ambient | TiSi ₂ /WSi ₂ |
|---------|------------|---------|------------|---------|-------------------------------------|
| KSURF.0 | 4.0e-11 | 3.5e-3 | 0.1 | 2.5e-4 | 4.0e-10 |
| KSURF.E | -1.75 | 0.0 | 0.0 | 0.0 | -1.75 |
| KSVEL.0 | 4.0e-11 | 3.5e-3 | 0.1 | 2.5e-4 | 4.0e-10 |
| KSVEL.E | -1.75 | 0.0 | 0.0 | 0.0 | -1.75 |
| GROWTH | false | false | false | false | false |
| A.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| A.E | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| T0.0 | 100.02 | 100.02 | 100.02 | 100.02 | 100.02 |
| T0.E | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| GPOW.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| GPOW.E | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| KPOW.0 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 |
| KPOW.E | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| THETA.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| THETA.E | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

Table A-16 Interstitial Parameters at Interfaces With Silicon (CM.SEC units)

| | Oxide [22] (26) | Nitride | Oxynitride | Ambient | TiSi ₂ /WSi ₂ |
|---------|-----------------|---------|------------|---------|-------------------------------------|
| KSURF.O | 1.4e-6 | 3.5e-3 | 0.1 | 2.5e-4 | 4.0e-10 |
| KSURF.E | -1.75 | 0.0 | 0.0 | 0.0 | -1.75 |
| KSVEL.O | 0.0 | 3.5e-3 | 0.1 | 2.5e-4 | 0.0 |
| KSVEL.E | -1.75 | 0.0 | 0.0 | 0.0 | -1.75 |
| GROWTH | true | false | false | false | false |
| A.O | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| A.E | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| T0.O | 0.0 | 100.02 | 100.02 | 100.02 | 100.02 |
| T0.E | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| GPOW.O | -0.7 | 0.0 | 0.0 | 0.0 | 0.0 |
| GPOW.E | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| KPOW.O | 0.0 | 0.5 | 0.5 | 0.5 | 0.5 |
| KPOW.E | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| THETA.O | 0.01 | 0.0 | 0.0 | 0.0 | 0.0 |
| THETA.E | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

Table A-17 Point Defect Parameters at Silicon/Polysilicon Interfaces (CM.SEC units) (27)

| | Interstitial | Vacancy |
|---------|--------------|---------|
| KSURF.O | 2.73 | 6.4e-5 |
| KSURF.E | 1.37 | 0.92 |

Table A-18 Interstitial Traps Parameters

| | |
|----------|-----|
| TRAP.CON | 0.0 |
| F.TRAP.O | 0.0 |
| F.TRAP.E | 0.0 |
| K.TRAP.O | 0.0 |
| K.TRAP.E | 0.0 |

Table A-19 Interface Injection/Recombination Model

| | Model | V.NORM.O | V.NORM.E |
|---------------|---------|----------|----------|
| Vacancies | V.MAXOX | | |
| Interstitials | V.NORM | 1.0 | 0.0 |

Table A-20 Dislocation Loop Coefficients

| | |
|---------|--------------|
| KLOOP.O | 29.8047 [24] |
| KLOOP.E | 0.4 [25] |
| RLMIN | 100e-8 |

Table A-21 Interstitial Clustering ({311} Model) Coefficients (28)

| | |
|----------|----------|
| CL.KFC.O | 5.207e14 |
| CL.KFC.E | 3.774 |
| CL.CF | 0.9398 |
| CL.IFC | 1 |
| CL.ISFC | 1 |
| CL.KFI.O | 0 |
| CL.KFI.E | 3.774 |
| CL.KFCI | 1.0 |
| CL.IFI | 2 |
| CL.ISFI | 2 |
| CL.KR.O | 9.431e13 |
| CL.KR.E | 3.017 |
| CL.CR | 1 |

Table A-22 Small Point Defect Clusters

| | Interstitial | Vacancy |
|----------|--------------|---------|
| ECLUST.O | 0 | 0 |
| ECLUST.E | 0 | 0 |
| ECLUST.N | 0 | 0 |
| KCV.O | 0 | |
| KCV.E | 0 | |
| KIC.O | | 0 |
| KIC.E | | 0 |

Oxidation

Table A-23 Ambient Definitions

| | DRYO2 | WETO2 | N2O | STREAM | INERT |
|----------|-------|----------------|-----|--------|-------|
| F.O2 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| F.H2O | 0.0 | 0.92 [26] (30) | 0.0 | 1.0 | 0.0 |
| F.N2O | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| F.H2 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| F.N2 | 0.0 | 0.08 | 0.0 | 0.0 | 1.0 |
| F.HCL | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| PRESSURE | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| HCL (%) | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

Table A-24 Linear Oxidation Rate Coefficients

| | O2 | H2O | N2O |
|---------------------|---------------------|---------------------|--------|
| L.LIN.O | | | |
| <111> | 1.038e5 [26] | 3.450e4 [27] | 8.06 |
| <110> | 8.650e4 (30) | 2.870e4 (30) | 8.22 |
| <100> | 6.176e4 | 2.058e4 | 5.87 |
| POLYSILI | 8.650e4 (31) | 2.870e4 (31) | 8.22 |
| L.LIN.E | | | |
| <111> | 2.0 | 1.60 | 1.09 |
| <110> | 2.0 | 1.60 | 1.09 |
| <100> | 2.0 | 1.60 | 1.09 |
| POLYSILI | 2.0 | 1.60 | 1.09 |
| H.LIN.O | | | |
| <111> | 1.038e5 | 2.950e6 | 8.06 |
| <110> | 8.650e4 (30) | 2.457e6 (30) | 8.22 |
| <100> | 6.176e4 | 1.755e6 | 5.87 |
| POLYSILI | 8.650e4 (31) | 2.457e6 (31) | 8.22 |
| H.LIN.E | | | |
| <111> | 2.0 | 2.05 | 1.09 |
| <110> | 2.0 | 2.05 | 1.09 |
| <100> | 2.0 | 2.05 | 1.09 |
| POLYSILI | 2.0 | 2.05 | 1.09 |
| LIN.BREA | 0.0 | 900.0 | 0.0 |
| LIN.PDEP | 0.75 | 1.0 | 2.48 |
| GAMMA.O [28] | 2.63e3 | 2.63e3 | 1.59e3 |
| GAMMA.E | 1.1 | 1.1 | 1.3 |

Table A-25 Parabolic Oxidation Rate Coefficients

| | O2 | H2O | N2O |
|-----------------|------------------------|-------------------------|------|
| L.PAR.O | 12.87 [26] (30) | 2.83e2 [27] (30) | 4.73 |
| L.PAR.E | 1.23 | 1.17 | 1.42 |
| H.PAR.O | 12.87 (30) | 7.00 (30) | 4.73 |
| H.PAR.E | 1.23 | 0.78 | 1.42 |
| PAR.BREA | 0.0 | 950.0 | 0.0 |
| PAR.PDEP | 1.0 | 1.0 | 1.0 |

Table A-26 Thin Oxide Growth Coefficients [29]

| | O2 | H2O | N2O |
|-----------------|--------------------|-----|--------|
| THINOX.O | | | |
| <111> | 6.58e6 | 0.0 | 2.39e2 |
| <110> | 5.32e4 | 0.0 | 5.37e4 |
| <100> | 7.48e6 | 0.0 | 5.37e4 |
| POLYSILI | 5.32e4 (31) | 0.0 | 5.37e4 |
| THINOX.E | | | |
| <111> | 2.33 | 0.0 | 1.09 |
| <110> | 1.80 | 0.0 | 1.85 |
| <100> | 2.38 | 0.0 | 1.85 |
| POLYSILI | 1.80 | 0.0 | 1.85 |
| THINOX.L | | | |
| <111> | 7.8e-3 | 1.0 | 1.2e-3 |
| <110> | 6.0e-3 | 1.0 | 6.5e-3 |
| <100> | 6.9e-3 | 1.0 | 6.5e-3 |
| POLYSILI | 6.0e-3 | 1.0 | 6.5e-3 |

Table A-27 Oxidation Rate for Chlorine Dependence

| For O2 [30] and NO2 | | | | | | | | | | | |
|----------------------------|------|------|------|------|------|------------------|------|------|------|------|------|
| Linear | | | | | | Parabolic | | | | | |
| Temperature: | 800 | 900 | 1000 | 1100 | 1200 | | 800 | 900 | 1000 | 1100 | 1200 |
| %HCl | | | | | | | | | | | |
| 0.0 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| 1.0 | 1.76 | 1.76 | 1.26 | 1.63 | 1.63 | | 1.09 | 1.09 | 1.67 | 1.37 | 1.37 |
| 3.0 | 1.79 | 1.79 | 1.52 | 2.26 | 2.26 | | 1.29 | 1.29 | 1.90 | 1.54 | 1.54 |
| 5.0 | 1.82 | 1.82 | 1.54 | 2.29 | 2.29 | | 1.52 | 1.52 | 2.18 | 1.73 | 1.73 |
| 7.0 | 1.85 | 1.85 | 1.57 | 2.33 | 2.33 | | 1.76 | 1.76 | 2.51 | 1.95 | 1.95 |
| 10.0 | 1.89 | 1.89 | 1.61 | 2.39 | 2.39 | | 2.25 | 2.25 | 3.07 | 2.34 | 2.34 |
| For H2O [26] | | | | | | | | | | | |
| Linear | | | | | | Parabolic | | | | | |
| Temperature: | 800 | 900 | 1000 | 1100 | 1200 | | 800 | 900 | 1000 | 1100 | 1200 |
| %HCl | | | | | | | | | | | |
| 0.0 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| 1.0 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| 3.0 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| 5.0 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| 7.0 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| 10.0 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |

Table A-28 Coefficients for Analytical Oxidation Models

| | |
|----------------------|---|
| INITIAL | 0.002 |
| MASK.EDG | -200.0 |
| SPREAD | 1.0 |
| NIT.THIC | (no default) |
| ERF.Q [31] | 0.05 |
| ERF.DELT [31] | 0.04 |
| ERF.LBB [31] | $8.25e-3 * (1580.3 - TOX) * (FOX^{0.67}) * (EOX^{0.3}) * \exp(-((EN - 0.08)^2) / 0.06)$ |
| ERF.H [31] | $402 * (0.445 - 1.75 * EN) * \exp(-TOX / 200)$ |

Table A-29 Coefficients for Numerical Oxidation Models

| | O2 | H2O | N2O |
|---------------------|-----------|------------|------------|
| HENRY.CO | 5.0e16 | 3.0e19 | 5.0e16 |
| TRANS.O (32) | 1.0e3 | 1.0e6 | 1.0e3 |
| THETA (33) | 2.2e22 | 2.2e22 | 2.2e22 |

Table A-30 Diffusivity of Oxidizing Species in Ambient (7)

| | O2 | H2O | N2O |
|------------|-----------|------------|------------|
| D.O | 3.5e9 | 2.5e7 | 3.5e9 |
| D.E | 1.25 | 0.8 | 1.25 |

Table A-31 Nitrogen-dependent Oxide Growth Coefficients (cm.sec)

| | O2 | H2O | N2O |
|---|-----|-----|---------|
| Oxide/Silicon, Oxide/Polysilicon | | | |
| N.G.O | 0.0 | 0.0 | 5.2e22 |
| N.G.E | 0.0 | 0.0 | 0.0 |
| N.V.O | 0.0 | 0.0 | 1.0 |
| N.V.E | 0.0 | 0.0 | 0.0 |
| N.GPOW.O | 0.0 | 0.0 | 0.0 |
| N.GPOW.E | 0.0 | 0.0 | 0.0 |
| N.THIN.O | | | |
| <111> | 0.0 | 0.0 | 1.55e14 |
| <110> | 0.0 | 0.0 | 1.55e14 |
| <100> | 0.0 | 0.0 | 1.55e14 |
| N.THIN.E | | | |
| <111> | 0.0 | 0.0 | 0.0 |
| <110> | 0.0 | 0.0 | 0.0 |
| <100> | 0.0 | 0.0 | 0.0 |

Table A-32 Parameters for Stress-Dependent Oxidation

| | |
|-----------------|--|
| STRESS.D | true |
| VC | 300.0 (oxide), 130.0 (nitride) (34) |
| VR | 15 [42] |
| VD | 75.0 [42] |
| VT | 0.0 |
| VDLIM | 1.2 |

Table A-33 Material Conversion Coefficient

| | ALPHA |
|---------------------------------|------------------|
| Silicon/oxide | 0.44 [49] |
| Polysilicon/oxide | 0.44 (38) |
| Other pairs of materials | 1.0 |

Diffusion Enhancement in Oxides

Table A-34 Diffusion Enhancement Factor of Boron in Oxides

$$1+4.55e-22*\exp(0.28/kt)*sio+2.23e-20*fluorine$$

Table A-35 Diffusivity of Silicon in Oxides (CM.SEC units)

| | |
|-----|------|
| D.0 | 13.0 |
| D.E | 4.5 |

Table A-36 Si and SiO Reactions in Oxides

| | | | | |
|----------|---------|---------|---------|---------|
| MAT.BULK | oxide | oxide | | |
| /MAT.L | | | oxide | oxide |
| /NM.L | | | 0 | 0 |
| MAT.R | | | silicon | ambient |
| NM.R | | | 0 | 0 |
| IMP.L | iox | sio | | |
| NI.L | 1.0 | 1.0 | | |
| EI.L | 1.0 | 1.0 | | |
| /IMP.L | O2 | O2 | iox | iox |
| /NI.L | 0.0 | 0.0 | 1.0 | 1.0 |
| /EI.L | 0.5 | 0.5 | 1.0 | 1.0 |
| IMP.R | sio | | interst | O2 |
| NI.R | 1.0 | | 1.0 | 0 |
| EI.R | 1.0 | | 1.0 | 1.0 |
| RATE.0 | 7.673e6 | 7.673e6 | 0.01 | 8.67e7 |
| RATE.E | 4.0 | 4.0 | 0.0 | 4.0 |
| EQUIL.0 | 0 | 0 | 2.95e-5 | 0 |
| EQUIL.E | 0 | 0 | -2.19 | 0 |

Silicidation

Table A-37 Diffusivity of Silicon in TiSi₂ and WSi₂ (CM.SEC units)

| | | |
|-----|-----------------------------------|---------------------------------------|
| | TiSi₂ (35) [32] | WSi₂ (36) [33]-[35] |
| D.0 | 2.0 | 6.86e3 |
| D.E | 1.86 | 2.72 |

Table A-38 Titanium Silicidation Reactions

| | | | |
|---------|---------|-------------|----------|
| /MAT.L | TiSi2 | TiSi2 | titanium |
| /NM.L | 0 | 0 | 0.5 |
| MAT.R | silicon | polysilicon | TiSi2 |
| NM.R | 1.0 | 1.0 | 0.5 |
| IMP.L | vacancy | | silicon |
| NI.L | 1.0e-3 | | 1.0 |
| EI.L | 0 | | 1.0 |
| /IMP.L | silicon | silicon | |
| /NI.L | 1.0 | 1.0 | |
| /EI.L | 1.0 | 1.0 | |
| RATE.0 | 1.0e-3 | 1.0e-3 | 104.0 |
| RATE.E | 0 | 0 | 1.0 |
| EQUIL.0 | 1.0e20 | 1.0e20 | 1.0 |
| EQUIL.E | 0 | 0 | 0 |

Table A-39 Tungsten Silicidation Reactions [33]-[35]

| | | | |
|---------|---------|-------------|--------------------|
| /MAT.L | WSi2 | WSi2 | tungsten |
| /NM.L | 0 | 0 | 0.5 |
| MAT.R | silicon | polysilicon | WSi2 |
| NM.R | 1.0 | 1.0 | 0.5 |
| IMP.L | vacancy | | silicon |
| NI.L | 1.0e-3 | | 1.0 |
| EI.L | 0 | | 1.0 |
| /IMP.L | silicon | silicon | |
| /NI.L | 1.0 | 1.0 | |
| /EI.L | 1.0 | 1.0 | |
| RATE.0 | 1.0e-1 | 1.0e-1 | 4.60e3 (37) |
| RATE.E | 0 | 0 | 1.64 (37) |
| EQUIL.0 | 1.0e20 | 1.0e20 | 1.0 |
| EQUIL.E | 0 | 0 | 0 |

Analytical Implant

Table A-40 Effective +n Model (in Silicon)

| | |
|----------|--------|
| D.PHDF | 0.0905 |
| D.PME | 0.85 |
| D.PLF | -2 |
| D.PLME | -0.5 |
| D.PCONST | 3.25 |
| D.PDME | -0.5 |
| D.PDLF | -2 |
| D.PDLME | -0.5 |
| D.PNORM | 3e12 |
| D.PDE | 1 |
| D.PRE | 0 |

Monte Carlo Implant

Table A-41 Electronic Stopping Power Pre-factor

| | Boron | Phosphorus | Arsenic | Antimony | Aluminum | Other |
|-------------------------|-------------|------------|---------|----------|----------|-------|
| ES.RAND (43) | | | | | | |
| Silicon | 2.0799 [45] | 2.5 | 1.7941 | 1.7456 | 0 | 0 |
| Polysilicon | 2.0799 | 1.7516 | 1.7941 | 1.7456 | 0 | 0 |
| Oxide | 1.8424 | 1.4609 | 1.4347 | 1.3674 | 0 | 0 |
| Oxynitride | 1.8424 | 1.4609 | 1.4347 | 1.3674 | 0 | 0 |
| Nitride | 1.8424 | 1.4609 | 1.4347 | 1.3674 | 0 | 0 |
| Aluminum | 2.0281 | 1.6857 | 1.7104 | 1.6564 | 0 | 0 |
| Photoresist | 1.0032 | 0.64016 | 0.5517 | 0.4957 | 0 | 0 |
| SiC | 1.1588 | 0 | 0 | 0 | 1.4341 | 0 |
| Other (44) | | 0 | 0 | 0 | | 0 |
| ES.100 [46]-[48] | | | | | | |
| Silicon | 1.500 | 1.7516 | 1.7941 | 1.7456 | 0 | 0 |
| Other | 0 | 0 | 0 | 0 | 0 | 0 |
| ES.110 | | | | | | |
| Silicon | 0.95 | 1.7516 | 1.7941 | 1.7456 | 0 | |
| Other | 0 | 0 | 0 | 0 | 0 | 0 |

Table A-42 Electronic Stopping Power Exponent

| | Boron | Phosphorus | Arsenic | Antimony | Aluminum | Other |
|----------------------|-------|------------|---------|----------|----------|-------|
| ES.F.RAN | | | | | | |
| Silicon | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 |
| Other | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 |
| ES.BREAK (45) | | | | | | |
| Silicon | 400 | 150 | 5000 | 0 | 0 | 0 |
| Polysilicon | 400 | 150 | 5000 | 0 | 0 | 0 |
| Oxide | 400 | 150 | 5000 | 0 | 0 | 0 |
| Oxynitride | 400 | 150 | 5000 | 0 | 0 | 0 |
| Nitride | 400 | 150 | 5000 | 0 | 0 | 0 |
| Aluminum | 400 | 150 | 5000 | 0 | 0 | 0 |
| Photoresist | 400 | 150 | 5000 | 0 | 0 | 0 |
| SiC | 600 | 0 | 0 | 0 | 1000 | 0 |
| Other | 0 | 0 | 0 | 0 | 0 | 0 |
| ES.F.H | | | | | | |
| Silicon | 0.1 | 0.4 | 1.0 | 0.5 | 0.5 | 0.5 |
| Polysilicon | 0.1 | 0.4 | 1.0 | 0.5 | 0.5 | 0.5 |
| Oxide | 0.1 | 0.4 | 1.0 | 0.5 | 0.5 | 0.5 |
| Oxynitride | 0.1 | 0.4 | 1.0 | 0.5 | 0.5 | 0.5 |
| Nitride | 0.1 | 0.4 | 1.0 | 0.5 | 0.5 | 0.5 |
| Aluminum | 0.1 | 0.4 | 1.0 | 0.5 | 0.5 | 0.5 |
| Photoresist | 0.1 | 0.4 | 1.0 | 0.5 | 0.5 | 0.5 |
| SiC | 0.1 | 0.5 | 0.5 | 0.5 | 0.3 | 0.5 |
| Other | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 |

Table A-42 Electronic Stopping Power Exponent

| | Boron | Phosphorus | Arsenic | Antimony | Aluminum | Other |
|-----------------|-------|------------|---------|----------|----------|-------|
| ES.F.100 | | | | | | |
| Silicon | 0.5 | 0.5 | 0.5 | 0.5 | | 0.5 |
| Other | 0.5 | 0.5 | 0.5 | 0.5 | | 0.5 |
| ES.F.110 | | | | | | |
| Silicon | 0.5 | 0.5 | 0.5 | 0.5 | | 0.5 |
| Other | 0.5 | 0.5 | 0.5 | 0.5 | | 0.5 |

Table A-43 Improved Electronic Stopping Power Model (Silicon)

| | Boron | Phosphorus | Arsenic | Antimony |
|----------|-------|------------|---------|----------|
| NLOC.PRE | 0.3 | 0.3 | 0.3 | 0.3 |
| NLOC.EXP | 0.2 | 0.2 | 0.2 | 0.2 |
| NLOC.MAX | 0.6 | 0.6 | 0.6 | 0.6 |
| NLOC.K | 1.2 | 1.2 | 1.2 | 1.2 |
| LOC.FAC | 0.8 | 0.8 | 0.8 | 0.8 |
| CHAN.CRI | 11.54 | 11.54 | 11.54 | 11.54 |
| CHAN.FAC | 2.0 | 2.0 | 2.0 | 2.0 |
| DISP.FAC | 0.5 | 0.5 | 0.5 | 0.5 |

Table A-44 Improved Electronic Stopping Power Model (SiC)

| | Boron | Phosphorus | Aluminum | Nitrogen |
|----------|-------|------------|----------|----------|
| NLOC.PRE | 1.0 | 0.8 | 0.825 | 0.9 |
| NLOC.EXP | 0 | 0 | 0 | 0 |
| NLOC.MAX | 1.0 | 0.6 | 1.0 | 0.6 |
| NLOC.K | 1.25 | 1.2 | 1.05 | 1.2 |
| LOC.FAC | 0.8 | 0.8 | 0.8 | 0.8 |
| CHAN.CRI | 11.54 | 11.54 | 11.54 | 11.54 |
| CHAN.FAC | 2.0 | 2.0 | 2.0 | 2.0 |
| DISP.FAC | 0.5 | 0.5 | 0.5 | 0.5 |

Electrical Parameters

Table A-45 Critical Electric Field (volts/cm) [36] [37]

| | |
|--------|--------|
| ECN.MU | 6.49e4 |
| ECP.MU | 1.87e4 |

Table A-46 Surface Mobility Degradation Factor

| | |
|--------|-----|
| GSURFN | 1.0 |
| GSURFP | 1.0 |

Table A-47 Arora Mobility Model [38]

| | |
|------|---------|
| MUN1 | 88.0 |
| MUN2 | 1252.0 |
| AN | 0.88 |
| CN | 1.26e17 |
| EXN1 | -0.57 |
| EXN2 | -2.33 |
| EXN3 | 2.4 |
| EXN4 | -0.146 |
| MUP1 | 54.3 |
| MUP2 | 407.0 |
| AP | 0.88 |
| CP | 2.35e17 |
| EXP1 | -0.57 |
| EXP2 | -2.23 |
| EXP3 | 2.4 |
| EXP4 | -0.146 |

Table A-48 Caughey Mobility Model [39]

| | |
|-----------|----------|
| MUN . MIN | 55.24 |
| MUN . MAX | 1429.23 |
| NREFN | 1.072e17 |
| NUN | -2.3 |
| XIN | -3.8 |
| ALPHAN | 0.733 |
| MUP . MIN | 49.705 |
| MUP . MAX | 479.37 |
| NREFP | 1.606e17 |
| NUP | -2.2 |
| XIP | -3.7 |
| ALPHAP | 0.70 |

Table A-49 Mobility Table

| Concentration (#/cm ³) | Mobility at 300K | |
|---------------------------------------|------------------|-------|
| | Electron | Hole |
| 1e14 | 1350 | 495 |
| 2e14 | 1345 | 495 |
| 4e14 | 1335 | 495 |
| 6e14 | 1320 | 495 |
| 8e14 | 1310 | 495 |
| 1e15 | 1300 | 491.1 |
| 2e15 | 1248 | 487.3 |
| 4e15 | 1200 | 480.1 |
| 6e15 | 1156 | 473.3 |
| 8e15 | 1115 | 466.9 |
| 1e16 | 1076 | 460.9 |
| 2e16 | 960 | 434.8 |
| 4e16 | 845 | 396.5 |
| 6e16 | 760 | 369.2 |
| 8e16 | 720 | 348.3 |
| 1e17 | 675 | 331.5 |
| 2e17 | 524 | 279.0 |
| 4e17 | 385 | 229.8 |
| 6e17 | 321 | 203.8 |
| 8e17 | 279 | 186.9 |
| 1e18 | 252 | 178.0 |
| 2e18 | 182.5 | 130.0 |
| 4e18 | 140.6 | 90.0 |
| 6e18 | 113.6 | 74.5 |
| 8e18 | 99.5 | 66.6 |
| 1e19 | 90.5 | 61.0 |
| 2e19 | 86.9 | 55.0 |
| 4e19 | 83.4 | 53.7 |
| 6e19 | 78.8 | 52.9 |
| 8e19 | 71.6 | 52.4 |
| 1e20 | 67.8 | 52.0 |
| 2e20 | 52.0 | 50.8 |
| 4e20 | 35.5 | 49.6 |
| 6e20 | 23.6 | 48.9 |
| 8e20 | 19.0 | 48.4 |
| 1e21 | 17.8 | 48.0 |

Material Coefficients

Table A-50 Semiconductor Material Coefficients

| | Silicon | Polysilicon (38) | SiC (39) |
|----------|--------------|------------------|----------|
| DENSITY | 2.33 | 2.33 | 3.21 |
| AT.NUM | 14.0 | 14.0 | 10.0 |
| AT.WT | 28.069 | 28.069 | 20.04 |
| MOL.WT | 28.069 | 28.069 | 40.08 |
| POLYCRYS | false | true | |
| NI.0 | 3.87e16 [40] | 3.87e16 | |
| NI.E | 0.605 | 0.605 | |
| NI.F | 1.5 | 1.5 | |
| YOUNG.M | 1.87e12 [42] | 1.87e12 [42] | |
| POISS.R | 0.28 | 0.28 | |
| VISC.0 | 1.0e30 | 1.0e16 | |
| VISC.E | 0.0 | 0.0 | |
| VISC.X | 0.499 | 0.499 | |
| EPSILON | 11.7 [42] | 11.7 | |
| E.FIELD | true | true | |
| AFFINITY | 4.17 | 4.17 | |
| BANDGAP | 1.08 | 1.08 | |
| EGALPH | 4.73e-4 | 4.73e-4 | |
| EGBETA | 636 | 636 | |
| N.CONDUC | 2.82e19 | 2.82e19 | |
| N.VALENC | 1.04e19 | 1.04e19 | |
| NC.F | 1.5 | 1.5 | |
| NV.F | 1.5 | 1.5 | |
| G.DONOR | 2.0 | 2.0 | |
| E.DONOR | 0.049 | 0.049 | |
| G.ACCEP | 4.0 | 4.0 | |
| E.ACCEP | 0.045 | 0.045 | |
| BOLTZMAN | false | false | |
| IONIZATI | true | true | |
| QM.BETA | 4.1e-8 | 4.1e-8 | |
| QM.YCRIT | 25e-8 | 25e-8 | |
| TIF.NAME | Si | Poly | SiC |
| MD.INDEX | 1 | 3 | 8 |
| ION.PAIR | true | true | |
| IP.OMEGA | 6.0 | 6.0 | |
| CL.MODEL | true | | |
| MAX.DAMA | 1.15e22 | | |
| DAM.GRAD | 10 | | |
| D.PCONST | 0.42 | | |

Table A-51 Polycrystalline Material Coefficients (40)

| | Polysilicon |
|----------|-------------|
| F11 | 1.0 |
| F22 | 2.0 |
| G.DENS | 2.0 |
| GRASZ.0 | 0.1 |
| GRASZ.E | 0 |
| TEMP.BRE | 600. |
| MIN.GRAI | 0.005 |
| FRAC.TA | 0.5 |
| GEOM | 6.0 |
| GAMMA.0 | 5.6e-6 |
| GAMMA.E | -1.73 |
| GBE.0 | 0.39125 |
| GBE.H | 3.0 |
| GBE.1 | 0 |
| ALPHA | 1.33 |
| DSIP.0 | 30.0 |
| DSIP.E | 5.09 |
| DSIX.0 | 0.015 |
| DSIX.E | 3.89 |
| DSIM.0 | 16.0 |
| DSIM.E | 4.54 |
| DSIMM.0 | 166.7 |
| DSIMM.E | 5.0 |
| NSEG | 2.0 |
| TBU.0 | 1.0 |
| TBU.E | -5.0 |
| TOXIDE | 0.0005 |
| DLGX.0 | 0 |
| DLGX.E | 4.4 |
| EAVEL.0 | 0 |
| EAVEL.E | 3.0 |

Table A-52 Insulator Material Coefficients

| | Oxide | Nitride | Oxynitride | Ambient |
|--------------|-------------|------------|------------|------------|
| DENSITY | 2.235 | 3.44 | 2.235 | |
| AT.NUM (avg) | 10.0 | 10.0 | 10.0 | |
| AT.WT (avg) | 20.023 | 20.030 | 20.023 | |
| MOL.WT | 60.069 | 140.21 | 60.069 | |
| POLYCRYS | false | false | false | |
| YOUNG.M | 6.6e11 [42] | 3.89e12 | 3.89e12 | |
| POISS.R | 0.17 [42] | 0.3 | 0.3 | |
| VISC.0 | 5.25e4 (41) | 1.3e6 (41) | 1.3e6 (41) | 1.0e9 (42) |
| VISC.E | -2.42 (41) | -2.42 (41) | -2.42 (41) | 0.0 (42) |
| VISC.X | 0.499 | 0.499 | 0.499 | 0.3 (42) |
| EPSILON | 3.9 [43] | 7.5 [44] | 7.5 | 1.0 |
| TIF.NAME | Ox | Nit | OxyNit | Ambient |
| MD.INDEX | -1 | -2 | -6 | -4 |
| IMPL.TAB | oxide | nitride | oxide | |

Table A-53 Other Material Coefficients

| | Aluminum | Photoresist | Titanium | TiSi ₂ | Tungsten | WSi ₂ |
|----------|----------|-------------|----------|-------------------|----------|------------------|
| DY.DEFAU | | | | 0.025 | | 0.02 |
| DENSITY | 2.702 | 1.06 | 4.5 | 4.0443 | 19.3 | 9.857 |
| AT.NUM | 13.0 | 2.875 | 22.0 | 16.67 | 74.0 | 34.0 |
| AT.WT | 26.98 | 5.125 | 47.90 | 34.68 | 183.85 | 80.0 |
| MOL.WT | 26.98 | 41.0 | 47.90 | 104.038 | 183.85 | 240.0 |
| POLYCRYS | false | false | false | false | | |
| YOUNG.M | | | 1.0e13 | 1.0e13 | 3.66e12 | 1.0e11 |
| POISS.R | | | 0.3 | 0.3 | 0.29592 | 0.29592 |
| EPSILON | 1.0 | 1.0 | | | | |
| WORKFUNC | 4.1 | | 4.80 | 4.8 | 4.80 | 4.80 |
| TIF.NAME | Al | Photo | Ti | TiSi2 | W | WSi2 |
| MD.INDEX | -5 | -4 | -5 | -5 | -5 | -5 |
| IMPL.TAB | aluminum | az-7500 | titanium | tisi2 | tungsten | wsic2 |

Table A-54 Linear Coefficients of Thermal Expansion

| | |
|-----------------|--------------------------------------|
| Silicon | $3.052e-6 + 2*6.206e-10*(T - 293)$ |
| Oxide | $1.206e-7 + 2*2.543e-10*(T - 293)$ |
| Nitride | 3.0e-6 |
| Aluminum | $2.438e-5 + 2*6.660e-9*(T - 293)$ |
| Other materials | 0.0 |

