

Silicon Integration Initiative, Inc.



Standard Interconnect Performance Parameters (SIPPs)

Version 1.06 01— 092200

Prepared and Distributed by:

Silicon Integration Initiative, Inc.
4030 Braker Lane Suite 550
Austin, TX 78759, USA

Standard Interconnect Performance Parameters (SIPPs)

Version 1.06.01— 092200

Prepared by: Silicon Integration Initiative, Inc. and Task Group participants from member companies.

Abstract: This document provides specifications for the Standard Interconnect Performance Parameters (SIPPs). SIPPs is an open, industry-wide, data specification that provides a standard method for representing the physical characteristics of any IC process that affects interconnect performance and thus, is required by parasitic extraction.

**Copyright (c) 2000 by Silicon Integration Initiative, Inc. (Si2)
(DUNS No. 62-191-1718).**

All Rights Reserved Worldwide. Permission to use, copy, and distribute for any purpose subject to the following restrictions:

1. Permission is granted for verbatim copy and distribution only, of this material, in any medium, provided that you publish on each copy this copyright notice and disclaimer of warranty.
2. Si2 makes no representation about the suitability for any purpose or regarding any claims made by products that use it. It is provided "as is" without expressed or implied warranty.
3. Si2 may revise this document from time to time without notice.

Change History

11/18/98 (V0.00) - M.Walker (Original SIPPs announcement)

03/17/99 (V0.01) - J.Tallinger

03/28/99 (V0.02) - J.Tallinger

1. Added the ability to add negative layer layers from the Reference Ground. They would be used for Diffusions, Epi layers, Lossy Dielectrics, etc. They may or may not have thicknesses or **GDSII_layers**.
2. Changed "epsilon" to "permittivity".
3. Added **resistivity** and **r_sheet** to Substrate layers to account for Diffusion Resistance and lossy dielectrics.
4. Added **GDSII_layers** to Substrate layers to define diffusion and well areas.
5. Added a general **user_property** to allow any special cases not covered in Standard. User may define as many of these as necessary.
6. Added **GDSII_layers** to Dielectric layers to define special dielectric areas.
7. Added **GDSII_layers** and **Resistivity** to Metal layers to merge resistance definition and metal together.
8. Made Via layers a type of layer to be more consistent with other layers.
9. Added *@Ctemp* to via resistances.

04/06/99(V0.03) - J.Tallinger/D.Cottrell

1. Clarified layer definition below reference capacitance ground plane
2. Added layer **type = via** definition
3. Expanded **user_property** definition
4. Added side wall coverage to layer type = metal
5. Added resistivity to layer type = substrate

05/27/99(V0.05) - Group Review

1. Changed "**planarize**" element name to "**max_height**" and add "*from_layer*" attribute - Dielectric Layer
2. Added "**max_height**" element to Metal Layer
3. Dropped "*inside*" attribute from "**corner_radius**" element of Via Layer
4. Dropped "**resistivity**" element from Via resistance Definition.
5. Dropped any reference to layer numbers. Only the **name** element is used to uniquely identify layers.
6. Clarified metal **height** element to include coating thickness.
7. Added note on meaning of **temperature** element within Resistance definitions.
8. Extended Dielectric **thickness** to be represented as a function of interconnect spacing.
9. Extended Metal **thickness** to be represented as a function of interconnect spacing.
10. Changed "**swcov**" to "**swthk**" on Metal Layer

08/17/99(V0.06) - J.Tallinger/D.Cottrell/K.J.Chang/M.Walker

1. Updated to reflect Abstract model elements from (Abstract Model V0.03) K.J.Chang
2. First Draft of formal Specification.

10/99(V0.07) - D.Cottrell

1. Updates made to reflect 9/1/99 meeting (see meeting Minutes)
 - Airgap syntax added
 - Reference_ground specification added
 - Clarified dielectric and metal types and heights
2. Added example SIPPs file from K.J. Chang

3. Updated contributor list

11/3/99 (V0.08) - J.Tallinger, D.Cottrell

1. Added Comments specification
2. Added default s for value type, planarity, and metal type.
3. Added a second SIPPs example from OEA.
4. Fixed typos and grammar errors identified by J Tallinger.
5. Simplified specification of the ground-0 reference.

11/12/99 (V0.09) - D. Cottrell, J.Tallinger, KJ Chang

1. Changes from 11/12 DWG Meeting
 - Added subtype attribute to substrate statement
 - Clarified use of height statement within substrate statements and metal/dielectric
 - Updated graphics and examples
 - Added table clarifying treatment for pair-wise combinations of dielectric types
2. Added examples of Abstraction from KJ

1/27/00 (V1.00) - SIPPs Work Group Meeting

1. Changes resulting from Si2 Ballot
2. Added Parameter and Mask Sections - Ordered SIPPs into Parameter, Mask, and Process sections
3. Moved concept of Abstract modeling to the Informative section and dropper notion of "Physical" model.
4. Clarified meaning of "spacing" keyword.
5. Eliminated multiple dependant variables as an option in the dependant (**D**)value specification.
6. Added library identification commands (SIPPs and Technology).

2/18/00 (V1.01) - Ertugrul Demircan

1. Cleaned up relationship of GDSII reference and derived mask reference. Added **mask** attribute to layer statements.
2. Cleaned up syntax and description of **mask** section.
3. Removed **swslope** and modified dielectric sidewalls instead to use metal **edge_shape** syntax.
4. Added thickness property to via layer.
5. Removed **edge_bias** from **minimum_metal_thickness**.
6. Typo corrections.
7. Expanded and corrected examples.
8. Removed **trench** type for metal and dielectric since this can be covered by the coating type.

2/26/00 (V1.02) - Ertugrul Demircan

1. Corrections to mask examples
2. Corrections to SIPPs Sample Data

3/11/00 (V1.03) - Jerry Tallinger, Marek Turowski

1. Modifies syntax to define a variable withiin parameters section - \$ not required within the definition of a substitution variable, only with the use.
2. Removed compliance requirement for any GDSII reference in a substrate layer.
3. Modified compliance requirement to allow either r_sheet or resistivity within a metal layer.
4. Modified syntax for units specifications.
5. Added ohms, centimeters and meters as allowed units attributes.
6. Added XOR as a mask operator.

7. Modified mask syntax to disallow complex expressions within a single statement.
8. Added "buried" to description in 6.1.5.2.
9. Corrected Figure 6.
10. Modified Informative section on Abstract model to make more generic.
11. Corrections to SIPPs example 1 and 2 sample data, including figures (wrt FOX bottom).

3/30/00 (V1.04) - Ertugrul et.al.

1. Change the definition of **height** from "Distance from the top of bare die to the bottom surface of this layer" to "Distance (Z-direction) from the lowest point of the top surface of the chip after all previous process steps."
2. Modify the syntax for VIA Layers to make them more to METAL to simplify the specification while the same time adding more capability for specification of vias.

A. Eliminate the "from_layer" and "to_layer" notions from VIA statements replace them with the "height" attribute. Thus, via layers would now also be order dependant and their from-to points would be determined by the position of the their layer statement in the SIPPs and their "height" specification.

B. Add the following attributes to the VIA Layer (to support the above, make VIA layers more flexible, and consistent with METAL):

planarity
 swthk
 height
 measured_from
 edge_pattern
 min_width
 min_spacing

3. Si2 Board of Directors approval

5/23/00 (V1.05) - Michal Furrmanczyk

1. Clarifications on measured_from statement definition.
2. Eliminated requirement for both height and thickness on a substrate layer to be compliant.
3. Fixed errors in Example 2.
4. Typo on page 19 - "...or diffusions.." changed to "...for diffusions..".

6/25/00 (V1.06)

1. Si2 Approved Version

9/22/00 (V 1.06.01)

1. Clarification edits based on customer usage
2. Minor typos corrected

Contributors

The following individuals contributed to the SIPPs development and specification.

A special thanks goes to Frequency Technology (Marty Walker and KJ Chang) and OEA International (Jerry Tallinger) for the initial technology contributions.

Contributor	Company
Kun-Ming Yang	3S Silicon Technology
Amrish Chokhavatia	AMD
Vivek Raghavan	Avant!
Chong Wai Chang Chan	Avant!
Sridhar Panchapakesan	Avant!
Ric Borges	Avant!
James Chen	BTA
Juan Rey	Cadence
Anand Bariya	Cadence
Jim McCanny	CADMOS
Vergheese Nishath	CADMOS
Golkar Nata	CADMOS
Michal Furmanczyk	CFD Research Corp.
Marek Turowski	CFD Research Corp.
Per Ljung	Cyote Systems
Keh-Jeng Chang	Frequency Technology
Marty Walker	Frequency Technology
John Beatty	IBM
Joe Morrell	IBM
Ashok Kapur	LSI Logic
William Loh	LSI Logic
Mitch Hines	Mentor Graphics
Ann Spratt	Mentor Graphics
Ertugrul Demircan	Motorola
Jerry Tallinger	OEA International
Gary Panzer	Peerless Systems
Gerard Vogels	Philips
Ralph Iverson	Random Logic
Don Cottrell	Si2
Mike Beunder	Silicon Access
Mike Newman	Silicon Technology

Corrine Cregut	ST Microelectronics
Tak Young	Monterey Design Systems
Boon-Khim Liew	TMSC
Sheng-Hsing Yang	UMC
S C Wong	Winbond
Min-Chie Jeng	WSMC

Table of Contents

1.0	Introduction.....	11
2.0	Glossary of Terms	12
3.0	Conventions	13
4.0	Assumptions and Limitations	14
5.0	SIPPs Model	15
5.1	Parameters Section Overview.....	15
5.2	Mask Section Overview.....	16
5.3	Process Section Overview	16
5.4	SIPPs Compliance Requirements	18
5.4.1	Minimum Substrate Layer Parameters	18
5.4.2	Minimum Dielectric Layer Parameters	18
5.4.3	Minimum Metal Layer Parameters.....	18
5.4.4	Minimum Via Layer Parameters.....	18
6.0	Syntax	19
6.1	General Syntax	19
6.1.1	Comments.....	19
6.1.2	Case Sensitivity	19
6.1.3	Units of Measure	19
6.1.4	Variables	20
6.1.5	Process Variations and Dependant Variables.....	20
6.1.6	User Properties	22
6.1.7	GDSII Reference	22
6.1.8	Names	22
6.2	Library Identification Commands	23
6.2.1	SIPPs File Identifier.....	23
6.2.2	Technology Identification.....	23
6.3	Parameters Section Syntax	23
6.4	Mask Section Syntax	24
6.4.1	BOUNDARY Mask.....	24
6.4.2	Custom Masks	24
6.4.3	Mask Section Examples.....	26
6.5	Process Section Syntax.....	27
6.5.1	Substrate Layer Syntax	27
6.5.2	Dielectric Layer Syntax	31
6.5.3	Metal Layer Syntax	38
6.5.4	Via Layer Syntax	44
7.0	Informative.....	49
7.1	SIPPs Abstract Model Techniques	49
7.1.1	Equivalent-brick model for conductors	50
7.1.2	Equivalent model for multiple dielectrics	51
7.2	Sample SIPPs Data	52
7.2.1	Example 1	52
7.2.2	Example 2.....	57

8.0	Follow-On	63
8.1	Substrate Modeling	63
8.2	Anisotropic Dielectrics	63

List of Figures

Figure 1—Typical design flow using SIPPs	11
Figure 2—SIPPs Model	17
Figure 3—Mask Layer Boolean Operations	25
Figure 4—Dielectric Cap	26
Figure 5—Substrate Layers	28
Figure 6—Height vs. Thickness	30
Figure 7—Dielectric Types	33
Figure 8—Coating Height, Thickness, and Sidewall Thickness	34
Figure 9—Dielectric Sidewall	36
Figure 10—Dielectric Edge Shape	36
Figure 11—Air Gaps Defined	37
Figure 12—Air Gaps	38
Figure 13—Metal Layer	39
Figure 14—Metal Edge Shape	42
Figure 15—Via Layer	45
Figure 16—Abstract Modeling	49
Figure 17—Three Trapezoidal-Shape Conductors	50
Figure 18—Equivalent-Brick Conductors	50
Figure 19—Multiple Dielectrics	51
Figure 20—Reduction of Multiple Dielectrics	51
Figure 21—Example 0.18 micron technology	52
Figure 22—SIPPs parameters for a sample 0.25mm 5M1P technology	58

1.0 Introduction

This document provides specifications for an open-industry standard format, known as the *Standard Interconnect Performance Parameters* or SIPPs, that semiconductor foundries shall use in supplying semiconductor process characteristic data to circuit designers. All circuit design tools must also adhere to the SIPPs standard to ensure accurate physical representation of IC processes that affect interconnect performance and thus, is required by parasitic extraction.

