

=====
I B I S I N T E R C O N N E C T S P E C I F I C A T I O N
=====

Last Updated: 07-09-2002

- Editorial Updates per July 08 email

=====
Specification Revision level 0.99900
=====

Table of Contents
=====

I B I S I N T E R C O N N E C T S P E C I F I C A T I O N	1
Table of Contents	1
REVISION HISTORY	2
1.0 GENERAL INTRODUCTION	2
2.0 STATEMENT OF INTENT	2
3.0 GENERAL SYNTAX RULES AND GUIDELINES	3
4.0 KEYWORD TREE DIAGRAM	4
5.0 HEADER KEYWORDS	5
Keyword: [Begin Header]	5
Keyword: [IBIS ICM Model Ver]	6
Keyword: [File Name]	6
Keyword: [File Rev]	6
Keyword: [Date]	6
Keyword: [Source]	7
Keyword: [Notes]	7
Keyword: [Disclaimer]	7
Keyword: [Copyright]	8
Keyword: [Support]	8
Keyword: [Redistribution]	9
Keyword: [Redistribution Text]	9
Keyword: [End Header]	9
6.0 GLOBAL KEYWORDS	10
Keyword: [Comment Char]	10
7.0 INTERCONNECT MODEL FAMILY KEYWORDS	10
Keyword: [Begin ICM Model Family]	10
Keyword: [Manufacturer]	10
Keyword: [ICM Family Description]	11
Keyword: [ICM Model List]	11
Keyword: [Begin ICM Model]	13
Keyword: [ICM Model Description]	15
Keyword: [Tree Path Description]	15
Keyword: [Nodal Path Description]	20
Keyword: [ICM Swath Parameters]	26
Keyword: [ICM Swath Pin Numbers]	30
Keyword: [End ICM Model]	31
Keyword: [ICM Pin Map]	32
Keyword: [ICM Node Map]	34
Keyword: [End ICM Model Family]	38
8.0 MATRIX KEYWORDS	38
Keyword: [Begin ICM Section]	38
Keyword: [Derivation Method]	39
Keyword: [Resistance Matrix], [Inductance Matrix], [Conductance Matrix], [Capacitance Matrix]	39
Matrix Descriptions	41
Resistance Matrix Description:	42
Inductance Matrix Description:	42
Loop-Inductance Matrix Description (SLMs ONLY):	42
Partial Self Inductance Matrix Description (MLMs ONLY):	44
Capacitance Matrix Description	44
Conductance Matrix Description:	45

Loss Matrix Description:	46
Keyword: [Bandwidth]	46
Keyword: [Row]	46
Keyword: [ICM Sparameter]	47
Keyword: [End ICM Section]	47
Matrix Examples:	47
9.0 End Keyword	49
Keyword: [End]	49

=====
REVISION HISTORY

xx-xx-200x, v1.0 Specification
=====

=====
1.0 GENERAL INTRODUCTION

=====
This document contains a general purpose specification for interconnect modeling in an IBIS (I/O Buffer Information Specification) compatible format. It was written to provide means for modeling all electrical interconnect types, including connectors, cables, packages, and printed circuit boards.

This section gives a general overview of the remainder of this document. Sections 2 to 5 contain general information about the IBIS versions and the general rules and guidelines. The remaining sections describe the interconnect keywords and usage rules.

=====
2.0 STATEMENT OF INTENT

=====
In order to enable an industry standard method to electronically transport IBIS interconnect modeling data between interconnect design and simulation tools. The intention of this specification is to specify a consistent format that can be parsed by software, allowing simulation vendors to derive models compatible with their own products.

One goal of this template is to represent the current state of IBIS data, while allowing a growth path to more complex models / methods (when deemed appropriate). This would be accomplished by a revision of the base template, and possibly the addition of new keywords or categories.

Another goal of this template is to ensure that model creation is simple enough for interconnect vendors and customers to use, while ensuring that it is rigid enough for simulation vendors to write reliable parsers.

This template is meant to contain a complete description of the elements of an entire interconnect.

This specification is intended to provide a basis for future versions which are expected to be backward compatible with this initial version.

=====

3.0 GENERAL SYNTAX RULES AND GUIDELINES

=====

1) Only ASCII characters, as defined in ANSI Standard X3.4-1986, may be used in an IBIS-Interconnect file. The use of characters with codes greater than hexadecimal 07E is not allowed. Also, ASCII control characters (those numerically less than hexadecimal 20) are not allowed, except for TAB characters and line termination sequences.

NOTE: the use of TAB characters is strongly discouraged.

2) Except for keywords (words enclosed by a square bracket []), the content of an IBIS Interconnect Model (ICM) file is case sensitive.