Numerical Methods

Table A-55 Block Solution Parameters

| | Impurities | Point Defects | Oxidant | Traps |
|--------------------|-------------------|----------------------|-----------------|-----------------|
| REL.ERR | 0.01 | 0.01 | 0.01 | 0.01 |
| ABS.ERR | 1.0e9 | 1.0e5 | 1.0e9 | 1.0e9 |
| Factoring | TIME.STE | TIME.STE | TIME.STE | TIME.STE |
| SYMMETRY | false | false | false | false |
| Fill | PART | FULL | PART | NONE |
| Block solve | LU | LU | LU | LU |

Table A-56 System Solution Parameters

| | |
|---------------------|--------------|
| MIN.FILL | true |
| MIN.FREQ | 1.1 |
| MF.METH | 1 |
| MF.DIST | 0.5 |
| System solve | CG |
| BACK | 18 |
| BLK.ITLI | 20 |
| Integration | TRBDF |
| INIT.TIM | 0.002 |

Table A-57 Models

| | |
|---------------------------|-----------------|
| Oxidation | VERTICAL |
| Point defect | PD.FERMI |
| Bulk recombination | KB.LOW |
| SKIP.SIL | true |
| OX.REL | 1.0e-6 |
| CONTIN.M | 2 |
| VE.SMOOT | 0.04 |
| V.COMPAT | current version |

Table A-58 Program Options

| | |
|-----------------|---------------|
| INFORMAT | false |
| DIAGNOST | false |
| ECHO | true |
| EXECUTE | true |
| Output | NORMAL |

Automatic Grid Generation Parameters

Table A-59 Automatic Grid Generation Parameters

| | | | |
|----------|-------|----------|-------|
| DX.MIN | 0.2 | | |
| DX.MAX | 0.5 | | |
| DX.RATIO | 1.5 | | |
| DY.SURF | 0.05 | LY.SURF | 0.1 |
| DY.ACTIV | 0.5 | LY.ACTIV | 4.0 |
| DY.BOT | 100.0 | LY.BOT | 200.0 |
| DY.RATIO | 1.5 | | |

Table A-60 Global Grid Spacing Factors

| | |
|----------|------|
| GRID.FAC | 1.0 |
| DY.OXIDE | 0.1 |
| DY.EXACT | true |
| DY.LOCAL | true |

Adaptive Grid Parameters

Table A-61 Dopants

| For Antimony, Arsenic, Boron, Indium, and Phosphorus | | | | |
|--|---------|-------------|--------|------------|
| | Silicon | Polysilicon | Oxide | Other (46) |
| REL.ADAP | 0.2 | 0.3 | 2.0 | 0.0 |
| ABS.ADAP | 1.0e14 | 1.0e15 | 1.0e16 | 0.0 |
| MIN.SPAC | 0.025 | 0.03 | 0.1 | 0.0 |

Table A-62 Silicon in TiSi₂ and WSi₂

| | |
|----------|--------|
| REL.ADAP | 0.1 |
| ABS.ADAP | 1.0e15 |
| MIN.SPAC | 0.01 |

Table A-63 O₂ and H₂O in Oxide

| | O ₂ | H ₂ O |
|----------|----------------|------------------|
| REL.ADAP | 0.2 | 0.2 |
| ABS.ADAP | 1.0e14 | 1.0e17 |
| MIN.SPAC | 0.01 | 0.01 |

Table A-64 Unrefinement Parameters

| | Silicon | Other (46) |
|-----------------|----------------|-------------------|
| MAX.SPAC | 0.25 | 0.0 |

Table A-65 Control Parameters

| | |
|-----------------|------|
| IMP.ADAP | true |
| DIF.ADAP | true |
| OX.ADAPT | true |
| ERR.FAC | 1.0 |
| UNREFINE | 0.25 |

Table A-66 Grid Parameters for Electrical Extraction

| | |
|-----------------|--------|
| E.TSURF | 0.005 |
| E.RSURF | 1.2 |
| E.DSURF | 0.0002 |
| E.REGRID | true |

Default Coefficient References

- [1] Fair, R. B., *Impurity Doping Processes in Silicon*, ed. F. F. Y. Wang, Amsterdam: North Holland, 1981.
- [2] *TMA Newsletter*, June 7, 1984.
- [3] C. S. Fuller and J. A. Ditzenberger, "Diffusion of donor and acceptor elements in silicon," *J. Appl. Phys.*, Vol. 27, pp. 544–53, 1956.
- [4] R. N. Ghoshtagore, "Dopant diffusion in silicon. III. Acceptors," *Phys Rev. B*, Vol. 3, pp. 2507–14, 1971.
- [5] T. Skotnicki, L. Guerin, D. Mathiot, M. Gauneau, A. Grouillet, C. D'Anter-roches, E. Andre, P. Bouillon, and M. Haond, "Channel engineering by heavy ion implants," *ESSDERC Tech. Dig.*, pp. 671–4, 1994.
- [6] K. Suzuki, H. Tashiro, T. Aoyama, "Diffusion coefficient of indium in Si substrates and analytical redistribution profile model," *Solid-State Elec-tron.*, Vol. 43, pp. 27–31, 1999.
- [7] R. D. Chang, P. S. Choi, D. L. Kwong, D. Y. C. Lie, and G. Wilk, "Diffu-sion modeling of implanted indium durign rapid thermal annealing," 1999 Semiconductor Technology CAD Workshop & Exhibition, May 4–6, 1999, Hsinchu, Taiwan, p. 16-1.
- [8] M. Ghezzeo and D. M. Brown. "Diffusivity Summary of B, Ga, P, As, and Sb in SiO₂," *J. Electrochem. Soc.*, Vol. 120, No. 1, pp. 146-148, Jan. 1973.
- [9] C. L. Chu, K. C. Saraswat, S. S. Wong. "Measurement of Lateral Dopant Diffusion in Thin Silicide Layers," *IEEE Trans. Elec. Dev.*, Vol. 39, No. 10, pp. 2333-2340, Oct. 1992.
- [10] V. Probst, H. Schaber, A. Mitwalsky, and H. Kabza. "WSi₂ and CoSi₂ as Diffusion Sources for Shallow-Junction Formation in Silicon," *J. Appl. Phys.*, Vol. 70, No. 2, pp. 708-719, July 1991.
- [11] M. Y. Tsai, F. M. d'Heurle, C. S. Petersson, and R. W. Johnson. "Properties of Tungsten Silicide Film on Polycrystalline Silicon," *J. Appl. Phys.*, Vol. 52, No. 8, pp. 5350-5355, Aug. 1981.
- [12] G. Giroult, A. Nouailhat, and M. Gauneau. "Study of a WSi₂/Polycrystal-line Silicon/Monocrystalline Silicon Structure for a Complementary Metal-Oxide-Semiconductor for a Compatible Self-Aligned Bipolar Transistor Emitter," *J. Appl. Phys.*, Vol. 67, No. 1, pp. 515-523, Jan. 1990.
- [13] D. A. Antoniadis, A. G. Gonzalez, and R. W. Dutton. "Boron in Near-Intrinsic <100> and <111> Silicon under Inert and Oxidizing Ambients—Diffusion and Segregation," *J. Electrochem. Soc.*, Vol. 125, No. 5, pp. 813-819, May 1978.
- [14] H. F. Wolf. *Semiconductors*, Interscience, p. 361, 1971.

- [15] I. C. Kizilyalli, T. L. Rich, F. A. Stevie, and C. S. Rafferty, "Diffusion parameters of indium for silicon process modeling," *J. Appl. Phys.*, Vol. 80, no. 9, pp. 4944-7, 1 Nov 1996.
- [16] P. H. Langer and J. I. Goldstein. "Boron Autodoping During Silane Epitaxy," *J. Electrochem. Soc.*, Vol. 124, No. 4, pp. 591-598, April 1977.
- [17] F. Lau, L. Mader, C. Mazure, C. Werner, and M. Orlowski. "A Model for Phosphorus Segregation at the Silicon-Silicon Dioxide Interface," *Appl. Phys. A.*, 49, pp. 671-675, 1989.
- [18] H. Vuong, C. Rafferty, S. Eshraghi, J. Lentz, P. Zeitzoff, M. Pinto, and S. Hillenius, "Effects of Oxide Interface Traps and Transient Enhanced Diffusion on the Process Modeling of PMOS Devices," *IEEE Trans. Electron Devices*, Vol. ED-43, No. 7, pp. 1144-1152, July 1996.
- [19] G. L. Vick and K. M. Whittle. "Solid Solubility and Diffusion Coefficients of Boron in Silicon," *J. Electrochem. Soc.*, Vol. 116, No. 8, pp. 1142-1144, Aug. 1969.
- [20] F. A. Trumbore. "Solid Solubilities of Impurity Elements in Germanium and Silicon," *Bell System Tech. J.*, Vol. 39, pp. 205-233, Jan. 1960.
- [21] M. R. Kump. "Two-Dimensional Computer Simulation of Diffusion in Silicon," Doctoral Dissertation, Stanford University, Stanford, California, March 1988.
- [22] S. T. Dunham. *J. Appl. Phys.*, Vol. 71, No. 1, 1992.
- [23] J. M. Poate, D. J. Eaglesham, G. H. Gilmer, H. J. Gossmann, M. Jaraiz, C. S. Rafferty, and P. A. Stolk. "Ion Implantation and Transient Enhanced Diffusion," *1995 IEDM Tech. Dig.*, pp. 77-80, 1995.
- [24] R. Y. S. Huang and R. W. Dutton. "Experimental Investigation and Modeling of the Role of Extended Defects During Thermal Oxidation," *J. Appl. Phys.*, Vol. 74, No. 9, pp. 5821-5827, Nov. 1993.
- [25] S. M. Hu. *Defects in Semiconductors*, ed. J. Narayan and T. Y. Tan, Amsterdam: North-Holland, pp. 333-354, 1981.
- [26] Deal, B. E., "Thermal Oxidation Kinetics of Silicon in Pyrogenic H₂O and 5% HCl/H₂O Mixtures," *J. Electrochem. Soc.*, Vol. 125, No. 4, pp. 576-579, Apr 1978.
- [27] R. R. Razouk, L. N. Lie, and B. E. Deal. "Kinetics of High Pressure Oxidation of Silicon in Pyrogenic Steam," *J. Electrochem. Soc.*, Vol. 128, No. 10, pp. 2214-2220, Oct. 1981.
- [28] C. P. Ho and J. D. Plummer. "Si/SiO₂ Interface Oxidation Kinetics: A Physical Model for the Influence of High Substrate Doping Levels," *J. Electrochem. Soc.*, Vol. 126, No. 9, pp. 1516-1522, Sept. 1979.
- [29] H. Z. Massoud. "Thermal Oxidation of Silicon in Dry Oxygen—Growth Kinetics and Charge Characterization in the Thin Regime," Technical

- Report, Stanford Electronics Laboratories, Stanford University, Stanford, California, June 1983.
- [30] D. W. Hess and B. E. Deal. "Kinetics of the Thermal Oxidation of Silicon in O₂/HCl Mixtures," *J. Electrochem. Soc.*, Vol. 124, No. 5, pp. 735-739.
- [31] N. Guillemot, G. Pananakakis, and P. Chenevier. "A New Analytical Model of the 'Bird's Beak'," *IEEE Trans. Electron Devices*, Vol. ED-34, No. 7, pp. 744-749, July 1983.
- [32] L. S. Hung, *et al.* *J. Appl. Phys.*, Vol. 54 p. 5076, 1983.
- [33] S. L. Zhang, R. Buchta, and M. Ostling. "A Study of Silicide Formation from LPCVD-Tungsten Films: Film Texture and Growth Kinetics," *J. Mater. Res.*, Vol. 6, pp. 1886-1891, Sept. 1991.
- [34] L. D. Locker and C. D. Capio. "Reaction Kinetics of Tungsten Thin Films on Silicon (100) Surfaces," *J. Appl. Phys.*, Vol. 44, No. 10, pp. 4366-4369, Oct. 1973.
- [35] E. Ma, B. S. Lim, M. A. Nicolet, N. S. Alvi, and A. H. Hamdi. *J. Electron. Mater.*, Vol. 17, p. 207, 1988.
- [36] K. Yamaguchi. "Field-dependent Mobility Model for Two-Dimensional Numerical Analysis of MOSFETs," *IEEE Trans. Electron Devices*, Vol. ED-26, pp. 1068-1074, July 1979.
- [37] K. Yamaguchi. "A Mobility Model for Carriers in the MOS Inversion Layer," *IEEE Trans. Electron Devices*, Vol. ED-30, pp. 658-663, June 1983.
- [38] N. D. Arora, J. R. Hauser, and D. J. Roulston. "Electron and Hole Mobilities in Silicon as a Function of Concentration and Temperature," *IEEE Trans. Electron Devices*, Vol. ED-29, pp. 292-295, Feb. 1982.
- [39] D. M. Caughey and R. E. Thomas. "Carrier Mobilities in Silicon Empirically Related to Doping and Field," *Proc. IEEE*, Vol. 55, pp. 2192-2193, 1967.
- [40] F. J. Morin and J. P. Maita. "Electrical Properties of Silicon Containing Arsenic and Boron," *Phys. Rev.*, Vol. 96, No. 1, pp. 28-35, Oct. 1954.
- [41] A. S. Grove. *Physics and Technology of Semiconductor Devices*, New York: John Wiley and Sons, pp. 102-103, 1967.
- [42] V. Senez, P. Ferreira, and B. Baccus. "Two-Dimensional Simulation of Local Oxidation of Silicon: Calibrated Viscoelastic Flow Analysis," *IEEE Trans. Electron Dev.*, Vol. 43, No. 5, pp. 720-731, May 1996.
- [43] *Handbook of Chemistry and Physics*, 53rd Edition, Cleveland: The Chemical Rubber Company, 1972.
- [44] R. S. Muller, and T. I. Kamins. *Device Electronics for Integrated Circuits*, New York: John Wiley and Sons, 1977.

- [45] J. P. Biersack and L. G. Haggmark. "A Monte Carlo Computer Program for the Transport of Energetic Ions in Amorphous Targets," *Nucl. Instr. and Meth.*, Vol. 174, p. 257, 1980.
- [46] R. G. Wilson. "Random and Channeled Implantation Profiles and Range Parameters for P and Al in Crystalline and Amorphized Si," *J. Appl. Phys.*, Vol. 59, p. 2797-2805, Oct. 1986.
- [47] J. Comas and R. G. Wilson. "Channeled and Random Equivalent Depth Distributions of 150keV Li, Be, and B Implanted in Si," *J. Appl. Phys.*, Vol. 51, No. 7, pp. 3697-3701, July 1980.
- [48] R. G. Wilson. "Channeling of 20-800 keV Arsenic Ions in the <110> and the <100> Directions of Silicon, and the Roles of Electronic and Nuclear Stopping," *J. Appl. Phys.*, Vol. 52, No. 6, pp. 3985-3988, June 1981.
- [49] E. Bassous, H. N. Yu, and V. Maniscalco. "Topology of Silicon Structure with Recessed SiO₂," *J. Electrochem. Soc.*, Vol. 123, No. 11, pp. 1729-1737, Nov. 1976.
- [50] T. I. Kamins and E. L. MacKenna. "Thermal Oxidation of Polycrystalline Silicon Films," *Metallurgical Transactions*, Vol. 2, pp. 2292-2294, Aug. 1971.
- [51] P. Fahey, G. Barbuscia, M. Moslehi, and R. W. Dutton, "Kinetics of Thermal Nitridation Processes in the Study of Dopant Diffusion Mechanisms in Silicon," *Appl Phys. Lett.*, Vol. 46, No. 8, p. 784, April 1985.
- [52] T. E. Seidel, et al. "Rapid Thermal Annealing in Si," *Mat. Res. Symp. Proc.*, Vol. 35, p. 329, 1985.
- [53] A. D. Buonaquisti, W. Carter, and P. H. Holloway. *Thin Solid Films*, Switzerland, Vol. 100, p. 235, 1983.
- [54] T. Kamins, et al. *J. Appl. Phys.*, Vol. 43, p. 83, 1972.
- [55] K. Sakamoto, K. Nishi, T. Yamaji, T. Miyoshi, and S. Ushio. *J. Electrochem. Soc.*, Vol. 132, p. 2457, 1985.
- [56] W. Rausch, R. Lever, and R. Kastl. *J. Appl. Phys.*, Vol. 54, No. 8, p. 4405, 1983.
- [57] H. Schaber, R. Criegern, and I. Weitzel. *J. Appl. Phys.*, Vol. 58, No. 11, p. 4036, 1985.
- [58] M. M. Mandurah, K. C. Saraswat, R. W. Dutton, and T. I. Kamins. "Dopant Segregation in Polysilicon," *J. Appl. Phys.*, Vol. 51, p. 5755, 1981.
- [59] M. J. van Dort, et al. *IEDM Tech. Digest*, p. 865, 1994.
- [60] P. B. Griffin. "Physics and Modeling of Two-Dimensional Diffusion in SUPREM-IV," Doctoral Dissertation, Stanford University, Palo Alto, California, Dec. 1989.

- [61] A. G. O'Neill, C. Hill, J. King, and C. Please. "A New Model for the Diffusion of Arsenic in Polycrystalline Silicon," *J. Appl. Phys.*, Vol. 64, No. 1, p. 167, 1988.
- [62] C. Hill and S. Jones. "Growth and Microcrystalline Structure of Poly-Si," *Properties of Silicon: EMIS Data Review*, No. 4, Inspec., p. 933, 1988.
- [63] C. P. Ho, J. D. Plummer, S. E. Hansen, and R. W. Dutton. "VLSI Process Modeling—SUPREM III," *IEEE Trans. Elec. Dev.*, Vol. ED-30, No. 11, p. 1438, 1983.
- [64] L. Mei and R. W. Dutton. "A Process Simulation Model for Multilayer Structures Involving Polycrystalline Silicon," *IEEE Trans. Elec. Dev.*, Vol. ED-29, No. 11, p. 1726, 1982.
- [65] D. Gupta, D. R. Campbell, and P. S. Ho. *Thin Films—Interdiffusion and Reaction*, New York: John Wiley and Sons, p. 161, 1980.
- [66] R. B. Fair. "Concentration Profiles of Diffused Dopants in Silicon," *Impurity Doping Processes in Silicon*, ed. F. F. Y. Wang, Amsterdam: North Holland, 1981.

Default Coefficient Notes

The notes listed in this section appear in Default Coefficient Values, [Page A-1](#) through [A-25](#).

- (1) The ratio of the interstitial component to the vacancy component of diffusion in silicon has been set according to the fractional interstitialcy factors given in [51]. In materials other than silicon, the vacancy component is zero.
- (2) The diffusivity of indium in silicon is taken to be an average of several reported values [3]–[6]; larger values ($1.6\exp(-3.5/kT)$) have been reported for RTA[7].
- (3) Due to the wide spread in reported diffusivities in oxide, the values for phosphorus and arsenic diffusivity in oxide were taken to be an average of those reported in reference [8]. Diffusivity in oxide is reported to be very sensitive to diffusion ambient and impurity concentration, but this dependence is not well characterized and is not modeled in TSUPREM-4. In particular, diffusion from a gaseous source into oxide can result in a “melt-through” process with a much larger effective diffusivity than the value used in TSUPREM-4. The diffusivity of indium in oxide is known to be very fast; it has been approximated here as 10× the diffusivity in silicon.
- (4) These values are for diffusion in the interior of a polysilicon grain. The diffusivities for boron, phosphorus, and antimony are identical to the values for single-crystal silicon. The values for arsenic are the same as for silicon but with the addition of the **DIMM.O** and **DIMM.E** terms. These values were taken from the data of Siedel et al. [52] for bulk single-crystal silicon at high concentration.
- (5) The diffusivity of boron is taken from the analysis of Buonaquisti et al. [53] of data measured by Kamins et al. [54], but the activation energy has been increased to that of single-crystal so as to reproduce measured values at low temperature. For phosphorus, the activation energy for single-crystal silicon is used, while the pre-exponential factor is taken to reproduce measured data at low temperature. For arsenic, the activation energy has been taken from Sakamoto et al. [55] and the pre-exponential factors have been fitted to diffusion profiles at 900°C for various implant doses. For antimony a diffusivity of 10^4 times the value in single-crystal silicon has been assumed. The polycrystalline diffusion model has not been enabled for indium in polysilicon for lack of data.
- (6) Tungsten silicide is assumed to be present in polycrystalline form, with diffusion dominated by grain boundary effects. The diffusivity within the crystalline grains is used here, however. The diffusivity can be increased to approximate the effect of diffusion along grain boundaries, but numerical difficulties may arise if very large values are used.
- (7) The diffusivity of impurities in ambient is used for calculating impurity diffusion through included voids in a structure. These values were chosen to

be about an order of magnitude larger than the diffusivity in solid materials at normal processing temperatures.

- (8) The diffusivities of pairs are estimated to be the same as the diffusivities of the corresponding point defects. Arbitrary (but physically reasonable) values have been chosen for the capture radii and barrier energies.
- (9) Reported values for segregation constants of n-type impurities between silicon and silicon dioxide range between 10 and 1000. The value used here is a rough average of the reported values.
- (10) Due to the absence of data for the segregation constants between polysilicon and silicon dioxide, these segregation constants were set equal to the segregation constants between silicon and silicon dioxide.
- (11) The transport of indium is known to be very fast; the value used here is computed by dividing the diffusivity (in the faster-diffusing material) by a distance of 1 angstrom.
- (12) The values for transport constants between silicon and ambient are taken from the CASPER process modeling program referred to in reference [16]. The values are meant to characterize the impurity transport across a microscopically clean silicon surface such as is present during epitaxial deposition. These values may be too large if a native oxide is present on the surface of the silicon.
- (13) Due to the absence of data for the transport constants between silicon dioxide and ambient and between polysilicon and ambient, these transport constants were set equal to the transport constants between silicon and ambient.
- (14) The transport rate between polysilicon and silicon has been set sufficiently high so that it is not the rate-limiting step in diffusion between the two materials. In the case where the behavior of a thin oxide diffusion barrier between the polysilicon and silicon must be modeled, the oxide layer should be explicitly included in the structure.
- (15) The values for constants between silicon dioxide and silicon are extracted to fit the SIMS data for 30KeV~70KeV, $1E12/cm^2 \sim 1E14/cm^2$ ion implantations followed by thermal processes for several different time conditions at $900^\circ C \sim 1200^\circ C$. Due to the absence of data for the constants between silicon dioxide and polysilicon, these constants were set equal to the constants between silicon dioxide and silicon.
- (16) The values for constants between silicon dioxide and silicon are extracted to fit the Lau et.al. data [17] and the SIMS data for 30KeV and 40 KeV $1E14/cm^2$ ion implantation followed by the RTA for several different time conditions at $800^\circ C \sim 1000^\circ C$. The value of **D . PLUS** parameter on the implantation is set to 1.5 for the simulation. Due to the absence of data for the constants between silicon dioxide and polysilicon, these constants were set at the same values as the constants between silicon dioxide and silicon.

- (17) The values for constants between silicon dioxide and silicon are extracted to fit the SIMS data for 20KeV 5E14/cm² and 32KeV 1E15/cm² ion implantations followed by the RTA for several different time conditions at 1000°C~1050°C. The value of **MAX.DAM** parameter on the implantation is set to 1.15E22/cm³ and the values for the clustering coefficients, **CTN.O** and **CTN.E** are set to 5.35E-20 and -0.874, respectively. Due to the absence of data for the constants between silicon dioxide and polysilicon, these constants were set at the same values as the constants between silicon dioxide and silicon.
- (18) The values for boron are appropriate for concentrations below solid solubility and are derived from the results of Rausch et al. [56] which are confirmed by Schaber et al. [57] to match measured polysilicon/silicon interfacial segregation. The values for phosphorus and arsenic are from Mandurah et al. [58], except that **GSEG.O** has been reduced to match measured polysilicon/silicon interfacial segregation. The initial segregation parameters **GSEG.INI** are set to unity so that there is no initial segregation of dopant into the grain boundary after implant. The interfacial segregation velocities have been set to large values.
- (19) The default values of **FGB** were chosen to give approximately the same low-concentration diffusivity in polysilicon at 1000°C as the default values used in versions of TSUPREM-4 prior to version 6.3.
- (20) The relationship between electrically active and chemical concentrations is assumed to be an equilibrium relationship that is attained immediately. For high concentrations of arsenic in silicon it has been found that the equilibrium clustering relationship may actually require many hours to be reached at low diffusion temperatures.
- (21) The activation energy for the arsenic clustering coefficient has not been well characterized. The value for the activation energy is such that this coefficient agrees with the value in reference [1] at a temperature of 1050°C.
- (22) **DDCS.N** for boron has been set to 1.0 so that parameter sets for modeling BICs in version 1999.2 will give the same results in version 1999.4.
- (23) In Versions 6.0 and above the rate of bulk recombination between interstitials and vacancies is determined by the values of **KB.O** and **KB.E** specified on the **INTERSTITIAL** statement. The values of **KB.O** and **KB.E** specified on the **VACANCY** statement are ignored.
- (24) The bulk recombination rate between interstitials and vacancies was chosen to give good results at lower temperatures. The value has been reduced at high temperatures to speed up convergence of the solution algorithms.
- (25) The **KIV.O** are set indirectly by specifying **KB.LOW**. See [“Recombination of Interstitials with Vacancies” on page 2-35](#)
- (26) A surface recombination velocity that is independent of oxidation rate was chosen to fit the results reported by Professor Scott Dunham of Boston University.

- (27) These values have been chosen to reproduce the values of K_s/D reported by van Dort [59] for the silicon/oxide interface at 900°C. The activation energies are chosen to match the values of K_s/D reported by Griffin [60] for the silicon/oxide interface.
- (28) These values were extracted to fit the data of Poate, et al. [23] using the default parameters for point defect kinetics. Only data for a single silicon implant at a single dose and energy were used; these values may need to be adjusted to provide satisfactory results for other implant conditions.
- (29) Wet oxygen produced from a 95°C bubbler at one atmosphere has in the past been thought to produce an effective oxidant pressure equal to the vapor pressure of water at 95°C: 640 torr, which is equal to 0.84 atmospheres. More recent work [27] has demonstrated oxidation rates closely approximating that of a bubbler by using a pyrogenic steam process with O₂ and H₂ flow rates of 1.175 and 2 liters/minute, respectively. Complete reaction for these flow rates would produce an ambient containing 92% H₂O and 8% O₂. Because TSUPREM-4 cannot model ambients containing both H₂O and O₂, the wet oxygen ambient is defined as 92% H₂O and 8% N₂. This effective mixture of H₂O and N₂ will depend upon the gas flow rates, the diffusion furnace configuration, and even the altitude of the fabrication facility. As a result, these values frequently must be adjusted in order to achieve agreement with the oxide thicknesses grown at a particular facility.
- (30) Oxidation rates for <110> silicon were not determined in references [23] and [27]. From other studies it has been found that the parabolic oxidation rates are not dependent upon the silicon orientation, but that the linear oxidation rates for <110> silicon are approximately a factor of 1.4 greater than those for <100> silicon. This factor has been used to calculate linear oxidation rates for <110> silicon from the rates for <100> silicon.
- (31) The oxidation rates for polysilicon have been set equal to those for <110> silicon [50].
- (32) These values are given in centimeters per second.
- (33) Only one value of **THETA** may be given; the value applies to both O₂ and H₂O.
- (34) The default value for **VC** is the recommended value at 900°C. Recommended values at other temperatures are 225 at 800°C, 425 at 1000°C, and 1100 at 1100°C and higher; for nitride, use 120 at 800°C, 170 at 1000°C, and 350 at 1100°C and higher.
- (35) These values are valid for reaction in argon or vacuum. Value will be smaller at high n-type concentrations.
- (36) These values apply when the amount of interfacial oxygen is minimal after tungsten deposition, e.g., after chemical vapor deposition. If there is significant interfacial oxygen between the tungsten and the silicon (e.g., after deposition by sputtering or evaporation), the values **DIX.0**=9.89e4 and **DIX.E**=3.13 are more recommended.

- (37) These values apply when the amount of interfacial oxygen is minimal after tungsten deposition, e.g., after chemical vapor deposition. If there is significant interfacial oxygen between the tungsten and the silicon (e.g., after deposition by sputtering or evaporation), the values **RATE . 0**=5.22e6 and **RATE . E**=2.46 are recommended.
- (38) Due to the absence of data for the material properties of polysilicon, the values of many material coefficients were set equal to the coefficients for silicon.
- (39) The only models available for silicon carbide in TSUPREM-4 are for ion implantation using the Monte Carlo implant model.
- (40) The polycrystalline microstructure parameters **F11**, **F22**, and **G . DENS** have been set to values appropriate for columnar grains with square cross-section, as in O'Neill et al. [61]. The deposition parameters have been set for the case of LPCVD polysilicon as reported in the review by Hill and Jones [62]. The parameter **FRAC . TA** for the recrystallized grain size after amorphous deposition or implant amorphization has been set to approximately reproduce measured values. The grain growth parameters **GEOM**, **GBE . 0**, and **GBE . H** are taken directly from the model of Mei and Dutton [63] and [64] (with appropriate adjustments to the units). The silicon self-diffusivity values are as stated in Fair [66] with the exception of **DSIMM . 0** and **DSIP . 0**, which have been fitted to the measured doping dependence of the normal grain growth rate. **GAMMA . 0** and **GAMMA . E** have been fit to known undoped normal grain growth rates. The solute drag coefficient **NSEG** is set to the value quoted by Gupti et al. [65] and used in the model of Mei and Dutton [64]. The geometric factor relating grain boundary velocity to average grain growth **ALPHA** has been set to the value recommended by O'Neill et al. [61]. The interface break-up and epitaxial regrowth parameters have been set to have been set effectively disable the models pending further calibration work.
- (41) The viscosities for oxide and nitride were fit to the values reported by Senez et al. [42]. The fits are very good, except for the value for oxide at 900°C (Senez's value is higher) and the value for nitride at 800°C (Senez's value is lower).
- (42) The mechanical properties of ambient are used for calculating the deformation of included voids during oxidation. They describe a material that is very soft and compressible compared to solid materials.
- (43) The values for the electronic stopping prefactor for amorphous materials and random directions in crystalline silicon, **ES . RAND**, are calculated as documented in [Chapter 2, "Crystalline Implant Model" on page 2-94](#). The value for **ES . RAND** for boron implants has been multiplied by a factor of 1.59 as suggested by experimental data. The value for phosphorus in silicon has been adjusted to 2.5 to provide better agreement with experimentally measured implant profiles. Values for **ES . 100**, **ES . 110**, **ES . F . 100**, and **ES . F . 110** for boron have been adjusted to correspond with experimental data for intentionally channeled implants contained in references [45] through [47]. The values for these parameters were selected by comparing

the results of Monte Carlo implant calculations with the experimental implant profiles.

- (44) When a value of 0.0 is specified for **ES .RAND**, **ES .100**, or **ES .110**, the value is calculated as documented in [Chapter 2, “Crystalline Implant Model”](#) on page 2-94.
- (45) Values of **ES .BREAK** and **ES .F .H** for silicon were obtained by fitting to measured implant profiles. The same values are assumed for other materials.
- (46) The adaptive grid parameters for materials other than silicon, polysilicon, and oxide and for solution variables other than antimony, arsenic, boron, and phosphorus are all set to zero. This disables adaptive grid for these materials and solution variables.

Appendix B: Graphics Devices

Determining the Device

TSUPREM-4 supports a number of graphics terminals and hardcopy plotting devices. The file *s4pcap* contains a description of each device known to the program.

The graphics device to be used is determined as follows:

1. If a valid device name has been specified with the **DEVICE** parameter on the **OPTION** statement (see [Page 3-39](#)), that device is used.
2. Otherwise, if the environment variable *DEFPDEV* is defined as the name of a valid plot device, its value is used as the plot device.
3. Otherwise, if the environment variable *TERM* is defined as the name of a valid plot device, its value is used as the plot device.
4. Otherwise, the **DEFAULT** device in the *s4pcap* file is used. Note that the *s4pcap* file may be modified to make the “default” device refer to any available real plotting device.

If any step the sequence above produces an invalid device name, the search continues with the next step.

Output

For interactive display devices (and terminal emulation windows in a windowing system), graphics output is sent to the standard output device (normally the user’s terminal); for hardcopy devices, output is sent to a file, as noted in the descriptions below; for windowed displays, output is sent to a window created by the program. In any case, output may be redirected to a user-specified file (for later display) with the **PLOT.OUT** parameter on the **OPTION** statement. The current output

device is closed and a new output device is opened whenever an **OPTION** statement with the **DEVICE** parameter is executed. This feature can be used to close the current output file or create a new graphics window without exiting TSUPREM-4. Note, however, that if the new graphics device specifies an output file that already exists, the old contents of the output file will be lost unless the file name is preceded by the “+” character in the **PLOT.OUT** parameter on the **OPTION** statement.