Prior to the arrival of SIPPs, semiconductor foundries were required to supply process characteristic data in a number of different formats to meet the requirements of various design tool implementations, in particular, parasitic extractors. This required significant conversion, qualification expense, and manpower on the part of the semiconductor industry. Also, the lack of a standard made it difficult for new state-of-the-art extraction tools to enter into the marketplace because semiconductor suppliers were reluctant to go through the expense of supplying yet another format for this information.

The objectives of the SIPPs data model and exchange standard include:

- To be tool-independent, imposing no assumptions or constraints on design tools
- To provide for electrically accurate descriptions and characterization
- To be flexible yet consistent across different abstraction levels
- Provide true physical representation of process layers
- Provide abstracted views of processing layers that are electrically equivalent
- To be extensible enough to accommodate future process technology

Figure 1 contains a simplified block diagram of the process characterization to design flow and the relationship of the SIPPs Standard Exchange library.

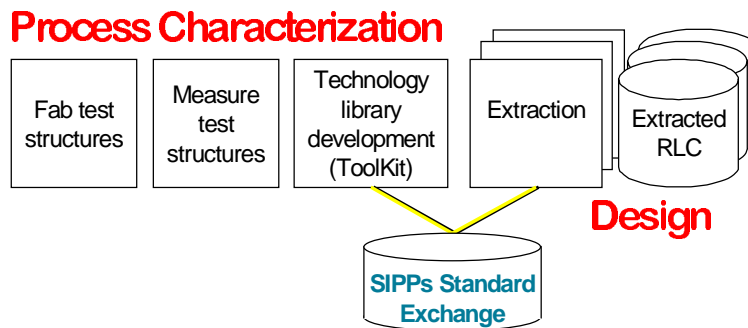


Figure 23—Typical design flow using SIPPs

2.0 Glossary of Terms

anisotropic *adj* : having properties that vary according to the direction in which they are measured.

CMP (chemical-mechanical polishing): process used to planarize top surface of metal or dielectric material.

Damascene *adj* : a manufacturing process where-by interconnect metal is inlaid into trenched etched in a dielectric.

dielectric *n* : a nonconductive material; an insulator.

dielectric constant (ϵ_r) (K) *n* : that property of a dielectric material which determines how much electrostatic energy stored per unit volume for unit potential gradient relative to that in a vacuum.

interconnect *n* : a highly conductive material, such as copper, aluminum or polysilicon, that carries electrical signals to different parts of a die thus providing the wiring between circuit elements.

OPC (optical proximity correction): corrections made to masks to compensate for negative effects of lithography and assure actual lines adhere to drawn geometry.

parasitics *n* : by product circuit components (for example, capacitance, inductance or resistance) present in a design.

permittivity (ϵ) *n* : that property of a dielectric material that determines how much electrostatic energy can be stored per unit of volume when unit voltage is applied.

resistivity (ρ) *n* : the ratio of the potential gradient parallel with the current in the material to the current density. Units are Ω *cm.

$$\text{Interconnect_Resistance} = (\rho * \text{Int_Length}) / \text{Int_Cross_Sectional_Area}$$

sheet resistance (R_S) (Ω/o) *n* : the ratio of the potential gradient (electric field) parallel with the current to the product of the current density and thickness.

$$\text{Interconnect_Resistance} = R_S * (\text{Int_Length} / \text{Int_Width})$$

substrate *n* : a wafer that is the basis for subsequent processing operations in the fabrication of semiconductor devices or circuits.

via *n* : an opening in the dielectric layer(s) in which the walls are made conductive, thus providing an electrical pathway from one metal layer to the metal layer above or below.

3.0 Conventions

The following conventions are used throughout this specification.

1. Keywords appear in **bold**.
2. Keywords are not case sensitive.
3. Variables appear in *italic*.
4. Square brackets ([]) enclose optional items.
5. Braces ({ }) enclose a list from which you must choose one or more items.
6. OR-bars (|) separate optional items in a list of choices.
7. Horizontal ellipsis (...) indicate that the preceding item can be repeated one or more times.

4.0 Assumptions and Limitations

1. Positional or die-to-die proximity effects are not included since they seem to be issues that for the most part are corrected during manufacturing by the use of OPC tools. From what has been learned to date, these effects are not predictable. These effects include, but are not limited to, optical proximity effects, etch proximity effects, CMP and CVD proximity effects, and phase shift effects. Within-die proximity effects, a.k.a. local variations are added where they are significant.
2. Top and bottom via shapes are assumed identical.
3. Anisotropic dielectrics are not supported in the first release of SIPPs. It is recognized that such support will be necessary for Low-K dielectrics so it will be added in the second release revision.
4. Modeling of substrate diffusions and layers is limited to that necessary for interconnect performance without consideration of coupling between interconnect and substrate features. This will be extended in the second release revision.
5. The scope of the SIPPs data model is that required for parasitic extraction and field solvers. It is not intended to cover process information required by all design or analysis tools. Over time, the SIPPs model may be extended to cover additional process characteristics and, thus, support other classes of EDA tools.

5.0 SIPPs Model

The SIPPs model is organized into three discrete sections:

- Parameters (optional)
- Mask (optional)
- Process (required)

The optional **Parameters** section defines macro variables that may be used throughout other sections to substitute the value of the macro definition for a value specification.

The optional **Mask** section defines virtual GDSII elements based on arithmetic or logical operations of real GDSII elements in order to precisely define physical characteristics that are dependant on combinations of elements.

The **Process** section shall be mandatory and describes the characteristics of each layer of the IC in the order that the processing steps are performed in the manufacturing process. Each of these sections is preceded by a **Begin** statement and followed by an **End** statement as follows:

```
begin_parameters
. . .
end_parameters or end

begin_mask
. . .
end_mask or end

begin_process
. . .
end_process or end
```

5.1 Parameters Section Overview

The **parameters** section is optional. It allows the SIPPs library developer to assign a name to a value and thereafter use that name within the value specification in other places through the SIPPs library. The application shall substitute the value assigned to the name wherever that name appears within the SIPPs library. This is useful in cases where a value is used in many places, but which may change in time. In this case, the value need be re-specified only once in the library (within the **parameters** section) rather than requiring it be changed in every place it is used. For example, should the metal thickness be the same on many personality layers, then the value for the **thickness** attribute may be assigned a name (e.g. M1). Thereafter, that name may be used as a synonym for the assigned value by preceding the name with a dollar (\$) sign. Thus, in the above example, \$M1 may be used to specify a metal thickness in many places. By changing the value for M1 in the parameters section, the new value shall be substituted in all places where \$M1 is used throughout the SIPPs library by the application reading the SIPPs library. This section can be used to define any design parameters, such as minimum width and spacing, that is allowed for the design layers in the process.

New variables may also be defined using algebraic operations of addition (+), subtraction (-), multiplication (*), and division (/) on previously defined variables. The order of operator precedence of algebraic operations shall be compliant with that specified in ANSI C.

The **parameters** section shall be preceded by:

begin_parameters

The **parameters** section shall be followed by either of the following:

end_parameters|end

5.2 Mask Section Overview

The **mask** section is optional. It allows the SIPPs to represent special conditions that cannot otherwise be represented by means of the **process** statements. The **mask** section is used to create binary masks that are applied to specified layers to create actual patterns for objects on those layers.

The **mask** section shall be preceded by:

begin_mask

The **mask** section shall be followed by either of the following:

end_mask | end

5.3 Process Section Overview

The Process section of the SIPPs data model provides a complete description of relevant characteristics of each manufacturing process layer. A complete description of all parameters affecting interconnect performance is contained within this data model. The SIPPs Model is useful for all levels of extraction including field-solver extraction tools where actual 3-D physical models may be used.

The SIPPs model may also be used to represent an abstracted view of the process steps. In this case multiple process steps may be combined into a single "abstracted" step with electrically equivalent values. Other simplifying assumptions may also be made for physical characteristics of features (see Section 7.1 in the Informative portion of this specification for examples). This may be useful for high-speed extraction tools where electrically equivalent parameters can be used to predict R's and C's with the goal of matching those measured from silicon.

The choice of what level of abstraction to use in a SIPPs instance is not part of the specification, but is left to the discretion of the user. There is, however, a minimum set of SIPPs model elements that must be represented in any SIPPs instance to be considered to be compliant with the specification and SIPPs "compliant" tools must be able to process this minimum set (specified in Section 5.4).

The data model is represented as an ordered list of [manufacturing] steps beginning with the wafer base, followed with the substrate layers, and proceeding up through the dielectric and metal layers.

1. Each layer has a type, a name, and physical properties depending on type.

SIPPs Model
Process Section Overview

2. Layer types allowed are **substrate**, **dielectric**, **metal**, and **via**.
3. Processing of SIPPs data is dependent on the order layer statements within the **process** section. For instance, by default, the height of a layer is defined relative to the top surface of the previous layers. Thus, the SIPPs exchange begins with substrate layers beginning at the bare die followed by sequences of dielectric and metal layers.
4. The ground-zero reference surface shall be at the bottom surface of the last layer containing the **reference_ground** attribute.
5. The name for each layer is arbitrary (has no semantic meaning) but must be unique. These names may or may not be case sensitive depending on the setting of the **case_sensitivity** statement.
6. Allowable physical properties that can be specified for any layer are dependent on the type of layer being represented i.e., **substrate**, **dielectric**, **metal**, or **via**.
7. User-defined property values may be specified on any layer - **user_property**.

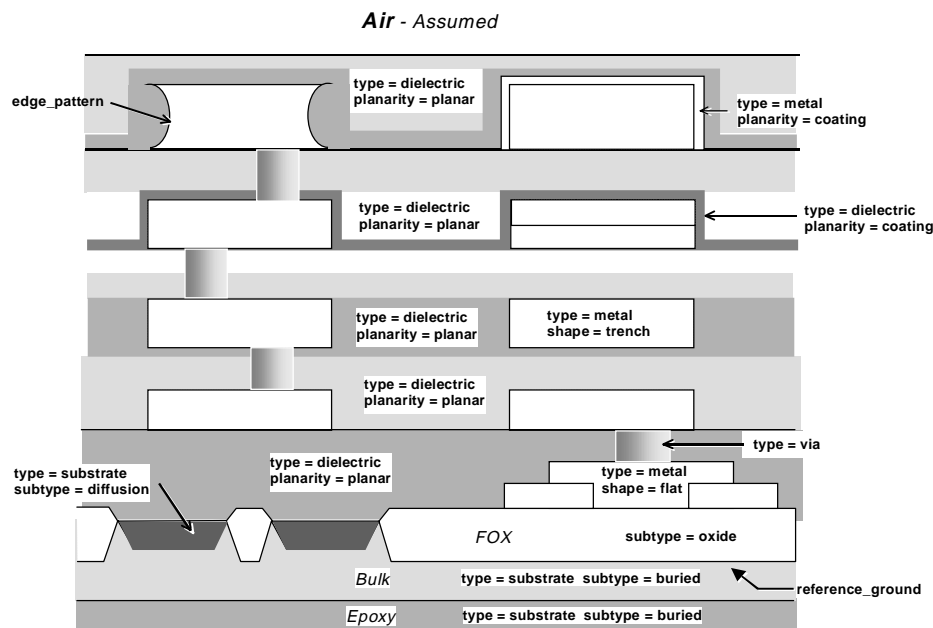


Figure 24— SIPPs Model

The **process** section shall be preceded by:

begin_process

The **process** section shall be followed by either of the following:

end_process | end

5.4 SIPPs Compliance Requirements

A SIPPs library may contain any or all of the data elements (keywords and attributes) defined within the **Parameters**, **Mask**, and **Process** sections.