3) Keywords must be enclosed in square brackets, [], and must start in column 1 of the line. No space is allowed immediately after the opening bracket "[" or immediately before the closing bracket "]". If used, only one space (" ") or underscore ("_") character separates the parts of a multi-word keyword. Spaces and underscores are equivalent within square bracket. Some keywords may be followed by an argument. Keyword arguments are of three types: either a text string, a file name, or a text block. Text string and file name arguments start on the same line as the keyword and are terminated with a line termination sequence. A text block starts on the same line as the keyword, may extend over multiple lines, and is terminated by the occurrence of the next keyword.

4) Keywords and subparameters must begin with a letter, and shall consist only of alphanumeric characters and the underscore "_". Spaces are not allowed in subparameter names.

5) Each IBIS Interconnect data file is divided into sections. Each section is delimited by keywords of the form [Begin "section name"] and [End "section name"]. Within each section only specific keywords are legal (note, however, a keyword may be legal in more than one section). Unless otherwise noted, a section's keywords can appear in any order.

6) To facilitate portability between operating systems, file names used in the IBIS Interconnect file must only have lower case characters. File names should have a basename followed by a period ("."), followed by a file name extension of no more than three characters. There is no length restriction on the basename. The basename and extension must use characters from the following set (space, " ", 0x20 is not included):

```
a b c d e f g h i j k l m n o p q r s t u v w x y z
0 1 2 3 4 5 6 7 8 9 _ -
```

7) A line of the file may have at most 120 characters, followed by a line termination sequence. The line termination sequence must be one of the following two sequences: a linefeed character, or a carriage return followed by a linefeed character.

8) Anything following the comment character is ignored and considered a comment on that line. The default "|" (pipe) character can be changed by the keyword [Comment Char] to any other character. The [Comment Char] keyword can be used throughout the file as desired.

9) Valid scaling factors are:

T = tera	k = kilo	n = nano
G = giga	m = milli	p = pico
M = mega	u = micro	f = femto

When no scaling factors are specified, the appropriate base units are assumed. (These are volts, amperes, ohms, farads, henries, and seconds). The parser looks at only one alphabetic character after a numerical entry; therefore it is enough to use only the prefixes to scale the parameters. However, for clarity, it is allowed to use full abbreviations for the units, (e.g., pF, nH, mA, mOhm). In addition, scientific notation IS allowed (e.g., 1.2345e-12).

10) All temperatures are represented in degrees Celsius.

11) All lines that occur before the [Begin Header] keyword or after the [End] keyword shall be treated as comments and ignored by the parser. The intent of this rule is to facilitate the inclusion of HTML tags, revision control headers, etc. at the beginning of the file.

12) The following words are reserved words and must not be used for any other purposes in the document:

- POWER - reserved model name, used with power supply pins,
- GND - reserved model name, used with ground pins,
- NC - reserved model name, used with no-connect pins,
- NA - used where data not available.

=====

4.0 KEYWORD TREE DIAGRAM

=====

```
/-- Start of File
|
|-----
|
|--/--[Begin Header]
| |
| | |--[IBIS ICM Model Ver]
| | |--[Comment Char]
| | |--[File Name]
| | |--[File Rev]
| | |--[Date]
| | |--[Source]
| | |--[Notes]
| | |--[Disclaimer]
| | |--[Copyright]
| | |--[Support]
| | |--[Redistribution]
| | |--[Redistribution Text]
| | \--[End Header]
|
|--/--[Begin ICM Model Family] | (Chapter 7)
| |
| | |--[Manufacturer]
| | |--[ICM Family Description]
| | |--[ICM Model List]
|
| |--/--[Begin ICM Model]
| |
| | ICM_Model_Type
| | SGR
| | Ref_impedance
| | |--[ICM Model Description]
| | |--[Tree Path Description]
| | Model_PinMap
| | Fork, EndFork
| | Section
```


=====
Keyword: [IBIS ICM Model Ver]
Required: Yes
Argument: Text String
Used By: [Begin Header]
Description: This keyword allows electronic parsers to immediately determine that this file contains an IBIS interconnect model. The version number is used to inform the parser what keywords are valid for this model and to allow backward support as new keywords are added.
Usage Rules: This keyword must appear directly after the [Begin Header] keyword.

[IBIS ICM Model Ver] 1.0

=====
Keyword: [File Name]
Required: Yes
Argument: Text String (file name)
Used by: [Begin Header]
Description: Specifies the name of the IBIS Interconnect Model (ICM) file
Usage Rules: The purpose of this keyword is to document a single filename the file uses even if the file is transferred from one computer to another. The file name must conform to the rules given in section 4, "General Syntax Rules and Guidelines".

The [File Name] keyword may appear only once between the [Begin Header]/[End Header] keyword pair.