The following sections list the devices defined in *s4pcap* by Avant! TCAD. Names in parentheses are alternate names for the device.

Supported Devices

The following devices are fully supported by Avant! TCAD:

X (WINDOW) X-Windows graphics window. Character sequences are generated and piped to the *tmaplot* program, which is executed by the program. Support is provided for 14-color output and filled polygons with a white background. The *tmaplot* program generates a X-Windows graphics window in which graphical output is produced. Multiple graphics images are retained.

If *tmaplot* is executed on a Sun Sparc system *and* the *TMAPLOT_XLIB* environment variable is not set, the display window is controlled through a control panel. Complete help information is provided as part of the user interface. The help information is only available if the directory where the file *studio_view.info* is located is included in the directory list set in the *HELPPATH* environment variable. The display window can also be controlled with the following keys:

- *f*: display next page
- *b*: display previous page
- *w*: print the cursor location in image coordinates
- *d*: use *replot* to convert the current image to a formatted plot file using the device specified by the *TMAPLOT_REPLOT* environment variable (the default is “FORMAT”)

If *tmaplot* is executed on any system other than a Sun Sparc *or* the *TMAPLOT_XLIB* environment variable is set, no control panel is available. The display window can be controlled with the following keys and mouse buttons:

- *left mouse button*: display next page
- *right mouse button*: display previous page
- *z*: Zoom—magnify the image by a factor of 2 and center it at the cursor location
- *Z*: Unzoom—demagnify the image by a factor of 2 and center it at the cursor location
- *p*: Pan—center the image at the cursor location
- *r*: Reset—restore the image to its initial state

- *u*: Undo—undo the previous Zoom, Unzoom, Pan, or Reset operation
- *w*: print the cursor location in image coordinates
- *d*: use *replot* to convert the current image to a formatted plot file using the device specified by the *TMAPLOT_REPLOT* environment variable (the default is “FORMAT”)
- *control-C*: exit

This version of the *tmaplot* program is only provided with versions of TSUPREM-4 for the UNIX operating system.

If STUDIO has been installed, specifying the X Window device activates STUDIO View. Instructions for using STUDIO View are provided in the STUDIO reference manual.

| | |
|--------------------------------------|---|
| I/X | Like x , but with white lines on a black background. |
| X/BW | Like x , but uses software crosshatch patterns instead of color area fill. |
| POSTSCRIPT (PS,PS-P) | Apple LaserWriter and other PostScript printers, with grayscale for area fills. Output is sent to file <i>plotfile.ps</i> . This driver plots in “portrait” mode, i.e., with the short side of the paper along the <i>x</i> axis and the long side along the <i>y</i> axis. |
| L/POSTSCRIPT (PS-L) | Same as POSTSCRIPT , except that it plots in “landscape” mode, i.e., with the long side of the paper along the <i>x</i> axis and the short side along the <i>y</i> axis. |
| PS-INSERT | Same as POSTSCRIPT , except that no output sequences are generated for displaying the plot. The output file produced by this driver can be included in other PostScript documents. |
| C/POSTSCRIPT (PS-C,PS-CP) | Color version of the POSTSCRIPT driver (portrait mode). |
| CL/POSTSCRIPT (PS-CL) | Color version of the L/POSTSCRIPT driver (landscape mode). |
| C/PS-INSERT | Color version of the PS-INSERT driver. |
| REPLOT | Produces a text file <i>s4ofil</i> which can be read by <i>Avant!</i> TCAD's REPLOT utility program. The REPLOT program can redisplay the plot on a number of graphics devices and can interface to graphics libraries such as DISSPLAY, DI-3000, and GDDM. |

| | |
|-----------------------------|--|
| HP2648 (2648) | Hewlett-Packard 2648 (or similar) graphics terminals. |
| HP2623 (2623) | Hewlett-Packard 2623 (or similar) graphics terminals. |
| TEK4100 (4100) | Tektronix 4100-series color graphics terminals. |
| TEKBW | Same as TEK4100 , except uses software-defined crosshatch patterns instead of different colors for area fills (i.e., the COLOR statement). (This entry shows how selected hardware capabilities may be disabled for a device.) |
| TEK4010 (4010) | Tektronix 4010-series graphics terminals. |
| XTERM | Puts graphics in the Tektronix-compatible window of the <i>xterm</i> terminal emulator available with some versions of X Window system. |
| REGIS (VT240, VT241) | DEC and other REGIS-compatible graphics terminals. |
| HP7550 (7550) | Hewlett-Packard 7550 (and other HPGL) pen plotters. Output sent to file <i>plotfile.hpp</i> . This driver plots in “landscape” mode, i.e., with the long side of the paper along the <i>x</i> axis and the short side along the <i>y</i> axis. |
| HP7550-P (7550-P) | Same as HP7550 , except that it plots in “portrait” mode, i.e., with the short side of the paper along the <i>x</i> axis and the long side along the <i>y</i> axis. |
| PRINTRONIX | Printronic line printer in plot mode. Output is sent to file <i>plotfile.lp</i> . |
| SELANAR | Selamar HiREZ 100XL and some Plessey graphics terminals. (This driver can be easily modified to support other terminals that switch between text and Tektronix-compatible graphics modes.) |
| SUN (SUNVIEW) | <p>SunView graphics window. Character sequences are generated and piped to the <i>tmaplot</i> program, which is executed by the program. Support is provided for 14-color output and filled polygons with a white background. The <i>tmaplot</i> program generates a SunView graphics window in which graphical output is produced. Multiple graphics images are retained. The display window can be controlled with the following keys and mouse buttons:</p> <ul style="list-style-type: none"> • <i>left mouse button</i>: display next page • <i>right mouse button</i>: display previous page • <i>z</i>: Zoom—magnify the image by a factor of 2 and center it at the cursor location |

- *Z*: Unzoom—demagnify the image by a factor of 2 and center it at the cursor location
- *p*: Pan—center the image at the cursor location
- *r*: Reset—restore the image to its initial state
- *u*: Undo—undo the previous Zoom, Unzoom, Pan, or Reset operation
- *w*: print the cursor location in image coordinates
- *d*: use *replot* to convert the current image to a formatted plot file using the device specified by the *TMAPLOT_REPLOT* environment variable (the default is “FORMAT”)
- *control-C*: exit

This version of the *tmaplot* program is only provided with versions of TSUPREM-4 for the UNIX operating system. A Sun version of the *tmaplot* program which supports SunView graphics must be executed on the local Sun computer which controls the monitor display.

I / SUN Like **SUN**, but with white lines on a black background.

APOLLO Apollo GPR frame mode window. Character sequences are generated and piped to the *tmaplot* program, which is executed by the program. Support is provided for 8-color output and filled polygons with a white background. The *tmaplot* program generates a GPR graphics window in which graphical output is produced. Multiple graphics images are retained. The display window can be controlled with the following keys:

- *downward vertical scroll*: display next page
- *upward vertical scroll*: display previous page
- *exit* or *abort*: exit

This version of the *tmaplot* program is only provided with versions of TSUPREM-4 for the UNIX operating system. An Apollo version of the *tmaplot* program must be executed on the local Apollo computer which controls the monitor display.

I / APOLLO Like **APOLLO**, but with white lines on a black background.

Unsupported Devices

The *s4pcap* file provided by *Avant!* TCAD defines a number of devices not described above. These devices are not supported by *Avant!* TCAD, but are provided for the benefit of those who may find use for them. Many of these devices have not been tested. If you try one of these devices and find that it works satisfactorily, let us know at *Avant!* TCAD. If you need one of these devices and are willing to help us make it work, let us know; we may be able to add it to the list of supported devices.

The following devices are not supported, but may be useful:

| | |
|---|---|
| TEK4510 (4510) | Tektronix 4510 rasterizer. Output is sent to file <i>plotfile.tek</i> . |
| HPJET (THINKJET) | Hewlett-Packard Thinkjet printer. Output is sent to file <i>plotfile.hpj</i> . |
| HPDESK (HP2671G, HP2673) | Hewlett-Packard 2671G and 2673A desktop printers. |
| HPLP (LP2563) | Hewlett-Packard 256X series line printers. Output is sent to file <i>lp.out</i> . |
| IMAGEN | Imagen laser printer, using ImPress. Output is sent to file <i>plotfile.ip</i> . |
| DITROFF | Generates ditroff (device-independent troff) codes. |
| TGPLOT | Generates a binary file containing plotting information. On some machines, these files are compatible with <i>Avant!</i> TCAD's REPLOT program. Output is sent to file <i>s4pfil</i> . This driver plots in "landscape" mode, i.e., with the long side of the paper along the <i>x</i> axis and the short side along the <i>y</i> axis. |
| TGPLOT-P | Same as TGPLOT , except that it plots in "portrait" mode, i.e., with the short side of the paper along the <i>x</i> axis and the long side along the <i>y</i> axis. |

The Default Device

| | |
|----------------|---|
| DEFAULT | A special device, used to specify a default graphics device from within the <i>s4pcap</i> file. The <i>s4pcap</i> entry for DEFAULT should consist of a link (using the LIKE keyword) to one of the devices listed above. |
|----------------|---|

Modifying *s4pcap*



CAUTION

Modify *s4pcap* at your own risk. Mistakes in the format of the file can produce some rather mysterious and unhelpful messages. Always save a good copy of the file before making any changes. And be sure that every device entry and every parameter in each entry ends with a ":" character.

The device descriptions in *s4pcap* can be modified by you. At present, complete documentation on the format of this file is not available, but some helpful hints are given in the following paragraphs, and much can be inferred by examining the existing entries. Some of the most useful parameters are the following:

- PIXX and PIXY** **PIXX** and **PIXY** specify the total number of addressable points in the *x* and *y* directions. Changing these values changes the size of the plotting area available to the program. (Note that some devices may require corresponding changes to the **INIT** entry.)
- PUNX and PUNY** **PUNX** and **PUNY** specify the number of addressable points per inch in the *x* and *y* directions. These values are used by the program to infer the spacing between points, so that absolute dimensions (i.e., dimensions in centimeters) may be specified in the plotting statements. You may wish to change these values if drawn dimensions are different from the dimensions specified in the plotting statements.
- FILE** **FILE** specifies the default output file for graphics output. If no **FILE** parameter is given, output is sent to the standard output (usually the user's terminal). If the file name is preceded by a "+", output is appended to an existing file (rather than overwriting the file). If the file name is preceded by "!", output is directed to the named program; (this only works on UNIX or UNIX-like systems).
- LIKE** The **LIKE** parameter is used to define one device to be similar to another. The current device uses the same specifications as the device named on the **LIKE** parameter, except for the items listed in the current entry. There are many examples of the **LIKE** parameter in the *s4pcap* file.
- PEN and AREA** **PEN** and **AREA** are examples of entries to control the device. **PEN** specifies the sequence of characters used to change the hardware line type (pen or color), and **AREA** specifies the sequence of characters to use to produce area fill (e.g., for the **COLOR** statement). The important thing about these parameters is that they can be used to turn off hardware line types and area fills (software area fills are used instead). To do this, you may specify the **PEN** or **AREA** parameter preceded by the "!" character and without a value (or "="). An example of this may be found in the *tekbw* entry.
- BFSZ** **BFSZ** sets the size of the internal buffer used to generate the plot output. It may be used to limit the maximum record length written to the plot output file.

Appendix C: Version 2000.2 Enhancements

This appendix describes the new features in TSUPREM-4 version 2000.2 and how to use them. It also discusses the compatibility of version 2000.2 with older versions of TSUPREM-4.

Improved Physical Models

TSUPREM-4 version 2000.2 offers the following improvements in physical modeling:

Cobalt Silicide Model

A model for cobalt silicide has been added. The model is identical in design to the models for other silicides, but uses parameter values appropriate for cobalt. Parameters have been chosen to fit data reported in the literature. The new materials are named *cobalt* and *CoSi2*.

Improved Effective +n Implant Damage Model

The effective +n implant damage model has been enhanced to include the results of recent research by Hobler and Moroz (see [Chapter 2, Implant Damage Model on page 2-98](#)). The enhanced model includes two factors, one to model the effect of ion mass and energy in the high-dose limit

$$f_{pl\infty} = 1 + \mathbf{D} \cdot \mathbf{PHDF} m^{\mathbf{D} \cdot \mathbf{PME}} E^{\lambda_{\infty}} \quad \text{Equation C-1}$$

and one to model the increased effect at lower doses

Equation C-2

$$f_{pld} = \mathbf{D.P1} + \mathbf{D.PCONST} m^{\mathbf{D.PDME}} E^{\lambda_d} \left(\frac{\mathbf{D.PNORM}}{d} \right)^{\mathbf{D.PDE}} R_p^{\mathbf{D.PDRE}}$$

where

$$\lambda_{\infty} = \mathbf{D.PLF} m^{\mathbf{D.PLME}} \quad \text{Equation C-3}$$

$$\lambda_d = \mathbf{D.PDLF} m^{\mathbf{D.PDLME}} \quad \text{Equation C-4}$$

With appropriate parameter values, this model can reproduce the results of Hobler and Moroz

$$n = (1 + 0.0905 m^{0.85} E^{\lambda}) \left(1 + \frac{3.25 C}{\sqrt{m}} E^{\lambda} \frac{3 \times 10^{12}}{N_d} \right), \quad \text{Equation C-5}$$

those of Pelaz *et al.*

$$n = 1 + \frac{0.42}{R_p^{3/4}} \sqrt{Em}, \quad \text{Equation C-6}$$

or the model used in TSUPREM-4 1999.4. The parameters for the Hobler and Moroz model have been made the new default. These values are expected to improve the accuracy of TED simulation, especially for heavier ions. You can use the old values instead if compatibility with TSUPREM-4 1999.4 is required. For more information see [Chapter 2](#).

Dislocation Loops in Amorphized Regions

When a portion of the silicon substrate is amorphized by an ion implantation step, any dislocation loops in the amorphized region will be removed at the start of the next high-temperature processing step.

Trapped Nitrogen Dependent Surface Recombination

The point defect recombination at interfaces depends on trapped nitrogen atoms. Since the recombination rate at interfaces is proportional to the number of kink sites which is supposed to be reduced by trapping nitrogen, the recombination of interstitials and vacancies at interfaces is given by:

$$R = K_s \left(1 - \frac{\sigma_N}{\mathbf{N.MAX.0} \exp\left(-\frac{\mathbf{N.MAX.E}}{kT}\right)} \right) (C - C^*) \quad \text{Equation C-7}$$

where **N.MAX.0** and **N.MAX.E** are parameters on the **INTERSTITIAL** and **VACANCY** statements. The model is taken into account by specifying the **PD.NREC** parameter on the **METHOD** statement.

New Applications Using USEIT

Five Stream Diffusion Model

The 3-stream diffusion model (**PD.FULL**) in TSUPREM-4 has been proven to be accurate enough to simulate most silicon process technologies. In the **PD.FULL** model, the quasi-static assumption under local equilibrium eliminates the equations for dopant-defect pairs. However, it needs to be verified that the assumption is still applicable at a very low temperature by implementing the whole 5-stream diffusion system.

$$\frac{\partial C}{\partial t} = -(\vec{\nabla} \cdot \vec{J}_m + \vec{\nabla} \cdot \vec{J}_n) \quad \text{Equation C-8}$$

$$\frac{\partial M}{\partial t} = -\vec{\nabla} \cdot \vec{J}_m + K_m \left(S \frac{I}{I^*} - \frac{M}{\alpha_m} \right) + K_{mv} \left(S - \frac{M}{\alpha_m} \frac{V}{V^*} \right) \quad \text{Equation C-9}$$

$$\frac{\partial N}{\partial t} = -\vec{\nabla} \cdot \vec{J}_n + K_n \left(S \frac{V}{V^*} - \frac{N}{\alpha_n} \right) + K_{ni} \left(S - \frac{N}{\alpha_n} \frac{I}{I^*} \right) \quad \text{Equation C-10}$$

$$\frac{\partial I}{\partial t} = -\vec{\nabla} \cdot \vec{J}_I - R_b - K_m \left(S \frac{I}{I^*} - \frac{M}{\alpha_m} \right) + K_{ni} \left(S - \frac{N}{\alpha_n} \frac{I}{I^*} \right) \quad \text{Equation C-11}$$

$$\frac{\partial V}{\partial t} = -\vec{\nabla} \cdot \vec{J}_V - R_b - K_n \left(S \frac{V}{V^*} - \frac{N}{\alpha_n} \right) + K_{mv} \left(S - \frac{M}{\alpha_m} \frac{V}{V^*} \right) \quad \text{Equation C-12}$$

where I and V denote an interstitial and a vacancy respectively, and S, M and N denote a substitutional dopant, a dopant-interstitial pair and a dopant-vacancy pair respectively. Since the dopant-defect pairs are mobile, their boundary conditions (recombination or transporting flux) must be specified. Since the point defects, and also dopant-defect pairs, are ignored in the other materials but silicon, the annihilation of dopant-defect pairs at interfaces needs to be specified. The recombinations are given by:

$$F_m = h \left(\frac{M}{m} - M^* \right) \quad \text{Equation C-13}$$

$$F_n = h \left(\frac{N}{n} - N^* \right) \quad \text{Equation C-14}$$

$$\text{where } M^* = \frac{\alpha_m C_a}{1 + \alpha_m + \alpha_n} \text{ and } N^* = \frac{\alpha_n C_a}{1 + \alpha_m + \alpha_n}.$$

For more information about the implementation using USEIT, see [Chapter 6, 5-Stream Diffusion Model on page 6-48](#).

SiGe Model for Boron

The model is based on Lever, Bonar and Willoughby's work. In SiGe, the isovalent atoms like Ge can be as traps for boron atoms so that the number of mobile boron atoms is reduced. Trapping a boron atom forms an immobile complex Ge-B.

$$\frac{C_{GeB}}{C_a} = \frac{C_{as} - C_a}{C_a} = \frac{C_{Ge}}{C_0} \quad \text{Equation C-15}$$

where C_0 is the characteristic concentration, $3.2 \times 10^{21} / \text{cm}^3$, and C_{Ge} is the germanium concentration. C_{as} is the active concentration when ignoring the boron trapping. The intrinsic carrier concentration increases with the Ge content because of the energy band-gap reduction.

$$n_i = n_{i, Si} \exp\left(\frac{\chi \delta E_g}{2kT}\right) \quad \text{Equation C-16}$$

where χ is the fraction of Ge content, and δE_g is the experimental constant energy, 0.7eV. Assuming the equilibrium point defect concentrations, from Equation C-15 and Equation C-16, the boron diffusion flux can be calculated by:

$$\vec{J} = -D_B \left(\vec{\nabla} C_a + C_a \frac{\vec{\nabla} N_T}{Q} + C_a \frac{2n}{Q} \cdot \vec{\nabla} \ln(n_i) \right) \quad \text{Equation C-17}$$

where N_T and Q mean the net doping concentration, $N_a - N_d$, and the total carrier concentration, $n + p$, respectively. The boron diffusivity D_B is given by:

$$D_B = \left(D_x + D_P \cdot \frac{p}{n_{i, Si}} \right) \cdot \exp\left(-\frac{\epsilon s \chi}{kT}\right) \quad \text{Equation C-18}$$

where ϵ and s mean the energy per unit strain (40 eV/strain) and the characteristic strain in the plane of the SiGe layer (0.042), respectively. Equation C-18 includes the Cowern *et.al.*'s work, which shows that the diffusion of B in compressively strained SiGe decreases due to an increase in the activation energy of B diffusion. Note that the intrinsic carrier concentration in Equation C-18 is acting just as the convenient normalization factor so that $n_{i, Si}$ must be used instead of n_i . For more information on the implementation using USEIT, see [Chapter 6, SiGe Model on page 6-58](#).

Carbon-Doped Silicon Model

Doping carbon in silicon results in the retardation of boron and phosphorus diffusion and the enhancement of arsenic and antimony diffusion. The carbon atoms play a role to drain out interstitials from the carbon-rich region through the kick-out and Frank-Turnbull reactions so that it causes the undersaturation of interstitials and the oversaturation of vacancy there. The following example shows how to implement the carbon impurity by using the built-in capability (**PD.FULL**).

```
$ carbon definition
impurity imp=carbon new c.init=1e15
method   imp=carbon lu part ^symm time.ste
         rel.err=0.01 abs.err=1.0e9
impurity imp=carbon mat=silicon cm.sec +
         dix.0=1.9 dix.e=3.1 +
         dipair.0=4.4 dipair.e=0.88 +
         dvpair.0=4.4 dvpair.0=0.88 +
         r.i.s=10 r.v.s=10 r.ip.v=10 r.vp.i=10
$ assumptions
impurity imp=carbon mat=ambient +
         dix.0=190 dix.e=3.1 cm.sec
impurity imp=carbon mat=silicon /mat=ambient
         seg.0=30 seg.e=0.0
impurity imp=carbon mat=silicon /mat=ambient
         trans.0=0.1 trans.e=0.0
```

Like the 5-stream diffusion model, each equation for an interstitial carbon and a substitutional carbon can be implemented with USEIT. For more information, see the Application Note “Models for diffusion retardation and enhancement due to carbon”.

Improved Accuracy, Flexibility, and Ease of Use

Several changes have been made in TSUPREM-4 version 2000.2 to improve accuracy, flexibility, and ease of use.

Mesh Unrefinement

The mesh spacing required for accurate numerical modeling may vary during the course of a simulation. The fine grid needed to resolve an implanted profile or a diffusing front may not be needed once the profile has been driven in. With version 2000.2, TSUPREM-4 has been enhanced to allow removal of these unneeded mesh points. This *unrefinement* of the mesh can be done automatically or at user-specified points in the simulation.

Automatic Unrefinement

By default, TSUPREM-4 will now examine the mesh during every high-temperature processing step and remove mesh points that are not required for an accurate

solution. The criteria for unrefinement are similar to those for adaptive grid refinement, and the parameters **REL .ADAP**, **ABS .ADAP**, and **ERR .FAC** are used for both. Points can be removed provided that the numerical error introduced is less than **UNREFINE** \times **REL .ADAP**, where **UNREFINE** is a parameter on the **METHOD** statement. By default, **UNREFINE** has a value of 0.25, so unrefinement can occur when the estimated error is less than one fourth the error that triggers mesh refinement. Unrefinement will not occur if it would result in a mesh spacing greater than **MAX .SPAC** (specified for each material on the **METHOD** statement).

Automatic mesh unrefinement can be disabled by setting **UNREFINE** to zero:

```
METHOD UNREFINE=0
```

Unrefinement only occurs when adaptive gridding during diffusion is enabled, i.e., when **DIF .ADAP** is set.

Manual Unrefinement

You can specify the removal of unneeded mesh points at any point in a simulation using the **UNREFINE** parameter on the **STRUCTURE** statement. The **REPEAT** parameter specifies that maximum number of unrefinement steps. The **ROWS** and **COLUMNS** parameters allow removal of rows and columns of grid nodes in the undoped (or background doped) substrate. The unrefinement can be limited to a portion of the structure with the **X .MIN**, **X .MAX**, **Y .MIN**, and **Y .MAX** parameters. The same criteria are used for manual unrefinement as for automatic unrefinement. Manual unrefinement can be used to remove specific nodes or groups of nodes in a structure.

For more information on mesh unrefinement, see [Chapter 2, Adaptive Gridding on page 2-10](#), [Chapter 3, STRUCTURE on page 3-77](#), and [Chapter 3, METHOD on page 3-188](#)

Improved Mesh Refinement

Adaptive mesh refinement has been improved in TSUPREM-4 version 2000.2. The new version can resolve two-dimensional doping profiles with a smaller number of mesh points. In particular, it reduces the number of points added in response to slight lateral variations in impurity concentration. The new version gives reduced run times for simulations that depend on adaptive gridding.

More Flexible MOMENT Statement

In calculating implantation profiles, there are two dopant distributions of interest: the distribution along the direction of the ion beam and the distribution in the direction perpendicular to the wafer; these distributions will be different if the wafer is tilted relative to the ion beam during the implantation process. TSUPREM-4 uses the distribution along the ion beam to calculate the as-implanted dopant distribution, while the distribution perpendicular to the wafer is the only distribution that can be practically measured (e.g., by SIMS). In the past, there was little need to distinguish between the two distributions because small tilt

angles were the norm and subsequent thermal processing made small differences in as-implanted distributions irrelevant. But now, with large tilt angles and low thermal budget processes, the distinction can be significant.

Until now, TSUPREM-4 has assumed that the distribution described by moments on the **MOMENT** statement was that along the ion beam. But moments along the beam are not readily available for tilted implants; they must be calculated from the measured distribution perpendicular to the surface. TSUPREM-4 version 2000.2 eliminates this problem. When the **TILT** parameter is specified on the **MOMENT** statement, the program calculates the moments along the ion beam from the moments measured perpendicular to the surface (assuming the specified tilt angle between the two).

The **TILT** parameter on the **MOMENT** statement lets you use the same moments in an implant data file and on the **MOMENT** statement without the need to convert between along-the-beam and perpendicular-to-the-wafer directions. You can use the **MOMENT** statement in an optimization loop to fit moments to measured implant data, and then include the extracted values in an implant moments file without conversion. Similarly, no conversion is needed to use moments from a moments file in the **MOMENT** statement. The only requirement is that the angle between the ion beam and the direction along which the dopant distribution is measured is specified with the **TILT** parameter on the **MOMENT** statement.

For more information see [“MOMENT” on page 3-224](#).

Ion Implantation Enhancements

1. Calling Taurus Implant Models from TSUPREM-4:

TSUPREM-4 is now capable of calling Taurus implant models directly by using keyword **TAUR.MOD** in the implant statement. The value of **TAUR.MOD** is one of the following Taurus models: Gauss, Pearson, DualPearson, or MonteCarlo. If the name of the Taurus executable is not "taurus" (default), **TAUR.EXE**=<your_taurus_executable_name> will also have to be specified. The other parameters which can be passed along to Taurus are described in [Chapter 3, IMPLANT on page 3-103](#).

Example:

```
IMPLANT BORON TILT=7 ROT=30 ENERGY=10 DOSE=1e14
TAUR.MOD=MonteCarlo +

    TAUR.EXE=my_taurus_executable NION=5000
```

2. Using PAI (Pre-Amorphization Implant):

The accumulated damage from previous implants can have a large effect on the channeling of the subsequent implants. TSUPREM-4 offers the solution to this problem by implementing the PAI model, which converts the accumulated dam-

age to the equivalent oxide thickness and then uses the oxide thickness-dependent implant table to reduce the channeling.

The PAI model can be selected by specifying **IMPL.TAB=PAI.*** (where "*" is implant species).

Please note that since the PAI model uses the TR.* tables (screening oxide thickness dependent tables), this model is not available for all implant species. Currently only the TR.BORON table is available.

Example:

```
IMPLANT ARSENIC TILT=7 ROT=30 ENERGY=50 DOSE=5e14
IMPL.TAB=TR.ARS DAMAGE
```

```
IMPLANT BORON TILT=0 ROT=0 ENERGY=10 DOSE=1e13
IMPL.TAB=PAI.BORON DAMAGE
```

Miscellaneous Improvements and Other Changes

The following miscellaneous improvements and other changes have been made in TSUPREM-4 version 2000.2:

1. The algorithm for adjusting the vertical and lateral spreads to account for rotation of the wafer has been improved. The new algorithm assumes that the specified lateral spread is the spread perpendicular to the direction of the beam, regardless of the tilt used in the experiment from which the moments are extracted.
2. The ability to save and load structures with the **STRUCTURE** statement has been eliminated. The **SAVEFILE** and **LOADFILE** statements must be used instead.
3. The maximum number of regions allowed in a structure has been increased from 40 to 100.
4. The interpolation of the channeling fraction is changed from linear to logarithmic interpolation, which is more accurate under normal conditions. The simulation results will now be identical to those of Taurus under the same implant conditions.
5. The dose reduction due to tilt implants is now implemented for the Monte Carlo model. Now all implant models (including both TSUPREM-4 and Taurus) employ the same dose reduction under tilt implant conditions.

Error Corrections

The following TSUPREM-4 errors have been corrected in version 2000.2:

1. The calculation of beamwidth of the ion beam for the Monte Carlo implant model was incorrect; the calculated width in the direction perpendicular to the tilt was too small.
2. The threshold for Frenkel pair generation was set to zero for temperatures above about 145°C; while this suppresses Frenkel pair generation completely, it also eliminates the dependence of the implant on pre-existing damage (e.g., preamorphization).
3. Floating-point exception or “moments out of range” error on implant with some combinations of tilt, rotation and moments.
4. Discretization of area was incorrect for some obtuse triangles.
5. “Unable to retriangulate” errors during oxidation.
6. Small errors in impurity dose when removing a node from the mesh.
7. Duplicate entries for active concentrations saved in TIF files when **ACT . FULL** is used.
8. “Could not open Topography file” when Taurus Topography input file does not end with a newline character.
9. “Out of memory” error (requesting 0 bytes) during **STRUCTURE REFLECT** with some mask files.
10. “Flat triangle” or “unable to retriangulate” error during deposition.
11. Roundoff errors could cause cutting of the time step and slow simulation when **ACT . FULL** is used.
12. The projected range for the damage distribution during an arsenic implant has been corrected; the range was too high at low energies. The range for damage caused by high energy (above 1 MeV) phosphorus implants has been reduced to at most 0.9 times the range of the phosphorus distribution.
13. Some depositions and etches with Taurus Topography produced incorrect results.
14. “Segment enters triangle but does not leave” error during etch.

Compatibility with TSUPREM-4 Version 1999.4

Input files that ran correctly with version 1999.4 should run correctly with version 2000.2 but may give somewhat different results. In most cases, the results obtained with version 2000.2 are more accurate, but in some cases it may be desirable to reproduce the results given by version 1999.4. The changes that might cause version 2000.2 to give different results are listed below. In most cases, the changes can be disabled. As noted below, the input file *s4compat994* (shipped with TSUPREM-4 version 2000.2) includes the commands to disable these changes. Other changes can be disabled by setting **V . COMPAT=1999.4** on the **OPTION** statement.

Accuracy Issues

The following differences between versions 2000.2 and 1999.4 may affect the accuracy of your results:

1. The algorithms for handling the **BEAMWIDT** parameter and threshold for Frenkel pair generation at elevated temperatures (above about 145°C) during Monte Carlo implants have been corrected. You should get slightly less channeling when **BEAMWIDT** is specified and more dependence of channeling on pre-existing damage at elevated temperatures.
2. The improved algorithm for adjusting the vertical and lateral spreads at large rotation angles for analytical implants may produce slightly different profiles. The old algorithm is used when **V.COMPAT=1999.4** is specified.
3. The effective +n implant damage model of Hobler and Moroz is now used by default. This should give more accurate simulation of TED than the previous default. The old defaults can be restored by specifying

```
MATERIAL MAT=SILICON D.PHDF=0 D.PCONST=0.42 +
  D.PDME=0.5 D.PDLF=0.5 D.PDLME=0 D.PNORM=1e14 +
  D.PDE=0.5 D.PDRE=-0.75
```

or by including the file *s4compat994*. (The statement above assumes that **D.PNORM=1e14** and **D.PDE=0.5** should be used for all impurities, while TSUPREM-4 1999.4 only specified values for boron, phosphorus, antimony, and arsenic; for full compatibility when other impurities are implanted, the corresponding statements from the *s4compat994* file should be used instead.)

4. Dislocation loops are now removed from regions amorphized by a subsequent ion implantation. This should improve the accuracy of the simulation.
5. The correction of the damage profiles for low energy arsenic implants and high energy phosphorus implants may cause noticeable change in the amount of TED during subsequent annealing.
6. Automatic grid unrefinement may reduce the accuracy in some simulations, but not below the accuracy specified by the product of **REL.ADAP** and **ERR.FAC**. Unrefinement can be disabled by setting **UNREFINE** to zero on the **METHOD** statement


```
METHOD UNREFINE=0
```

 or by including the file *s4compat994*.
7. The changes to the grid refinement algorithms may cause slight changes in simulated results, but there should be no significant difference in the accuracy of the results. The old algorithms are used when **V.COMPAT=1999.4** is specified.
8. The ability to save and load structures with the **STRUCTURE** statement has been eliminated. The **SAVEFILE** and **LOADFILE** statements must be used instead. Input files that use the **STRUCTURE** statement to save or load files will need to be modified.