A SIPPs compliant library shall contain a **SIPPs** statement and, as a minimum requirement, the **Process** section shall contain the following set of parameters:

5.4.1 Minimum Substrate Layer Parameters

type = **substrate**

name = layer name

height = height of the substrate feature or layer, or

thickness = thickness of the substrate layer

reference_ground = capacitance ground-0 reference plane (on one layer only)

5.4.2 Minimum Dielectric Layer Parameters

type = **dielectric**

name = layer name

thickness = dielectric thickness

K = dielectric constant

5.4.3 Minimum Metal Layer Parameters

type = **metal**

name = layer name

mask = mask object reference

thickness = metal thickness (z-direction)

r_sheet = sheet resistance | **resistivity** = resistivity

5.4.4 Minimum Via Layer Parameters

type = **via**

name = layer name

mask = mask object reference

from_layer = bottom layer of via

to_layer = top layer of via

r_per_via = via resistance | **resistivity** = via resistivity

6.0 Syntax

6.1 General Syntax

6.1.1 Comments

Comments may be placed anywhere within a SIPPs file. Comments begin with the pound-sign character (#), and end with a line return (cr) or end-of-line (eol).

6.1.2 Case Sensitivity

SIPPs **keyword** names are not case sensitive. For example, **Type = type = TYPE**.

User-defined names for process layers and properties may or may not be case sensitive, depending on the setting of the **case_sensitivity** statement. The **case_sensitivity** statement must precede any name variables for which it applies. If no **case_sensitivity** statement precedes the use of a name variable, then case sensitivity shall be considered **OFF**.

```
case_sensitivity = ON|OFF
```

Where:

- **ON** means uppercase and lowercase letters shall be treated as unique values.
- **OFF** means that uppercase and lowercase letters shall be treated as equal values.

6.1.3 Units of Measure

Definition of the assumed units for all variables are defined by units definitional statements. These statements must precede the variable value for which they apply. Multiple units definitional statements may be present within the SIPPs instance in which case the value in the latter statement shall override that in any preceding units definitional statement for the same units dimension. If no units definition statement is present, then default units shall be as follows:

Variable Dimension	Default Units
Distance	Microns
Temperature	Degrees Celsius
Resistance	Ohms
Resistivity	Ohm.Centimeters
Sheet Resistance	Ohms per Square
Angles	Degrees

Syntax General Syntax

Units definitional statements override any of the defaults. These statements may reside anywhere in a SIPPs library and the latest (sequentially in the SIPPs file) shall always supercede any previous value for statements that follow it. The syntax is as follows:

```
distance = distanceunits
temperature = C | F
resistance = resistanceunits
resistivity = resistanceunits.distanceunits
sheetresistance = resistanceunits
```

Where:

- *distanceunits* = **m** | **cm** | **mm** | **um** | **nm** | **pm**
m = meters
cm = centimeters ($\text{meters} \times 10^{-2}$)
mm = millimeters ($\text{meters} \times 10^{-3}$)
um = micrometers ($\text{meters} \times 10^{-6}$)
nm = nanometers ($\text{meters} \times 10^{-9}$)
pm - picometers ($\text{meters} \times 10^{-12}$)
- *temperatureunits* = **C** | **F**
C = Celsius
F = Fahrenheit
- *resistanceunits* = **ohm** | **mohm** | **uohm** | **nohm** | **pohm**
ohm = ohm (Ω)
mohm = milli-ohms ($\Omega \times 10^{-3}$)
uohm = micro-ohms ($\Omega \times 10^{-6}$)
nohm = nano-ohms ($\Omega \times 10^{-9}$)
pohm = pico-ohms ($\Omega \times 10^{-12}$)

6.1.4 Variables

The variable names defined in the **parameters** or **mask** sections may be numeric or alphanumeric strings. When used, the name must be preceded by a "\$". Values for variables can be signed *real* or *integer* numbers.

6.1.5 Process Variations and Dependant Variables

To account for process variations, most values may be specified as either a fixed value, as a minimum-nominal-maximum triplet, or a mean-variation pair. The representation is defined by preceding the value(s) with **F**, **R**, or **N** for fixed, range, or normal distribution respectively.

These are specified as follows:

Syntax
General Syntax

value

F (*value*)

R (*min_value, nom_value, max_value*)

N (*mean_value, variation_value*)

D (see description below)

Where:

- If only one value is specified, it shall be **fixed (F)** format.
- *variation_value* is a signed number in the same units as the mean. The *variation_value* shall be added (+) or subtracted (-) from the *mean_value* to account for process variation. If the number is unsigned, (+-) shall be assumed (thus the *variation_value* is added and subtracted from the *mean_value*)
- Values that begin with a dollar sign (\$) are substitution variables defined within the **parameters** and **mask** sections.

Variables whose value is dependent on another variable(s) are indicated by preceding the values with a **D** (dependant). In this case, the variable value is specified as a piecewise linear function of the dependent variable(s). The general syntax for this case is given below.

```
D (keyword)  
  (function_value1, var_value1  
  [function_value2, var_value2]  
  . . . )
```

Where:

- **keyword** defines the variable(s) on which the value is dependent.
- The value of a variable between any two specified points shall be determined by a linear interpolation of the values at those two points.
- The value of a variable that falls outside of the specified range of points shall be the value at the lowest specified point (if below the range) and of the highest point (if above the specified range).
- Values that begin with "\$" are substitution variables defined within the **parameters** or **mask** section.

6.1.6 User Properties

User assigned data may be assigned to layers within the model. The syntax for these is:

```
user_property = name, {string|int|real} [,range # #]  
                ( {string_value/value}  
                  [...] )
```

Where:

- **string** indicates the value to be a text string.
- **int** indicates the value to be a signed integer value.
- **real** indicates the value to be a signed real number.
- *string_value* is enclosed in quotes.
- Quotes (") within a string are written as \".
- A backslash (\) within a string is written as \\.
- **range** is used to specify the limits of **real** or **int** values.

6.1.7 GDSII Reference

SIPPs **process** layers allow for reference to GDSII objects (**mask** attribute) to be applied as a mask for that layer. Syntax for GDSII objects is as follows:

```
(GDSII# [;GDSIIdatatype] [,GDSII# [;GDSIIdatatype]] [...])
```

Where:

- *GDSII#* is the GDSII Layer. If no GDSIIdatatype is specified, then all GDSII datatypes for the specified layer shall be used.
- *GDSIIdatatype* is the GDSII Data Type

6.1.8 Names

User-assigned name specifications may be any alphanumeric string of arbitrary length, but must be unique across the scope of the **parameters** and **mask** sections (if they are substitution variable names) or within the **process** section (if defined there).

Names containing special characters or whitespace shall be enclosed in double quote marks (").

Quotes (") within a string are written as \".

A backslash (\) within a string is written as \\.

6.2 Library Identification Commands

6.2.1 SIPPs File Identifier

The SIPPs File Identifier identifies the version and level of the SIPPs specification to which this library adheres.

```
SIPPs = (version = version#,  
        iteration = iteration#)
```

Where:

- *version#* is the version number of the SIPPs Specification to which this library adheres.
- *iteration#* is the iteration number of the SIPPs Specification to which this library adheres.

6.2.2 Technology Identification

All process technology in the library is identified by process name, creation date, and contact information as follows:

```
technology = (process = process_name,  
             date = creation_date,  
             contact = contact_info)
```

Where:

- *process_name* name is an alphanumeric text string identifying the process this library describes.
- *Creation_date* name is an alphanumeric text string identifying the date this library was created.
- *Contact_info* name is an alphanumeric text string identifying contact information for this library.

6.3 Parameters Section Syntax

Substitution variable definitions within the Parameters section shall adhere to the following syntax:

```
begin_parameters  
{substitution_variable_name = value_spec |  
  
 substitution_variable_name = value_spec [+|-|*|/]  
                                     value_spec [[+|-|*|/]  
                                     value_spec]  
}
```

```
{end_parameters | end}
```

Where:

- *substitution_variable_name* may be any alphanumeric string.
- *substitution_variable_name* must be unique within the scope of all substitution variable names (in both the **parameters** and **mask** sections).
- *value_spec* is a signed real or integer number.
- The operators +, -, *, and / mean add, subtract, multiply, or divide respectively. Operator precedence shall be in accordance with that of ANSI C.

6.4 Mask Section Syntax

The **mask** section specifies a derived mask to be applied to objects used within the **process** section. Operations may be logical (such as the AND or OR of object areas), or arithmetic (such as adding or subtracting of a value to the object's size).

6.4.1 BOUNDARY Mask

\$_BOUNDRY is a predefined mask that represents the bounding box of all objects in the GDSII file thus, the chip area boundary. It is useful for defining boundary conditions for mask definition.

6.4.2 Custom Masks

```
begin_mask
{
  derived_mask_name = mask_spec |
  derived_mask_name = [( ] mask_spec [ ) ]
    {AND | OR | XOR | NOT}
    [( ]mask_spec [ ) ] |
  derived_mask_name = NOT mask_spec |
  derived_mask_name = SIZE(mask_spec, {sizing_value/
    $size_parameter })
}
end_mask | end
```

Where

- *derived_mask_name* is any alphanumeric.
- *derived_mask_name* must be unique within the scope of all *derived_mask_names* within the **mask** section and variable names within the **parameters** section.
- *mask_spec* = { *GDSII_spec* | *\$derived_mask_name* }
- *GDSII_spec* is defined as in Section 6.1.7

Syntax
Mask Section Syntax

- **AND, OR, XOR** (exclusive-OR), and **NOT** are logical operations to be performed on the surrounding GDSII or derived masks.
- **SIZE** is an arithmetic operator that adds the specified *sizing_value* to each side of the specified GDSII or derived mask object.
- *sizing_value* is a signed real or integer number.