[File Name] iconm_hdi_202.icm

=====
Keyword: [File Rev]
Required: Yes
Argument: Text String
Used By: [Begin Header]
Description: Tracks the revision level of a particular ICM file
Usage Rules: Revision level is set at the discretion of the engineer defining the file. The following guidelines are suggested:

0.x interconnect and/or file in development
1.x pre-manufacturing interconnect and/or model
2.x mature interconnect and/or model, no more changes likely

The [File Rev] keyword may appear only once between the [Begin Header]/[End Header] keyword pair.

[File Rev] 1.0 | Used for ICM file versions

=====
Keyword: [Date]
Required: No
Argument: Text String (Date)
Used By: [Begin Header]
Description: Date this file was last modified
Usage Rules: This keyword is provided to insure the last changed date for

this file is not lost if the file is transmitted between computer systems. The format is fixed. The pattern to be used is "Month Day, Year". Month is fully written out. Day uses two digits. Year uses four digits. See example.

The [Date] keyword may appear only once between the [Begin Header]/[End Header] keyword pair.

[Date] July 01, 2000 | The latest file revision date

=====

Keyword: [Source]
Required: No
Argument: Text Block
Used by: [Begin Header]
Description: Records the originating source of model data
Usage Rules: Use this keyword to record how the model information was obtained (physical measurement of device, simulations, data book, etc.). This field contain the name of the company or entity that created the model.

NOTE: It is recommended that the argument to the [Source] keyword be limited to a maximum of 24 lines of text.

The [Source] keyword may appear only once between the [Begin Header]/[End Header] keyword pair.

[Source] Put originator and the source of information here. For example:
Results from field simulation

=====

Keyword: [Notes]
Required: No
Argument: Text Block
Used By: [Begin Header]
Description: Optional notes regarding the file
Usage Rules: The keyword provides a place for the model maker to record important notes about the file or model data that are not included elsewhere. Such information may include notes on validation level, model limits, usage assumptions, etc.

NOTE: It is recommended that the argument to the [Notes] keyword be limited to a maximum of 24 lines of text.

The [Notes] keyword can only be used once.

The [Notes] keyword may appear only once between the [Begin Header]/[End Header] keyword pair.

[Notes] Use this section for any special notes related to the file.

=====

Keyword: [Disclaimer]
Required: No
Argument: Text Block
Used By: [Begin Header]

Description: Legal disclaimer and copyright information
Usage Rules: This keyword provides a place for the user to add a legal disclaimer.

NOTE: It is recommended that the argument to the [Disclaimer] keyword be limited to a maximum of 96 lines of text.

The [Disclaimer] keyword may appear only once between the [Begin Header]/[End Header] keyword pair.

[Disclaimer] This information is for modeling purposes only, and is not guaranteed.

=====
Keyword: [Copyright]
Required: No
Argument: Text Block
Used By: [Begin Header]
Description: Legal copyright information
Usage Rules: Because model writers may consider the information in these keywords essential to users, and sometimes legally required, design automation tools should make this information available. Any text following the [Copyright] keyword must be included in any derivative models verbatim.

NOTE: It is recommended that the arguments to the [Copyright] keywords be limited to a maximum of 96 lines of text.

The [Copyright] keyword may appear only once between the [Begin Header]/[End Header] keyword pair.

[Copyright] Copyright 2000, XYZ Corp., All Rights Reserved

=====
Keyword: [Support]
Required: No
Argument: Text Block (URL path name)
Used By: [Begin Header]
Description: Specifies a web site that can be visited to get the latest version of the file
Usage Rules: Following the [Support] keyword is the URL of a web site the user may visit for more information on the model or model(s). The entire link, including any file extension, is required. The [Support] keyword may appear only once between the [Begin Header]/[End Header] keyword pair.

NOTE: It is recommended that a separate IBIS model web site (not page) be maintained to prevent accidental changes of page name from breaking this link.

The [Support] keyword may appear only once between the [Begin Header]/[End Header] keyword pair.