Performance Issues

Some of the changes between versions 2000.2 and 1999.4 affect the speed of the simulation:

1. The combination of automatic grid unrefinement and the improvements in grid refinement will increase the speed of many simulations. Simulation times have been reduced by 25 to 50% for moderately complex examples, and speedups by a factor of 5 or more have been obtained for some large structures.
2. Improved treatment of roundoff errors in the **ACT . TRAN** model will improve the speed of some simulations using that model.

Appendix D: Format of Mask Data Files

Introduction

TSUPREM-4 can read mask layout data from specially formatted files. Taurus Layout — IC Layout Interface can be used to create these mask data files interactively, extracting the appropriate information from a GDS description of a layout. This appendix describes the format of the mask data files used by TSUPREM-4. It is intended for use by experienced programmers who wish to generate mask data files for use by TSUPREM-4. It can also be used for generating simple mask files by hand.

Format

The figure below shows an example of a mask data file. The first line identifies the format of the file. It contains the characters “TL1” followed by a space and a 4-digit number. The number represents the version of Taurus Layout that created the file. Current versions of Taurus Layout specify values from 0000 to 0100.

```

TL1 0100
/ Mask definition file s4ex4m.tl1, for use with s4ex4[abc].
/
%%
%%
1e3
  0 5000
4
Field 1
  0 3900
Poly 1
  0 0650
Contact 2
  0 1950
3250 5000
Metal 1
1300 5000

```

Figure D-1 Example of a mask data file

The identification line is followed by any number of comment lines, identified by a “/” character in the first column. The comments are printed when the file is read with the **MASK** statement in TSUPREM-4. The comments are followed by any number of lines beginning with a “%” character. These lines are for use by Taurus Layout and are ignored by TSUPREM-4. The first line following the “%” lines contains a single floating-point number that represents the scale factor (units per micron) used for coordinates in the mask file. In the example of [Figure D-1](#) there are 1000 units per micron, so the quantity 3250 is used in the mask file to represent a value of 3.25 microns. Following the scale factor is a line containing the minimum and maximum coordinates in the mask specification. In this example, the mask specification extends from 0 microns to 5.0 microns.

The next line gives the number of masks in the file. It is followed by the data for each mask. The first line of data for each mask gives the name of the mask and the number of opaque segments. Following lines give the minimum and maximum coordinates of each opaque segment, in scaled units. TSUPREM-4 does not distinguish between upper and lower case in mask names.

Appendix E: Data Format of Saved Structure Files

Introduction

TSUPREM-4 can save simulated structures using the TSUPREM-4 format and the Medici format. It reads saved structures in TSUPREM-4 format. Although these files contain formatted text, they are not intended to be understood or modified by users. The documentation that follows is provided for experienced programmers who wish to interface TSUPREM-4 with other programs.

TSUPREM-4 can also read and write Technology Interchange Format (TIF) files. For more information on the format of TIF files, please contact *Avant!* TCAD.

TSUPREM-4 uses other file formats for communicating with the Taurus-Lithography and MINIMOS 5 programs. These formats contain subsets of the solution information provided by the TSUPREM-4 and Medici formats and are not described here.

TSUPREM-4 Structure File Format

The TSUPREM-4 structure file format has evolved over time. The following section describes the format used by the latest version of the program, while the section [“Older Versions of TSUPREM-4” on page E-6](#) describes differences between the current format and that used by older version of TSUPREM-4.

Current Version

This section describes the data format of the TSUPREM-4 structure file. Indented lines denote lines that appear in the saved structure file. They are not indented in the file—the first code letter must be the first character of the line. Names in angle brackets (<>) denote quantities (usually numbers) that are described in the text. Floating-point values may use exponential notation, with the exponent denoted by “e” or “E” (e.g., 1.23e21).

Program Identification

```
v TMA TSUPREM-4 (1999.2S)
```

The version number (1999.2) and platform code (S) depend on the particular release of the program that produced the file. This line should appear exactly as shown, except for the version number and platform code.

Coordinates

```
c <index> <x-coordinate> <y-coordinate> 0
```

There is one “c” line for each mesh point in the grid. <index> is the integer index of the mesh point, starting at 1 and increasing to the total number of points in the grid. <x-coordinate> and <y-coordinate> are the (floating-point) coordinates of the mesh point, in microns.

Edges

```
e <index> <point-1> <point-2> <bcode>
```

There is one “e” line for each edge segment in the structure, where an edge is defined as a mesh line that lies on the interface between two regions or on the boundary of the structure. <index> is the (integer) index of the edge segment. <index> starts at one and increases to the total number of edges in the mesh. <point-1> and <point-2> are the (integer) indices of the coordinates that define the starting and ending points of the edge. <bcode> has the (integer) value 2 for edges on exposed boundaries, and zero otherwise. The “e” lines are produced by TSUPREM-4 when writing a structure file but are ignored when the structure file is read; they need not be present in the file.

Regions

```
r <index> <material>
b <edge>
. .
. .
. .
```

Each region is defined by an “r” line and a series of “b” lines. The regions are specified in the order of their region *<index>*, starting at one. The *<material>* of each region is specified by name. The name *e_photoreซิส* is used for exposed photoresist.

The “b” lines specify the list of edges, in order, that define the region. Each (integer) value *<edge>* refers to the *<index>* value on an “e” line. The “b” lines define the boundary of the region in a clockwise direction when using a left-handed coordinate system (*x* increasing towards the right, and *y* increasing from top to bottom). The “b” lines are produced by TSUPREM-4 when writing a structure file but are ignored when the structure file is read; they need not be present in the file.

Triangles

t *<index>* *<region>* *<c1>* *<c2>* *<c3>* *<t1>* *<t2>* *<t3>*

The triangles are specified in order of their (integer) *<index>*, starting at one. The (integer) *<region>* specifies the region to which the triangle belongs. *<c1>*, *<c2>*, and *<c3>* are the indices of the “c” lines that define the corners of the triangle. They are ordered to produce a clockwise triangle in a left-handed coordinate system. *<t1>*, *<t2>*, and *<t3>* are the (integer) indices of the neighboring triangles. *<t1>* is the neighbor on the side opposite corner *<c1>*; *<t2>* is the neighbor opposite *<c2>*; and *<t3>* is the neighbor opposite *<c3>*. A code of -1024 is used for edges on a reflecting boundary, and -1022 is used for edges on an exposed boundary.

Model Parameters

M *<subornt>* *<nadd>* *<temperature>* *<oxmodel>* *<pdmodel>*
<subrotc> *<phototype>* *<activmod>* *<orVoid>* *<customMod>*
<clModel> *<kbModel>* *<shModel>* *<itModel>*

All parameters appear on a single line in the structure file.

The “M” line specifies miscellaneous information about the structure and models used. The (integer) value *<subornt>* specifies the substrate orientation:

Table E-1

| <i><subornt></i> | Substrate Orientation |
|------------------------|-----------------------|
| 0 | <i><100></i> |
| 1 | <i><110></i> |
| 2 | <i><111></i> |

The integer *<nadd>* specifies how many additional parameters appear on the “M” line; in version 6.6 *<nadd>* is 12, and all of the parameters listed above are present. *<temperature>* is the last processing temperature, in degrees Kelvin.

<oxmodel> indicates which oxidation model was selected by the user. The values of *<oxmodel>* are as follows:

Table E-2

| Oxmodel | Oxidation Model |
|---------|-----------------|
| 0 | ERFC |
| 1 | VERTICAL |
| 2 | COMPRESS |
| 3 | VISCOUS |
| 4 | VISCOELA |
| 5 | ERF1 |
| 6 | ERF2 |
| 7 | ERFG |

<pdmodel> indicates what point defects models are to be used. Its value is calculated by adding 8 to the value of **NSTREAMS** (1 or 3) then adding an additional value for each of the following point defect parameters that is set true:

Table E-3

| Parameter | Value Added to <i><pdmodel></i> if True |
|-----------------|---|
| PAIR.GRA | 16 |
| PAIR.SAT | 32 |
| PAIR.REC | 64 |
| PD.PFLUX | 256 |
| PD.PTIME | 512 |
| PD.PREC | 1024 |

<subrotc> gives the orientation of the *x* axis of the simulation—it is the cosine of the angle between the *x* axis and a $\langle 110 \rangle$ direction. *<phototype>* specifies the polarity of any photoresist in the structure—it is 1 for positive photoresist and 0 for negative photoresist. *<activmod>* indicates whether the transient activation model is in effect; it has the value 1 when the **ACT.TRAN** model is enabled and 0 when the **ACT.EQUI** model is in effect. *<orVoid>* is the current void size at silicon/polysilicon interfaces. *<customMod>* indicates any custom or experimental models that are in effect; it is reserved for future use. *<clModel>* is 1 if the interstitial clustering model is enabled and 0 otherwise. *<kbModel>* indicates which point defect bulk recombination is selected:

Table E-4

| <i><kbModel></i> | Recombination Model |
|------------------------|---------------------|
| -1 | none |
| 0 | KB.LOW |
| 1 | KB.MED |
| 2 | KB.HIGH |

where “none” indicates that the user has explicitly specified values of **KIV.0** and/or **KIV.E**. Note that *<kbModel>* is ignored when reading the file unless compati-

bility with version 1998.4 or earlier has been specified with **V.COMPAT**.
 <shModel> is 1 if the stress history model is enabled and 0 otherwise; <itModel>
 is 1 if the interface trap model is enabled and 0 otherwise.

Solution Values

```
s <nsol> <sol1> <sol2> . . . <soln>
```

The “s” line specifies which solution values are associated with each node of the structure. The (integer) value <nsol> specifies how many solution values are given, while <sol1> through <soln> give the names of the solutions that are present. Valid names in version 6.4 are: the names of the impurities (including user-defined impurities); the names of the impurities preceded by “a_”, indicating the active portion of the impurity concentration; the names of the impurities preceded by “gb_”, indicating the concentration of the impurity associated with polycrystalline grain boundaries; the names *vacancy*, *interstitial*, *x.vel*, *y.vel*, *o2*, *h2o*, *trap*, *sxx*, *syy*, *sxy*, *rloop*, *dloop*, *lgrain*, *gorient*, and *tpoly*. Note that this list of solutions is subject to change in future revisions of the program.

Nodes

```
n <cindex-1> <region> <val1> . . . <valn>
```

The “n” lines specify the solution values at each node. For each point in the mesh, there is an “n” line for each material at the point. The (integer) value <cindex-1> gives the index of the coordinate, minus one, and the (integer) <region> specifies the region. Points on exposed surfaces must also have an *ambient* node, denoted by a region index of zero. For each node, the solution values are given by <val1> through <valn> (floating-point).

Example

The following input produces the structure file shown in [Figure E-1](#).

```
line x loc=0
line x loc=1
line y loc=0
line y loc=1
init boron=1e13
deposit oxide thick=.2
savefile out.file=test.str
stop
```

This structure contains 6 grid points, 2 regions (silicon and oxide), 4 triangles, and 10 nodes (4 in silicon, 4 in oxide, and 2 ambient nodes). Solution values for boron concentration are given at each node. No last processing temperature was specified, and the **VERTICAL** oxidation model and the **PD.FERMI** point defect model were used.

```

v TMA TSUPREM-4 (1999.2S)
c 1 0 0 0
c 2 0 0.99999997 0
c 3 0.99999997 0 0
c 4 0.99999997 0.99999997 0
c 5 0 -0.19999999 0
c 6 0.99999997 -0.19999999 0
e 1 2 1 0
e 2 4 2 0
e 3 3 4 0
e 4 5 6 2
e 5 1 5 0
e 6 6 3 0
e 7 3 1 0
r 1 silicon
b 1
b 7
b 3
b 2
r 2 oxide
b 4
b 6
b 7
b 5
t 1 1 1 3 2 2 -1024 4
t 2 1 3 4 2 -1024 1 -1024
t 3 2 1 5 6 -1022 4 -1024
t 4 2 1 6 3 -1024 1 3
M 0 12 1073.16 1 9 1.000000 0 0 0 0 1 0 0 1
s 2 boron a boron
n 0 1 1.000000e+13 1.0000e+13
n 0 2 1.000000e+05 1.0000e+05
n 1 1 1.000000e+13 1.0000e+13
n 2 1 1.000000e+13 1.0000e+13
n 2 2 1.000000e+05 1.0000e+05
n 3 1 1.000000e+13 1.0000e+13
n 4 2 1.000000e+05 1.0000e+05
n 4 0 1.000000e+05 1.0000e+05
n 5 2 1.000000e+05 1.0000e+05
n 5 0 1.000000e+05 1.0000e+05

```

Figure E-1 TSUPREM-4 structure file

Older Versions of TSUPREM-4

This section describes differences between the current TSUPREM-4 structure file format and that used in older versions of the program.

Regions In versions 6.0 and earlier, regions were identified by a `<material>` code:

Table E-5

| <code><material></code> | Material |
|-------------------------------|-------------------------|
| 1 | oxide |
| 2 | nitride |
| 3 | silicon |
| 4 | polysilicon |
| 5 | oxynitride |
| 6 | aluminum |
| 7 | photoresist |
| 8 | ambient (included void) |

Model Parameters

In versions 8926 and before `<nadd>` and the parameters that follow are omitted. In versions 9035 and 5.0 `<nadd>` is 3, and the `<temperature>`, `<oxmodel>`, and `<pmodel>` parameters are present. Starting with version 5.2, `<nadd>` is 5, and the "M" line includes the `<subrotc>` and `<phototype>` parameters. Starting with version 6.2, `<nadd>` is 6 and the "M" line includes the `<activmod>` parameter. Starting with version 6.3, `<nadd>` is 7 and the "M" line includes the `<orVoid>` parameter. Starting with version 6.4, `<nadd>` is 8 and the "M" line includes the `<customMod>` parameter. Starting with version 6.5, `<nadd>` is 10 and the "M" line includes the `<clModel>` and `<kbModel>` parameters. Starting with version 6.6, `<nadd>` is 12 and the "M" line includes the `<shModel>` and `<itModel>` parameters. Starting with version 1999.2, -1 is a valid value for `<kbModel>`.

In versions of TSUPREM-4 prior to version 6.0 the value of `<pmodel>` is interpreted as follows:

Table E-6

| <code><pmodel></code> | Point Defect Model |
|-----------------------------|--|
| 0 | TWO.DIM (PD.TRANS) |
| 2 | FERMI (PD.FERMI) |
| 4 | STEADY (Obsolete;PD.TRANS used instead) |

Solution Values

In versions 6.0 and earlier, `<sol1>` through `<soln>` are integer values indicating which solution values are present:

Table E-7

| <code><solx></code> | Solution |
|---------------------------|----------------------------|
| 0 | vacancy concentration |
| 1 | interstitial concentration |
| 2 | arsenic concentration |
| 3 | phosphorus concentration |
| 4 | antimony concentration |
| 5 | boron concentration |

Table E-7

| <i><solx></i> | Solution |
|---------------------|--|
| 6 | steady-state vacancy concentration |
| 7 | steady-state interstitial concentration |
| 8 | x component of flow velocity |
| 9 | y component of flow velocity |
| 10 | oxidant concentration (dry O2) |
| 11 | oxidant concentration (wet O2) |
| 12 | concentration of filled interstitial traps |
| 15 | Sxx component of stress |
| 16 | Syy component of stress |
| 17 | Sxy component of stress |
| 20 | active component of arsenic concentration |
| 21 | active component of phosphorus concentration |
| 22 | active component of antimony concentration |
| 23 | active component of boron concentration |

Nodes In version 6.0 and earlier, *<region>* specifies a material index rather than a region index. The same values are used to identify materials as are used in the “r” lines.

Medici Structure File Format

This section describes the output file format produced by the “**SAVEFILE MEDICT**” statement in TSUPREM-4. The TSUPREM-4 Medici file contains numerical values only. Floating point values use an exponent, denoted by “e”. Integer values do not have an exponent or a decimal point. Values are separated by spaces. The format of the file is as follows.

Line 1:

<npt> <ntri> <nelpt>

This line specifies the number of points in the grid *<npt>*, the number of triangular elements *<ntri>*, and the number of electrode points *<nelpt>*.

Line 2:

<nelec> -<nmat>

This line specifies the number of electrodes *<nelec>* and the number of materials in the structure *<nmat>*. Note that the number of materials is preceded by a minus sign. TSUPREM-4 converts all points on an interface with aluminum to electrodes.

Lines 3 through <npt>+2:

<xcoord> <ycoord> <cnet> <ctotal>

These lines specify the *x* and *y* coordinates <xcoord> and <ycoord> of each mesh point (in centimeters), the net n-type doping <cnet> (in atoms/cm³), and the total doping <ctotal> (in atoms/cm³). Note that a negative value of <cnet> indicates a p-type region. <xcoord>, <ycoord>, <cnet>, and <ctotal> are all floating-point values.

Lines <npt>+3 through <npt>+<ntri>+2:

<reg> <p1> <p2> <p3>

These lines specify the triangular elements that make up the mesh. <reg> is the region number for the element. (Regions are numbered consecutively, starting at one.) <p1>, <p2>, and <p3> are the indices of the points at the vertices of the triangular element. The points are specified in a clockwise order, when viewed in a left-handed coordinate system (i.e., *x* increasing from left to right and *y* increasing from top to bottom).

Lines <npt>+<ntri>+3 through <npt>+<ntri>+<nelpt>+2:

<ept> <elec>

These lines specify the electrode <elec> associated with each electrode point <ept>. <elec> is either a name (assigned with the **ELECTRODE** statement) or a number. (Electrode numbers start at one and increasing to <nelec>, as needed.)

Line <npt>+<ntri>+<nelpt>+3 (last line in file):

<mat1> <mat2> ... <matn>

This line specifies the material associated with each region in the structure:

Table E-8

| <matx> | Material |
|--------|---|
| -6 | oxynitride |
| -4 | unspecified insulator (e.g., photoresist) |
| -2 | nitride |
| -1 | oxide |
| 1 | silicon |
| 3 | polysilicon |
| 4 | unspecified semiconductor |

Example The following TSUPREM-4 input produces the structure file shown in [Figure E-2](#):

```

line x loc=0
line x loc=1
line y loc=0
line y loc=1
init boron=1e13
deposit oxide thick=.2
deposit aluminum thick=.1
select z=0 temp=800
savefile medici out.file=testpi.str
stop

```

```

6 4 2
1 -2
0.000000e+00 0.000000e+00 -1.000000e+13 1.000000e+13
0.000000e+00 1.000000e-04 -1.000000e+13 1.000000e+13
1.000000e-04 0.000000e+00 -1.000000e+13 1.000000e+13
1.000000e-04 1.000000e-04 -1.000000e+13 1.000000e+13
0.000000e+00 -2.000000e-05 0.000000e+00 0.000000e+00
1.000000e-04 -2.000000e-05 0.000000e+00 0.000000e+00
1 1 3 2
1 3 4 2
2 1 5 6
2 1 6 3
5 1
6 1
1 -1

```

Figure E-2 Medici structure file

Appendix F: Using the MINIMOS 5 Interface

This appendix explains how to send TSUPREM-4 results to the MINIMOS 5 device simulator. It should be used in conjunction with the **SAVEFILE** input statement described [on page 3-71](#).

Overview of the TSUPREM-4 Interface to MINIMOS 5

The interface is file-oriented and unidirectional (from TSUPREM-4 to MINIMOS 5), and consists of three separate steps.

1. TSUPREM-4 writes a formatted file (i.e., ASCII or EBCDIC). This file contains a two-dimensional doping profile of the source area (and optionally the drain area) of the device being simulated. (See “[Step 1: Directing TSUPREM-4 to Generate a Formatted File](#)” on page F-2.)
2. Use a separate stand-alone FORTRAN program (named *mmatob*) to convert the formatted file to a FORTRAN binary file. (See “[Step 2: Converting the Formatted File to FORTRAN Binary](#)” on page F-6)
3. Execute MINIMOS 5, which reads in the binary doping file. (See “[Step 3: Running MINIMOS 5](#)” on page F-6)

Throughout this discussion, use the convention that the source is on the left of the channel and the drain is on the right of the channel. Also, you should keep in mind that TSUPREM-4 expects user-specified values in microns, while MINIMOS 5 expects centimeters.

All references to MINIMOS 5 refer to the program supplied by the Technical University of Vienna, Austria, Institute for Microelectronics, and the accompanying MINIMOS 5 User’s Guide, dated March 2, 1990.

Step 1: Directing TSUPREM-4 to Generate a Formatted File

Central to the preparation of the MINIMOS 5 output file is the concept of the MINIMOS 5 simulation region. This region is a subset of a simulated TSUPREM-4 device. While a TSUPREM-4 MOS device simulation may encompass an entire device, including the gate oxide, field oxide, metallization layers, etc., MINIMOS 5 expects to read a two-dimensional doping file encompassing only a portion of the full device. The **X.MIN**, **X.MAX**, **Y.MIN**, and **Y.MAX** parameters on the **SAVEFILE** statement are used to define a rectangular subset of the complete device. This subset is referred to as the MINIMOS 5 simulation region.

Defining the MINIMOS 5 Simulation Region

The MINIMOS 5 simulation region should be defined to include the source, drain, and channel of the MOS transistor; gate and field oxide regions would normally be excluded. The MINIMOS 5 simulation region must be deep enough to include the source and drain junction depths, because MINIMOS 5 expects to see a change in the sign of the net doping concentration along both the left and right edges of its simulation space (except when simulating SOI devices).

The following TSUPREM-4 statement defines the MINIMOS 5 simulation region and generates a formatted MINIMOS 5 file for the device shown in [Figure F-1](#) (from [Chapter 5, “NMOS LDD Process”](#) on page 5-2):

```
SAVEFILE MINIMOS5 OUT.FILE=MMDOPF FULL.DEV+
X.MIN=-2 X.MAX=2 Y.MIN=0.0174 Y.MAX=1.5+
X.MASK.S=-0.57 X.MASK.D=0.57
```

The parameter **MINIMOS5** specifies that the file to be produced is a formatted two-dimensional doping file, to be read by MINIMOS 5. **OUT.FILE=MMDOPF** specifies that the formatted doping file is named *MMDOPF*. Because this is the filename expected by the file conversion program *mmatob* (discussed later in this section), it is the recommended name to use when directing TSUPREM-4 to prepare a MINIMOS 5 doping file.

Referring to [Figure F-1](#), you see a complete device with source, channel, and drain regions, so the **FULL.DEV** parameter should be specified. If the TSUPREM-4 simulation encompasses only the source half of a device, use **HALF.DEV**. If the TSUPREM-4 simulation is for the drain half of a device, reflect the TSUPREM-4 structure to create a complete device and specify **FULL.DEV**. Because the drain region is assumed to be to the right of the channel, use the statement

```
STRUCTURE REFLECT LEFT
```

(preceding the **STRUCTURE MINIMOS5** statement) to reflect the device at its left edge (the center of the channel), thus creating a complete device.

Example 4 - Complete NMOS Structure

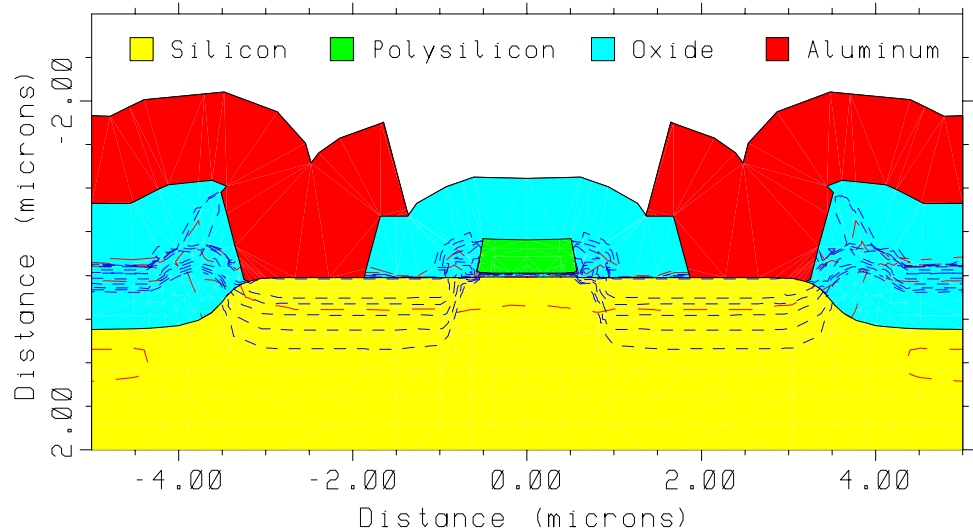


Figure F-1 NMOS structure to be transferred to MINIMOS 5

The **X.MIN**, **X.MAX**, **Y.MIN**, and **Y.MAX** parameters define the left, right, top, and bottom edges of the MINIMOS 5 simulation region, respectively. The assigned values for **X.MIN** and **X.MAX** are chosen by visual inspection of [Figure F-1](#) and need not be exact. **X.MIN** is roughly centered between the left-most field oxide and the gate, in an area where the impurity contours are essentially flat. If **X.MIN** is too far to the left, the field oxide region is included. **X.MAX** is assigned the corresponding value on the right side of the structure.

Similar to **X.MIN** and **X.MAX**, **Y.MAX** is determined based on visual inspection of [Figure F-1](#) and need not be exact. However, **Y.MAX** must be deep enough in the structure to include the source and drain junction depths, or MINIMOS 5 issues the error message

DOPING PROFILE INCONSISTENT

Y.MAX should also be deep enough in the structure to include the depletion region for the maximum bias to be applied, particularly if the dopant profile is still changing at that depth.

The values used for **X.MIN**, **X.MAX**, and **Y.MAX** are not critical, because if MINIMOS 5 needs a simulation region that is larger than the one defined by the doping file, MINIMOS 5 extends the profiles in the required direction. However, the value assigned to **Y.MIN** is critical, because MINIMOS 5 interprets **Y.MIN** as the gate oxide/silicon interface. Thus, **Y.MIN** has a physical significance and should be carefully determined. For TSUPREM-4 MOSFET simulations, the

gate oxide/silicon interface is typically found just below $y=0.0$. Therefore, as a first approximation, the default value (**Y.MIN**=0.0) could be used. Another approximation could be obtained from visual inspection of a two-dimensional plot. However, the most accurate value is obtained by using the following pair of TSUPREM-4 statements:

```
SELECT      Z=1
PRINT .1D  LAYERS  X.VALUE=<xchan>
```

where *<xchan>* is the x coordinate of the vertical slice along which values are to be printed. In a typical TSUPREM-4 MOSFET simulation (such as [Figure F-1](#)), the gate oxide/silicon interface is bowed slightly upwards and is closest to $y=0.0$ in the middle of the channel. For such cases, *<xchan>* in the preceding code fragment should be the x coordinate of the middle of the channel. The center of the channel for the device shown in [Figure F-1](#) is at $x=0.0$. Thus, the TSUPREM-4 statements

```
SELECT      Z=1
PRINT .1D  LAYERS  X.VALUE=0.0
```

are used to generate the following TSUPREM-4 output (results may vary slightly on your system):

| Num | Material | Top | Bottom | Thickness | Integral |
|-----|-------------|---------|----------|-----------|------------|
| 1 | oxide | -1.1111 | -0.4107 | 0.7004 | 7.0040e-05 |
| 2 | polysilicon | -0.4107 | -0.0245 | 0.3862 | 3.8615e-05 |
| 3 | oxide | -0.0245 | 0.0174 | 0.0419 | 4.1938e-06 |
| 4 | silicon | 0.0174 | 200.0000 | 199.9826 | 1.9998e-02 |

From this output, you see that the bottom of the oxide layer occurs at $y=0.0174$ microns, so **Y.MIN**=0.0174 is used. Also, note that the thickness of the insulating oxide is 0.0419 microns (419 Å). This value is used on the MINIMOS 5 **DEVICE** directive when specifying a value for the **TINS** parameter.

There is no need to specify **X.CHANNE** in this example, because the default value (midway between **X.MIN** and **X.MAX**) is appropriate. **X.CHANNE** is only needed if an asymmetric device is simulated and serves to divide the MINIMOS 5 simulation region into a source half and a drain half.

The two remaining parameters, **X.MASK.S** and **X.MASK.D**, are similar to **Y.MIN** in that they have a physical significance. **X.MASK.S** is the x coordinate of the mask edge in the source area of the MINIMOS 5 simulation region. MINIMOS 5 interprets this coordinate as the left edge of the gate electrode and treats **X.MASK.D** as the right edge of the gate electrode. Values for these parameters can be obtained by visual inspection of [Figure F-1](#). For more accuracy, use the following pair of TSUPREM-4 statements

```
SELECT      Z=Y
PRINT .1D  POLY  /OXIDE
```

to print the x and y coordinates of the interface between the polysilicon gate contact and the gate oxide. From this output (not shown), you find that **X.MASK.S**=-0.57 and **X.MASK.D**=0.57.

The **SGAP** and **DGAP** parameters on the MINIMOS 5 **DEVICE** directive are used to specify the gap between the gate and source contacts, and the gap between the gate and drain contacts, respectively. The TSUPREM-4 statements

```
SELECT      Z=Y
PRINT.1D    SILICON /OXIDE
```

can be used to print the x and y coordinates of the silicon/oxide interface, as an aid to determining **SGAP** and **DGAP** for the subsequent MINIMOS 5 run. Alternatively, the values could be approximated by visual inspection of [Figure F-1](#).

Notes on the Size of the MINIMOS 5 Simulation Region

The **X.MIN**, **X.MAX**, **Y.MIN**, and **Y.MAX** parameters determine the size of the MINIMOS 5 simulation region. If this region is large, containing many TSUPREM-4 grid points, the preparation of the MINIMOS 5 doping file could take several minutes. Additionally, the subsequent MINIMOS 5 run can take more execution time if the doping file defines a large simulation region. For these reasons, you should attempt to keep the MINIMOS 5 simulation region reasonably small when choosing **X.MIN**, **X.MAX**, **Y.MIN**, and **Y.MAX**. This also reduces the size of the doping file.

The standard version of MINIMOS 5 expects the doping file to contain from 10 to 80 horizontal grid lines and from 10 to 80 vertical grid lines. Normally, this requirement is satisfied automatically by TSUPREM-4 during the preparation of the doping file. However, certain choices for **X.MIN**, **X.MAX**, **Y.MIN**, and **Y.MAX** can result in a TSUPREM-4 warning message stating that there are too many (or too few) grid lines. Unless the version of MINIMOS 5 you run has been modified to accept expanded limits, you must modify the **X.MIN**, **X.MAX**, **Y.MIN**, or **Y.MAX** values as advised in the TSUPREM-4 warning message, and regenerate the doping file. Otherwise, when MINIMOS 5 attempts to read the doping file, it issues the error message

```
DOPING FILE FORMAT ERROR
```

and terminate execution. You can also modify the number of grid lines by specifying the **DX.MIN** and/or **DY.MIN** parameters. These change the minimum grid spacings in the x and y directions from their default values of $\min((\mathbf{X.MAX} - \mathbf{X.MIN})/80, 0.01)$ and $\min((\mathbf{Y.MAX} - \mathbf{Y.MIN})/80, 0.01)$, respectively.

Nonplanar Oxide Regions in MINIMOS 5

As stated in the *MINIMOS 5 User's Guide*, MINIMOS 5 can model nonplanar oxide regions. This is accomplished by including the **GEOMETRY** directive (and

associated parameters) in the MINIMOS 5 input command file. However, the doping file (as dictated by MINIMOS 5) makes no provision for the inclusion of geometry information. As discussed, the doping file uses a single coordinate (**Y.MIN**) to specify the gate oxide/silicon interface, implying a planar device. Hence, TSUPREM-4 cannot automatically export nonplanar geometry information to MINIMOS 5 by including it in the doping file—the MINIMOS 5 user must approximate the nonplanarities of the device simulated in TSUPREM-4 by hand, attempting to duplicate the TSUPREM-4 geometry in MINIMOS 5 by using the **GEOMETRY** directive. Fortunately, many devices (such as the MOSFET in [Figure F-1](#)) are approximately planar, and the MINIMOS 5 **GEOMETRY** directive is not required to simulate the intrinsic device operation. In such cases, assign **Y.MIN** to the top-most point of the silicon as discussed earlier. Because the gate oxide/silicon interface is not absolutely flat, the left and right portions of the top of the MINIMOS 5 simulation region contain very thin layers of oxide. To correct this, TSUPREM-4 automatically ignores the impurity concentration in the oxide and extends the doping profile of the underlying silicon upward (using the concentration at the oxide/silicon interface). In general, TSUPREM-4 ignores the dopants in any nonsilicon material within the MINIMOS 5 simulation region, using instead the doping of the underlying silicon.