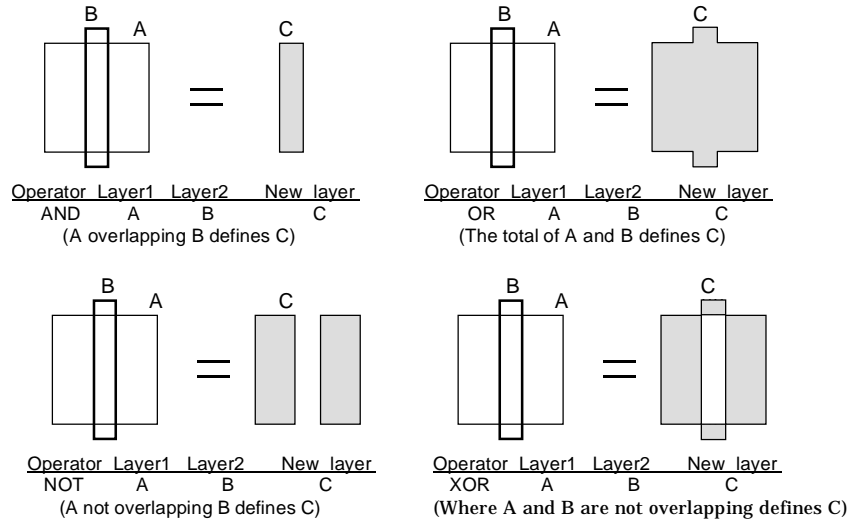


Figure 25— Mask Layer Boolean Operations

6.4.3 Mask Section Examples

6.4.3.1 Caps Example

Suppose the deposition of a dielectric layer on top of metal only covers the top surface area of the metal (and not the entire surface of the chip) as shown in Figure 4. In this case, using **mask**, it may be represented as follows:

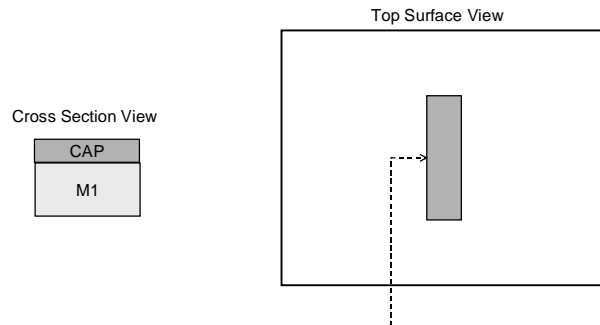


Figure 26— Dielectric Cap

```
begin_mask
M1GDSII = (10;1) (10;2)
Dielectric1 = (15;1) (15;2)
CAPmask = $M1GDSII AND $Dielectric1
. . .
end

begin_process
type = metal
name = M1
thickness = ..
mask = $M1GDSII

type = dielectric
name = CAP
mask = $CAPmask # Defines the dielectric to exist only on
                 # the top surface area of metal 1 lines.
thickness = ..
K = ..
. . .
end
```

6.4.3.2 "Dummy" (Fill) Metal Example

The rules under which "dummy" fill metal will be deposited may be defined by use of the **mask** section. In this example, "dummy" metal (for the metal 1 layer) is defined to exist wherever there is a gap between M1 lines of greater than 7 um.

```
distance = um

begin_parameters
M1_Half = 7
Metall = (1;16)
end

begin_mask
M1Buffer = SIZE($Metall, $M1_Half)      # Adds 7um to each
                                           # side of the
                                           # M1 GDSII_reference
                                           # metal (1,16).
Fill = NOT $M1Buffer                    # Defines the areas where there is no
                                           # metal within 7um.

. . .
end

begin_process
type = metal
name = M1
mask = $Metall
thickness = ..
. . .

type = metal
name = M1Fill
mask = $Fill
thickness = ..
. . .

end
```

6.5 Process Section Syntax

6.5.1 Substrate Layer Syntax

Substrate statements are used to characterize layers that comprise the bare die and layers that are diffused into the bare die. Examples of substrate layers are bulk, diffusions or wells, field oxides and SOI oxides, bottom epoxy, and epi.

```
type = substrate
subtype = well | diffusion | trench | oxide | buried
```

Syntax
Process Section Syntax

name = layer name
reference_ground = specifies that the bottom surface of this layer is the capacitance ground-0 reference.
mask = mask object reference
user_property = user-defined properties
height = Distance (Z-direction) from the lowest point of the top surface of the chip after all previous process steps
measured_from = name of layer from which the **height** measured.
thickness = applied or diffused thickness of this layer
k = dielectric constant
resistivity = resistivity
r_sheet = sheet resistance

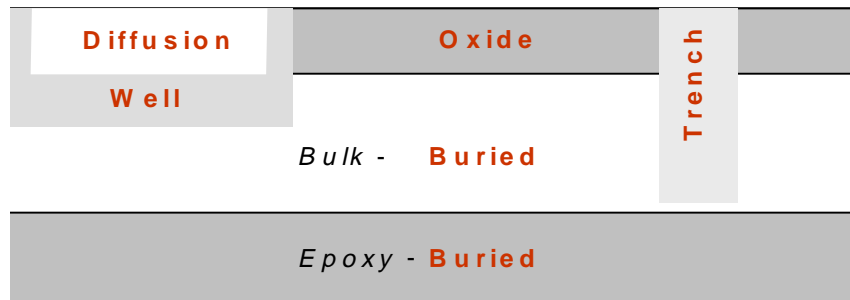


Figure 27—Substrate Layers

6.5.1.1 Layer Type

Layer Type is used to define this layer as a substrate.

```
type = substrate
```

6.5.1.2 Layer Subtype

Layer Subtype is used to defines the function of this substrate layer as either a well, diffusion, trench, oxide, or buried.

```
subtype = { well | diffusion | trench | oxide | buried }
```

6.5.1.3 Layer Name

Layer Name is a user-defined name identifying this layer. This name must be unique across the scope of the **process** section. Case sensitivity is determined by **case_sensitivity** statement.

```
name = username
```

6.5.1.4 Ground-Zero Reference Surface

Ground-Zero Reference Surface defines the bottom surface of this layer to be the capacitance ground-0 reference.

```
reference_ground
```

6.5.1.5 Mask Reference

Mask is used to identify areas where special geometries apply. If no **mask** is specified, the same definition is applied to the entire surface.

```
mask = mask_spec
```

Where:

- mask_spec = { GDSII_spec | derived_mask_name }

6.5.1.6 User Properties

User Properties define types and values for user-defined properties for this layer.

```
user_property = name, {string|int|real} [,range # #]  
                ( {string_value/value}  
                  [...] )
```

Where:

- string_value is enclosed in quotes.

6.5.1.7 Height

Height is the distance (Z-direction) from the lowest point of the top of the chip after all previous process steps (unless overridden by **measured_from**).

```
height = { [F] (value) |  
           R (min_value, nom_value, max_value) |  
           N (mean_value, variation_value) |  
           D (width)[(mask = mask_spec)]  
           (height_value1,width_value1  
            [height_value2,width_vale2]  
            [...] )  
         }
```

Where:

- **height** is a vector value that may be signed (+ or -) indicating a direction either up or down respectively from the reference surface. If no sign is specified, then the direction shall be defaulted to negative (-) or down for *diffusions* and positive (+) or up for all other instances.
- *width_value* is the width of the mask referenced object or the object below the dielectric.
- *mask_spec* = { GDSII_spec | derived_mask_name }

- height_value and width_value are Fixed

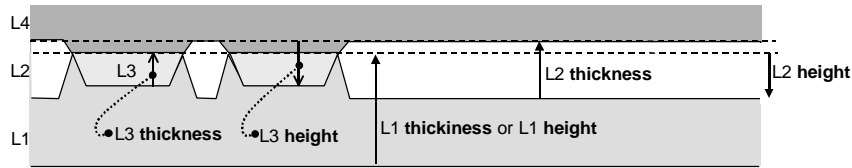


Figure 28— Height vs. Thickness

6.5.1.8 Height Reference

Height Reference specifies an override to the default reference surface for **height**. In this case, the lowest point of the top surface of the layer specified (considering all process steps previous to that layer) shall be used as the base from which the **height** is measured.

measured_from = *layername*

6.5.1.9 Thickness

Thickness is the applied or diffused thickness of this layer.

```
thickness = { [F] (value) |
               R (min_value, nom_value, max_value) |
               N (mean_value, variation_value) |
               D (spacing { [min | max | nom] })
               [ (mask = mask_spec) ]
               ( thickness_value1, spacing_value1
                 [ thickness_value2, spacing_value2 ]
                 [...] )
               }
```

Where:

- *spacing_value* is the **minimum**, **maximum**, or **nominal** (average) spacing between the *mask_spec* object and its neighbors, or the nearest objects (on this layer) to this location.
- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }
- *thickness_value* and *spacing_value* are Fixed

6.5.1.10 Permittivity

Permittivity is used to specifies the Relative permittivity (dielectric constant).

```
K = { [F] (value) |
       R (min_value, nom_value, max_value) |
       N (mean_value, variation_value) |
       D (spacing { [min | max | nom] })
```

Syntax
Process Section Syntax

```
[ ( mask = mask_spec )  
  ( permittivity_value1, spacing_value1  
    [ permittivity_value2, spacing_value2 ]  
    [ ... ] )  
}
```

Where:

- *spacing_value* is the **minimum**, **maximum**, or **nominal** (average) spacing between the *mask_spec* object and its neighbors, or the nearest objects (on this layer) to this location.
- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }
- *Permittivity_value* and *thickness_value* are Fixed

6.5.1.11 Resistance

Specification of the resistance for this layer may be specified as resistivity or sheet resistance. Multiple values may be specified, each for different temperature values.

```
resistivity = { [ F ] ( value ) , @temp_val |  
                R ( min_value, nom_value, max_value ), @temp_val |  
                N ( mean_value, variation ), @temp_val }  
                [ ... ]  
  
r_sheet =      { [ F ] ( value ) , @temp_val |  
                R ( min_value, nom_value, max_value ), @temp_val |  
                N ( mean_value, variation ), @temp_val }  
                [ ... ]
```

Where:

- @*temp_val* is the temperature at which the resistivity or sheet resistance are specified.

6.5.2 Dielectric Layer Syntax

Dielectric defines the electrical and physical characteristics of insulating layers sandwiched between metal on an interconnect layer, and between interconnect layers. Dielectrics may be planar or conformal.

```
type = dielectric  
name = layer name  
reference_ground = specifies the bottom surface of this layer to  
                    be the capacitance ground-0 reference.  
mask = mask reference  
user_property = user-defined properties  
  
thickness = the maximum thickness of dielectric layer over the  
             surface on which it is deposited.  
height = Distance (Z-direction) from the lowest point of the top  
          surface of the chip after all previous process steps.  
          The reference surface for height specifications may be
```

overridden by the **measured_from**.
planarity = type of dielectric
k = dielectric constant
swthk = conformal coating thickness on side walls
edge_pattern = shape or slope of side walls (if other than 90°)
measured_from = name of layer from which the **height** measured.
min_gap = minimum gap before which a dielectric fills the area
air_gap = specifies air gap (dielectric void) characteristics

6.5.2.1 Layer Type

Defines this layer to be a dielectric.

```
type = dielectric
```

6.5.2.2 Layer Name

User-defined name identifying this layer. This name must be unique across the scope of the **process** section. Case sensitivity is determined by **case_sensitivity** statement.

```
name = username
```

6.5.2.3 Ground-Zero Reference Surface

Defines the bottom surface of this layer to be the capacitance ground-0 reference.

```
reference_ground
```

6.5.2.4 Mask Reference

Identify areas where special geometries apply. If no **mask** is specified, the same definition is applied to the entire surface.

```
mask = mask_spec
```

Where:

```
- mask_spec = { GDSII_spec | derived_mask_name }
```

6.5.2.5 User Properties

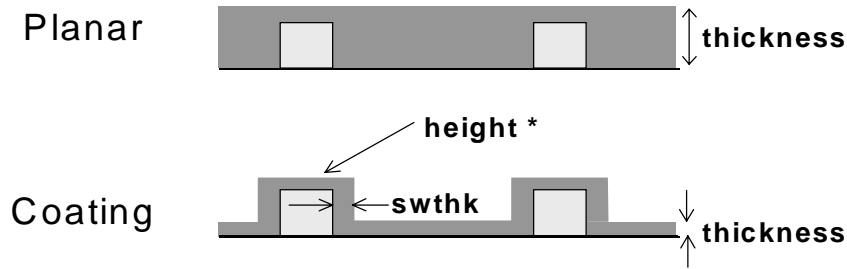
Define types and values for user-defined properties for this layer.

```
user_property = name, {string|int|real} [,range # #]  
    ( {string_value/value}  
    [...] )
```

Where:

```
- string_value is enclosed in quotes.
```

Syntax
Process Section Syntax



* *Note:* The reference plane for **height** is the lowest point of the top of the chip after all previous process steps (unless overridden by **measured_from**).