[Support] <http://www.VendorNameIbisModels.com> |example ibis web site root


```
=====
Keyword: [Redistribution]
Required: Yes
Argument: Text String (Yes, No, Specific)
Used By: [Begin Header]
Description: Indicates to EDA tool companies and model users who may use and
redistribute this file
Usage Rules: Following the [Redistribution] keyword is one of three
arguments: "Yes", "No" and "Specific".

An argument value of "Yes" means that a EDA tool vendor or end
user may freely distribute the model as long as no fee is
charged. A fee may be charged if authorized by the model
creator.

An argument value of "No" means that the model may not be
redistributed or retransmitted in any form.

An argument value of "Specific" means that specific license
information is contained in the [Redistribution Text] field.

Use of the "Specific" or "No" argument prevents automated
redistribution.

The [Redistribution] keyword may appear only once between the
[Begin Header]/[End Header] keyword pair.
```

```
-----
[Redistribution] Yes
```

```
=====
Keyword: [Redistribution Text]
Required: Yes, ONLY if [Redistribution] value is "Specific", otherwise it
is optional
Argument: Text block
Used By: [Begin Header]
Description: Allows model creator to supply specific redistribution
information
Usage Rules: If the argument to the [Redistribution] keyword is "Specific",
then the user must include additional information on licensing
details, or where to find them. Any text following the
[Redistribution Text] keyword must be included in any derivative
models verbatim.

Note: It is recommended that the argument to the [Redistribution
Text] keyword be limited to a maximum of 24 lines of text.

The [Redistribution Text] keyword may appear only once between
the [Begin Header]/[End Header] keyword pair.
```

```
-----
[Redistribution Text] Your text goes here.
```

```
=====
Keyword: [End Header]
Required: Yes
Used By: [Begin Header]
Description: Marks the end of an IBIS Interconnect header section
```

Usage Rules: [End Header] must be the last keyword in any IBIS Interconnect header section.

[End Header]

=====
6.0 GLOBAL KEYWORDS
=====

Keyword: [Comment Char]
Required: No
Argument: Text String
Description: Defines a new comment character to replace the default
"|" (pipe) character, if desired
Usage Rules: The new comment character to be defined must be followed by
the underscore character and the letters "char". For example:
"|_char" redundantly redefines the comment character to be
the pipe character. The new comment character is in effect
only following the [Comment Char] keyword. The following
characters MAY be used:

! " # \$ % & ' () * , : ; < > ? @ \ ^ ` { | } ~

The [Comment Char] keyword can only appear after the
[IBIS ICM Model Ver] keyword. A file may contain multiple
[Comment Char] keywords.

[Comment Char] |_char

=====
7.0 INTERCONNECT MODEL FAMILY KEYWORDS
=====

Keyword: [Begin ICM Model Family]
Required: Yes
Argument: Text String
Uses: [Manufacturer], [ICM Family Description], [ICM Model List],
[Begin ICM Model], [End ICM Model Family]
Description: This keyword marks the beginning of the interconnect family
description section. An interconnect "family" consists of one
or more similar interconnects. Note that each individual
interconnect in a family will have a separate interconnect
model.
Usage Rules: Following the keyword is the name of the interconnect family.
Only one [Begin ICM Model Family]/[End ICM Model Family]
keyword is allowed per file.

Interconnect family descriptions are terminated by the
occurrence of the [End ICM Model Family] keyword.

[Begin ICM Model Family] High_Speed_Interconnect

=====
Keyword: [Manufacturer]

Required: Yes
Argument: Text String
Used By: [Begin ICM Model Family]
Description: This keyword gives the name of the manufacturer of the physical connectors or interconnect being described in this file.
Usage Rules: Following the keyword is the interconnect manufacturer's name. Blank characters are allowed.

The [Manufacturer] keyword may appear only once between the [Begin ICM Model Family]/[End ICM Model Family] keyword pair.

[Manufacturer] XYZ Incorporated

=====
Keyword: [ICM Family Description]

Required: Yes
Argument: Text Block
Used By: [Begin ICM Model Family]
Description: Provides a human-readable description of this interconnect family
Usage Rules: Only a single occurrence of [ICM Family Description] is allowed in the file, and this keyword must be placed between the [Begin ICM Model Family] and [End ICM Model Family] keywords.

Note: It is recommended that the family description be limited to a maximum of 4 lines.

[ICM Family Description]
High Density 0.1 center square pin connector designed for use on IEEE 99999 buses.

=====
Keyword: [ICM Model List]

Required: Yes
Used By: [Begin ICM Model Family]
Description: This keyword lists the model name, mating conditions, minimum slew time and picture link for each individual interconnect model described in this file.
Usage Rules: The [ICM Model List] keyword may appear only once between the [Begin ICM Model Family]/[End ICM Model Family] keyword pair."

Following the keyword are four columns of data, as shown in the example below (the column headings are shown as comments and are not required). The "Name", "Mating" and "Min_Slew_Time" columns are required, and all columns must be in the left to right order shown.

The "name" column lists the model name for each interconnect model contained in the file. The model name must match one of the model names given as an argument to the [Begin ICM Model] keyword(s).

The "mating" column specifies the mating condition under which each model's data is valid. Valid mating choices are "Mated", "Unmated_Side_A" and "Unmated_Side_B". These choices are defined as follows:

Mated: Both halves of a two piece interconnect are mated together for the model.