Step 2: Converting the Formatted File to FORTRAN Binary

TSUPREM-4 writes a formatted (i.e., ASCII or EBCDIC) file for MINIMOS 5. However, MINIMOS 5 requires a FORTRAN binary doping file. The formatted file written by TSUPREM-4 is converted to FORTRAN binary by executing the stand-alone FORTRAN program named *mmatob*. This step is needed because of incompatibilities between binary files written by C programs (such as TSUPREM-4) and FORTRAN programs (such as MINIMOS 5) on some computer systems.

The current version of *mmatob* is executed by simply typing

```
mmatob
```

followed by a carriage return. Note that *mmatob* expects to read a formatted file named *MMDOPF* and write a binary file named *MMDOPB*. The easiest way to assure that the formatted file is named *MMDOPF* is to specify **OUT.FILE=MMDOPF** on the TSUPREM-4 **SAVEFILE** statement used to generate the MINIMOS 5 file. Alternatively, rename or copy the formatted file to *MMDOPF* before executing *mmatob*. Any existing file named *MMDOPB* is overwritten when *mmatob* is executed.

Step 3: Running MINIMOS 5

MINIMOS 5 reads the binary doping file from FORTRAN logical unit 20. Similarly, the input command file must be available on FORTRAN unit 15. Refer to

your MINIMOS 5 documentation for instructions on how to associate these input files with the correct FORTRAN logical unit numbers.

Figure F-2 shows a listing of a sample MINIMOS 5 input command file (*EX2D.INP*, included with standard versions of MINIMOS 5).

```
EXAMPLE FOR CLASSICAL TWO-DIMENSIONAL MODEL (EX2D.INP)
DEVICE      CHANNEL=N GATE=NPOLY TINS=150.E-8 W=1.E-4 L=0.85E-4
BIAS        UD=4. UG=1.5
PROFILE     NB=5.2E16 ELEM=AS DOSE=2.E15 TOX=500.E-8 AKEV=160.
+           TEMP=1050. TIME=2700
IMPLANT     ELEM=B DOSE=1.E12 AKEV=12 TEMP=940 TIME=1000
OPTION      MODEL=2-D
OUTPUT      ALL=YES
END
```

Figure F-2 Listing of MINIMOS 5 command file *EX2D.INP*

This file uses the **PROFILE** and **IMPLANT** directives to generate a doping profile. To convert *EX2D.INP* so that it reads an external two-dimensional doping file, the **PROFILE** statement is changed to

PROFILE FILE=2-D

The **IMPLANT** directive, used in *EX2D.INP* to specify a channel implantation, is removed. In addition, the parameters and values specified on the **DEVICE** statement are modified as needed to match the geometry of the device simulated in TSUPREM-4. The resulting modified MINIMOS 5 input command file is shown in Figure F-3.

```
EXAMPLE FOR CLASSICAL TWO-DIMENSIONAL MODEL (READS 2-D DOPING FILE)
DEVICE      CHANNEL=N GATE=NPOLY TINS=419E-8 W=1.E-4 L=1.14E-4
+           DGAP=1.87E-4 SGAP=1.87E-4
BIAS        UD=4. UG=1.5
PROFILE     FILE=2-D
OPTION      MODEL=2-D
OUTPUT      ALL=YES
END
```

Figure F-3 Listing of MINIMOS 5 command file *EX2D.INP*, modified to read doping profiles produced by TSUPREM-4

Recall that MINIMOS 5 expects dimensions in centimeters, not microns. To use MINIMOS 5 to simulate a nonplanar device, specify the **GEOMETRY** directive, as discussed earlier, in “Nonplanar Oxide Regions in MINIMOS 5” on page F-5.

Notes on Using MINIMOS 5

Particular care must be taken when specifying the **L** parameter (gate electrode length) on the MINIMOS 5 **DEVICE** directive. The value assigned to **L** should correspond to the distance between the values assigned to TSUPREM-4 parameters **X.MASK.S** and **X.MASK.D**. For example, if **X.MASK.S**=-0.57 (microns) and **X.MASK.D**=0.57 (microns), the distance between them is 1.14 microns, so **L**=1.14e-4 (centimeters). In this case, specifying a value of **L** greater than 1.14e-4 centimeters has the effect of stretching the channel, and should not cause problems in MINIMOS 5. However, a value of **L** substantially smaller than 1.14e-4 centimeters could cause the source and drain profiles to overlap, effectively eliminating the channel. MINIMOS 5 responds with the error message

DOPING PROFILE INCONSISTENT

and terminates execution.

Interpreting Error Messages

The warning and error messages issued by TSUPREM-4 during the preparation of the doping file are clear and fairly detailed, recommending corrective action when necessary.

The following error messages from MINIMOS 5 may be encountered:

DOPING PROFILE INCONSISTENT

MINIMOS 5 issues this message for a variety of error conditions. A likely cause is that the **L** parameter on the MINIMOS 5 **DEVICE** directive is too small, causing the source and drain profiles to overlap and effectively eliminating the channel. Correct the error by using a value of **L** that is at least as large as the difference between the **X.MASK.S** and **X.MASK.D** parameters, as described above. The doping file does not need to be regenerated.

Another cause for the above message is that the MINIMOS 5 simulation region, as defined in TSUPREM-4, was not deep enough to include the junction depths in both the source and drain regions. Correct the error by using a larger value for **Y.MAX** on the TSUPREM-4 **SAVEFILE** statement. In this case, the doping file needs to be regenerated.

Lastly, to avoid the above message when using TSUPREM-4 and MINIMOS 5 to simulate SOI devices, the MINIMOS 5 **DEVICE** directive must include the **BULK**, **FILM**, and **KBULK** parameters.

When too many (or too few) horizontal or vertical grid lines are present in the doping file, the following message can appear:

DOPING FILE FORMAT ERROR

TSUPREM-4 should have issued a warning message, describing the problem and recommending corrective action. However, TSUPREM-4 went ahead and prepared a doping file after issuing the warning, in case your version of MINIMOS 5 has been modified to accept expanded limits. Correct the error by using different values for **X.MIN**, **X.MAX**, **Y.MIN**, or **Y.MAX** in TSUPREM-4, and regenerate the doping file.

Glossary

This glossary contains terms frequently used in the *TSUPREM-4 User's Manual*. A list of acronyms is included as the last section in the Glossary. For references to more information about a term, see the Index.

A

| | |
|-----------------------------------|--|
| activation of dopants | Movement of dopant atoms onto substitutional sites on the silicon lattice where they are electrically active. |
| adaptive gridding | Automatic adaptation of the simulation mesh by the program to improve solution accuracy. |
| amorphization | The complete disruption of the silicon lattice structure caused by ion implantation. The amorphized region will recrystallize (through epitaxial regrowth) during the next high-temperature step, but residual damage may remain, especially at the edges of the amorphized layer. |
| analytical oxidation model | A model that uses empirical descriptions of the two-dimensional oxide shape to predict the results of local oxidation steps. |
| automatic mesh generation | Generation of the simulation mesh automatically by the program based on general guidelines supplied by the user. |

C

| | |
|-------------------|---|
| channeling | Tendency of ions to travel along open directions in the silicon lattice during implantation. Channeling may cause implanted profiles to be deeper than expected, depending on the tilt and rotation of the wafer related to the ion beam. The effect is included in the dual Pearson implant tables and in the Monte Carlo implant model. |
| clustering | The formation of groups of atoms whose properties are different from the properties of the individual atoms within the group. Clustering usually refers to the clustering of dopant atoms to form immobile and electrically inactive complexes. (See interstitial clustering.) |

D

| | |
|--------------------------|---|
| deposition | One of the basic fabrication steps used for IC processing. TSUPREM-4 contains a simple model for deposition of conformal layers. If more advanced models are needed, TSUPREM-4 can be used with Taurus-Topography. |
| diffusion | Spreading of impurities due to random motion at high temperatures. The term also refers to any process step in which diffusion occurs. TSUPREM-4 includes detailed physical models for diffusion and other processes, such as oxidation and silicidation, that occur during high temperature fabrication steps. |
| damage annealing | Removal of implantation damage with high temperatures. |
| dislocation loops | A class of extended defect in the silicon lattice structure. The ED-AAM models end-of-range dislocation loops produced by amorphizing ion implants. |
| dopant | An impurity added to a structure to control its electrical properties. |

E

| | |
|---------------------------|---|
| epitaxial growth | Growth of single crystal silicon on top of an existing silicon structure. TSUPREM-4 includes a model for epitaxial growth on planar substrates. |
| epitaxial regrowth | Recrystallization of a silicon region amorphized during ion implantation. |

G

| | |
|-------------|-------------|
| grid | (See mesh.) |
|-------------|-------------|

I

| | |
|--------------------------------|--|
| implant damage | Damage to the silicon lattice structure caused by ion implantation. |
| impurity | Any atom other than silicon in a silicon structure. Often used to refer to a dopant. |
| interstitial | A type of point defect consisting of an extra silicon atom not on the lattice site. Interstitials promote diffusion by pairing with dopants. |
| interstitial clustering | Clustering of interstitials. Clustered interstitials are assumed to be immobile and do not promote the diffusion of impurities. In TSUPREM-4, the clustering model is used to simulate the effects of {311} defects. |
| interstitial trap model | A model describing the absorption and release of interstitials at a trapping site. |
| ion pairing | Coupling of dopant atoms of opposite charge to form an immobile pair. |
| ion implantation | Introduction of dopants into a wafer by a beam of accelerated ions. |

L

local oxidation Oxidation of a portion of the silicon surface, achieved by masking the portion that is not oxidized with a material such as silicon nitride.

M

mesh The array of points within the structure at which solution values are numerically calculated. (The term is used interchangeably with *grid*.)

Monte Carlo implant model A physics-based model that uses statistical techniques to calculate impurity and damage profiles produced by ion implantation.

N

numerical oxidation model A model that uses the numerical solution of the physical equations for oxide growth and flow to predict the results of arbitrary oxidation steps.

O

oxidation Creation of silicon dioxide by reaction of silicon or polysilicon with oxygen. Oxidation is modeled in TSUPREM-4 whenever diffusion occurs in an oxidizing ambient.

point defects A localized defect (as opposed to an extended defect) in a silicon lattice structure. In TSUPREM-4, the term *point defects* refers to interstitials (i.e. silicon self-interstitials) and vacancies.

polycrystalline model A model for polycrystalline materials used primarily for modeling polycrystalline silicon. It includes models for grain growth and the effect of grain boundaries on dopant diffusion.

P

PMEI Physical model and equation interface. PMEI is intended to be an universal language for scripting models and equations for semiconductor process and device simulation.

S

segregation Transport of an impurity across an interface between two materials due to a difference in chemical potential. This transport results in an equilibrium ratio of impurity concentration across a material boundary.

| | |
|-------------------------|--|
| silicidation | Interaction of silicon to form a metal silicide. Modeled in TSUPREM-4 by the DIFFUSION statement whenever necessary materials are present in the structure. |
| solid solubility | The concentration at which dopant atoms precipitate, becoming immobile and electrically active. |
| stress | TSUPREM-4 includes models for calculating the physical stresses produced during fabrication. The stress history model simulates the stresses produced by oxidation, thermal mismatch between materials, intrinsic strain in deposited layers, and surface tension. |

T

| | |
|------------------|---|
| TSUPREM-4 | The name originated from a combination of TMA (now the TCAD Business Unit of <i>Avant!</i>) and SUPREM-IV, the Stanford University Process Engineering Models. |
| TED | Transient-Enhanced Diffusion. Enhanced diffusion caused by implant damage. |

U

| | |
|--------------|--|
| USEIT | User-specified equation interface is an optionally available capability for TSUPREM-4 that helps to quickly develop and evaluate process models. |
|--------------|--|

Acronyms

| | |
|--------------|---------------------------------------|
| LDD | Lightly Doped Drain |
| MOS | Metal-Oxide Semiconductor |
| NMOS | N-channel Metal-Oxide Semiconductor |
| OED | Oxidation-Enhanced Diffusion |
| PMEI | Physical Model and Equation Interface |
| TED | Transient-Enhanced Diffusion |
| USEIT | User-Specified Equation Interface |

Index

Symbols

() 3-4
<> 3-4
{ } 3-4
| 3-4

Numerics

1D bipolar example

completing the active region simulation 4-9
DEPOSITION statement 4-5, 4-6
EPITAXY statement 4-5
ERFC oxidation model 4-4
ETCH statement 4-5
final structure 4-11
IMPLANT statement 4-5
initial active region 4-2
INITIALIZE statement 4-3
input file sequence 4-2
LABEL statement 4-6
mesh generation 4-2
model selection 4-4
overview 4-2
PD.FERMI point defect model 4-4
PD.TRANS point defect model 4-4
PLOT.1D statement 4-6, 4-8
plotting the results 4-6
point defect model 4-4
printing layer information 4-8
processing steps 4-5
SAVEFILE statement 4-6
saving the structure 4-6
SELECT statement 4-6, 4-7
VERTICAL oxidation model 4-4

1D simulation of simple structures

overview 2-12
tutorial example of 4-2

2D diffusion with point defects

automatic grid generation 4-19
contour of boron concentration 4-20
field implant 4-19
grid plot 4-20

lateral distribution of point defects 4-25
oxidation 4-19
shaded contours of interstitial concentration 4-25
using the **FOREACH** statement 4-23
vertical distribution of point defects 4-24
5-stream diffusion model 6-48

A

about the manual xxxvi
absorption of interstitial traps 2-37
activation of dopants 2-129
ACTIVE 6-25
active region simulation
LDD implant 5-8
modeling polysilicon 5-7
oxide spacer and source/drain implant 5-9
plots 5-11
source/drain contacts 5-9
adaptive grid parameters, default coefficients A-25
adaptive gridding
advanced examples of 5-4, 5-28
analytical implant models 3-200
disable 3-200
overview 2-10
tutorial example of 4-4
adding expressions to equations 6-21
advanced examples
CMOS process 5-30
DMOS power transistor 5-39
MOSFET with self-aligned silicides 5-50
NMOS LDD process 5-2
overview 5-1
poly-buffered LOCOS 5-27
polysilicon emitter study 5-54
SOI MOSFET 5-45
trench implant simulation 5-18
advanced input, physical model and equation interface 6-3

advantages of intermediates 6-24

AMBIENT 3-207

additional notes 3-222
chlorine 3-220
coefficients 3-220
coefficients chlorine examples 3-220
COMPRESS model 3-218
description 3-217
ERFC model 3-218
ERFG model 3-218
examples 3-222
INITIAL parameter, 1D bipolar example 4-4
orientation 3-221
oxidation models 3-217
oxidation models, polycrystalline silicon 3-218
oxidizing species 3-221
parameter dependencies 3-221
specified material 3-222
specified units 3-222
stress dependence 3-219
VERTICAL model 3-218
VISCOELA model 3-218
VISCOUS model 3-219

ambient gas

chlorine example 2-16
chlorine, coefficient tables 2-16
defaults 2-15
oxidation of materials 2-14
pressure 2-14
specifying characteristics 2-14

amorphization, damage dechanneling 2-97

amorphous implant calculation 2-91

electronic stopping 2-92
electronic stopping at high energies 2-93
ion beam width 2-94
nuclear stopping 2-92
total energy loss and ion deflection 2-93

analytic damage model 2-86

damage distribution calculations 2-86

recommended usage and limitations 2-87

analytic ion implant models 2-77

- analytic damage model 2-86
- depth-dependent lateral distribution 2-85
- dose-dependent implant profiles 2-81
- dual Pearson distribution 2-81
- Gaussian distribution 2-80
- implant moment tables 2-78
- implanted impurity distributions 2-77
- lateral distribution 2-85
- multilayer implants 2-84
- Pearson distribution 2-80
- tilt and rotation tables 2-84
- wafer tilt and rotation 2-85

analytical oxidation models 2-48

- ERF1** model 2-52
- ERF2** model 2-52
- ERFC** model 2-51
- ERFG** model 2-53
- linear rate 2-48
- overview 2-48
- oxide growth rate 2-48
- parabolic rate 2-49
- thin regime 2-48
- usage 2-50

annealing

- damage 2-97
- in situ* 2-97
- self-annealing 2-97

ANTIMONY 3-284

- additional notes 3-289
- description 3-288
- examples 3-289

application of user-specified models 6-24

Arora mobility model 2-123

Array command 6-9

- DefaultValue** parameter 6-10
- MinIndex** parameter 6-9
- Name** parameter 6-9
- Object** parameter 6-10
- Size** parameter 6-9
- Type** parameter 6-9

array values

- effect due to redefinition 3-35
- getting array values 3-35
- redefinition of array elements 3-35

ARSENIC 3-290

- additional notes 3-295
- description 3-294
- examples 3-295

ARSENIC parameter, **EPITAXY** statement

- 1D bipolar example 4-6

ASSIGN 3-25

- additional **ASSIGN** notes 3-30

ASSIGN and optimization 3-30

ASSIGN with mathematical expressions 3-29

assign with mathematical expressions, using **N.EXPRESS** 3-29

description 3-28

expansion of **ASSIGNED** variable 3-30

overriding variables 3-31

reading the external data file 3-31

varying during statement looping 3-29

authorization file *s4auth* 1-10

automatic grid generation

- custom defaults 2-6
- overview 2-5
- parameters, default coefficients A-25
- tutorial examples of 4-3, 4-19

AXES parameter, **PLOT.1D** statement

- 1D bipolar example 4-7

AXES parameter, **PLOT.2D** statement

- local oxidation example 4-18

B

BF2 implant 2-86

BF2 implant profile *Fig. 2-83*

binary scattering theory 2-87

- Coulomb potential 2-89
- dimensionless form 2-88
- energy loss 2-88
- scattering angle 2-88
- universal potential 2-90

bipolar emitter structure and as-implanted arsenic profiles, as plotted using *s4ex11c.inp Fig. 5-56*

Boltzmann statistics 2-121

BORON 3-296

- additional notes 3-301
- description 3-300
- examples 3-301

boron diffusion

- enhancement due to fluorine 2-67
- enhancement in oxides 2-65
- enhancement in thin oxides 2-65

BORON parameter, **INITIALIZE** statement

- 1D local oxidation example 4-19

boron-interstitial clustering model with small clusters 6-41

BOTTOM parameter, **PLOT.1D** statement

- local oxidation example 4-24

BOUNDARY 3-60

- description 3-60
- example 3-61
- limitations 3-61

boundary conditions, ion implantation

2-103

BoundaryCondition command 6-11

- Material1** parameter 6-12
- Material2** parameter 6-12
- Type** parameter 6-12

built-in keywords 6-27

C

capacitance calculation 2-125

- DC method 2-126
- MOS capacitances 2-126

Caughey mobility model 2-124

C.COMPRE parameter, **PLOT.2D** statement

- local oxidation example 4-16

CENTER parameter, **LABEL** statement

- CMOS process example 5-37

C.GRID parameter, **PLOT.2D** statement

- local oxidation example 4-14

channel and source/drain doping profiles for SOI MOSFET *Fig. 5-49*

channel doping profile for NMOS transistor *Fig. 5-35*

Channeling 3-111

channeling 2-94

charge state fractions 2-33

chemical predeposition 2-16

CLEAR parameter, **PLOT.1D** statement

- 1D bipolar example 4-7

CLEAR parameter, **PLOT.2D** statement

- local oxidation example 4-26

clustering and solid solubility, default coefficients A-7

clustering model 2-27

CM parameter, **LABEL** statement

- 1D bipolar example 4-7

CMOS process example

- 0.8 micron device 5-37
- 1.2 micron device 5-38
- arsenic profiles in gate 5-37
- channel doping plot 5-34
- COMPRESS** model 5-34
- contacts 5-34
- end of main loop 5-35
- END** statement 5-35
- final mesh 5-37
- FOREACH** loop 5-30, 5-35
- FOREACH** statement 5-34
- INITIALIZE** statement 5-35
- lightly doped drain structure 5-34
- main loop 5-32
- mesh generation 5-32
- models 5-34
- overview 5-30
- PD.TRANS** model 5-34
- plotting results 5-35
- processing 5-34

SAVEFILE statement 5-35
 saving the structure 5-35
VERTICAL oxidation model 5-34
 cobalt silicide model C-1
 coefficient default values A-1
COLOR 3-150
 description 3-151
 examples 3-151
 plot device selection 3-151
COLOR parameter, **PLOT . 1D** statement
 1D bipolar example 4-7
 command input files 1-4
COMMENT 3-8
 description 3-8
 examples 3-8
 notes 3-8
 compatibility with TSUPREM-4 version
 1999.4 C-9
 accuracy issues C-10
 performance issues C-11
 complete NMOS structure, plotted by
 input file *s4ex4c.inp* Fig. 5-13
 completing the active region simulation
 4-9
 field oxidation 4-9
 reading a saved structure 4-9
COMPRESS model 2-57
 boundary conditions 2-58
 compressible viscous flow 2-57
 material parameters 2-58
 recommended usage 2-58
 compressible viscous flow 2-57
 concentration dependence 2-55
 concentration of point defects
 vs. depth, as plotted by input file
 s4ex2b.inp Fig. 4-24
 vs. width, as plotted by input file
 s4ex2b.inp Fig. 4-25
Condition function 6-16
CONTOUR 3-148
 additional notes 3-149
 description 3-148
 example 3-149
 line type and color 3-149
 contours of boron
 after analytic implant Fig. 5-21
 after Monte Carlo implant Fig. 5-25
 concentration produced by input file
 s4ex2b.inp Fig. 4-23
 contours of hydrostatic pressure
 final poly-buffered LOCOS
 structure Fig. 5-29
 plotted by statements in input file
 s4ex2a.inp Fig. 4-18
 contours of interstitial concentration, as
 plotted by input file *s4ex2b.inp* Fig.
 4-26
 contours of net doping for 1-micron and
 2-micron emitter stripes Fig. 5-58

contours of total arsenic concentration
 and poly grain size after RTA Fig.
 5-58
CPULOG 3-46
 description 3-46
 examples 3-46
 limitations 3-46
 creating the initial structure
 adaptive gridding 5-4
 masking information 5-4
 setting the grid density 5-3
 creating the test structure
 automatic grid generation 4-29
 outline of example 4-29
CRYSTAL parameter, **IMPLANT**
 statement
 trench implant simulation example
 5-23, 5-25
 crystalline implant model 2-94
 BF2 implantation 2-97
 channeling 2-94
 damage annealing 2-97
 damage dechanneling 2-96
 lattice damage 2-96
 lattice temperature 2-96
 number of ions 2-97
C . TENSIO parameter, **PLOT . 1D**
 statement
 local oxidation example 4-16

D

damage annealing 2-97
 damage dechanneling
 amorphization 2-97
 Monte Carlo implant 2-97
 DC method 2-126
DEEP parameter, **ELECTRICAL**
 statement
 NMOS LDD process example 5-15
 default coefficient notes A-32
 default coefficient tables
 adaptive grid parameters A-25
 automatic grid generation
 parameters A-25
 clustering and solid solubility A-7
 electrical parameters A-18
 impurity parameters A-2
 material coefficients A-21
 Monte Carlo implant A-24
 notes A-32
 numerical methods A-24
 oxidation A-10
 point defect parameters A-8
 polysilicon grain segregation A-6
 references A-27
 default file names 1-3
DEFINE 3-42

description 3-42
 examples 3-42
 format and syntax 3-42
 usage notes 3-43
DEPOSITION 3-90
 additional notes 3-94
 description 3-92
 examples 3-94
 photoresist 3-93
 polycrystalline materials 3-93
 structure bottom 3-92
 deposition
 incorporation of impurities 2-106
 layer thickness 2-106
 overview 2-105
 photoresist type 2-106
 deposition parameters 3-93
DEVELOP 3-97, 3-98
 description 3-97
 example 3-97
DEVICE parameter, **OPTION** statement
 1D bipolar example 4-6
 device structure specification, input
 statements
 BOUNDARY 3-60
 ELECTRODE 3-87
 ELIMINATE 3-57
 INITIALIZE 3-64
 LINE 3-55
 LOADFILE 3-68
 MASK 3-82
 MESH 3-50
 PROFILE 3-84
 REGION 3-62
 SAVEFILE 3-71
 STRUCTURE 3-77
 diagnostic output file *s4dia* 1-6
DIFFUSION 3-115
 ambient gas 3-118
 ambient gas parameters 3-118
 description 3-118
 DRY02 parameter, 1D bipolar
 example 4-6
 examples 3-120
 oxidation limitations 3-119
 reflow 3-119
 STEAM parameter, 1D bipolar
 example 4-5
 THICKNES parameter, 1D bipolar
 example 4-5
 diffusion
 activation of impurities 2-24
 along grain boundaries 2-113
 anisotropic 2-113
 clustering model 2-27
 diffusion of impurities 2-17
 DIFFUSION statement 2-13
 grain boundary structure 2-113
 grain interiors 2-112

impurities 2-105
 impurity fluxes 2-17
 injection and recombination of point defects at interfaces 2-37
 interstitial traps 2-41
 modeling of diffusivities 2-18
 overview 2-13
 point defects 2-32
 segregation of impurities 2-28
 solution of equations 2-16
 temperature 2-14
 tutorial example of 4-18
 diffusion along boundaries 6-23
 diffusion enhancement in thinner oxides 6-45
 diffusion in **PD.TRANS** 6-38
 diffusion of point defects
 absorption of interstitial traps 2-37
 charge state fractions 2-33
 equations 2-33
 equilibrium concentrations 2-32
 interstitial and vacancy diffusivities 2-34
 net recombination rate of interstitials 2-35
 reaction of pairs with point defects 2-35
 diffusivities 2-20
 dislocation loop model 2-126, 6-36
 creation of 2-128
 effects of 2-129
 DMOS power transistor example
 after p-type body diffusion *Fig. 5-42*
 COMPRESS oxidation model 5-40
 mesh generation 5-40
 overview 5-39
 PD.TRANS point defect model 5-44
 processing DMOS structure 5-40
 summary 5-44
 documentation and control, input statements
 ASSIGN 3-25
 COMMENT 3-8
 CPULOG 3-46
 DEFINE 3-42
 ECHO 3-38
 FOREACH/END 3-16
 HELP 3-47
 IF/ELSEIF/ELSE/IF.END 3-23
 INTERACTIVE 3-12
 INTERMEDIATE 3-32
 L.MODIFY 3-22
 LOOP/L.END 3-18
 OPTION 3-39
 PAUSE 3-14
 RETURN 3-10
 SOURCE 3-9
 STOP 3-15
 UNDEFINE 3-45

dopants
 activation of 2-129
 advanced tutorial examples of 5-20, 5-23
 diffusion of atoms equations 2-17
 dislocation loop model 2-126
 paired fractions 2-22
 polycrystalline materials 2-112
 solid solubility 2-24
 transient clustering model 2-129
DOSE parameter, **IMPLANT** statement
 trench implant simulation example 5-23
DRYO2 parameter, **DIFFUSION** statement
 1D bipolar example 4-6
 dual Pearson distribution 2-81
DY parameter, **DEPOSITION** statement
 SOI MOSFET example 5-46
DY.OXIDE parameter, **METHOD** statement
 local oxidation example 4-14
DY.SURF parameter, **MESH** statement
 CMOS process example 5-32

E

ECHO 3-38
 description 3-38
 examples 3-38
 effective range model 2-84
 electric field 2-19
ELECTRICAL 3-174
 additional notes 3-183
 description 3-179
 E.RVCAP parameter, NMOS LDD process example 5-16
 examples 3-180
 files and plotting 3-179
 JUNCTION parameter, NMOS LDD process example 5-16
 MOSCAP parameter, NMOS LDD process example 5-14, 5-15
 optimization examples 3-181
 quantum effect in CV plot 3-181
 RESISTAN parameter, NMOS LDD process example 5-14
 THRESHOLD parameter, NMOS LDD process example 5-14
 electrical calculations
 Arora mobility model 2-123
 Boltzmann statistics 2-121
 carrier mobility 2-123
 Caughey mobility model 2-124
 Fermi-Dirac statistics 2-121
 ionization of impurities 2-121
 overview 2-120
 Poisson's equation 2-120
 solution methods 2-122
 tabular form 2-123
 electrical characteristics, plotted by input file *s4ex4d.inp Fig. 5-17*
 electrical data output files 1-8
 electrical extraction
 MOS capacitance 5-15
 plotting results of electrical extraction 5-17
 source/drain junction capacitance 5-16
 threshold voltage 5-14
 electrical parameters, default coefficients A-18
ELECTRODE 3-87
 additional notes 3-88
 description 3-87
 examples 3-88
ELIMINATE 3-57
 description 3-57
 examples 3-58
 initial structure generation 3-58
 overlapping regions 3-58
 reducing grid nodes 3-58
 enable/disable user-specified models 3-199
ENERGY parameter, **IMPLANT** statement
 trench implant simulation example 5-23
 enhancements
 error corrections C-8
 improved accuracy, flexibility, and ease of use C-5
 miscellaneous improvements C-8
 version 2000.2 C-1
 environment variables 1-3
 epitaxial growth
 layer thickness 2-105
 modeling 2-104
 overview 2-104
 epitaxial regrowth 2-117
EPITAXY 3-121
 ARSENIC parameter, 1D bipolar example 4-6
 description 3-123
 example 3-123
 mobility tables 3-123
 SPACES parameter, 1D bipolar example 4-5
 TEMPERAT parameter, 1D bipolar example 4-6
 TIME parameter, 1D bipolar example 4-6
EPITAXY parameter, **TEMPERAT** statement
 1D bipolar example 4-6
EQUATION 3-204
 adding expressions to equations 6-21
 diffusion along boundaries 6-23
 flux at interfaces 6-22

initialization 6-20
 new solution variable 6-20
 solving method 6-20
 update solution 6-23
Equation command 6-6
Array parameter 6-7
Boundary parameter 6-7
DopantCharge parameter 6-7
ErrorLimit parameter 6-7
Expression parameter 6-7
Loop parameter 6-7
Material parameter 6-6
Material2 parameter 6-7
Model parameter 6-7
Name parameter 6-6
Parameter parameter 6-7
Scaling parameter 6-7
VariableName parameter 6-6
 equilibrium concentrations 2-32
ERF1 model 2-52
 initial structure 2-51
 parameters 2-51
ERF2 model 2-52
 initial structure 2-51
 parameters 2-51
ERFC model 2-51
ERFG model 2-53
 initial structure 2-51
 parameters 2-51
 recommended usage 2-53
ERR.FAC parameter, **MESH** statement
 NMOS LDD process example 5-4
ERR.FAC parameter, **METHOD** statement
 1D bipolar example 4-4
 error corrections C-8
 error messages 1-3
E.RVCAP parameter, **ELECTRICAL**
 statement
 NMOS LDD process example 5-16
ETCH
 description 3-100
 examples 3-102
 generate simulation structure 3-101
 removing regions 3-100
 etching
 complex structures 2-111
 defining the region 2-107
 examples 2-109
 material removal 2-108
 overview 2-107
 simple structure 2-109
 structure with overhangs 2-110
 trapezoidal model 2-108
 tutorial example 4-5
 examples
 mask data file *Fig. D-2*
 sensitivity plot for target with
 multiple data *Fig. 3-21*

trapezoidal etch model *Fig. 2-110*
Exist function 6-16
EXPOSE 3-95
 description 3-95
 example 3-96
 OFFSET parameter 3-95
Expression command 6-8
 Material parameter 6-8
 MaterialType parameter 6-8
 extended defects AAM
 dislocation loop model 2-126
 overview 2-126
 transient clustering model 2-129
 extended defects Advanced Applications
 Module
 see extended defects AAM
EXTRACT 3-160
 description 3-165
 error calculation 3-168
 examples 3-169
 extraction procedure 3-166
 file formats 3-168
 optimization examples 3-171
 solution variables 3-166
 targets for optimization 3-168
 extract output files 1-8

F

Fermi-Dirac statistics 2-121
 field isolation simulation
 displaying the plot 5-5
 file specification 1-3
 files
 command input 1-4
 default names 1-3
 environment variables 1-3
 input 1-4
 library 1-8
 mask data 1-5
 profile 1-5
 terminal output 1-5
 types 1-3
 final grid
 LDD NMOS example, produced by
 input files *s4ex4a.inp* and
 s4ex4b.inp Fig. 5-11
 SOI MOSFET *Fig. 5-48*
 final mesh
 0.8 micron NMOS structure *Fig.*
 5-37
 DMOS simulation, showing the
 result of adaptive gridding *Fig. 5-44*
 final profiles produced by input file
 s4ex1b.inp Fig. 4-11
 final structure
 0.8 micron, NMOS structure, plotted
 by *s4ex7b.inp Fig. 5-37*