Figure 29— Dielectric Types

6.5.2.6 Dielectric Type

Defines the dielectric type to be either planar or conformal.

```
planarity = { P[lanar] | C[oating] }
```

Where:

- **P** or **Planar** = planar dielectric
- **C** or **Coating** = conformal dielectric
- If **planarity** is not specified, **planar** shall be assumed.

6.5.2.7 Thickness

Specifies the applied thickness of the dielectric (see Figure 6) as follows

```
thickness = { [F] (value) |
              R (min_value, nom_value, max_value) |
              N (mean_value, variation_value) |
              D (spacing { [min | max | nom] })
                [ (mask = mask_spec)
                  ( thickness_value1, spacing_value1
                    [ thickness_value2, spacing_value2 ]
                    [...] )
                ]
            }
```

Where:

- *spacing_value* is the **minimum**, **maximum**, or **nominal** (average) spacing between the *mask_spec* object and its neighbors, or the nearest objects (on this layer) to this location.
- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }
- *thickness_value* and *spacing_value* are Fixed

6.5.2.8 Height

Distance (Z-direction) from the lowest point of the top surface of the chip after all previous process steps. The reference surface may be overridden by a **measured_from** statement within this layer.

```

height =      { [ F ] ( value ) |
                  R ( min_value, nom_value, max_value ) |
                  N ( mean_value, variation_value ) |
                  D ( width ) [ ( mask = mask_spec ) ]
                    ( height_value1, width_value1
                      [ height_value2, width_value2 ]
                      [ ... ] )
                  }

```

Where:

- **height** is a vector value that may be signed (+ or -) indicating a direction either up or down respectively from the reference surface. If no sign is specified, then the direction shall be defaulted to negative (-) or down for *diffusions* and positive (+) or up for all other instances.
- *width_value* is the width of the mask referenced object or the object below the dielectric.
- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }
- *height_value* and *width_value* are Fixed

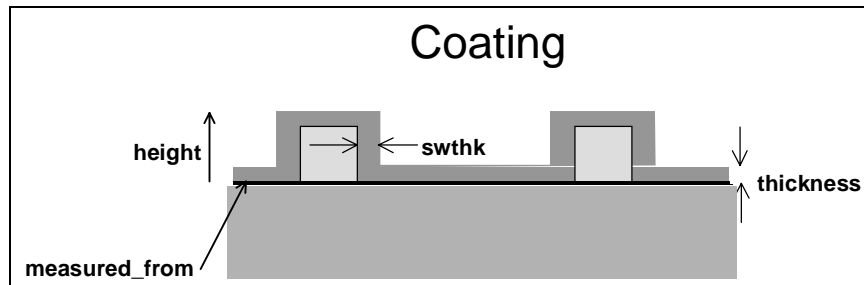


Figure 30— Coating Height, Thickness, and Sidewall Thickness

6.5.2.9 Height Reference

Specifies an override to the default reference surface for **height**. In this case, the lowest point of the top surface of the layer specified (considering all process steps previous to that layer) shall be used as the base from which the **height** is with respect to.

```

measured_from = layername

```

6.5.2.10 Permittivity

Relative permittivity (dielectric constant) value.

```

K =          { [F] (value) |
                R (min_value, nom_value, max_value) |
                N (mean_value, variation_value) |
                D (spacing { [min | max | nom] })
                  [(mask = mask_spec)]
                  (permittivity_value1, spacing_value1
                   [ permittivity_value2, spacing_value2]
                   [...] )
                }

```

Where:

- *spacing_value* is the **minimum**, **maximum**, or **nominal** (average) spacing between the *mask_spec* object and its neighbors, or the nearest objects (on this layer) to this location.
- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }
- *permittivity_value* and *spacing_value* are Fixed.

6.5.2.11 Sidewall Thickness

Specifies the thickness on the sides walls of coatings. Thickness shall be defined as the distance from a any point on the inside edge of the coating along the perpendicular projection from that point to the outside edge of the coating.

```

swthk =       { [F] (value) |
                R (min_value, nom_value, max_value) |
                N (mean_value, variation_value) |
                D (spacing { [min | max | nom] })
                  [(mask = mask_spec)]
                  (sw_thickness_value1, spacing_value1
                   [ sw_thickness_value2, spacing_value2]
                   [...] )
                }

```

Where:

- *spacing_value* is the **minimum**, **maximum**, or **nominal** (average) spacing between the *mask_spec* object and its neighbors, or the nearest objects (on this layer) to this location.
- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }
- *sw_thickness_value* and *spacing_value* are of type **F**

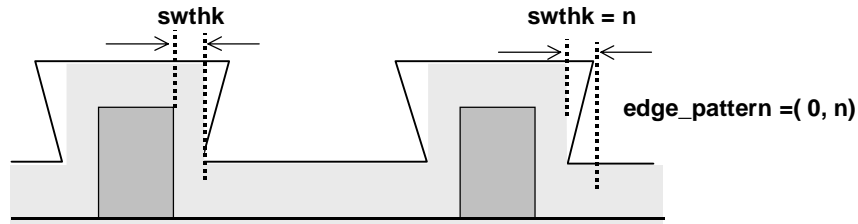


Figure 31— Dielectric Sidewall

6.5.2.12 Sidewall Edge Shape

Specifies the shape or slope of the conformal dielectric along the metal's sidewall. This attribute is optional and if not specified, the slope shall be assumed 90°.

edge_pattern = (*x_val1* [, *x_val2*] [...])

Where:

- x_val_n is a set of x-coordinates relative to the metal edge after biasing which describes the edge shape. These are specified for equidistant y-coordinates which are in equal increments (each equal to **thickness** / (the number of specified x_vals)) and are ordered sequentially beginning from the bottom surface of the metal. Specification of only two x_val values denotes a constant slope.

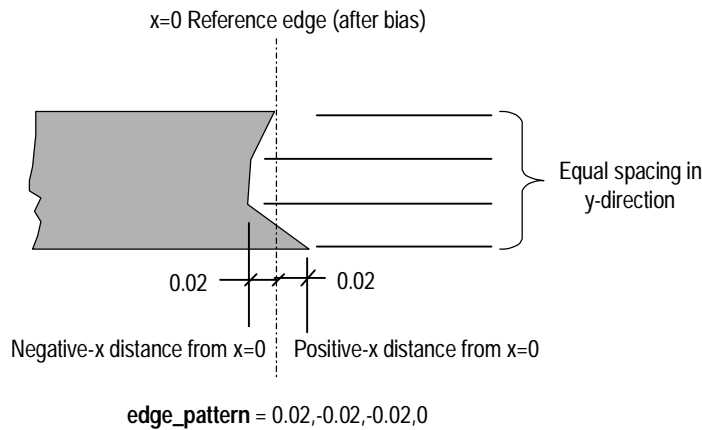


Figure 32— Dielectric Edge Shape

6.5.2.13 Fill Gap

Specifies the minimum gap between metals below which a coating dielectric will fill in the space between them.

mingap = { [**F**] (*value*) |
R (*min_value*, *nom_value*, *max_value*) |
N (*mean_value*, *variation_value*) }

6.5.2.14 Air Gap

Specifies the shape, size, and position where air-gaps (dielectric voids) may exist. An air-gap or “void” forms when a conformal dielectric with re-entrant sidewalls “pinches off”. When this occurs, a planar dielectric with an embedded void results. There is a threshold space beyond which the dielectric is conformal, and below which it is planar with a void. This is illustrated in the figure below:

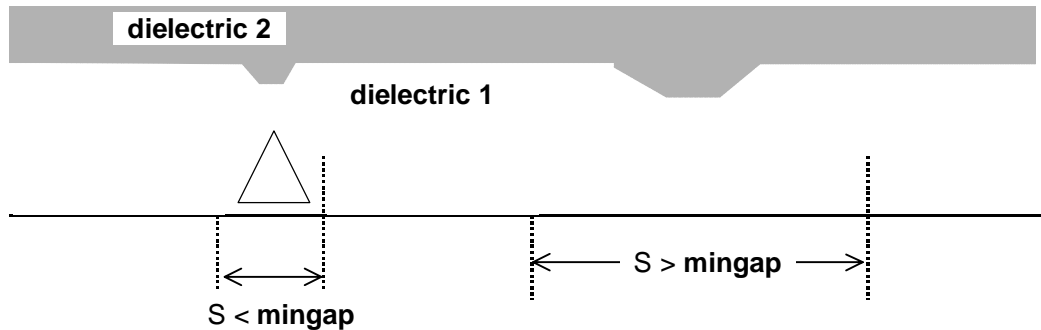


Figure 33— Air Gaps Defined

For $s < \text{mingap}$, the conformal dielectric (“diel1”) pinches off, becoming a planar dielectric with an embedded void. For $s > \text{mingap}$, diel1 does not pinch off, and remains conformal.

Referring to Figure 11, the following syntax defines air gaps:

```
airgap = { R[ectangle] | T[riangle] },
          (space1, w1, h1, x1, b1), (space2, w2, h2, x2, b2)
          [...]
```

Where:

- h = height of rectangle or triangle
- w = width of the air-gap at its base
- b = distance to base of air-gap from the bottom surface of the dielectric
- x = distance from a metal edge to the closest corner of the air-gap's base
- $space_n$ = typically equals **min_gap**

Syntax
Process Section Syntax

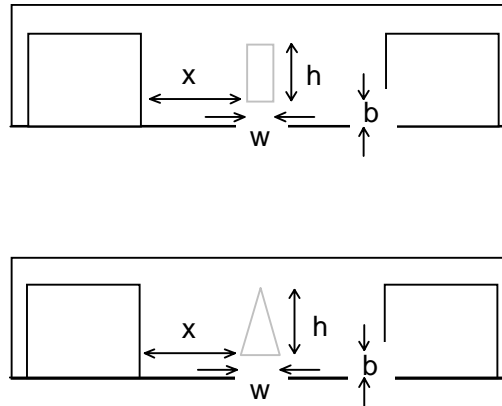


Figure 34—Air Gaps

6.5.3 Metal Layer Syntax

Metal defines the physical and electrical characteristics of the interconnect material (metal or poly) on a given layer.

```
type = metal
name = layer name
mask = mask object reference
user_property = user-defined properties
planarity = metal shape
thickness = metal thickness (z-direction)
edge_bias = edge bias relative to drawn width
resistivity = resistivity
r_sheet = sheet resistance
edge_pattern = shape or slope of side walls (if other than 90°)
swthk = conformal metal coating thickness on side walls
height = Distance (Z-direction) from the lowest point of the top
           surface of the chip after all previous process steps.
           The reference surface for height specifications may be
           overridden by the measured_from.
measured_from = name of layer from which the height measured.
corner_radius = radius of corners on the pattern edges
min_width = minimum allowed metal width
min_spacing = minimum allowed metal spacing
```

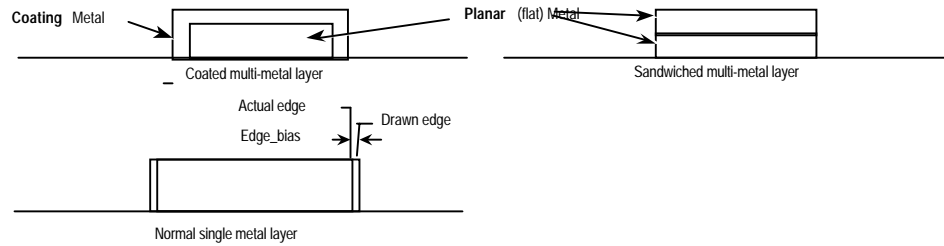


Figure 35—Metal Layer

6.5.3.1 Layer Type

Defines this layer to be metal.

```
type = metal
```

6.5.3.2 Layer Name

User-defined name identifying this layer. This name must be unique across the scope of the **process** section. Case sensitivity is determined by **case_sensitivity** statement.

```
name = username
```

6.5.3.3 Mask Reference

Identify areas where special geometries apply. If no **mask** is specified, the same definition is applied to the entire surface.

```
mask = mask_spec
```

Where:

- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }

6.5.3.4 User Properties

Define types and values for user-defined properties for this layer.

```
user_property = name, { string | int | real } [, range # # ]  
                ( { string_value / value }  
                  [ ... ] )
```

Where:

- *string_value* is enclosed in quotes.