UnMated_Side_A: A model of non-mated interconnect. Typically, this would be the connector half that is attached to a baseboard, or the socket which holds the socket of an IC.

UnMated_Side_B: A model of non-mated interconnect. Typically, this would be the connector half that is attached to a daughter card, cable, edgcard, or package of an IC.

Note that using two unmated models in series DOES NOT provide a correct model of a mated interconnect. Also, if the interconnect model describes a package, one should always make it "mated" since unmated packages don't have any practical use.

The Min_Slew_Time column specifies the minimum slew time (i.e. fastest edge rate) for which each model is valid. Slew time is defined as the time it takes a signal to transition from 20% to 80% of its final value. This term does not indicate the bandwidth of the actual interconnect. Rather, this term is used to define a usage limit of each model.

Note: It is strongly recommended that an EDA tool issue a warning if a signal with a lesser slew time (i.e. faster edge rate) is applied to the model during a simulation.

The optional image column specifies an associated file that contains an image of each interconnect. The image file shall be either a .jpg or .txt (ASCII) formatted picture. The interconnect model MUST be considered valid by EDA tools even if an image file is specified but is missing.

If present, images should be maintained in the same directory as the interconnect model file or stored in the same compressed file with the interconnect to ensure that the files remain paired.

The same image file may be repeated for all model types if desired.

Note: it is suggested that EDA tools provide a viewer for the .jpg format and that model builders provide at least one picture for an interconnect.

Suggested .jpg image attributes:

```
size = 160 horizontal by 120 vertical (pixels)
color = 24 bit
Image quality = 85%
```

Larger or smaller images may be used but are not recommended. A standard image size will allow EDA tools to provide uniform look and feel for images from all vendors. The size was chosen to be a small disk file size and be useful as either a thumb-nail in a model selection window or scaled up to give a larger picture. The 160x120 was chosen as a perfect 4:1 scale in both axis of a 640x480 picture.

```
-----
| [Begin ICM Model List]
|   Name           Mating       Min_Slew_Time  Image
|-----
```

HDI_TEST_202	Mated	100ps	HDI_TEST_202_Mated.jpg
HDI_TEST_202_UnMatedA	UnMated_Side_A	100ps	HDI_TEST_202_UnMatedA.jpg
HDI_TEST_202_UnMatedB	UnMated_Side_B	100ps	
HDI_TEST_202_ThruHole_to_Cable	Mated	25ps	HDI_TEST_202_Mated.jpg
HDI_TEST_202_SMT_to_Cable	Mated	25ps	HDI_TEST_202_Mated.jpg
HDI_TEST_202_SMT_to_ThruHole	Mated	25ps	HDI_TEST_202_Mated.jpg

=====

Keyword: [Begin ICM Model]

Required: Yes

Argument: Text String (Model name)

Used By: [Begin ICM Model Family]

Uses: [ICM Model Description], [Tree Path Description], [Nodal Path Description], [ICM Swath Parameters], [End ICM Model]

Description: This keyword marks the beginning of each individual interconnect model.

Sub-params: ICM_Model_Type, SGR, Ref_impedance

Usage Rules: Following the keyword is a single argument specifying the name of this particular interconnect model. The name must be one of the model names listed under the "model name" column of the preceding [ICM Model List] keyword. The [Begin ICM Model] keyword must appear after the [ICM Model List] keyword. Note that for every model name listed by the [ICM Model List] keyword there must be a corresponding model as defined by a [Begin ICM Model]/[End ICM Model] keyword pair.

The [Begin ICM Model Family] keyword must appear after the [End Header] keyword.

Subparameters:

ICM_Model_Type

Following the keyword line is a required subparameter ICM_Model_Type. This subparameter defines the type of the data (RLGC or S-parameter), and the conditions under which the RLGC data was gathered.

The legal values for this subparameter are "SLM_General", "SLM_Quiescent", "SLM_EvenMode", "SLM_OddMode", "MLM", and "S-parameter".

The subparameter and its value are separated by a single whitespace. Only one ICM_Model_Type is allowed per interconnect model.

An ICM_Model_Type subparameter value of "SLM_Quiescent" indicates that the model data was gathered with all other neighboring conductors non-switching and terminated by a reference impedance as given by the optional Ref_Impedance subparameter.

A value of "SLM_EvenMode" indicates that all other neighboring conductors are transitioning at the same time and in the same direction as the conductor being modeled (i.e. under even mode switching conditions). A value of "SLM_OddMode" indicates that all other neighboring conductors are transitioning at the same time but in the opposite direction as the conductor being modeled (i.e. under odd mode switching conditions).

A value of "SLM_General" indicates that the model data was

gathered with all other neighboring conductors non-switching and the ground to signal ratio is specified by SGR and the return path is unknown.