1.2 micron, NMOS, plotted by input
 file *s4ex7c.inp Fig. 5-38*
 after removal of remaining titanium
Fig. 5-53
 DMOS power transistor, produced
 by input file *s4ex8.inp Fig. 5-44*
 NMOS plotted by file *s4ex4b.inp*
Fig. 5-12
 showing contours of net doping for
 SOI MOSFET *Fig. 5-48*
 fine grids, final check 5-1
 first part of input file
 s4ex10.inp, NMOS transistor
 processing *Fig. 5-50*
 s4ex11c.inp, for plotting structure
 and contours of as-implanted
 arsenic concentration *Fig. 5-57*
 s4ex2a.inp, for determining LOCOS
 shape *Fig. 4-12*
 s4ex2b.inp, showing processing
 steps *Fig. 4-19*
 s4ex3.inp, showing processing and
 plotting using **PD.FERMI** point
 defect model *Fig. 4-27*
 s4ex4a.inp, setting up grid for
 simulating NMOS process *Fig. 5-3*
 s4ex4b.inp, showing polysilicon gate
 formation *Fig. 5-8*
 s4ex4d.inp, showing threshold
 voltage extraction *Fig. 5-15*
 s4ex5.inp, showing grid setup *Fig.*
 5-18
 s4ex6.inp, Poly-buffered LOCOS
 process *Fig. 5-27*
 s4ex7a.inp, to set up grid for
 simulating CMOS process *Fig.*
 5-30
 flat structure syntax 6-5
FLOW parameter, **PLOT.2D** statement
 local oxidation example 4-15
 flux at interfaces 6-22
 flux at interfaces *Fig. 6-22*
FOREACH/END 3-16
 description 3-16
 examples 3-16
 notes 3-17
 format
 input statements 3-2
 mask data files D-1
 saved structure files E-1
 statement description 3-4
 functions and operators 6-14

G
 gas
 see **DIFFUSION** statement, ambient
 gas
 Gaussian distribution 2-80
 grain size model 2-115

- concentration dependence 2-116
- grain growth 2-116
- grain surface energy 2-117
- initial grain size 2-115
- segregation drag 2-117
- graphical output files 1-8
- graphics devices
 - default device B-6
 - determining which to use B-1
 - modifying *s4pcap* B-6
 - output B-1
 - supported devices B-2
- grid
 - after formation of isolation region, plotted by *s4ex4a.inp* Fig. 5-6
 - plot produced by first **PLOT . 2D** statement input file *s4ex2b.inp* Fig. 4-20
 - poly-buffered LOCOS application Fig. 5-28
 - trench implant example Fig. 5-19
- grid generation parameters, automatic, default coefficients A-25
- grid lines
 - automatic generation 2-5
 - eliminating 2-4
 - generated 2-3
 - specifying 2-3
- GRID** parameter, **PLOT . 2D** statement
 - local oxidation example 4-14, 4-20
- grid parameters, adaptive, default coefficients A-25
- grid structure
 - 1D simulation of simple structures 2-12
 - adaptive gridding 2-10
 - automatic generation 2-5
 - changes to mesh during processing 2-7
 - defining 2-3
 - explicit specification 2-3
 - generated grid lines 2-3
 - initial impurity concentration 2-12
 - initial impurity concentration, sheet resistance 2-13
 - LINE** statement 2-3
 - mesh 2-2
 - nodes 2-2
 - overview 2-2
 - triangular elements 2-2
- GRID . FAC** parameter, **MESH** statement
 - CMOS process example 5-32
 - DMOS power transistor example 5-40
 - local oxidation example 4-13
 - NMOS DD process example 5-3

H

- HELP** 3-47
 - description 3-47
 - example 3-47
 - notes 3-47
- HIGH** parameter, **ELECTRICAL** statement
 - NMOS LDD process example 5-15
- H . RECTAN** parameter, **LABEL** statement
 - local oxidation example 4-18

I

- IF/ELSEIF/ELSE/IF . END** 3-23
 - conditional operators 3-24
 - description 3-23
 - expression for condition 3-24
- IMPLANT** 3-103
 - boundary conditions 3-111
 - channeling effects 3-112
 - CRYSTAL** parameter, trench implant simulation example 5-23
 - description 3-108
 - DOSE** parameter, trench implant simulation example 5-23
 - ENERGY** parameter, trench implant simulation example 5-23
 - examples 3-112
 - extended defects 3-111
 - Gaussian and Pearson distributions 3-109
 - Monte Carlo implant model 3-110
 - MONTECAR** parameter, trench implant simulation example 5-19
 - N . ION** parameter, trench implant simulation example 5-23
 - point defect generation 3-111
 - table of range statistics 3-109
 - TILT** parameter, trench implant simulation example 5-19
 - TSUPREM-4 version considerations 3-112
- implant damage model 2-98
 - net damage calculation 2-101
 - using 2-102
- implant, Monte Carlo, default coefficients A-24
- ImplantModel** command 6-13
- improved accuracy, flexibility, and ease of use C-5
- improved effective +n implant damage model C-1
- improved physical models C-1
- impurities
 - activation of 2-24
 - clustering of 2-24
 - diffusion 2-105
 - diffusion of 2-17
 - implantation 2-30
 - incorporation 2-105
 - interface trap model 2-29
 - ionization of 2-121
 - moving-boundary flux 2-29
 - segregation coefficient 2-29
 - segregation flux 2-28
 - segregation of 2-28
 - specifying 3-2
 - transport coefficient 2-28
- IMPURITY** 3-239
 - description 3-251
 - examples 3-252
 - further reading 3-252
 - impurity type 3-251
 - multiplication to diffusivity 3-252
 - other parameters 3-251
 - solution options 3-251
- impurity distributions in bipolar structure at end input file *s4ex1a.inp* Fig. 4-8
- impurity parameters, default coefficients A-2
- incorporation of impurities 2-105
- IN . FILE** parameter, **INITIALIZE** statement
 - 1D bipolar example 4-9
- informational output file *s4inf* 1-6
- Initial Condition** command 6-11
- initial conditions 2-73
- initial grid
 - accuracy 5-1
 - for simulating DMOS power transistor Fig. 5-40
 - for simulating SOI MOSFET Fig. 5-46
 - for the 0.8 micron NMOS transistor, produced by *s4ex7a.inp* Fig. 5-33
- INITIAL** parameter, **AMBIENT** statement
 - 1D bipolar example 4-4
- initialization 6-20
- initialization input file *s4init* 1-9
- INITIALIZE** 3-64
 - crystalline orientation 3-66
 - description 3-66
 - examples 3-67
 - IN . FILE** parameter, 1D bipolar example 4-9
 - Masetti's mobility table 3-67
 - mesh generation 3-66
 - previously saved structure files 3-66
 - specifying initial doping 3-67
- injection rate 2-39
- input files 1-4
 - command 1-4
 - mask data 1-5
 - other 1-5
 - profile 1-5

s4ex1a.inp, simulating buried layer and epitaxial deposition, bipolar transistor structure *Fig. 4-3*
s4ex4c.inp, plotting final LDD NMOS structure *Fig. 5-13*
s4ex7b.inp, plotting results *Fig. 5-36*
s4init 1-9
se4ex10p.inp, plotting results *Fig. 5-52*

input statement descriptions
 format 3-2
 introduction 3-1
 parameter types 3-3
 specifying materials and impurities 3-2
 statement description format 3-4
 syntax 3-2

installation xxxvii

INTERACTIVE 3-12
 description 3-12
 example 3-13
 interactive input mode 3-12

interface functions 6-31

interface oxide break-up and epitaxial regrowth
 oxide break-up 2-117

interface trap model 2-30, 6-39

interface trap model, impurities 2-29

InterfaceCondition command 6-12
Material1 parameter 6-13
Material2 parameter 6-13
Symmetric parameter 6-13

INTERMEDIATE 3-32
 advantages of **INTERMEDIATES** 3-32
 advantages of intermediates 6-24
 array values 3-34
 comparison to **ASSIGN** 3-36
 value type 3-33

INTERSTITIAL 3-265
 additional notes 3-275
 bulk and interface parameters 3-274
 description 3-274
 examples 3-275

interstitial and vacancy diffusivities 2-34

interstitial clustering model 2-42, 6-34

interstitial traps 2-41

interstitial traps, enabling, disabling, and initialization 2-42

interstitials
 absorption of traps 2-37
 net recombination rate 2-35

intrinsic stress in deposited layers
 stress history model 2-74
 stress model 2-74

introduction to TSUPREM-4 xxxv

Ion C-7

ion implant data file *s4imp0* 1-9

ion implantation
 advanced example of 5-18
 analytic models 2-77
 boundary conditions 2-103
 implant damage model 2-98
 into silicon carbide 2-104
 Monte Carlo model 2-87
 overview 2-77

ion pairing and mobile impurities 2-18

ION parameter, **IMPLANT** statement
 trench implant simulation example 5-23

J

JCAP parameter, **ELECTRICAL** statement
 NMOS LDD process example 5-14

JUNCTION parameter, **ELECTRICAL** statement
 NMOS LDD process example 5-16

K

key files *s4fky0* and *s4uky0* 1-10

L

LABEL 3-155
CM parameter, 1D bipolar example 4-7
 color 3-159
 description 3-158
 examples 3-159
 label placement 3-158
 line, symbol, and rectangle 3-158
X parameter, 1D bipolar example 4-7
Y parameter, 1D bipolar example 4-7

LAYERS parameter, **PRINT . 1D** statement
 DMOS power transistor example 5-42
 local oxidation example 4-8

library files 1-8
 authorization file *s4auth* 1-10
 initialization input file *s4init* 1-9
 ion implant data file, *s4imp0* 1-9
 key files *s4fky0* and *s4uky0* 1-10
 plot device definition file *s4pcap* 1-9

LINE 3-55
 additional notes 3-56
 default regions and boundaries 3-56
 description 3-55
 example 3-56
 maximum number of nodes and grid lines 3-56
 placing grid lines 3-55

structure depth and point defect models 3-56

LINE . TYP parameter, **CONTOUR** statement
 1D local oxidation example 4-23

LINE . TYPE parameter, **PLOT . 1D** statement
 1D bipolar example 4-7

listings
 input file *s4ex11a.inp* for simulating bipolar emitter structure *Fig. 5-55*
 input file *s4ex1b.inp*, showing statements simulating the field oxide, base, and emitter region processing for bipolar transistor *Fig. 4-10*
 mask information read from file *s4ex4m.tll* *Fig. 5-4*
 MINIMOS 5 command file *EX2D.INP* *Fig. F-7*
 MINIMOS 5 command file *EX2D.INP*, modified to read doping profiles produced by TSUPREM-4 *Fig. F-7*
PRINT . 1D command in file *s4ex1a.inp* *Fig. 4-9*
 produced by **PRINT . 1D** statement in input file *s4ex3.inp* *Fig. 4-31*

L . MODIFY 3-22
 description 3-22

LOADFILE 3-68
 description 3-68
 examples 3-69
 Taurus-Lithography files 3-69
 TSUPREM-4 files 3-68
 user-defined materials and impurities 3-69

local oxidation example
COMPRESS oxidation model 4-19
CONTOUR statement 4-17, 4-23
DIFFUSION statement 4-15, 4-19
 diffusion with point defects 4-18
END statement 4-17, 4-23
FOREACH statement 4-17, 4-23
INITIALIZE statement 4-13
LABEL statement 4-24, 4-25
 mesh generation 4-13
 model selection 4-14
 overview 4-12
 oxide shape 4-12
 pad oxide and nitride layers 4-14
PLOT . 1D statement 4-25
 plotting
 mesh 4-14
 results 4-15
 stresses 4-16
PRINT . 1D statement 4-18
SAVEFILE statement 4-15
SELECT statement 4-17
 summary 4-26

Loop command 6-10**Index** parameter 6-10**Name** parameter 6-10**Start** parameter 6-10**Steps** parameter 6-10**LOOP/L.END** 3-18

advantages 3-21

dependence and variability 3-20

description 3-19

example 3-20

parameter sensitivity 3-19

termination of optimization looping 3-19

LOW parameter, **ELECTRICAL** statement

NMOS LDD process example 5-15

LY.ACTIV parameter, **MESH** statement

CMOS process example 5-32

LY.SURF parameter, **MESH** statement

CMOS process example 5-32

M

manual overview xxxvi

MASK 3-82

description 3-82

examples 3-83

mask data files

format D-1

overview 1-5, D-1

MATERIAL 3-228

description 3-236

examples 3-237

POLYCRYS parameter, NMOS LDD process example 5-7

viscosity and compressibility 3-237

material coefficients, default coefficients A-21

materials, specifying 3-2

mathematical functions 6-28

Medici

MEDICI parameter 3-75

saved structure files overview 1-7

structure file format E-8

Medici structure file *Fig.* E-10**MESH** 3-50

1D mode 3-53

description 3-51

examples 3-54

grid creation methods 3-51

GRID.FAC parameter, 1D bipolar example 4-4

horizontal grid generation 3-52

scaling the grid spacing 3-53

scaling the grid spacing, **GRID.FAC** 3-53

vertical grid generation 3-52

mesh

changes during processing 2-7

DEPOSITION statement 2-7**DEVELOP** statement 2-8**EPITAXY** statement 2-7**ETCH** statement 2-8

overview 2-2

oxidation and silicidation 2-8

structure extension 2-7

mesh generation

depositing a layer with nonuniform grid spacing 5-46

DMOS power transistor, from input file *s4ex8.inp Fig.* 5-39SOI MOSFET, from input file *s4ex9.inp Fig.* 5-45

tutorial example 4-2

mesh used for oxidation simulation *Fig.* 4-13**METHOD** 3-188

adaptive gridding 3-200

analytical implant models 3-200

disable 3-200

block solution 3-202

customizing the point defect models 3-199

description 3-197

enable/disable user-specified models 3-199

ERR.FAC parameter, 1D bipolar example 4-4

error tolerances 3-203

examples 3-203, 3-205

fine control 3-201

grid spacing in oxide 3-197

initial time step 3-201

internal solution methods 3-201

internal solution methods, changing 3-201

matrix refactoring 3-203

matrix structure 3-202

minimum-fill reordering 3-202

oxidation models 3-197

PD.FERMI model 3-198**PD.FULL** model 3-199**PD.TRANS** model 3-199

point defect modeling 3-198

rigid vs. viscous substrate 3-198

solution method 3-202

system solutions 3-202

time integration 3-201

MINIMOS 5 interface F-1

miscellaneous functions 6-31

miscellaneous improvements C-8

MOBILE 6-26

mobile impurities and ion pairing 2-18

MOBILITY 3-259

analytic models 3-262

description 3-262

example 3-263

tables and analytic models 3-262

tables or model selection 3-263

Model command 6-7**Array** parameter 6-8**DataType** parameter 6-8**Expression** parameter 6-7**Loop** parameter 6-8**Model** parameter 6-8**Name** parameter 6-7**Parameter** parameter 6-8

models and coefficients, input statements

AMBIENT 3-207**ANTIMONY** 3-284**ARSENIC** 3-290**BORON** 3-296**EQUATION** 3-204**IMPURITY** 3-239**INTERSTITIAL** 3-265**MATERIAL** 3-228**METHOD** 3-188**MOBILITY** 3-259**MOMENT** 3-224**PHOSPHORUS** 3-302**REACTION** 3-254**VACANCY** 3-276

modification of built-in functions 6-25

ACTIVE 6-25**MOBILE** 6-26**MOMENT** 3-224

additional notes 3-227

description 3-226

examples 3-227

optional and required model parameters 3-226 using 3-226

Monte Carlo implant

amorphization 2-97

damage dechanneling 2-97

default coefficients A-24

model 2-87

model, binary scattering theory 2-87

overview 5-23

using 5-23

MONTECAR parameter, **IMPLANT** statement

trench implant simulation example 5-19, 5-23

MOS capacitance 2-126

MOSCAP parameter, **ELECTRICAL** statement

NMOS LDD process example 5-14

MOSFET with self-aligned silicides example

COMPRESS model 5-51

overview 5-50

PD.FERMI model 5-51

preparation for silicidation 5-50

silicidation 5-51

MOSFET, quantum mechanical model 2-125

moving-boundary flux 2-29, 2-41
 multilayer implants 2-84
 dose matching 2-84
 effective range model 2-84

N

N2O oxidation
 nitrogen trap and generation 2-63
 N2O oxidation 2-63
 net recombination rate of interstitials 2-35
 new features in version 2000.2 C-1
 new solution variable 6-20
 nitrogen trap and generation 2-63
 NMOS LDD process example
 active region simulation 5-7
 COMPRESS oxidation model 5-5
 creating initial structure 5-2
 DEPOSITION statement 5-7, 5-9
 DEVELOP statement 5-5, 5-7
 electrical extraction 5-14
 ELIMINATE statement 5-2
 ETCH statement 5-7
 formation of complete NMOS transistor 5-12
 INITIALIZE statement 5-2, 5-7
 LINE statement 5-2
 MASK statement 5-2, 5-7
 MESH statement 5-2, 5-7
 METHOD statement 5-7
 overview 5-2
 PD.FERMI model 5-4
 PD.TRANS model 5-5
 PLOT.2D statement 5-5
 remove photoresist 5-4
 SAVEFILE statement 5-11
 TRAP etch model 5-5
 VERTICAL model 5-4
 VISCOELA model 5-5
 NMOS parameter, **ELECTRICAL** statement
 NMOS LDD process example 5-14
 NMOS structure
 after LDD implant, as plotted by file *s4ex4b.inp* Fig. 5-9
 to transfer to MINIMOS 5 Fig. F-3
 nodes
 addition in growing oxide 2-8
 maximum number 2-2
 overview 2-2
 regions where oxide is forming 2-9
 removing in consumed silicon 2-8
 numerical methods, default coefficients A-24
 numerical oxidation models 2-54
 COMPRESS model 2-57
 compressible viscous flow 2-57
 concentration dependence 2-55

oxide growth rate 2-54
 thin regime 2-56
 usage 2-56
VERTICAL model 2-56
VISCOELA model 2-60
VISCOUS model 2-58

O

OnMat1 function 6-16
OnMat2 function 6-16
 operators 6-27
 optimization
 ASSIGN statement 3-25
 ELECTRICAL statement 3-181
 EXTRACT statement 3-168
 LOOP statement 3-18
OPTION 3-39
 DEVICE parameter, 1D bipolar example 4-6
 echoing and execution of input statements 3-40
 examples 3-41
 informational and diagnostic output 3-40
 printed output 3-40
 redirecting graphics output 3-40
 selecting a graphics device 3-40
 version compatibility 3-41
 other silicides 2-72
 output files 1-5
 electrical data 1-8
 extract 1-8
 graphical 1-8
 listing 1-6
 saved structure 1-6
 terminal 1-5
 output listing files 1-6
 s4dia 1-6
 s4inf 1-6
 s4out 1-6
 output, input statements
 COLOR 3-150
 CONTOUR 3-148
 ELECTRICAL 3-174
 EXTRACT 3-160
 LABEL 3-155
 PLOT.1D 3-136
 PLOT.2D 3-144
 PLOT.3D 3-152
 PRINT.1D 3-132
 SELECT 3-128
 VIEWPORT 3-184
 output, program
 graphical 1-2
 overview 1-2
 printed 1-2
 oxidation

advanced example of 5-28
 analytical models 2-48
 default coefficients A-10
 N2O 2-63
 numerical models 2-54
 overview 2-46
 polysilicon 2-63
 surface reaction rate 2-64
 theory of 2-46
 thin oxidation rate 2-64
 tutorial example 4-4, 4-12
 oxidation and plotting of impurity profiles
 doping and layer information 4-31
 PD.FERMI model 4-30
 PD.FULL model 4-30
 PD.TRANS model 4-30
 printing junction depth 4-31
 simulation procedure 4-29
 oxidation and silicidation 2-8
 oxide growth rate 2-54
 oxide reflow
 see reflow

P

P1.X parameter, **ETCH** statement
 1D local oxidation example 4-14
P1.Y parameter, **ETCH** statement
 1D local oxidation example 4-14
P2.X parameter, **ETCH** statement
 local oxidation example 4-14
P2.Y parameter, **ETCH** statement
 1D local oxidation example 4-14
 paired fractions of dopant atoms 2-22
Parameter command 6-9
 Boundary parameter 6-9
 Default parameter 6-9
 Name parameter 6-9
 parameters
 definition table 3-4
 deposition 3-93
 list syntax 3-4
 types 3-3
 parameters, **AMBIENT** statement
 /**AMBIENT** 2-54, 3-214
 /**MATERIA** 3-214
 /**NITRIDE** 3-214
 /**OXIDE** 3-214
 /**OXYNITR** 3-214
 /**POLYSIL** 3-214
 /**SILICON** 3-214
 <**100**> 2-48, 2-49, 2-64, 2-65, 3-209
 <**110**> 2-48, 2-49, 2-64, 2-65, 3-209
 <**111**> 2-48, 2-49, 2-64, 2-65, 3-209
 ALPHA 2-63, 3-215
 AMB.1 2-14, 3-208
 AMB.2 2-14, 3-208
 AMB.3 2-14, 3-208

AMB.4 2-14, 3-208
AMB.5 2-14, 3-208
AMBIENT 3-213
CLEAR 3-217
CM.SEC 3-217
COLUMN 2-49, 3-212
D.O 2-54, 3-213
D.E 2-54, 3-213
DRYO2 2-14, 3-208
ERF1 2-53
ERF2 2-53
ERF.DELT 2-51, 3-216
ERF.H 2-51, 2-52, 3-217
ERF.LBB 2-51, 2-52, 3-216
ERF.Q 2-51, 3-216
F.H2 2-14, 3-209
F.H2O 2-14, 3-209
F.HCL 2-14, 3-209
F.N2 2-14, 3-209
F.N2O 3-209
F.O2 2-14, 3-208
GAMMA.O 2-55, 3-212
GAMMA.E 2-55, 3-212
H2O 3-209
HCL 2-14, 3-209
HENRY.CO 2-54, 3-214
H.LIN.O 2-49, 3-210
H.LIN.E 2-49, 3-210
H.PAR.O 2-50, 3-211
H.PAR.E 2-50, 3-211
INERT 2-14, 3-208
INITIAL 2-50, 2-51, 3-216, 4-4
LIN.BREA 2-49, 3-211
LIN.CLDE 2-49, 3-212
LIN.PCT 2-49, 3-212
LIN.PDEP 2-49, 3-211
L.LIN.O 2-49, 3-210
L.LIN.E 2-49, 3-210
L.PAR.O 2-50, 3-211
L.PAR.E 2-50, 3-211
MASK.EDG 2-48, 3-216
MATERIAL 3-213
N.O 3-210
N2O 3-208
N.E 3-211
N.G.O 3-215
N.G.E 3-215
N.GPOW.O 3-215
N.GPOW.E 3-215
NITRIDE 3-213
NIT.THIC 2-51, 2-52, 3-217
N.THIN.O 3-211
N.THIN.E 3-211
N.V.O 3-215
N.V.E 3-215
O2 3-209
ORIENTAT 3-210
OXIDE 2-54, 3-213
OXYNITRI 3-213

PAR.BREA 2-50, 3-211
PAR.CLDE 2-50, 3-212
PAR.PCT 2-50, 3-212
PAR.PDEP 2-50, 3-212
POLYSILI 2-48, 2-49, 2-64, 2-65, 3-210, 3-213
PRESSURE 2-14, 3-209
SEG.O 3-214
SEG.E 3-214
SILICON 3-213
SPREAD 2-51, 3-216
STEAM 2-14, 3-208
STRESS.D 2-59, 3-215
TABLE 2-49, 3-213
TEMPERAT 2-49, 3-217
THETA 2-54, 3-214
THINOX.O 2-48, 3-210
THINOX.E 2-48, 3-210
THINOX.L 2-48, 3-210
TRANS.O 2-54, 3-215
TRANS.E 2-54, 3-215
VC 2-59, 3-214
VD 2-59, 3-216
VDLIM 2-59, 3-216
VR 2-59, 3-216
VT 2-59, 3-216
WETO2 2-14, 3-208

parameters, ANTIMONY statement

/AMBIENT 3-287
/MATERIA 3-286
/NITRIDE 3-287
/OXIDE 3-286
/OXYNITR 3-287
/POLYSIL 3-287
/SILICON 3-286
AMBIENT 3-284
CM.SEC 3-288
DIM.O 2-20, 3-285
DIM.E 2-20, 3-285
DIMM.O 2-20
DIMM.E 2-20
DIPAIR.O 2-22, 3-285
DIPAIR.E 2-22, 3-285
DIP.E 2-20
DIP.O 2-20
DIX.O 2-20, 3-285
DIX.E 2-20, 3-285
DVM.O 2-20, 3-285
DVM.E 2-20, 3-285
DVMM.O 2-20
DVMM.E 2-20
DVP.O 2-20
DVPAIR.O 2-22, 3-285
DVPAIR.E 2-22, 3-285
DVP.E 2-20
DVX.O 2-20, 3-285
DVX.E 2-20, 3-285
E.IP.V 2-23, 3-286
E.I.S 3-286

E.I.S. 2-23
ES.100 3-287
ES.110 3-288
ES.F.100 3-288
ES.F.110 3-288
ES.F.RAN 3-287
ES.RAND 3-287
E.VP.I 2-23, 3-286
E.V.S 3-286
E.V.S. 2-23
MATERIAL 3-284
NITRIDE 3-284
OXIDE 3-284
OXYNITRI 3-284
POLYSILI 3-284
R.IP.V 2-23, 3-286
R.I.S 2-23, 3-285
R.VP.I 2-23, 3-286
R.V.S 2-23, 3-286
SEG.O 3-287
SEG.E 3-287
SILICON 3-284
SS.CLEAR 3-286
SS.CONC 3-286
SS.TEMP 3-286
TRANS.O 3-287
TRANS.E 3-287

parameters, Array command

DefaultValue 6-10
MinIndex 6-9
Name 6-9
Object 6-10
Size 6-9
Type 6-9

parameters, ARSENIC statement

/AMBIENT 3-293
/MATERIA 3-292
/NITRIDE 3-293
/OXIDE 3-293
/OXYNITR 3-293
/POLYSIL 3-293
/SILICON 3-292
AMBIENT 3-290
CM.SEC 3-294
CTN.O 3-292
CTN.E 3-292
CTN.F 3-292
DIM.O 2-20, 3-291
DIM.E 2-20, 3-291
DIMM.O 2-20
DIMM.E 2-20
DIP.O 2-20
DIPAIR.O 2-22, 3-291
DIPAIR.E 2-22, 3-291
DIP.E 2-20
DIX.O 2-20, 3-291
DIX.E 2-20, 3-291
DVM.O 2-20, 3-291
DVM.E 2-20, 3-291

- DVMM.0 2-20
- DVMM.E 2-20
- DVP.0 2-20
- DVPAIR.0 2-22, 3-291
- DVPAIR.E 2-22, 3-291
- DVP.E 2-20
- DVX.0 2-20, 3-291
- DVX.E 2-20, 3-291
- E.IP.V 2-23, 3-292
- E.I.S 2-23, 3-292
- ES.100 3-293
- ES.110 3-294
- ES.F.100 3-294
- ES.F.110 3-294
- ES.F.RAN 3-293
- ES.RAND 3-293
- E.VP.I 2-23, 3-292
- E.V.S 3-292
- E.V.S. 2-23
- MATERIAL 3-290
- NITRIDE 3-290
- OXIDE 3-290
- OXYNITRI 3-290
- POLYSILI 3-290
- R.IP.V 2-23, 3-292
- R.I.S 3-291
- R.I.S. 2-23
- R.VP.I 2-23, 3-292
- R.V.S 3-292
- R.V.S. 2-23
- SEG.0 3-293
- SEG.E 3-293
- SILICON 3-290
- TRANS.0 3-293
- TRANS.E 3-293
- parameters, ASSIGN statement
 - ARRAY 3-28
 - C1 3-27
 - C10 3-28
 - C2 3-27
 - C3 3-27
 - C4 3-27
 - C5 3-27
 - C6 3-27
 - C7 3-27
 - C8 3-27
 - C9 3-28
 - C.COUNT 3-28
 - C.FILE 3-26
 - COLUMN 3-28
 - C.VALUE 3-26
 - DATA 3-28
 - DELETE 3-25
 - DELTA 3-26
 - IN.CVALU 3-28
 - IN.FILE 3-28
 - IN.NVALU 3-28
 - LINE 3-26
 - LOG 3-26
- LOWER 3-26
- NAME 3-25
- N.EXPRES 3-25
- N.VALUE 3-26
- PRINT 3-25
- PROMPT 3-25
- RATIO 3-26
- ROW 3-28
- TIF 3-28
- UPPER 3-26
- parameters, BORON statement
 - /AMBIENT 3-299
 - /MATERIA 3-298
 - /NITRIDE 3-299
 - /OXIDE 3-298
 - /OXYNITR 3-299
 - /POLYSIL 3-299
 - /SILICON 3-298
 - AMBIENT 3-296
 - CM.SEC 3-300
 - DIM.0 2-20
 - DIM.E 2-20
 - DIMM.0 2-20
 - DIMM.E 2-20
 - DIP.0 2-20, 3-297
 - DIPAIR.0 2-22, 3-297
 - DIPAIR.E 2-22, 3-297
 - DIP.E 2-20, 3-297
 - DIX.0 2-20, 3-297
 - DIX.E 2-20, 3-297
 - DVM.0 2-20
 - DVM.E 2-20
 - DVMM.0 2-20
 - DVMM.E 2-20
 - DVP.0 2-20, 3-297
 - DVPAIR.0 2-22, 3-297
 - DVPAIR.E 2-22, 3-297
 - DVP.E 2-20, 3-297
 - DVX.0 2-20, 3-297
 - DVX.E 2-20, 3-297
 - E.IP.V 2-23, 3-298
 - E.I.S 3-298
 - E.I.S. 2-23
 - ES.100 3-299
 - ES.110 3-300
 - ES.F.100 3-300
 - ES.F.110 3-300
 - ES.F.RAN 3-299
 - ES.RAND 3-299
 - E.VP.I 2-23, 3-298
 - E.V.S 3-298
 - E.V.S. 2-23
 - MATERIAL 3-296
 - NITRIDE 3-296
 - OXIDE 3-296
 - OXYNITRI 3-296
 - POLYSILI 3-296
 - R.IP.V 2-23, 3-298
 - R.I.S 3-297
- R.I.S. 2-23
- R.VP.I 2-23, 3-298
- R.V.S 3-298
- R.V.S. 2-23
- SEG.0 3-299
- SEG.E 3-299
- SILICON 3-296
- SS.CLEAR 3-298
- SS.CONC 3-298
- SS.TEMP 3-298
- TRANS.0 3-299
- TRANS.E 3-299
- parameters, BOUNDARY statement
 - EXPOSED 3-60
 - REFLECTI 3-60
 - XHI 3-60
 - XLO 3-60
 - YHI 3-60
 - YLO 3-60
- parameters, BoundaryCondition command
 - Material1 6-12
 - Material2 6-12
 - Type 6-12
- parameters, COLOR statement
 - ALUMINUM 3-150
 - COLOR 3-150
 - MATERIAL 3-150
 - MAX.VALU 3-150
 - MIN.VALU 3-150
 - NITRIDE 3-150
 - OXIDE 3-150
 - OXYNITRI 3-150
 - PHOTORES 3-150
 - POLYSILI 3-150
 - SILICON 3-150
- parameters, CONTOUR statement
 - COLOR 3-148
 - LINE.TYP 3-148, 4-23
 - SYMBOL 3-148
 - VALUE 3-148, 4-23
- parameters, CPULOG statement
 - LOG 3-46
 - OUT.FILE 3-46
- parameters, DEPOSITION statement
 - ALUMINUM 2-105, 3-90
 - ANTIMONY 2-106, 3-91
 - ARC.SPAC 2-106, 3-92
 - ARSENIC 2-106, 3-91
 - BORON 2-106, 3-91
 - CONCENTR 3-91
 - DY 2-7, 3-91, 5-46
 - GSZ.LIN 2-115, 3-92
 - I.CONC 2-106, 3-91
 - IMPURITY 2-106, 3-91
 - I.RESIST 3-91
 - MATERIAL 3-90
 - NEGATIVE 3-90
 - NITRIDE 2-105, 3-90