6.5.3.5 Metal Shape

Defines the metal to be either planar or a coating.

planarity = P[lanar] | C[oating]

Where:

- **P** or **Planar** = flat metal
- **C** or **Coating** = coating (conformal) metal

Note:

If **planarity** is not specified, **Planar** shall be assumed.

6.5.3.6 Metal Thickness

Specify the thickness of the metal in the z-direction.

```
thickness = { [F] (value) |  
                R (min_value, nom_value, max_value) |  
                N (mean_value, variation_value) |  
                D (spacing { [min | max | nom] })  
                [(mask = mask_spec)]  
                (thickness_value1, spacing_value1  
                [thickness_value2, spacing_value2]  
                [...] )  
            }
```

Where:

- *spacing_value* is the **minimum**, **maximum**, or **nominal** (average) spacing between the *mask_spec* object and its neighbors, or the nearest objects (on this layer) to this location.
- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }
- *thickness_value* and *spacing_value* are Fixed

Note:

- **Planar** metal shall be assumed if only *value* is given.

6.5.3.7 Metal Height

Distance (Z-direction) from the lowest point of the top surface of the chip after all previous process steps to the topmost surface of this metal. The reference may be overridden by a **measured_from** statement within this layer.

```

height =      { [F] (value) |
                  R (min_value, nom_value, max_value) |
                  N (mean_value, variation_value) |
                  D (width) [(mask = mask_spec)]
                    ( height_value1, width_value1
                      [ height_value2, width_value2]
                      [...] )
                  }

```

Where:

- **height** is a vector value that may be signed (+ or -) indicating a direction either up or down respectively from the reference surface. If no sign is specified, then the direction shall be defaulted to negative (-) or down for *diffusions* and positive (+) or up for all other instances.
- *width_value* is the width of the mask referenced object or the object below the metal.
- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }
- *height_value* and *width_value* are Fixed

6.5.3.8 Height Reference

Specifies an override to the default reference surface for **height**. In this case, the lowest point of the top surface of the layer specified (considering all process steps previous to that layer) shall be used as the base from which the **height** is with respect to.

measured_from = *layername*

6.5.3.9 Metal Edge Bias

Specify the bias on the edge of metal objects with respect to the drawn width. Values may be positive or negative. Actual metal width is equal to the drawn width plus 2 x (*edge_bias*).

```

edge_bias =    [F] (value) |
                  R (min_value, nom_value, max_value) |
                  N (mean_value, variation_value) |
                  D (spacing {[min | max | nom]})
                    [(mask = mask_spec)]
                    ( bias_value1, spacing_value1
                      [ bias_value2, spacing_value2]
                      [...] )
                  }

```

Where:

- *spacing_value* is the **minimum**, **maximum**, or **nominal** (average) spacing between

the *mask_spec* object and its neighbors, or the nearest objects (on this layer) to this location.

- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }
- *bias_value* and *spacing_value* are Fixed

6.5.3.10 Metal Edge Shape

Specify the actual shape of the interconnect metal edges.

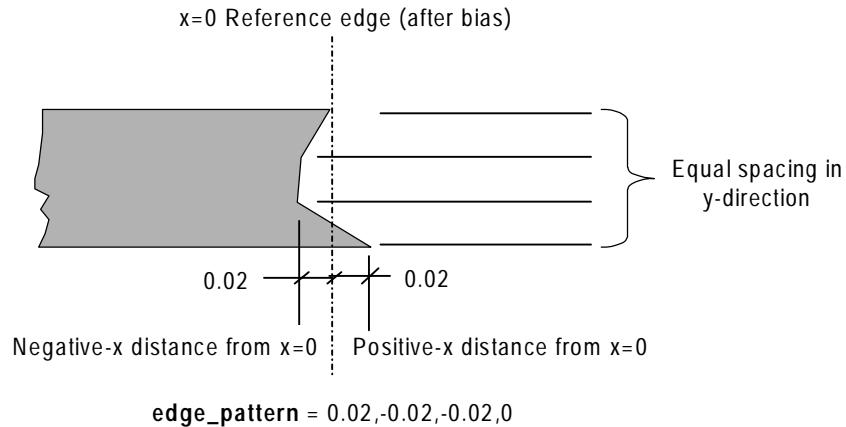


Figure 36—Metal Edge Shape

`edge_pattern` = (*x_val1* [, *x_val2*] [...])

Where:

- *x_val_n* is a set of x-coordinates relative to the metal edge after biasing which describes the edge shape. These are specified for equidistant y-coordinates which are in equal increments (each equal to [**thickness** / (the number of specified *x_val*s)]) and are ordered sequentially beginning from the bottom surface of the metal.
- Specification of only two *x_val* values denotes a constant slope.

6.5.3.11 Metal Resistance

Specification of the metal resistance for this layer (may be specified as resistivity or sheet resistance). Multiple values may be specified, each for different temperature values.

```

resistivity = {[F](value),@temp_val |
                [R(min_value,nom_value,max_value),@temp_val |
                [N(mean_value,variation),@temp_val
                [D (width {[min | max | nom]})
                  [(mask = mask_spec)]
                  (resistivity_value1, width_value1
                  [resistivity_value2, width_value2]
                  [...])
                ]
    }

```

Where:

- *width_value* is the **minimum**, **maximum**, or **nominal** (average) width of the *mask_spec* object or the metal width for which the resistivity is based.
- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }
- *resistivity_value* and *width_value* are Fixed
- *@temp_val* is the temperature at which the resistivity is specified.

```
r_sheet =      {[F](value) , @temp_val |
                R(min_value,nom_value,max_value), @temp_val|
                N(mean_value,variation), @temp_val} |
                D (width {[min | max | nom]})
                  [(mask = mask_spec)]
                  ( r_sheet_value1, width_value1
                    [ r_sheet_value2, width_value2]
                    [...] )
                }
```

Where:

- *width_value* is the **minimum**, **maximum**, or **nominal** (average) width of the *mask_spec* object or the metal width for which the sheet resistance is based.
- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }
- *r_sheet_value* and *width_value* are Fixed.
- *@temp_val* is the temperature at which the sheet resistance is specified.

6.5.3.12 Sidewall Thickness

Specifies the thickness of coating metals. Thickness shall be defined as the distance from a any point on the inside edge of the coating along the perpendicular projection from that point to the outside edge of the coating.

```
swthk =      {[F] (value) |
                R (min_value, nom_value, max_value) |
                N (mean_value, variation_value) |
                D (spacing {[min | max | nom]})
                  [(mask = mask_spec)]
                  ( sw_thickness_value1, spacing_value1
                    [ sw_thickness_value2, spacing_value2])
                }
```

Where:

- *sw_thickness_value* is the **minimum**, **maximum**, or **nominal** (average) thickness of the *mask_spec* object or the coating metal on a side wall.
- *mask_spec* = { *GDSII_spec* | *derived_mask_name* }
- *sw_thickness_value* and *spacing_value* are Fixed

6.5.3.13 Metal Corner Shape

Specifies the radius of the corners of metal objects on their pattern edges.

```
corner_radius = (outside_radius [, inside_radius])
```

Where:

- If only *outside_radius* is specified, then that value shall be used for both the outside and inside radii.

6.5.3.14 Minimum Metal Width

Specifies the minimum width of metal objects on this layer.

```
min_width = value
```

Where:

- *value* is the minimum allowed metal width and is of type Fixed

6.5.3.15 Minimum Metal Spacing

Specifies the minimum allowed spacing between conductors on this layer.

```
min_spacing = value [, layer_name]
```

Where:

- *value* is the minimum allowed metal pitch and is of type Fixed
- *layer_name* is optional and represents a second layer and the spacing refers to the minimum allowed spacing between metal on this layer and the second layer specified. If *layer_name* is not specified, the spacing value is for metals on this metal layer.

6.5.4 Via Layer Syntax

Via layers define the physical and electrical properties of connections between metal layers.

```
type = via  
name = layer name  
mask = mask object reference  
user_property = user-defined properties  
planarity = planar or coating  
edge_pattern = shape or slope of via side walls (if other than  
90°)  
swthk = conformal coating thickness on the via side walls  
height = distance (Z-direction) from the lowest point of the top  
surface of the after all previous process steps chip to  
the top surface of the via. The reference surface for  
height specifications may be overridden by  
measured_from.
```

Syntax
Process Section Syntax

measured_from = name of layer from which the **height** measured.
thickness = via thickness (z-direction)
corner_radius = radius of corners on the pattern edges
min_width = minimum allowed via width
min_spacing = minimum allowed via spacing
corner_radius = radius of the via corners
resistivity = resistivity
r_per_via = via resistance

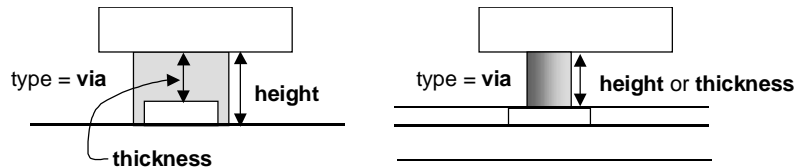


Figure 37— Via Layer

6.5.4.1 Layer Type

Defines this layer to be a via.

```
type = via
```

6.5.4.2 Layer Name

User-defined name identifying this layer. This name must be unique across the scope of the **process** section. Case sensitivity is determined by **case_sensitivity** statement.

```
name = username
```

6.5.4.3 Mask Reference

Identify areas where special geometries apply.

```
mask = mask_spec
```

Where:

```
- mask_spec = { GDSII_spec | derived_mask_name }
```

6.5.4.4 User Properties

Define types and values for user-defined properties for this layer.

```
user_property = name, {string|int|real} [,range # #]  
    ( {string_value/value}  
      [...] )
```

Where:

- *string_value* is enclosed in quotes.

6.5.4.5 Via Shape

Defines the via shape to be either planar or coating.

```
planarity = P[lanar] | C[coating]
```

Where:

- **P** or **Planar** = via stud
- **C** or **Coating** = coating (conformal)
- If **planarity** is not specified, **Planar** shall be assumed.

6.5.4.6 Via Thickness

Specify the thickness of the via in the z-direction.

```
thickness = { [F] (value) |  
              R (min_value, nom_value, max_value) |  
              N (mean_value, variation_value)  
            }
```

Note:

- **Planar** metal shall be assumed if only value is given.

6.5.4.7 Via Height

Distance (Z-direction) from the lowest point of the top surface of the chip after all previous process steps to the topmost surface of this via. The reference may be overridden by a **measured_from** statement within this layer.

```
height =      { [F] (value) |  
               R (min_value, nom_value, max_value) |  
               N (mean_value, variation_value)
```

Where:

- **height** is a vector value that may be signed (+ or -) indicating a direction either up or down respectively from the reference surface. If no sign is specified, then the direction shall be defaulted to negative (-) or down for *diffusions* and positive (+) or up for all other instances.

6.5.4.8 Height Reference

Specifies an override to the default reference surface for **height**. In this case, the lowest point of the top surface of the layer specified (considering all process steps previous to that layer) shall be used as the base from which the **height** is with respect to.

measured_from = *layername*

6.5.4.9 Via Edge Shape

Specify the actual shape of the via sidewalls.

```
edge_pattern = (x_val1 [, x_val2][...])
```

Where:

- x_val_n is a set of x-coordinates relative to the via sidewall that describes the edge shape. These are specified for equidistant y-coordinates which are in equal increments (each equal to **thickness** / (the number of specified x_vals)) and are ordered sequentially beginning from the bottom surface of the via. Specification of only two x_val values denotes a constant slope.