Note that the use of an SLM_* value indicates that this is a "single line model" (SLM) and no electrical data is supplied to model the coupling between pins. Specifically, the corresponding RLGC matrix must contain only a single diagonal term. SLMs can be used to evaluate for Prop Delay, bandwidth, and impedance where ideal ground can be assumed. SLMs should not be used when lossy return paths contributions are required for simulations.

"SLM_Quiescent", "SLM_EvenMode" and "SLM_OddMode" have a defined reference or return path. SLM_General does not have specified Return Path given in the matrix.

An ICM_Model_Type value of "MLM" indicates that this model is a "multi line model" and the electrical data suitable for modeling pin to pin coupling is supplied. Any combination of Diagonal_matrix, Sparse_matrix, Banded_matrix or Full_matrix may be used in an MLM model. In general, MLMs should not be used in simulations with perfect ground planes on both sides of the interconnect model.

An ICM_Model_Type value of "S-parameter" indicates that this model is a "multi line model" and the electrical data suitable for modeling pin to pin coupling is supplied in a frequency domain format. Specifically, the model is a secondary file in "Touchstone" format.

Example: ICM_Model_type SLM_Quiescent

SGR

If the ICM_Model_Type subparameter is equal to SLM_General, then this SGR subparameter is required. This subparameter specifies the signal to ground ratio used when generating the SLM_General model. The signal to ground ratio is indicated as "number of signal pins per ground pin", followed by a colon (:), followed by the number one (1). For example, if the data for an 18 pin connector was taken with 6 of the pins grounded, the signal to ground ratio is 18/6 or 3 to 1. The SGR subparameter argument would then be listed as 3:1.

This subparameter is only valid when the ICM_Model_Type subparameter has a value of "SLM_General".

If this subparameter is present and the ICM_Model_Type subparameter is NOT "SLM_General" a warning should be issued.

Ref_impedance

An optional subparameter that can be used to specify the impedance of the system used to extract SLM values. If not given the Ref_Impedance value is assumed to be 50 ohms. The reference impedance for an "S-parameter" model is located in the model file.

=====

Keyword: [ICM Model Description]
Required: No
Argument: Text Block
Used By: [Begin ICM Model]
Description: Provides a concise yet easily human-readable description of this interconnect model
Usage Rules: This keyword may appear multiple times in the file however it may appear only once between each pair of [Begin_ICM_Model]/[End_ICM_Model] keywords.

[Begin ICM Model Description]
High Density 0.1 center square pin with PCB effects

=====
Keyword: [Tree Path Description]
Required: Yes, if [Nodal Path Description] does not exist
Sub-params: Model_PinMap, Section, Fork, EndFork
Used By: [Begin ICM Model]
Uses: [ICM Pin Map]
Description: This keyword describes the connection path between the pin or pins on one side of the interconnect and the corresponding set of pins on the other side. This pin to pin interconnect is divided into one or more "sections", where each section consists of RLGC and/or S-parameter data. Sections are generally cascaded in series, but the Fork and EndFork subparameters allow sections to branch off the main interconnect path as a stub or tee.
A [Tree Path Description] is used when all of the sections in a given interconnect have one to one mapping. These straight through connections are relative to the matrices. This description is useful when all of the matrices used in each section can be connected in a conductor by conductor fashion. In all cases, before swathing (if used), all the Resistance, Inductance, Conductance, and Capacitance matrices must all have the same number of diagonal terms. In all cases, after swathing (if used), all the Resistance, Inductance, Conductance, and Capacitance matrices must all have the same number of diagonal terms. Further, all diagonal terms must be in the same order.
Usage Rules: There must be one and only one path description keyword between a [Begin ICM Model] and [End ICM Model] keyword pair. The [Nodal Path Description] and [Tree Path Description] keywords are mutually exclusive and only one of them can be used between a [Begin ICM Model] and [End ICM Model] keyword pair. However, these keywords may appear more than once and may be used together within one file.

A path description begins with the subparameter Model_PinMap, which points to the [ICM Pin Map] used for one end of the interconnect. This is followed by one or more Section subparameters. The electrical properties of an interconnect are contained in the [ICM Section] matrix(s) pointed to by the Section subparameter. Branches or "T" connections are indicated by the "Fork" and "EndFork" subparameters. Finally, a path description is terminated by a Model_PinMap subparameter.

These subparameters are described in detail below.

Model_PinMap

A path description begins with the required subparameter Model_PinMap. This subparameter is followed by the name of a "pin map" which names the pins on that end of the interconnect. A second Model_PinMap is used at the end of the path description to reference the pin map used for the other end of the interconnect. The Model_PinMap subparameter is also used to reference a pin map for the pins at the end of any Fork. Note that the pin map name given as an argument to a Model_PinMap subparameter must match the name of a pin map defined by a subsequent [ICM Pin Map] keyword.