- OXIDE 2-105, 3-90
 OXYNITRI 2-105, 3-90
 PHOSPHOR 2-106, 3-91
 PHOTORES 2-105, 3-90
 POLYSILI 2-105, 3-90
 POSITIVE 3-90
 RESISTIV 3-91
 SILICON 2-105, 3-90
 SPACES 2-7, 3-91, 5-46
 TEMPERAT 3-92
 THICKNES 2-7, 3-91, 4-5
 TOPOGRAP 3-92
 YDY 2-7, 3-92, 5-46
- parameters, **DEVELOP** statement
 NEGATIVE 2-106
 POSITIVE 2-106
- parameters, **DIFFUSION** statement
 AMB. 1 3-116
 AMB. 2 3-116
 AMB. 3 3-116
 AMB. 4 3-116
 AMB. 5 3-116
 ANTIMONY 2-16, 3-117
 ARSENIC 2-16, 3-117
 BORON 2-16, 3-117
 CONTINUE 3-115
 D.RECOMB 3-117
 DRYO2 2-46, 3-115, 4-6
 DUMP 3-118
 F.H2 3-116
 F.H2O 3-116
 F.HCL 3-117
 F.N2 3-116
 F.N2O 3-116
 F.O2 3-116
 HCL 2-16, 3-117
 I.CONC 2-16, 3-117
 IMPURITY 2-16, 3-117
 INERT 3-116
 MOVIE 3-118
 N2O 2-46, 3-116
 P.FINAL 2-14, 3-117
 PHOSPHOR 2-16, 3-117
 P.RATE 2-14, 3-117
 PRESSURE 3-117
 STEAM 2-46, 3-116, 4-5
 TEMPERAT 2-14, 3-115
 T.FINAL 2-14, 3-115
 TIME 2-13, 3-115
 T.RATE 2-14, 3-115
 WETO2 2-46, 3-115
- parameters, **ELECTRICAL** statement
 ANGLE 3-175
 BIAS.REG 3-175
 BULK.LAY 3-177
 BULK.REG 3-176
 DEEP 3-176, 5-15
 DEPTH 3-175
 DISTRIB 3-177
- E.RVCAP 5-16
 EXT.REG 3-175
 GATE.ELE 3-176
 GATE.WF 3-176
 HIGH 3-176, 5-15
 JCAP 3-175, 5-14
 JUNCTION 3-176, 5-16
 LOW 3-176, 5-15
 MIN.ABS 3-179
 MIN.REL 3-179
 MOSCAP 3-176, 5-14
 NAME 3-177
 NMOS 3-176, 5-14
 OUT.FILE 2-120, 3-177
 PITCH 3-175
 PMOS 3-176
 POINT 3-175
 PRINT 3-177
 QM 2-125, 3-176
 QSS 3-176, 5-14
 RESISTAN 3-175, 5-14
 SENSITIV 3-177
 SRP 3-174
 TARGET 3-177
 T.COLUMN 3-178
 TEMPERAT 2-121, 3-177
 T.FILE 3-177
 THRESHOL 3-176
 THRESHOLD 5-14
 T.LOWER 3-178
 TOLERANC 3-179
 T.TRANSF 3-178
 T.UPPER 3-178
 V 3-175
 VB 3-176
 V.COLUMN 3-177
 V.LOWER 3-178
 V.SELECT 3-177
 VSTART 3-175
 VSTEP 3-175
 VSTOP 3-175
 V.TRANSF 3-178
 V.UPPER 3-178
 WEIGHT 3-179
 X 3-174, 5-16
 Y.SURFAC 3-175
 Z.VALUE 3-178
- parameters, **ELECTRODE** statement
 ALL 3-87
 BOTTOM 3-87
 CLEAR 3-87
 MERGE 3-87
 NAME 3-87
 PRINT 3-87
 X 3-87
 Y 3-87
- parameters, **ELIMINATE** statement
 COLUMNS 3-57
 ROWS 3-57
- X.MAX 3-57
 X.MIN 3-57
 Y.MAX 3-57
 Y.MIN 3-57
- parameters, **EPITAXY** statement
 ANTIMONY 2-105, 3-121
 ARC.SPAC 3-122
 ARSENIC 2-105, 3-121, 4-6
 BORON 2-105, 3-122
 CONCENTR 3-122
 DY 2-7, 3-122
 I.CONC 2-105, 3-121
 IMPURITY 2-105, 3-121
 I.RESIST 3-121
 PHOSPHOR 2-105, 3-122
 RESISTIV 3-122
 SPACES 2-7, 3-122, 4-5
 TEMPERAT 2-104, 3-121, 4-6
 T.FINAL 2-104, 2-105, 3-121
 THICKNES 2-7, 3-122
 TIME 2-104, 3-121, 4-6
 T.RATE 2-104, 2-105, 3-121
 YDY 2-7, 3-122
- parameters, **Equation** command
 Array 6-7
 Boundary 6-7
 DopantCharge 6-7
 ErrorLimit 6-7
 Expression 6-7
 Loop 6-7
 Material 6-6
 Material2 6-7
 Model 6-7
 Name 6-6
 Parameter 6-7
 Scaling 6-7
 VariableName 6-6
- parameters, **EQUATION** statement
 /MATERIAL 3-204
 ADDTOEXP 3-204
 BOUNDARY 3-204
 INITIAL 3-204
 INTERFACE 3-204
 MATERIAL 3-204
 MODEL 3-204
 TIF.NAME 3-204
 TRANSPOR 3-205
 UPDATE 3-204
 VARIABLE 3-204
- parameters, **ETCH** statement
 ALL 2-108, 3-100
 ALUMINUM 2-107, 3-98
 ANGLE 2-107, 3-99
 CONTINUE 2-108, 3-99
 DONE 2-108, 3-99
 ISOTROPI 2-107, 3-99
 LEFT 2-108, 3-99
 MATERIAL 3-98
 NITRIDE 2-107, 3-98

OLD.DRY 2-108, 3-100
OXIDE 2-107, 3-98
OXYNITRI 2-107, 3-98
P1.X 2-108, 3-99, 4-14
P1.Y 2-108, 3-99, 4-14
P2.X 2-108, 3-99, 4-14
P2.Y 2-108, 3-99, 4-14
PHOTORES 2-107, 3-98
POLYSILI 2-107, 3-98
RIGHT 2-108, 3-99, 4-14
SILICON 2-107, 3-98
START 2-108, 3-99
THICKNES 2-107, 3-99
TOPOGRAP 3-100
TRAP 5-34
TRAPEZOI 2-107, 3-98
UNDERCUT 2-107, 3-99
X 2-108, 3-100
Y 2-108, 3-100
 parameters, **EXPOSE** statement
MASK 3-95
OFFSET 3-95
SHRINK 3-95
 parameters, **Expression** command
Material 6-8
MaterialType 6-8
 parameters, **EXTRACT** statement
/ALUMINU 3-161
/AMBIENT 3-161
/MATERIA 3-161
/NITRIDE 3-161
/OXIDE 3-161
/OXYNITR 3-161
/PHOTORE 3-161
/POLYSIL 3-161
/SILICON 3-161
ALUMINUM 3-161
APPEND 3-165
AREA.EXT 3-162
ASSIGN 3-163
AVG.EXTR 3-162
CLOCKWIS 3-161
CLOSE 3-165
D.EXTRAC 3-162
DISTANCE 3-162
INT.EXTR 3-162
MATERIAL 3-160
MAXIMUM 3-162
MIN.ABS 3-165
MINIMUM 3-162
MIN.REL 3-165
NAME 3-163
NITRIDE 3-160
OUT.FILE 3-165
OXIDE 3-160
OXYNITRI 3-160
P1.X 3-161
P1.Y 3-161
P2.X 3-161

P2.Y 3-161
PHOTORES 3-161
POLYSILI 3-160
PREFIX 3-162
PRINT 3-163
SENSITIV 3-164
SEPARAT 3-163
SILICON 3-160
SUFFIX 3-163
TARGET 3-163
T.COLUMN 3-164
T.FILE 3-163
T.LOWER 3-164
TOLERANC 3-164
T.UPPER 3-164
VAL.EXTR 3-162
VALUE 3-162
V.COLUMN 3-163
V.LOWER 3-163
V.TRANSF 3-164
V.UPPER 3-164
WEIGHT 3-165
WRITE 3-163
X 3-162
X.EXTRAC 3-162
Y 3-162
Y.EXTRAC 3-162
Z.VALUE 3-164
 parameters, **IMPLANT** statement
AMOR.PAR 3-107
ANTIMONY 2-77, 3-104
ARSENIC 2-77, 3-104
BACKSCAT 2-77, 3-105
BEAMWIDT 2-94, 3-105
BF2 2-77, 3-104
BF2 2-97
BORON 2-77, 3-104
CRIT.110 2-95, 3-106
CRIT.F 2-95, 3-106
CRIT.PRE 2-95, 3-106
CRYSTAL 2-94, 3-105, 5-23
DAMAGE 2-86, 3-107
DAM.MOD 3-107
DOSE 2-77, 3-103, 5-23
D.P1 3-107
D.PLUS 2-102, 3-107
D.RECOMB 2-101, 3-108
D.SCALE 2-102, 3-108
E.LIMIT 2-96, 3-106
ENERGY 2-77, 3-103, 5-23
GAUSSIAN 2-78, 3-104
IMPL.TAB 2-78, 3-105
IMPURITY 3-103
IN.FILE 2-78, 3-105
INTEREST 3-107
ION 5-23
L.DENS 2-128, 3-108
L.DMAX 3-108

L.DMIN 2-128, 3-108
L.FRAC 3-108
L.RADIUS 2-128, 3-108
LSS.PRE 3-107
L.THRESH 3-108
MAX.DAMA 2-97, 3-108
MOMENTS 3-105
MONTECAR 2-87, 3-105, 5-19
N.ION 2-91, 3-105
NLOC.EXP 3-107
NLOC.PRE 3-107
PD.FULL 2-102
PD.PTIME 2-102
PEARSON 2-78, 3-104
PERIODIC 2-103, 3-106
PHOSPHOR 2-77, 3-104
POLY.GSZ 3-107
PRINT 3-108
REC.FRAC 2-96, 3-106
REFLECT 2-103, 3-106
ROTATION 2-85, 3-104
RP.EFF 2-84, 3-104
SEED 3-105
SURV.RAT 3-107
TAUR.EXE 3-107
TAUR.MOD 3-107
TEMPERAT 2-96, 3-105
THRESHOL 2-96, 3-106
TILT 2-77, 3-104, 5-19
VACUUM 2-103, 3-107
VIBRATIO 2-96, 3-106
X.RMS 2-96, 3-106
 parameters, **IMPURITY** statement
/MATERIA 2-29, 3-246
/RATIO.0 2-32, 3-247
/RATIO.E 2-32, 3-248
/SEG.0 2-32, 3-247
/SEG.E 2-32, 3-247
/SEG.EQ2 2-32, 3-248
/SEG.SS 3-248
/TRANS.0 3-247
/TRANS.E 3-247
ACCEPTOR 3-240
ACT.MIN 2-129, 3-250
ALPHA 2-115
AT.NUM 3-240
AT.WT 3-240
CG.MAX 2-114, 3-245
CHAN.CRI 2-95, 3-250
CHAN.FAC 2-95, 3-250
CL.INI.A 3-244
CM.SEC 3-251
C.STATE 3-242
CTN.0 2-27, 3-244
CTN.E 2-27, 3-244
CTN.F 2-27, 3-244
DDC.F.0 3-244
DDCF.D.N 3-244
DDC.F.E 3-244

DDCF . I . N 3-245
 DDCF . N . N 3-245
 DDCR . I . N 3-245
 DDCR . N . N 3-245
 DDCS . O 3-245
 DDCS . E 3-245
 DDCS . N 3-245
 DDC . T . O 3-244
 DDC . T . E 3-244
 DIC . O 2-21, 3-242
 DIC . E 2-21, 3-242
 DI . FAC 3-242
 DIFFUSE 3-242
 DIM . O 3-241
 DIM . E 3-241
 DIMM . O 3-241
 DIMM . E 3-241
 DIP . O 3-241
 DIPAIR . O 2-22, 3-243
 DI . PAIR . E 2-22
 DIPAIR . E 3-243
 DIP . E 3-241
 DISP . FAC 2-97, 3-250
 DIX . O 2-70, 3-241
 DIX . E 2-70, 3-241
 D . MODEL 3-242
 DONOR 3-240
 D . PF 3-250
 D . PNORM 3-250
 DVC . O 2-21, 3-242
 DVC . E 2-21, 3-242
 DV . FAC 3-242
 DVM . O 3-242
 DVM . E 3-242
 DVMM . O 3-242
 DVMM . E 3-242
 DVP . O 3-241
 DVPAIR . O 2-22, 3-243
 DVPAIR . E 2-22, 3-243
 DVP . E 3-241
 DVX . O 3-241
 DVX . E 3-242
 E . IP . V 3-243
 E . I . S 3-243
 ES . 100 2-95, 3-249
 ES . 110 2-95, 3-249
 ES . BREAK 2-93, 3-249
 ES . F . 100 2-95, 3-249
 ES . F . 110 2-95, 3-249
 ES . F . H 2-93, 3-249
 ES . F . RAN 2-93, 3-249
 ES . RAND 2-93, 3-248
 E . VP . I 3-244
 E . V . S 3-243
 FGB 2-119, 3-243
 GSEG . O 3-246
 GSEG . E 3-246
 GSEG . INI 2-114, 3-246
 IFRACM 3-245

IFRACS 3-245
 IMP . ACT 3-240
 IMP . GB 3-240
 IMP . IT 3-240
 IMPL . TAB 3-241
 IMPURITY 3-240
 LOC . FAC 2-93, 3-250
 MATERIAL 2-26, 3-241
 MODEL 3-240
 NEW 3-240
 NLOC . EXP 2-93, 3-249
 NLOC . K 2-93, 3-250
 NLOC . MAX 2-93, 3-250
 NLOC . PRE 3-249
 NLOC . PREN 2-93
 Q . INI . O 2-32, 3-248
 Q . INI . E 2-32, 3-248
 Q . MAX . O 2-32, 3-248
 Q . MAX . E 2-32, 3-248
 Q . SITES 2-114, 3-245
 RATIO . O 2-32, 3-247
 RATIO . E 2-32, 3-247
 R . IP . V 3-243
 R . I . S 3-243
 R . VP . I 3-243
 R . V . S 3-243
 SEG . O 2-29, 3-246
 SEG . E 2-29, 3-246
 SEG . EQ2 2-32, 3-248
 SEG . EQ3 3-248
 SEG . O 2-32
 SEG . SEGSS 2-32
 SEG . SS 3-247
 SOLVE 3-240
 SS . CLEAR 2-26, 3-244
 SS . CONC 2-26, 3-244
 SS . TEMP 2-26, 3-244
 STEADY 2-70, 3-241
 T . ACT . O 2-129, 3-250
 T . ACT . E 2-129
 TIF . NAME 3-240
 TRANS . O 2-29, 3-246
 TRANS . E 2-29, 3-247
 TWO . PHAS 2-32, 3-248
 VELIF . O 3-246
 VELIF . E 3-246

parameters, INITIALIZE statement

<100> 3-64
 <110> 3-64
 <111> 3-64
 ANTIMONY 3-65
 ARSENIC 3-65
 BORON 3-65, 4-19
 CONCENTR 3-66
 DX 2-5, 3-64
 FLIP . Y 3-64
 I . CONC 3-65
 IMPURITY 3-65
 IN . FILE 2-6, 3-64, 4-9

I . RESIST 3-65
 LINE . DAT 3-65
 MATERIAL 3-65
 ORIENTAT 3-65
 PHOSPHOR 3-66
 RATIO 2-3, 3-65
 RESISTIV 3-66
 ROT . SUB 2-95, 3-65
 SCALE 3-64
 TIF 3-64
 WIDTH 2-5, 3-64, 4-19
 X . ORIENT 2-95, 3-65

parameters, InterfaceCondition command

Material1 6-13
 Material2 6-13
 Symmetric 6-13

parameters, INTERMEDIATE statement

DEFAULT 3-32
 DELETE 3-32
 EXPRESS 3-32
 MININDEX 3-32
 NAME 3-32
 PRINT 3-32
 SIZE 3-32

parameters, INTERSTITIAL statement

/AMBIENT 3-271
 /MATERIA 3-271
 /NITRIDE 3-271
 /OXIDE 3-271
 /OXYNITR 3-271
 /POLYSIL 3-271
 /SILICON 3-271
 A . O 2-41, 3-273
 A . E 2-41, 3-273
 AMBIENT 3-266
 C . ALL 3-268
 CEQUIL . O 2-33, 3-266
 CEQUIL . E 2-33, 3-266
 CL . CF 2-43, 3-270
 CL . CR 2-43, 3-271
 CL . IFC 2-43, 3-270
 CL . IFI 2-43, 3-270
 CL . ISFC 2-43, 3-270
 CL . ISFI 2-43, 3-270
 CL . KFC . O 2-43, 3-270
 CL . KFC . E 2-43, 3-270
 CL . KFCI 3-270
 CL . KFI . O 2-43, 3-269
 CL . KFI . E 2-43, 3-270
 CL . KR . O 2-43, 3-270
 CL . KR . E 2-43, 3-270
 CL . MODEL 3-266
 CM . SEC 3-274
 C . STATE 2-33, 3-268
 C . V 3-268
 D . O 2-35, 3-266
 DC . O 3-268
 DC . E 3-268

D.E 2-35, 3-266
 DNEG.O 3-267
 DNEG.E 3-267
 DPOS.O 3-267
 DPOS.E 3-267
 ECLUST.O 3-269
 ECLUST.E 3-269
 ECLUST.N 3-271
 FRAC.O 3-268
 FRAC.E 3-268
 F.TRAP.O 2-42, 3-269
 F.TRAP.E 2-42, 3-269
 GPOW.O 2-40, 3-274
 GPOW.E 2-40, 3-274
 GROWTH 2-39, 3-273
 KB.O 2-36, 3-266
 KB.E 2-36, 3-266
 KB.HIGH 2-37, 3-266
 KB.LOW 2-37, 3-266
 KB.MED 2-37, 3-266
 KCV.O 3-269
 KCV.E 3-269
 KIV.O 3-268
 KIV.E 3-268
 KIV.NORM 3-266
 KLOOP.O 2-127, 3-271
 KLOOP.E 2-127, 3-271
 KPOW.O 2-38, 3-273
 KPOW.E 2-38, 3-273
 KSRAT.O 2-38, 3-272
 KSRAT.E 2-38, 3-272
 KSURF.O 2-38, 3-272
 KSURF.E 2-38, 3-272
 KSVEL.O 2-38, 3-272
 KSVEL.E 2-38, 3-272
 K.TRAP 2-42
 K.TRAP.O 3-269
 K.TRAP.E 2-42, 3-269
 MATERIAL 3-265
 N.MAX.O 3-274
 N.MAX.E 3-274
 NEG.O 3-267
 NEG.E 3-267
 NEU.O 3-267
 NEU.E 3-267
 NITRIDE 3-265
 OXIDE 3-265
 OXYNITRI 3-265
 POLYSILI 3-266
 POS.O 3-267
 POS.E 3-267
 RLMIN 3-271
 SILICON 3-265
 TO.O 2-41, 3-273
 TO.E 2-41, 3-273
 THETA.O 2-40, 3-273
 THETA.E 2-40, 3-273
 TRAP.CON 2-41, 2-42, 3-269
 V.INITOX 2-38, 3-272

V.MAXOX 2-38, 3-271
 VMOLE 2-40, 3-267
 V.NORM 2-38, 3-272
 VNORM.O 2-39, 3-272
 VNORM.E 2-39, 3-272
 parameters, LABEL statement
 CENTER 3-156, 5-37
 C.LINE 3-156
 CM 3-155, 4-7
 COLOR 3-156
 C.RECTAN 3-157
 C.SYMBOL 3-157
 H.RECTAN 3-157, 4-18
 LABEL 3-156
 LEFT 3-156
 LENGTH 3-156
 LINE.TYP 3-156
 RECTANGL 3-157
 RIGHT 3-156, 5-37
 SIZE 3-156, 4-18
 SYMBOL 3-157
 W.RECTAN 3-157, 4-18
 X 3-155, 4-7
 X.CLICK 3-155
 Y 3-155, 4-7
 Y.CLICK 3-155
 parameters, LINE statement
 LOCATION 2-3, 3-55
 SPACING 2-3, 3-55, 4-13
 TAG 3-55, 5-45
 X 3-55, 4-13
 Y 3-55, 4-13
 parameters, L.MODIFY statement
 BREAK 3-22
 NEXT 3-22
 STEPS 3-22
 parameters, LOADFILE statement
 DEPICT 3-68
 FLIP.Y 3-68
 IN.FILE 3-68
 SCALE 3-68
 TIF 3-68
 parameters, Loop command
 Index 6-10
 Name 6-10
 Start 6-10
 Steps 6-10
 parameters, LOOP/L.END statement
 DNORM 3-18
 DSSQ 3-18
 INDEX 3-18
 OPTIMIZE 3-18
 PLOT 3-18
 START 3-18
 STEPS 3-18
 parameters, MASK statement
 DX.MAX 2-5
 G.EXTENT 3-82
 GRID 2-5, 3-82

IN.FILE 3-82
 PRINT 3-82, 5-4
 SCALE 3-82
 parameters, MATERIAL statement
 "CL.IMI.A 2-129
 AFFINITY 2-122, 3-231
 ALPHA 3-234
 ALUMINUM 2-78, 3-229
 AMBIENT 3-229
 AT.NUM 2-70, 3-230
 AT.WT 2-70, 3-230
 BANDGAP 2-121, 3-231
 BOLTZMAN 2-121, 3-232
 CONDUCTO 3-232
 DAM.GRAD 3-232
 DENSITY 2-68, 3-230
 DLGX.O 2-118, 3-236
 DLGX.E 2-118, 3-236
 D.PCONST 3-233
 DSIM.O 3-235
 DSIM.E 2-117, 3-235
 DSIMM.O 2-117, 3-235
 DSIMM.E 2-117, 3-235
 DSIP.O 2-117, 3-235
 DSIP.E 2-117, 3-235
 DSIX.O 2-117, 3-235
 DSIX.E 2-117, 3-235
 DY.DEFAU 3-229
 E.ACCEP 2-122, 3-232
 EAVEL.O 2-118, 3-236
 EAVEL.E 2-118, 3-236
 E.DONOR 2-122, 3-232
 E.FIELD 3-229
 EGALPH 3-231
 EGBETA 3-231
 EPSILON 2-120, 3-230
 E.SEG 2-114
 F11 2-113, 3-234
 F22 2-113, 3-234
 FRAC.TA 2-115, 3-234
 G.ACCEP 2-122, 3-232
 GAMMA.O 2-116, 3-234
 GAMMA.E 2-116, 3-234
 GBE.O 2-117, 3-235
 GBE.1 2-117, 3-235
 GBE.H 2-117, 3-235
 G.DENS 2-113, 3-234
 G.DONOR 2-122, 3-232
 GEOM 2-116, 3-234
 GRASZ.O 2-115, 3-234
 GRASZ.E 2-115, 3-234
 GSEG.O 2-114
 IMPL.TAB 2-80, 3-229
 INTRIN.S 2-74, 3-231
 IONIZATI 2-121, 3-232
 ION.PAIR 2-19, 3-229
 IP.OMEGA 2-19, 3-229
 LCTE 2-73, 2-76, 3-231
 MATERIAL 3-228

- MAX.DAMA** 3-232
MD.INDEX 3-229
MIN.GRAI 2-115, 3-234
MOL.WT 2-68, 3-230
NC.F 3-232
N.CONDUC 2-121, 3-231
NEW 3-228
NI.O 2-20, 3-229
NI.E 2-20, 3-230
NI.F 2-20, 3-230
NITRIDE 2-78, 3-229
NSEG 2-117, 3-235
N.VALENC 2-121, 3-231
NV.F 3-232
OXIDE 2-78, 3-228
OXYNITRI 3-228
PHOTORES 2-78, 3-229
POISS.R 2-57, 3-231
POLYCRYS 2-119, 3-233, 5-7
POLY.FAC 3-236
POLYSILI 2-78, 3-229
QM.BETA 2-125, 3-232
QM.YCRIT 2-125, 3-232
SEMICOND 3-231
SILICON 2-78, 3-228
SURF.TEN 2-63, 3-231
TBU.O 2-118, 3-235
TBU.E 2-118, 3-236
TEMP.BRE 2-115, 3-234
TEMPERAT 3-230
TIF.NAME 3-229
TOXIDE 2-118, 3-236
VC 3-230, 5-28
VELIF.O 2-115
VELIF.E 2-115
VISC.O 2-58, 3-230
VISC.E 2-58, 3-230
VISC.X 2-58, 3-230
WORKFUNC 2-122, 3-232
YOUNG.M 2-57, 3-231
- parameters, **MESH** statement
- DX.MAX** 2-5, 3-50
DX.MIN 2-5, 3-50
DX.RATIO 2-5, 3-50
DY.ACTIV 2-6, 3-51
DY.BOT 2-6, 3-51
DY.RATIO 2-6, 3-51
DY.RATION 2-6
DY.SURF 2-6, 3-50, 5-32
ERR.FAC 5-4
FAST 2-12, 3-51
GRID.FAC 2-3, 3-50, 4-13, 5-3, 5-32
LY.ACTIV 2-5, 2-6, 3-50, 5-32
LY.BOT 2-6, 3-51
LY.SURF 2-5, 3-50, 5-32
- parameters, **METHOD** statement
- ABS.ADAP** 2-10, 3-195
ABS.ERR 2-17, 3-194
ACT.EQUI 2-28, 3-191
ACT.FULL 3-191
ACT.TRAN 2-28, 3-191
ALUMINUM 3-194
ANTIMONY 3-192
ARSENIC 3-192
BACK 3-192
BLK.ITLI 3-192
BORON 3-193
CG 3-191
COMPRESS 2-57, 3-189
CONTIN.M 3-195
DIF.ADAP 2-11, 3-190
DY.EXACT 3-189
DY.LOCAL 3-189
DY.OXIDE 2-8, 3-189, 4-14
E.AVCAP 3-196
E.DSURF 3-196
E.ITMAX 2-122, 3-195
E.ITMIN 2-122, 3-195
ENABLE 3-197
E.REGRID 2-120, 3-196
E.RELERR 2-122, 3-196
ERF1 2-51, 3-188
ERF2 2-51, 3-188
ERFC 2-48, 3-188
ERFG 2-48, 3-188
ERR.FAC 2-10, 3-191, 4-4
ERROR 3-194
E.RSURF 3-196
E.RVCAP 3-196
E.TSURF 3-196
E.USEAVC 3-196
FORMULA 3-191
FULL 3-193
GAUSS 3-191
GRID.OXI 2-8, 3-189
HYBRID 3-191
ICCG 3-193
IMP.ADAP 2-11, 3-190
IMPURITY 3-192
INIT.TIM 2-16, 3-191
INTERSTI 3-192
IT.ACT 2-29, 3-197
IT.CPL 2-30, 3-196
ITRAP 2-29, 3-196
IT.STEAD 2-30, 3-197
IT.THERM 2-30, 3-197
IT.ZERO 2-30, 3-197
LU 3-193
MATERIAL 3-194
MF.DIST 3-192
MF.METH 3-192
MILNE 3-191
MIN.FILL 3-192
MIN.FREQ 3-192
MIN.SPAC 2-10, 2-11, 3-195
MOB.AROR 2-124, 3-196
MOB.CAUG 2-124, 3-196
MOB.TABL 2-123, 3-196
MODEL 3-197
NEWTON 3-194
NITRIDE 3-194
NONE 3-193
NSTREAMS 2-21, 3-190
OX.ADAP 2-11
OX.ADAPT 3-190
OXIDANT 3-193
OXIDE 3-194
OX.REL 3-195
OXYNITRI 3-194
PAIR.GRA 2-24, 3-190
PAIR.REC 2-21, 3-190
PAIR.SAT 2-21, 3-190
PART 3-193
PD.FERMI 2-69, 3-189
PD.FULL 2-21, 3-190
PD.PFLUX 2-24, 3-190
PD.PREC 2-32, 3-190
PD.PTIME 2-32, 3-190
PD.TRANS 2-21, 3-189
pd.TRANS 4-19
PHOSPHOR 3-193
PHOTORES 3-194
POLYSILI 3-194
REL.ADAP 2-10, 3-195
REL.ERR 2-17, 3-194
SILICON 3-194
SIP 3-193
SKIP.SIL 2-62, 3-189
SOR 3-193
ST.HISTO 2-59, 3-189
SYMMETRY 3-193
TIME 2-16
TIME.STE 3-193
TRAP 3-193
TRBDF 3-191
VACANCY 3-192
VERTICAL 2-56, 3-188
VE.SMOOT 2-62, 3-195
VISCOELA 2-60, 3-189, 4-14
VISCOUS 2-59, 3-189
- parameters, **MOBILITY** statement
- ALPHAN** 3-262
ALPHAP 3-262
AN 3-260
AP 3-261
CN 3-260
CP 3-261
ECN.MU 2-123, 3-260
ECP.MU 2-123, 3-260
EXN 3-260
EXN2 3-260
EXN3 3-260
EXN4 3-260
EXP1 3-261
EXP2 3-261
EXP3 3-261

- EXP4 3-261
- GSURFN 2-123, 3-260
- GSURFP 2-123, 3-260
- KELVIN 3-259
- MUN1 3-260
- MUN2 3-260
- MUN.MAX 3-261
- MUN.MIN 3-261
- MUP1 3-261
- MUP2 3-261
- MUP.MAX 3-262
- MUP.MIN 3-262
- NREFN 3-261
- NREFP 3-262
- NUN 3-261
- NUP 3-262
- TAB.CLEA 3-259
- TAB.CONC 3-259
- TAB.E.MU 3-259
- TAB.H.MU 3-259
- TAB.TEMP 3-259
- XIN 3-261
- XIP 3-262
- parameters, **Model** command
 - Array 6-8
 - DataType 6-8
 - Expression 6-7
 - Loop 6-8
 - Model 6-8
 - Name 6-7
 - Parameter 6-8
- parameters, **MOMENT** statement
 - CLEAR 3-224
 - D.FRAC 3-225
 - D.GAMMA 3-225
 - D.KURTOS 3-225
 - D.LSIGMA 3-225
 - D.LSLOPE 3-225
 - D.RANGE 3-225
 - D.SIGMA 3-225
 - GAMMA 3-224
 - KURTOSIS 3-225
 - LSIGMA 3-225
 - LSLOPE 3-225
 - MATERIAL 3-224
 - NITRIDE 3-224
 - OXIDE 3-224
 - OXYNITRI 3-224
 - PHOTORES 3-224
 - POLYSILI 3-224
 - RANGE 3-224
 - SIGMA 3-224
 - SILICON 3-224
- parameters, **OPTION** statement
 - DEBUG 3-39
 - DEVICE 3-39, 4-6
 - DIAGNOST 3-39
 - ECHO 3-39
 - EXECUTE 3-39
- INFORMAT 3-39
- NORMAL 3-39
- PLOT.OUT 3-39
- QUIET 3-39
- V.COMPAT 2-97, 3-39
- VERBOSE 3-39
- parameters, **Parameter** command
 - Boundary 6-9
 - Default 6-9
 - Name 6-9
- parameters, **PHOSPHORUS** statement
 - /AMBIENT 3-305
 - /MATERIA 3-305
 - /NITRIDE 3-305
 - /OXIDE 3-305
 - /OXYNITR 3-305
 - /POLYSIL 3-305
 - /SILICON 3-305
 - AMBIENT 3-302
 - CM.SEC 3-306
 - DIM.O 2-20, 3-303
 - DIM.E 2-20, 3-303
 - DIMM.O 2-20, 3-303
 - DIMM.E 3-303
 - DIP.O 2-20
 - DIPAIR.O 2-22, 3-304
 - DIPAIR.E 2-22, 3-304
 - DIP.E 2-20
 - DIX.O 2-20, 3-303
 - DIX.E 2-20, 3-303
 - DVM.O 2-20, 3-303
 - DVM.E 2-20, 3-303
 - DVMM.O 2-20, 3-303
 - DVMM.E 2-20, 3-303
 - DVP.O 2-20
 - DVPAIR.O 2-22, 3-304
 - DVPAIR.E 2-22, 3-304
 - DVP.E 2-20
 - DVX.O 2-20, 3-303
 - DVX.E 2-20, 3-303
 - E.IP.V 2-23, 3-304
 - E.I.S 3-304
 - E.I.S. 2-23
 - ES.100 3-306
 - ES.110 3-306
 - ES.F.100 3-306
 - ES.F.110 3-306
 - ES.F.RAN 3-306
 - ES.RAND 3-306
 - E.VP.I 2-23, 3-304
 - E.V.S 3-304
 - E.V.S. 2-23
 - MATERIAL 3-302
 - NITRIDE 3-302
 - OXIDE 3-302
 - OXYNITRI 3-302
 - POLYSILI 3-302
 - R.IP.V 2-23, 3-304
 - R.I.S 3-304
- R.I.S. 2-23
- R.VP.I 2-23, 3-304
- R.V.S 3-304
- R.V.S. 2-23
- SEG.O 3-305
- SEG.E 3-305
- SILICON 3-302
- SS.CLEAR 3-305
- SS.CONC 3-305
- SS.TEMP 3-305
- TRANS.O 3-305
- TRANS.E 3-305
- parameters, **PLOT .1D** statement
 - /ALUMINU 3-138
 - /AMBIENT 3-138
 - /MATERIA 3-137
 - /NITRIDE 3-137
 - /OXIDE 3-137
 - /OXYNITR 3-137
 - /PHOTORE 3-138
 - /POLYSIL 3-137
 - /REFLECT 3-138
 - /SILICON 3-137
 - ALUMINUM 3-137
 - AXES 3-140, 4-7
 - BOTTOM 3-141, 4-24
 - BOUNDARY 3-139
 - CLEAR 3-139, 4-7
 - COLOR 3-140, 4-7
 - COLUMN 3-138
 - CURVE 3-140
 - ELECTRIC 3-139
 - IN.FILE 3-138
 - LEFT 3-140
 - LINE.TYP 3-140
 - LINE.TYPE 4-7
 - MATERIAL 3-136
 - NITRIDE 3-137
 - OXIDE 3-137
 - OXYNITRI 3-137
 - PHOTORES 3-137
 - POLYSILI 3-137
 - RIGHT 3-141, 4-24
 - ROW 3-138
 - SILICON 3-137
 - SYMBOL 3-140
 - TIF 3-138
 - TOP 3-141
 - T.SIZE 3-141
 - X.AXIS 3-138
 - X.COLUMN 3-138
 - X.LABEL 3-139
 - X.LENGTH 3-141
 - X.LOG 3-139
 - X.OFFSET 3-141
 - X.ROW 3-138
 - X.SCALE 3-139
 - X.SHIFT 3-139
 - X.SIZE 3-141