6.5.4.10 Sidewall Thickness

Specifies the thickness of the via coating. Thickness shall be defined as the distance from a any point on the inside edge of the coating along the perpendicular projection from that point to the outside edge of the coating.

```
swthk = { [F] (value) |  
          R (min_value, nom_value, max_value) |  
          N (mean_value, variation_value)  
        }
```

6.5.4.11 Minimum Via Width

Specifies the minimum width of vias on this layer.

```
min_width = value
```

Where:

- $value$ is the minimum allowed via width and is of type Fixed

6.5.4.12 Minimum Via Spacing

Specifies the minimum allowed spacing between vias on this layer.

```
min_spacing = value
```

Where:

- $value$ is the minimum allowed via spacing and is of type Fixed

6.5.4.13 Via Corner Shape

Specifies the radius of the corners of via objects on their pattern edges.

```
corner_radius = (outside_radius [, inside_radius])
```

Where:

- *If only outside_radius is specified, then that value shall be used for both the outside and inside radii.*

6.5.4.14 Via Resistance

Specification of the via resistance (may be specified as resistivity or resistance). Multiple values may be specified, each for different temperature values.

```
resistivity = { [F] (value, @temp_val ...) |
               R (min_value, nom_value, max_value,
                  @temp_val [...])          |
               N (mean_value, variation_value, @temp_val [...])
               }

r_per_via =   { [F] (value, @temp_val, side1_value, side2_value
                   [...])          |
               R (min_value, nom_value, max_value,
                  @temp_val, side1_value, side2_value [...]) |
               N (mean_value, variation_value, @temp_val,
                  side1_value, side2_value [...])
               }
```

Where:

- *side1_value* and *side2_value* are the via's cross-sectional length and width.
- If only one *side_value* is specified, then width shall be equal to diameter (circular via).
- *@temp_val* is the temperature at which the resistivity or via resistance are specified.

7.0 Informative

7.1 SIPPs Abstract Model Techniques

A SIPPs abstract model uses a minimum set of parameters to provide an electrical prediction of the interconnect parasitics.

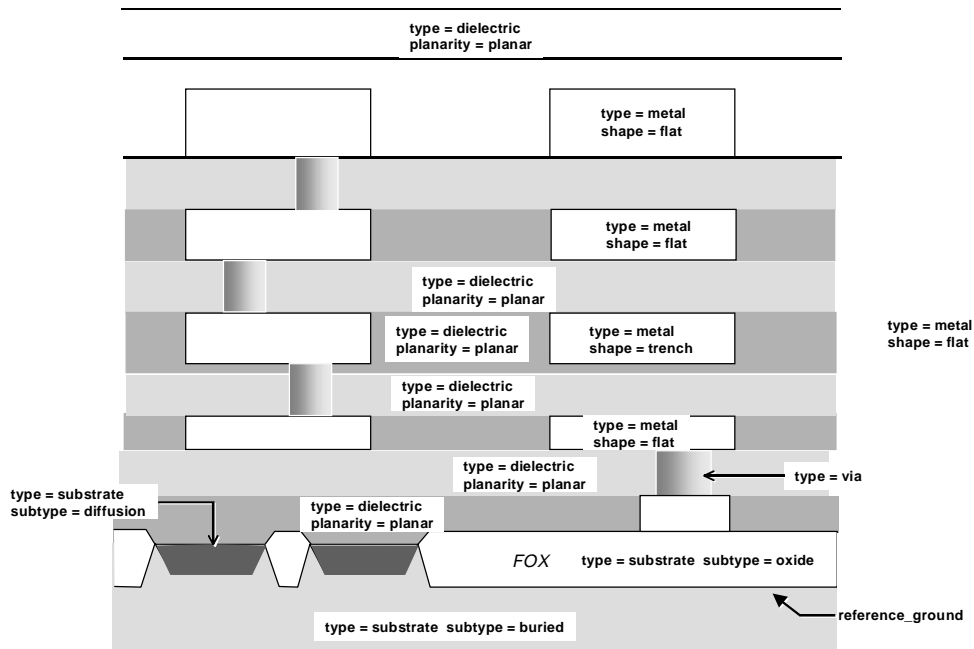


Figure 38— Abstract Modeling

The cross-section of an abstract SIPPs representation is shown in the above figure. Conceptually, it is composed of **substrate** layers, planar **dielectric-metal** layer pairs, and **vias**. Each of these is represented by a subset of the SIPPs Physical data model elements where the values may represent an abstraction that models the effects of several process layers into one. Thus, the parameters are electrically predictive parameters, not exact representation of process geometry.

1. Each interconnect layer is considered as a **dielectric-metal** pair, where there are two planar dielectrics. One lies between the conducting lines on the same layer (same-layer dielectric), and the other is between the conductors of a layer and the layer above (inter-layer dielectric). The dielectric constants of the two need not to be the same.
2. A **via** is modeled as a cylinder connecting two conductor polygons at adjacent layers. The height of the via is the thickness for the inter-layer dielectric.
3. The cross-sectional views of interconnects are modeled as rectangles only.

The model closely approximates, but does not exactly replicate, the interconnect geometry of current IC processes. Instead, it intentionally simplifies the actual physical geometry. Nonetheless, the model may provide an electrically equivalent model for the interconnect system when dimensioned with properly cho-

sen parameter values. Using this model, a field solver operating on suitable test structures will predict R's or C's whose values model those measured from silicon in an abstract form.

7.1.1 Equivalent-brick model for conductors

In certain cases, trapezoidal-shape conductors, as shown in Figure 16, can be simplified with equivalent "bricks" as shown in Figure 17.

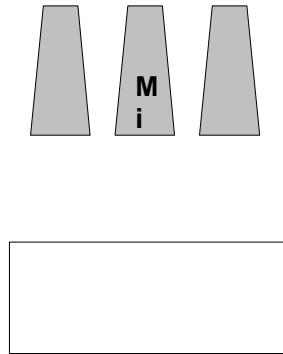


Figure 39— Three Trapezoidal-Shape Conductors

The equivalent-brick model must be chosen so that the coupling capacitance from M_i to its neighbors and the ground capacitance are equivalent to that of the actual trapezoids

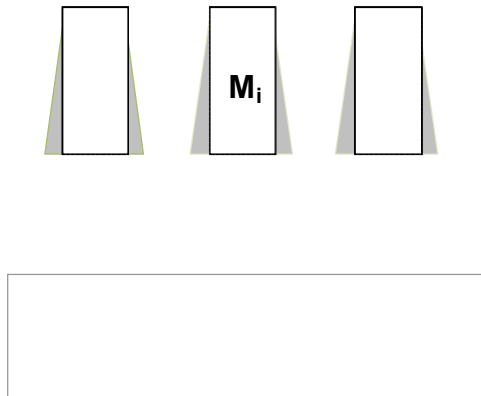


Figure 40—Equivalent-Brick Conductors

7.1.2 Equivalent model for multiple dielectrics

In certain cases, multiple dielectrics between metal layers, such as the three between M_{i+1} and M_i in Figure 18, can be simplified to two equivalent dielectrics.

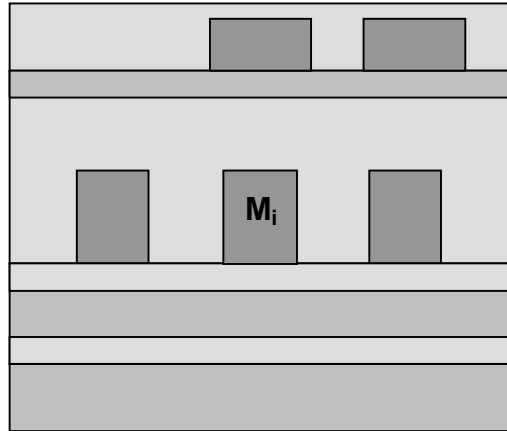


Figure 41—Multiple Dielectrics

Figure 20 below depicts the equivalent simplified model for the three dielectric layers in the example shown above in Figure 19

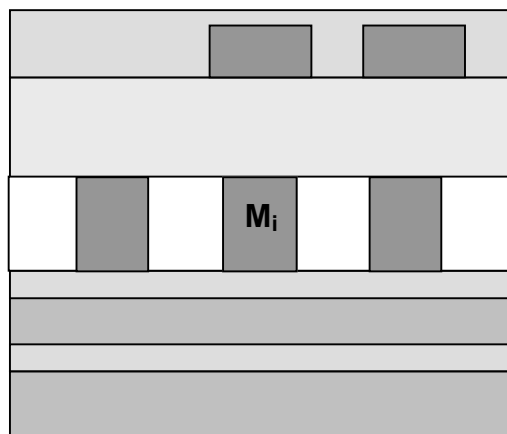


Figure 42—Reduction of Multiple Dielectrics

7.2 Sample SIPPs Data

7.2.1 Example 1

The following describes the SIPPs parameters of a sample 0.18 μm process. This file was created by OEA International.

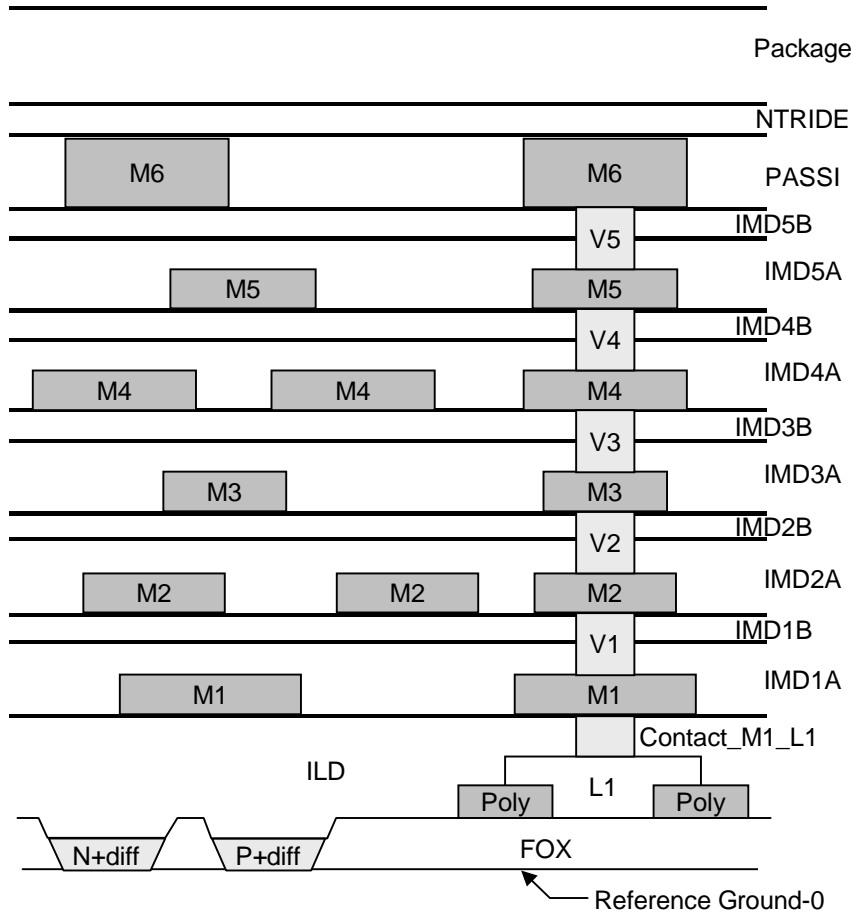


Figure 43— Example 0.18 micron technology

```
# Process XYZ_018CMOS
```

```
SIPPs = (version = 1, iteration = 00)
Technology = (process = "XYZ_018CMOS", contact = OEA)
resistance = ohm
resistivity = ohm.cm
distance = um
```

```
begin_parameters
poly_min_width = 0.16
poly_min_spacing = 0.25
```