The argument to the Model_PinMap subparameter and the subparameter itself are separated by whitespace. There can be as many Model_PinMap subparameters as required to represent an interconnect, but each must be on a single line.

In general, an interconnect will have a single pin map that is referenced at the beginning and end of a path description. However, two different pin maps will be used if the interconnect contains cross over pins.

Section

Following the opening Model_PinMap subparameter is one or more instances of the required subparameter Section. This subparameter defines which [ICM Section] matrix is part of this interconnect model. Matrix sections are serially interconnected and may be re-used to form a complete interconnect model.

Arguments "multiplier" and "section name" must follow the Section parameter.

The "multiplier" argument contains a single numeric value which allows a matrix section to be re-used as a scaled value. This argument can either be a multiplier factor determining how many times the section should be repeated in series (cascaded), or defines what its length is. The value of the second argument may be any positive value greater than zero. If a section is intended to be repeated (cascaded) a certain amount of times, the syntax of this argument is:

Mult = value

where "value" must be an integer number.

If this argument contains the length of a section, its syntax is:

Len = value

where "value" is a floating point decimal real number. (Note that spaces around the "=" sign are not required).

Some sections within a particular path description can use Mult while others use Len, if so desired. If "Mult=" is used the [Derivation Method] keyword for the corresponding [ICM Section] must have a value of "Lumped" else it is an error. If "Len=" is used the [Derivation Method] keyword for the corresponding [ICM Section] must have a value of "Distributed" else it is an error.

The scale factor "Len" is applied to each parameter of all matrices of a section resulting in scaled RLGC matrices. When S-parameters are used to describe a section, the usage of scale factor "Len" is not allowed.

Mult and Len are separate arguments. Mult and Len are not allowed to be used in the same section.

The "section name" is the name of the section as specified under the [Begin ICM Section] keyword.

Section may be used as many times as required to describe the interconnect. Each scale factor and section name MUST be separated by at least one space.

Fork, EndFork

The Fork and EndFork subparameters are used if there are forks or "stubs" in the interconnect being modeled. Fork marks the beginning of the stub while EndFork marks the end of the stub. These subparameters do not have any arguments.

At a fork, a "tee" section is created. Matrix sections are connected to the previous section and interconnected to zero or more sections between Fork and EndFork. If no pin map name is placed between an Fork and EndFork block, then the fork constitutes a stub. However, if a pin map name is placed at the end of the Fork and EndFork block, an externally available port is created on the interconnect.

Any number of Fork subparameters may be used per path description, however only one subparameter is allowed per line.

Example: Fork
 EndFork

"ICM Model" Examples

Example 1: (A simple single line model with one section).

```
[Begin ICM Model] MyModelExample1
  ICM_Model_Type SLM_General
  SGR 3:1
  Ref_Impedance 50
```

```
[Tree Path Description]
  Model_PinMap Example1_pinmap
  Section Mult=1 Diagonal_matrix1
  Model_PinMap Example1_pinmap
  .
  .
  .
```

```
[End ICM Model]
```

Example 2: (2 ports per pin, 5 sections straight through).

```
          SectA      SectB      SectC      SectD      SectE
Port1 >-----< >-----< >-----< >-----< >-----< Port2
[Begin ICM Model] MyModelExample2
```

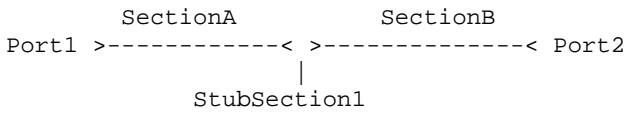
```

ICM_Model_Type SLM_Quiescent

[Tree Path Description]
Model_PinMap MyModelPinMapA
Section Mult=1 SectA
Section Mult=1 SectB
Section Mult=1 SectC
Section Mult=1 SectD
Section Mult=1 SectE
Model_PinMap MyModelPinMapB
.
.
.
[End ICM Model]

```

Example 3: (2 ports per pin, 2 series sections with one stub).