- X.VALUE** 3-136, 4-7
Y.AXIS 3-138
Y.COLUMN 3-138
Y.LABEL 3-139
Y.LENGTH 3-141
Y.LOG 3-139
Y.OFFSET 3-141
Y.ROW 3-139
Y.SCALE 3-139
Y.SHIFT 3-139
Y.SIZE 3-141
Y.VALUE 3-136, 4-7
- parameters, **PLOT . 2D** statement
- AXES** 3-144, 4-18
BOUNDARY 3-145
C.BOUND 3-145
C.COMPRES 3-145, 4-16
C.GRID 3-145, 4-14
CLEAR 3-144, 4-26
C.TENSIO 3-146, 4-16
DIAMONDS 3-146
FLOW 3-145, 4-15
GRID 3-145, 4-14
L.BOUND 3-145
L.COMPRES 3-145
L.GRID 3-145
L.TENSIO 3-146
SCALE 3-144, 4-14
STRESS 3-145, 4-16
T.SIZE 3-146
VLENG 3-145, 4-15
VMAX 3-145
X.LENGTH 3-146
X.MAX 3-144, 4-15
X.MIN 3-144, 4-15
X.OFFSET 3-146
X.SIZE 3-146
Y.LENGTH 3-146
Y.MAX 3-144, 4-15, 5-33
Y.MIN 3-144, 4-15
Y.OFFSET 3-146
Y.SIZE 3-146
- parameters, **PLOT . 3D** statement
- BOUNDARY** 3-153
C.BOUND 3-153
CLEAR 3-152
COLOR 3-153
L.BOUND 3-153
LINE.TYP 3-153
NUM.CNTR 3-153
PHI 3-152
THETA 3-152
X.MAX 3-152
X.MIN 3-152
Y.MAX 3-152
Y.MIN 3-152
Z.MAX 3-153
Z.MIN 3-152
- parameters, **PRINT . 1D** statement
- /ALUMINU** 3-133
/AMBIENT 3-133
/MATERIA 3-133
/NITRIDE 3-133
/OXIDE 3-133
/OXYNITR 3-133
/PHOTORE 3-133
/POLYSIL 3-133
/REFLECT 3-133
/SILICON 3-133
ALUMINUM 3-133
APPEND 3-133
LAYERS 3-133, 4-8, 5-42
MATERIAL 3-132
NITRIDE 3-132
OUTFILE 3-133
OXIDE 3-132
OXYNITRI 3-132
PHOTORES 3-132
POLYSILI 3-132
SILICON 3-132
SPOT 3-133, 4-31
X.MAX 3-134
X.MIN 3-134
X.VALUE 3-132
Y.VALUE 3-132
- parameters, **PROFILE** statement
- ANTIMONY** 3-84
ARSENIC 3-84
BORON 3-84
IMPURITY 3-84
IN.FILE 3-84
OFFSET 3-84
PHOSPHOR 3-84
REPLACE 3-84
- parameters, **REACTION** statement
- /EI.L** 3-255
/EI.R 3-255
/IMP.L 3-255
/IMP.R 3-255
/MAT.L 3-254
/NI.L 3-255
/NI.R 3-255
/NM.L 3-255
DELETE 3-254
EI.L 3-255
EI.R 3-255
EQUIL.O 3-256
EQUIL.E 3-256
IMP.L 3-254
IMP.R 3-255
MAT.BULK 3-254
MAT.NEW 3-256
MAT.R 3-254
MODEL 3-254
NAME 3-254
NI.L 3-254
NI.R 3-255
NM.R 3-255
- RATE.O** 3-255
RATE.E 3-255
REPLACE 3-254
THICKNES 3-256
- parameters, **REGION** statement
- ALUMINUM** 3-62
MATERIAL 3-62
NITRIDE 3-62
OXIDE 3-62
OXYNITRI 3-62
PHOTORES 3-62
POLYSILI 3-62
SILICON 3-62
XHI 3-62
XLO 3-62
YHI 3-62
YLO 3-62
- parameters, **SAVEFILE** statement
- ACTIVE** 3-73
CHEMICAL 3-73
DEFECT 3-73
DEPICT 3-72
DX.MIN 3-73
DY.MIN 3-73
ELEC.BOT 3-72
FLIP.Y 3-71
FULL.DEV 3-72
HALF.DEV 3-72
MEDICI 3-72
MINIMOS5 3-72
MISC 3-73
OUT.FILE 3-71
OXID 3-73
POLY.ELE 3-72
SCALE 3-71
TEMPERAT 3-71
TIF 3-71
TIF.VERS 3-71
WAVE 3-73
X.CHANNE 3-72
X.MASK.D 3-72
X.MASK.S 3-72
X.MAX 3-73
X.MIN 3-72
Y.MAX 3-73
Y.MIN 3-73
- parameters, **SELECT** statement
- LABEL** 3-128
TEMPERAT 3-128
TITLE 3-128
Z 3-128, 4-8, 5-28
- parameters, **Selector** command
- Default** 6-14
Name 6-14
- parameters, **STRESS** statement
- NEL** 3-124
TEMP1 2-76, 3-124
TEMP2 2-76, 3-124
- parameters, **STRUCTURE** statement

BOTTOM 3-77
DX 2-7, 3-78
EXTEND 2-7, 3-78
LEFT 3-77
REFLECT 3-77, 5-19
RIGHT 3-77
SPACES 3-78
TEMPERAT 3-79
TOP 3-77
TRUNCATE 3-77, 5-33
WIDTH 2-7, 3-78
X 3-77, 3-78, 3-79
XDX 2-7, 3-78
Y 3-77
Y.ELIM 3-78
 parameters, **VACANCY** statement
 /**AMBIENT** 3-280
 /**MATERIA** 3-279
 /**NITRIDE** 3-279
 /**OXIDE** 3-279
 /**OXYNITR** 3-279
 /**POLYSIL** 3-280
 /**SILICON** 3-279
A.0 2-41, 3-281
A.E 2-41, 3-281
AMBIENT 3-276
C.ALL 3-278
CEQUIL.0 2-33, 3-277
CEQUIL.E 2-33, 3-277
C.I 3-278
CM.SEC 3-282
C.STATE 2-33, 3-278
D.0 2-35, 3-277
DC.0 3-278
DC.E 3-278
D.E 2-35, 3-277
DNEG.0 3-277
DNEG.E 3-277
DPOS.0 3-278
DPOS.E 3-278
ECLUST.0 3-279
ECLUST.E 3-279
ECLUST.N 3-279
FRAC.0 3-278
FRAC.E 3-279
GPOW.0 2-40, 3-282
GPOW.E 2-40, 3-282
GROWTH 2-39, 3-281
KIC.0 3-279
KIC.E 3-279
KPOW.0 2-38, 3-282
KPOW.E 2-38, 3-282
KSRAT.0 2-38, 3-280
KSRAT.E 2-38, 3-281
K.SURF.0 2-38
KSURF.0 3-280
KSURF.E 2-38, 3-280
KSVEL.0 2-38, 3-280
KSVEL.E 2-38, 3-280

MATERIAL 3-276
NEG.0 3-277
NEG.E 3-277
NEU.0 3-277
NEU.E 3-277
NITRIDE 3-276
OXIDE 3-276
OXYNITRI 3-276
POLYSILI 3-276
POS.0 3-278
POS.E 3-278
SILICON 3-276
TO.0 2-41, 3-281
TO.E 2-41, 3-281
THETA.0 2-40, 3-281
THETA.E 2-40, 3-281
V.INITOX 2-38, 3-280
V.MAXOX 2-38, 3-280
VMOLE 3-277
V.NORM 2-38, 3-280
VNORM.0 2-39, 3-281
VNORM.E 2-39, 3-281
 parameters, **VIEWPORT** statement
 X.MAX 3-184
 X.MIN 3-184
 Y.MAX 3-184
 Y.MIN 3-184
PAUSE 3-14
 description 3-14
 example 3-14
PD.FERMI model 2-21
PD.FULL model 2-21
PD.TRANS model 2-21
PD.TRANS parameter, **METHOD**
 statement
 local oxidation example 4-19
 Pearson distribution 2-80
PHOSPHORUS 3-302
 additional notes 3-307
 description 3-306
 examples 3-307
 photoresist
 masking, exposure, and
 development 2-107
 physical functions 6-29
 physical model and equation interface 6-3
 flat structure syntax 6-5
 functions and operators 6-14
 PMEI commands 6-6
 PMEI functions 6-15
 syntax 6-3
 plot device definition file *s4pcap* 1-9
 plot produced by
 PLOT.2D FLOW statement input
 file *s4ex2a.inp* Fig. 4-16
 PLOT.2D STRESS statement input
 file *s4ex2a.inp* Fig. 4-17
PLOT.1D 3-136

AXES parameter, 1D bipolar
 example 4-7
CLEAR parameter, 1D bipolar
 example 4-7
COLOR parameter, 1D bipolar
 example 4-7
 description 3-141
 examples 3-143
IN.FILE parameter 3-142
 line type and color 3-142
LINE.TYP parameter, 1D bipolar
 example 4-7
X.VALUE parameter, 1D bipolar
 example 4-7
Y.VALUE parameter, 1D bipolar
 example 4-7
PLOT.2D 3-144
 description 3-146
 examples 3-147
 line type and color 3-147
 X.MIN parameter, local oxidation
 example 4-15
PLOT.3D 3-152
 additional notes 3-154
 description 3-153
 examples 3-154
 line type and color 3-154
 plotting results, Monte Carlo method
 boron contours 5-23
 vertical profiles 5-24
 plotting the results 4-6
 after implant 5-56
 doping and grain size 5-57
 doping vs. stripe width 5-58
 labels 4-7
 PLOT.1D statement 4-7
 SELECT statement 4-6
 specifying a graphics device 4-6
 PMEI
 commands 6-6
 flat structure syntax 6-5
 function and operators 6-14
 functions 6-15
 syntax 6-3
 PMEI commands 6-6
 Array 6-9
 BoundaryCondition 6-11
 Equation 6-6
 Expression 6-8
 ImplantModel 6-13
 InitialCondition 6-11
 InterfaceCondition 6-12
 Loop 6-10
 Model 6-7
 Parameter 6-9
 Selector 6-14
 PMEI functions 6-15
 Condition 6-16
 Exist 6-16

- OnMat1** 6-16
 - OnMat2** 6-16
 - proposed language additions 6-16
 - Sum** 6-15
 - point defect diffusion equations 2-33
 - point defect models example
 - choosing 4-32
 - creating the test structure 4-29
 - LABEL** statement 4-30
 - LOADFILE** statement 4-29
 - overview 4-27
 - oxidation and plotting of impurity profiles 4-29
 - PD.FERMI** model 4-27
 - PD.FULL** model 4-27
 - PD.TRANS** model 4-27
 - point defect profiles 4-31
 - point defect parameters, default coefficients A-8
 - point defects
 - diffusion of 2-32
 - enhancement 2-21
 - injection and recombination at interfaces 2-37
 - injection rate 2-39
 - profiles plotted by *s4ex3.inp Fig.* 4-32
 - reaction of pairs 2-35
 - tutorial example of 4-18
 - Poisson's equation 2-120, 3-177, 3-195, 3-196
 - Poisson's ratio 2-57, 3-218
 - poly-buffered LOCOS example
 - COMPRESS** model 5-28
 - METHOD** statement 5-28
 - overview 5-27
 - plotting results 5-28
 - structure generation 5-27
 - VISCOEL** model 5-28
 - VISCOUS** model 5-28
 - POLYCRYSTALS** parameter, **MATERIAL** statement
 - NMOS LDD process example 5-7
 - polycrystalline materials, modeling
 - advanced example of 5-27
 - epitaxial regrowth 2-117
 - grain interior and boundaries, segregation 2-114
 - grain size model 2-115
 - interface oxide breakup 2-117
 - overview 2-112
 - using the model 2-32, 2-119
 - polysilicon
 - enhancement 2-21
 - oxidation 2-63
 - polysilicon emitter study example
 - overview 5-54
 - PD.FERMI** diffusion model 5-54
 - PD.FULL** model 5-54
 - plotting results 5-56
 - process simulation 5-54
 - polysilicon grain segregation, default coefficients A-6
 - polysilicon Monte Carlo implant model 2-103
 - ion implantation into silicon carbide 2-104
 - PRINT** parameter, **MASK** statement
 - NMOS LDD process example 5-4
 - PRINT . 1D** 3-132
 - description 3-134
 - examples 3-135
 - interface values 3-134
 - layers 3-134
 - LAYERS** parameter
 - 1D bipolar example 4-8
 - DMOS power transistor example 5-42
 - point defect models example 4-31
 - printed output 1-2
 - printing layer information 4-8
 - PRINT . 1D** statement 4-8
 - using **PRINT . 1D** layers 4-8
 - problems and troubleshooting xxxvii
 - process simulation
 - models 5-54
 - processing 5-54
 - process steps, input statements
 - DEPOSITION** 3-90
 - DEVELOP** 3-97
 - DIFFUSION** 3-115
 - EPITAXY** 3-121
 - ETCH** 3-98
 - EXPOSE** 3-95
 - IMPLANT** 3-103
 - STRESS** 3-124
 - processing of SOI MOSFET, from input file *s4ex9.inp Fig.* 5-47
 - processing steps xxxv, 4-5
 - buried layer 4-5
 - buried layer masking oxide 4-5
 - epitaxial layer 4-5
 - pad oxide and nitride mask 4-6
 - processing the DMOS power transistor
 - gate processing 5-42
 - source processing 5-42
 - processing, CMOS process example 5-34
 - PROFILE** 3-84
 - description 3-84
 - example 3-86
 - IMPURITY** parameter 3-86
 - interpolation 3-85
 - OFFSET** parameter 3-85
 - profiles
 - active and total arsenic concentration through the poly gate *Fig.* 5-38
 - with **PD.FERMI** and **PD.FULL** models, from *s4ex3.inp Fig.* 4-30
 - program execution and output 1-1
 - proposed language additions 6-16
 - parameterized functions 6-16
 - physical functions 6-17
- ## Q
- QSS** parameter, **ELECTRICAL** statement
 - NMOS LDD process example 5-14
 - quantum effect 2-125
 - quantum effect in MOS capacitance *Fig.* 3-182
 - quantum mechanical model for MOSFET 2-125
- ## R
- REACTION** 3-254
 - defining and deleting 3-256
 - description 3-256
 - effects 3-258
 - insertion of native layers 3-256
 - parameters 3-257
 - reaction equation 3-257
 - reaction of pairs with point defects 2-35
 - reaction rate constants 2-23
 - references
 - default coefficients A-27
 - TSUPREM-4 models 2-131
 - REFLECT** parameter, **STRUCTURE** statement
 - trench implant simulation example 5-19
 - reflow 2-63, 3-119
 - REGION** 3-62
 - description 3-63
 - example 3-63
 - related publications
 - Data Visualizer Programming Guide* 1-7
 - TCAD Products and Utilities Installation Manual* xxxvii
 - RESISTAN** parameter, **ELECTRICAL** statement
 - NMOS LDD process example 5-14
 - RETURN** 3-10
 - description 3-10
 - example 3-11
 - exiting interactive input mode 3-10
 - returning from batch mode 3-10
 - RIGHT** parameter, **LABEL** statement
 - CMOS process example 5-37
 - RIGHT** parameter, **PLOT . 1D** statement
 - local oxidation example 4-24

S

saved structure files

Medici E-8
 MINIMOS 5 1-7
 older versions of TSUPREM-4 E-6
 overview 1-6, E-1
 TIF 1-7
 TSUPREM-4 1-7, E-1
 wave 1-7

SAVEFILE 3-71

description 3-74
 examples 3-76
 Medici files 3-75
 MINIMOS 3-76
 Taurus-Lithography files 3-75
 temperature 3-76
 TIF files 3-74
 TSUPREM-4 files 3-74

saving the structure 4-6

SCALE parameter, **PLOT . 2D** statement

local oxidation example 4-14

Schrödinger equation 2-125

second part of input file

s4ex10.inp, silicide growth *Fig. 5-51*
s4ex2a.inp, showing statements
 plotting results of LOCOS process
Fig. 4-15
s4ex2b.inp, showing statements for
 plotting the results of diffusion
 simulation *Fig. 4-22*
s4ex3.inp, using the full
 two-dimensional point defect
 model *Fig. 4-28*
s4ex4a.inp, for simulating an NMOS
 process *Fig. 5-6*
s4ex4b.inp, showing source/drain
 processing and metallization *Fig.*
 5-10
s4ex4d.inp, showing the MOS
 capacitance extraction *Fig. 5-16*
s4ex5.inp, showing tilted
 implantation using analytic implant
 model *Fig. 5-20*
s4ex6.inp, plotting final
 poly-buffered LOCOS structure
Fig. 5-29
s4ex7a.inp, showing statements for
 simulating CMOS process *Fig.*
 5-31
s4ex8.inp, processing of DMOS
 power transistor, through body
 diffusion *Fig. 5-41*

segregation and transport coefficients,
 default coefficients A-4

segregation coefficient 2-29

segregation flux 2-28

SELECT 3-128

description 3-128
 examples 3-131

mathematical operations and
 functions 3-130

solution values 3-128

TEMPERAT specified 3-130

SELECT statement 4-6

Z expression 4-7

Selector command 6-14

Default parameter 6-14

Name parameter 6-14

sidewall profiles

after Monte Carlo implant *Fig. 5-26*
 produced by analytic implant *Fig.*
 5-22

SiGe model 6-58

silicidation and oxidation 2-8

silicide models

advanced example of 5-50
 impurities and point defects 2-69
 other silicides 2-72
 overview 2-68
 TiSi₂ growth kinetics 2-68
 titanium silicide model 2-69
 tungsten silicide model 2-72

simple grids, initial checkout 5-1

simulation structure xxxv

advanced examples of 5-2, 5-18,
 5-27

coordinates, overview 2-1

initial structure, restrictions 2-2

materials, overview 2-2

regions, overview 2-2

SIZE parameter, **LABEL** statement

local oxidation example 4-18

SOI MOSFET example

COMPRESS model 5-48

INITIALIZE statement 5-47

LINE statement 5-46

mesh generation 5-45

overview 5-45

process simulation 5-48

REGION statement 5-45

VERTICAL model 5-48

VISCOEL model 5-48

solid solubility clustering model 2-31

solid solubility model 2-26

solid solubility tables 2-26

solving method 6-20

SOURCE 3-9

description 3-9

examples 3-9

generating templates 3-9

reusing combinations of statements
 3-9

SPACES parameter, **DEPOSITION**
 statement

SOI MOSFET example 5-46

SPACES parameter, **EPITAXY** statement

1D bipolar example 4-5

SPACING parameter, **LINE** statement

1D local oxidation example 4-13

specifying silicide models and parameters

impurities 2-70, 2-72

materials 2-70

reactions 2-70

SPOT parameter, **PRINT . 1D** statement

local oxidation example 4-31

standard output file *s4out* 1-6

starting TSUPREM-4 1-1

statements

AMBIENT 3-207

ANTIMONY 3-284

ARSENIC 3-290

ASSIGN 3-25

BORON 3-296

BOUNDARY 3-60

COLOR 3-150

COMMENT 3-8

CONTOUR 3-148

CPULOG 3-46

DEFINE 3-42

DEPOSITION 3-90

DEVELOP 3-97

DIFFUSION 3-115

ECHO 3-38

ELECTRICAL 3-174

ELECTRODE 3-87

ELIMINATE 3-57

EPITAXY 3-121

EQUATION 3-204

ETCH 3-98

EXPOSE 3-95

EXTRACT 3-160

FOREACH/END 3-16

HELP 3-47

IMPLANT 3-103

IMPURITY 3-239

INITIALIZE 3-64

INTERACTIVE 3-12

INTERMEDIATE 3-32

INTERSTITIAL 3-265

LABEL 3-155

LINE 3-55

LOADFILE 3-68

LOOP/L . END 3-18

MASK 3-82

MATERIAL 3-228

MESH 3-50

METHOD 3-188

MOBILITY 3-259

MOMENT 3-224

OPTION 3-39

PAUSE 3-14

PHOSPHORUS 3-302

PLOT . 1D 3-136

PLOT . 2D 3-144

PLOT . 3D 3-152

PRINT . 1D 3-132

- PROFILE** 3-84
 - REACTION** 3-254
 - REGION** 3-62
 - RETURN** 3-10
 - SAVEFILE** 3-71
 - SELECT** 3-128
 - SOURCE** 3-9
 - STOP** 3-15
 - STRESS** 3-124
 - STRUCTURE** 3-77
 - UNDEFINE** 3-45
 - VACANCY** 3-276
 - VIEWPORT** 3-184
 - STEAM** parameter, **DIFFUSION** statement
 - 1D bipolar example 4-5
 - STEAM** parameter, **ETCH** statement
 - 1D local oxidation example 4-14
 - STOP** 3-15
 - description 3-15
 - example 3-15
 - STRESS** 3-124
 - description 3-124
 - example 3-125
 - printing and plotting of stresses and displacements 3-124
 - reflecting boundary limitations 3-125
 - replace oxidation values 3-124
 - stress
 - calculation of 2-75
 - thermal mismatch 2-75
 - stress history model, initial conditions 2-73
 - STRESS** parameter, **PLOT . 2D** statement
 - local oxidation example 4-16
 - STRUCTURE** 3-77
 - description 3-79
 - examples 3-80
 - order of operations 3-79
 - REFLECT** parameter, trench implant simulation example 5-19
 - truncation cautions 3-80
 - TSUPREM-4 version compatibility 3-80
 - structure
 - after silicide growth step *Fig. 5-53*
 - immediately before silicide growth step *Fig. 5-52*
 - with contours of boron concentration, after first p-well diffusion *Fig. 5-41*
 - with contours of boron concentration, after formation of isolation region, plotted by file *s4ex4a.inp Fig. 5-7*
 - Sum** function 6-15
 - summary of statements 3-309
 - surface reaction rate 2-64
 - surface recombination rate depending on nitrogen 6-47
 - surface recombination velocity models 2-38
 - surface tension 2-63
 - syntax 6-3
 - input statements 3-2
 - parameter lists 3-4
 - parameters 3-3
- T**
- TAG** parameter, **LINE** statement
 - SOI MOSFET example 5-45
 - Taurus Layout, mask data files D-1
 - Taurus-Lithography **DEPICT** parameter 3-75
 - TCAD Products and Utilities Installation Manual* xxxvii
 - Technology Interchange Format
 - see TIF* files
 - terminal output 1-5
 - thermal stress model equations 2-73, 2-74
 - boundary conditions 2-73
 - effect of etching on stress 2-74
 - initial conditions 2-73
 - limitations 2-75
 - stress history model 2-74
 - THICKNES** parameter, **DEPOSITION** statement
 - 1D bipolar example 4-5
 - thin oxidation rate 2-64
 - thin regime 2-56
 - third part of input file
 - s4ex4d.inp*, showing junction capacitance extraction *Fig. 5-17*
 - s4ex5.inp*, using Monte Carlo implantation model *Fig. 5-24*
 - s4ex7a.inp*, for simulating CMOS process *Fig. 5-32*
 - s4ex8.inp*, final processing and plotting *Fig. 5-43*
 - THRESHOLD** parameter, **ELECTRICAL** statement
 - NMOS LDD process example 5-14
 - TIF files
 - overview 1-7
 - TIF** parameter 3-74
 - TSUPREM-4 version limitations 3-74
 - user defined materials or impurities 3-74
 - tilt and rotation tables, accuracy after implantation 2-84
 - TILT** parameter, **IMPLANT** statement
 - trench implant simulation example 5-19
 - TIME** parameter, **EPITAXY** statement
 - 1D bipolar example 4-6
 - TiSi₂ growth kinetics 2-68
 - diffusion of silicon 2-68
 - initialization 2-69
 - material flow 2-69
 - reaction at TiSi₂/Si interface 2-68
 - titanium silicide model 2-69
 - transient clustering model 2-129, 6-35
 - transport coefficient 2-28
 - TRAP** parameter, **ETCH** statement
 - CMOS process example 5-34
 - trapezoidal etch model 2-108
 - etch examples 2-109
 - etch steps 2-109
 - parameters 2-109
 - trench implant simulation example
 - analytic implant 5-19
 - COLOR** statement 5-20
 - ELIMINATE** statement 5-19
 - END** statement 5-20
 - ETCH** statement 5-19
 - FOREACH** statement 5-20
 - LABEL** statement 5-20
 - LINE** statement 5-19
 - Monte Carlo implant 5-23
 - Monte Carlo summary 5-26
 - overview 5-18
 - PLOT . 1D** statement 5-21
 - PLOT . 2D** statement 5-20
 - plotting results of Monte Carlo method 5-23
 - SAVEFILE** statement 5-19
 - SELECT** statement 5-20
 - structure generation 5-18
 - triangles
 - flipping 2-9
 - maximum number 2-2
 - numerical integrity 2-9
 - overview 2-2
 - troubleshooting and problems xxxvii
 - TRUNCATE** parameter, **STRUCTURE** statement
 - CMOS process example 5-33
 - TSUPREM-4 models
 - deposition 2-105
 - diffusion 2-13
 - electrical calculations 2-120
 - epitaxial growth 2-104
 - etching 2-107
 - extended defects AAM 2-126
 - grid structure 2-2
 - introduction 2-1
 - ion implantation 2-77
 - modeling polycrystalline materials 2-112
 - oxidation 2-46
 - photoresist 2-107
 - polysilicon Monte Carlo implant model 2-103
 - references 2-131

- silicide models 2-68
- simulation structure 2-1
- stress history 2-73
- TSUPREM-4 structure file *Fig. E-6*
- tungsten silicide model 2-72
- tutorial examples
 - 1D bipolar example 4-2
 - input file syntax and format 4-1
 - local oxidation, 2D simulation 4-12
 - overview 4-1
 - point defect models 4-27
- typeface conventions xxxvii

U

- UNDEFINE** 3-45
 - description 3-45
 - example 3-45
 - redefined parameter names 3-45
- update solution 6-23
- USEIT** 6-1, 6-48
 - examples 6-33
 - new applications using USEIT C-3
 - overview 6-1
 - physical model and equation
 - interface (PMEI) 6-3
 - using 6-19
 - working diagram *Fig. 3-205*, 6-2
- USEIT** examples
 - 5-stream diffusion model 6-48
 - boron-interstitial clustering model
 - with small clusters 6-41
 - diffusion enhancement in thinner oxides 6-45
 - diffusion in **PD . TRANS** 6-38
 - dislocation loop model 6-36
 - interface trap model 6-39
 - interstitial clustering model 6-34
 - SiGe model 6-58
 - surface recombination rate
 - depending on nitrogen 6-47
 - transient clustering model 6-35
- User-Specified Equation Interface (USEIT) 6-1
- using TSUPREM-4
 - file specification 1-3
 - input files 1-4
 - introduction 1-1
 - library files 1-8
 - output files 1-5
 - program output 1-2
 - starting TSUPREM-4 1-1
- using USEIT 6-19
 - application of user-specified models 6-24
 - built-in keywords 6-27
 - EQUATION** 6-20
 - interface functions 6-31
 - INTERMEDIATE** 6-24

- mathematical functions 6-28
- modification of built-in functions 6-25
- operators 6-27
- physical functions 6-29

V

- VACANCY** 3-276
 - additional notes 3-283
 - bulk and interface parameters 3-282
 - description 3-282
 - examples 3-283
- VALUE** parameter, **CONTOUR** statement
 - local oxidation example 4-23
- VC** parameter, **MATERIAL** statement
 - poly-buffered LOCOS example 5-28
- version 2000.2 enhancements C-1
- VERTICAL** model 2-56
- VERTICAL** model, recommended usage 2-56
- vertical profiles
 - after Monte Carlo implant *Fig. 5-25*
 - produced by analytic implant *Fig. 5-22*
- VIEWPORT** 3-184
 - description 3-184
 - examples 3-185
 - scaling plot size 3-184
- V . INITOX** model 2-39
- VISCOELA** model 2-60
 - parameters 2-62
 - recommended usage 2-62
 - viscoelastic flow 2-60
- VISCOELA** parameter, **METHOD** statement
 - local oxidation example 4-14
- VISCOUS** model 2-58
 - incompressible viscous flow 2-58
 - recommended usage 2-60
 - stress dependence 2-59
- VLENG** parameter, **PLOT . 2D** statement
 - local oxidation example 4-15
- V . MAXOX** model 2-38
- V . NORM** model 2-39

W

- warnings 1-3
- WIDTH** parameter, **INITIALIZE** statement
 - 1D local oxidation example 4-19
- W . RECATN** parameter, **LABEL** statement
 - 1D local oxidation example 4-18

X

- X direction
 - automatic grid generation 2-5
 - column elimination 2-6
- X grid
 - MASK** statement 2-5
 - WIDTH** parameter 2-5
- X parameter, **ELECTRICAL** statement
 - NMOS LDD process example 5-16
- X parameter, **LABEL** statement
 - 1D bipolar example 4-7
- X parameter, **X** statement
 - local oxidation example 4-13
- X . MAX** parameter, **PLOT . 2D** statement
 - local oxidation example 4-15
- X . VALUE** parameter, **PLOT . 1D** statement
 - 1D bipolar example 4-7
 - local oxidation example 4-24

Y

- Y parameter, **LABEL** statement
 - 1D bipolar example 4-7
- Y parameter, **LINE** statement
 - local oxidation example 4-13
- YDY** parameter, **DEPOSITION** statement
 - SOI MOSPET example 5-46
- Y . MAX** parameter, **PLOT . 2D** statement
 - CMOS process example 5-33
 - local oxidation example 4-15
- Y . MIN** parameter, **PLOT . 2D** statement
 - local oxidation example 4-15
- Y . VALUE** parameter, **PLOT . 1D** statement
 - 1D bipolar example 4-7

Z

- Z parameter, **METHOD** statement
 - 1D bipolar example 4-8
- Z parameter, **SELECT** statement
 - poly-buffered LOCOS example 5-28