Informative
Sample SIPPs Data

```
li_min_width = 0.25
li_min_spacing = 0.25
m1_min_width = 0.25
m1_min_spacing = 0.25
m2_min_width = 0.28
m2_min_spacing = 0.28
m3_min_width = 0.28
m3_min_spacing = 0.28
m4_min_width = 0.28
m4_min_spacing = 0.28
m5_min_width = 0.28
m5_min_spacing = 0.28
m6_min_width = 0.28
m6_min_spacing = 0.28
    end

begin_process
type = substrate
subtype = diffusion
name = P+diff
mask = 4
height = 0.20# Note: Height defaults to inus (-) because
              #           this is a diffusion subtype
r_sheet = R(4.2,7.2,11.2)

type = substrate
subtype = diffusion
name = N+diff
mask = 4
height = 0.20
r_sheet = R(4.2,7.2,11.2)

type = substrate
subtype = oxide
reference_ground
name = FOX
height = F(-0.20)
thickness = R(0.298,0.350,0.402)
K = 3.9

type = metal
name = poly
mask = 6;16
thickness = 0.200
edge_bias = -0.01
```

Informative
Sample SIPPs Data

```
resistivity = R(3200,3600,4000)  
min_width = $Poly_min_width  
min_spacing = $Poly_min_spacing
```

```
type = metal  
name = L1  
mask = 7  
thickness = 0.45  
resistivity = R(3.5,4.6,5.7)  
min_width = $Li_min_width  
min_spacing = $Li_min_spacing
```

```
type = via  
name = contact_M1_LI  
mask = 31  
height = R(6.4,7.0,8.6)  
R_per_via = 8.0
```

```
type = dielectric  
name = ILD  
thickness = R(6.4,7.0,8.6)  
K = 4.0  
type = metal  
name = M1  
mask = 8;18  
thickness = 0.500  
resistivity = R(2.3,3.5,4.7)  
min_width = $m1_min_width  
min_spacing = $m1_min_spacing
```

```
type = via  
name = V1  
mask = 32  
height = R(1.14, 1.40, 1.66)  
R_per_via = 6.5
```

```
type = dielectric  
name = IMD1A  
thickness = R(0.95,1.20,1.45)  
K = 3.8
```

```
type = dielectric  
name = IMD1B  
thickness = R(0.19,0.20,0.21)  
K = 4.2
```

Informative
Sample SIPPs Data

type = metal
name = M2
mask = 9;19
thickness = 0.500
resistivity = R(2.3,3.5,4.7)
min_width = \$m2_min_width
min_spacing = \$m2_min_spacing

type = via
name = V2
mask = 33
height = R(1.14, 1.40, 1.66)
R_per_via = 6.5

type = dielectric
name = IMD2A
thickness = R(0.95,1.20,1.45)
K = 3.8

type = dielectric
name = IMD2B
thickness = R(0.19,0.20,0.21)
K = 4.2

type = metal
name = M3
mask = 10;20
thickness = 0.500
resistivity = R(2.3,3.5,4.7)
min_width = \$m3_min_width
min_spacing = \$m3_min_spacing

type = via
name = V3
mask = 34
height = R(1.14, 1.40, 1.66)
R_per_via = 6.5

type = dielectric
name = IMD3A
thickness = R(0.95,1.20,1.45)
K = 3.8

type = dielectric

Informative
Sample SIPPs Data

```
name = IMD3B
thickness = R(0.19,0.20,0.21)
K = 4.2

type = metal
name = M4
mask = 11;21
thickness = 0.500
resistivity = R(2.3,3.5,4.7)
min_width = $m4_min_width
min_spacing = $m4_min_spacing

type = via
name = V4
mask = 35
height = R(1.14, 1.40, 1.66)
R_per_via = 6.5

type = dielectric
name = IMD4A
# Since planarity is not specified, the dielectric is Planar
thickness = R(0.95,1.20,1.45)
K = 3.8

type = dielectric
name = IMD4B
thickness = R(0.19,0.20,0.21)
K = 4.2

type = metal
name = M5
mask = 12;22
thickness = 0.500
resistivity = R(2.3,3.5,4.7)
min_width = $m5_min_width
min_spacing = $m5_min_spacing

type = via
name = V5
mask = 36
height = R(1.14, 1.40, 1.666)
R_per_via = 6.5

type = dielectric
name = IMD5A
```

Informative
Sample SIPPs Data

```
thickness = R(0.95,1.20,1.45)  
K = 3.8
```

```
type = dielectric  
name = IMD5B  
thickness = R(0.19,0.20,0.21)  
K = 4.2
```

```
type = metal  
name = M6  
mask = 13;23  
thickness = 0.95  
resistivity = R(2.1,3.4,4.7)  
min_width = $m6_min_width  
min_spacing = $m6_min_spacing
```

```
type = dielectric  
name = PASS1  
thickness = R(0.90,1.10,1.30)  
K = 4.2
```

```
type = dielectric  
name = Nitride  
thickness = R(0.29,0.35,0.41)  
K = 7.5
```

```
type = dielectric  
name = package  
thickness = 4.0  
K= 3.5  
End
```

```
# END
```

7.2.2 Example 2

The following describes the SIPPs parameters of a sample 0.25 μ m 5M1P technology. This file is created by Frequency Technology.

Informative
Sample SIPPs Data

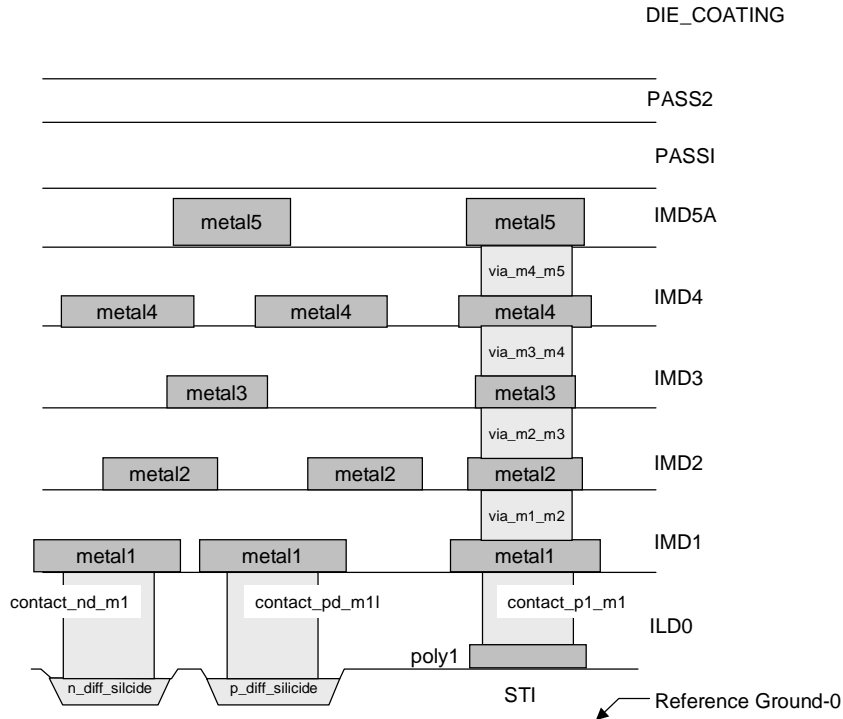


Figure 44— SIPPs parameters for a sample 0.25µm 5M1P technology

SIPPs Model

```

SIPPs = (version = 1, iteration = 00)
Technology = (process = 5M1P, contact = Frequency)
resistance = ohm
distance = um
temperature = F

begin_parameters
T1 = 0.7
m5_min_width = 0.5
m5_min_spacing = 0.5
end

begin_process

type = substrate
subtype = oxide
name = STI
thickness = 0.4
reference_ground
k = 4.0

type = substrate

```

Informative
Sample SIPPs Data

```
subtype = diffusion
name = n_diff_silicide
gdsII_layer = 2
height = -0.3
thickness = 0.2
r_sheet = F (5), 150
```

```
type = via
name = cont_nd_m1
gdsII_layer = 6
height = 1.0
r_per_via = F (5, 150, 0.25, 0.25)
```

```
type = substrate
subtype = diffusion
name = p_diff_silicide
gdsII_layer = 3
height = -0.3
thickness = 0.2
r_sheet = F (5), 150
```

```
type = via
name = cont_pd_m1
gdsII_layer = 6
height = 1.1
r_per_via = F (5, 150, 0.25, 0.25)
```

```
type = metal
name = poly1
gdsII_layer = 5
thickness = 0.25
r_sheet = F (5), 150
```

```
type = via
name = cont_p1_m1
gdsII_layer = 6
thickness = 0.75
r_per_via = F (5, 150, 0.25, 0.25)
```

```
type = dielectric
name = ILD0
height = 1.1
k = 4.2
# Since planarity is not specified, planarity = p is assumed
```

Informative
Sample SIPPs Data

```
type = metal
name = metal1
gdsII_layer = 7
thickness = $T1 #Substitution parameter $T1 defined in
                # parameters section
r_sheet = 0.07, 150
# In the above statement, the r_sheet value (0.07) is Fixed

min_width = $m5_min_width
min_spacing = $m5_min_spacing

type = via
name = via_m1_m2
gdsII_layer = 8
height = 0.70
r_per_via = F (2, 150, 0.25, 0.25)

type = dielectric
name = IMD1
thickness = 1.4
k = 3.6
planarity = p

type = metal
name = metal2
gdsII_layer = 9
thickness = $T1
r_sheet = F (0.07), 150
min_width = $m5_min_width
min_spacing = $m5_min_spacing

type = via
name = via_m2_m3
gdsII_layer = 10
height = 0.70
r_per_via = F (2, 150, 0.25, 0.25)

type = dielectric
name = IMD2
thickness = 1.4
k = 3.6
planarity = p

type = metal
name = metal3
```

Informative
Sample SIPPs Data

```
gdsII_layer = 11
thickness = $T1
r_sheet = F (0.07), 150
min_width = $m5_min_width
min_spacing = $m5_min_spacing

type = via
name = via_m3_m4
gdsII_layer = 12
height = 0.70
r_per_via = F (2, 150, 0.25, 0.25)

type = dielectric
name = IMD3
thickness = 1.4
k = 3.6
planarity = p

type = metal
name = metal4
gdsII_layer = 13
thickness = $T1
r_sheet = F (0.07), 150
min_width = $m5_min_width
min_spacing = $m5_min_spacing

type = via
name = via_m4_m5
gdsII_layer = 14
height = 0.70
r_per_via = F (2, 150, 0.25, 0.25)

type = dielectric
name = IMD4
thickness = 1.4
k = 3.6
planarity = p

type = metal
name = metal5
gdsII_layer = 15
thickness = 1.0
r_sheet = F (0.05), 150

type = dielectric
```

Informative
Sample SIPPs Data

```
name = PASS1
thickness = 1.0
k = 4.0
planarity = p

type = dielectric
name = PASS2
thickness = 0.5
k = 7.0
planarity = p

type = dielectric
name = DIE_COATING
thickness = 10
k = 3.0
planarity = p

end

# END SIPPS
```

8.0 Follow-On

8.1 Substrate Modeling

The second release of the SIPPs Specification will provide for more detailed modeling within the substrate area on the chip. This is an important consideration for analysis of noise into the substrate.

8.2 Anisotropic Dielectrics

Anisotropic dielectrics have different values of permittivity in the xy-direction than in the z-direction. This condition will be able to be represented in a future version of the specification.