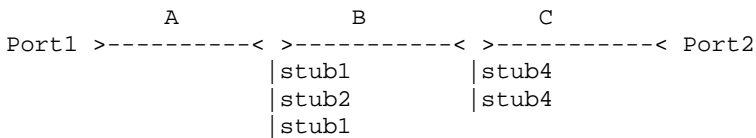


```

[Begin ICM Model] MyModelExample3
ICM_Model_Type MLM
[Tree Path Description]
Model_PinMap MyModelPinMapA
Section Mult=1 SectionA
Fork
  Section Mult=1 StubSection1
EndFork
Section Mult=1 SectionB
Model_PinMap MyModelPinMapB
.
.
.
[End ICM Model]

```

Example 4: (2 ports per pin, 2 stub sections, one stub matrix is re-used,
and another stub is repeated).



```

[Begin ICM Model] MyModelExample4
ICM_Model_Type MLM
[Tree Path Description]
Model_PinMap MyModelPinMapA
Section Mult=1 A
Fork
  Section Mult=1 stub1
  Section Mult=1 stub2
  Section Mult=1 stub1
EndFork
Section Mult=1 B
Fork
  Section Mult=2 stub4
EndFork

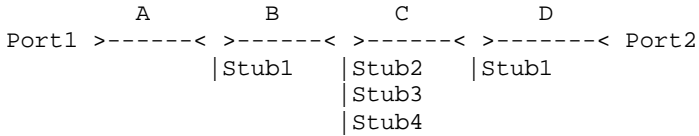
```

```

Section Mult=1 C
Model_PinMap MyModelPinMapB
.
.
.
[End ICM Model]

```

Example 5: (2 ports per pin, 3 stubs).

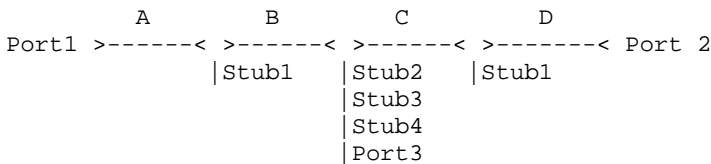


```

[Begin ICM Model] MyModel4
  ICM_Model_Type MLM
[Tree Path Description]
  Model_PinMap MyModelPinMapA
  Section Mult=1 A
  Fork
    Section Mult=1 Stub1
  EndFork
  Section Mult=1 B
  Fork
    Section Mult=1 Stub2
    Section Mult=1 Stub3
    Section Mult=1 Stub4
  EndFork
  Section Mult=1 C
  Fork
    Section Mult=1 Stub1
  EndFork
  Section Mult=1 D
  Model_PinMap MyModelPinMapB
  .
  .
  .
[End ICM Model]

```

Example 6: (3 ports per pin, 2 stubs).



```

[Begin ICM Model] MyModelExample6
  ICM_Model_Type MLM
[Tree Path Description]
  Model_PinMap MyModelPinMapA
  Section Mult=1 A
  Fork
    Section Mult=1 Stub1
  EndFork
  Section Mult=1 B
  Fork
    Section Mult=1 Stub2
    Section Mult=1 Stub3

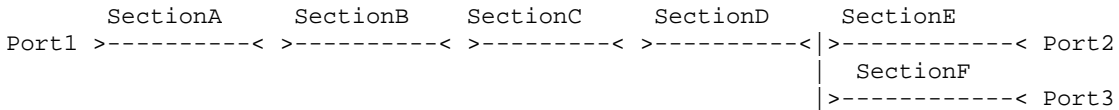
```

```

    Section Mult=1 Stub4
    Model_PinMap MyModelPinMapC
EndFork
Section Mult=1 C
Fork
    Section Mult=1 Stub1
EndFork
Section Mult=1 D
Model_PinMap MyModelPinMapB
.
.
.
[End ICM Model]

```

Example 7: (3 ports per pin, 5 sections straight through).



```

[Begin ICM Model] MyModelExample7
  ICM_Model_Type SLM_Quiescent
[Tree Path Description]
  Model_PinMap = MyModelPinMapA
  Section Mult=1 SectionA
  Section Mult=1 SectionB
  Section Mult=1 SectionC
  Section Mult=1 SectionD
  Fork
    Section Mult=1 SectionF
    Model_PinMap MyModelPinMapB
  EndFork
  Section Mult=1 SectionE
  Model_PinMap MyModelPinMapC
.
.
.
[End ICM Model]

```

=====
Keyword: [Nodal Path Description]
Required: Yes, if [Tree Path Description] does not exist
Sub-params: N_section, Model_NodeMap
Used By: [Begin ICM Model]
Uses: [ICM Node Map]
Description: This keyword marks the beginning of a nodal path description. It contains the information on how two or more nodes are connected. The nodal description style allows for any arbitrary topologies to be described.

Nodes define the connection points where the conductors of two or more sections are joined together, or where the path is interfaced to the outside world. Please note that single connections (dangling ends) at nodes are also permitted. Thus nodes can represent any connection points, including pins (on either side of connectors, packages, or sockets), pads and bumps of a die (where the connections to the package or printed circuit board are made), or the ends of either side of interconnect sections of printed circuit boards.

An interconnect path may have one or more sections where each section is described by RLGC, or S-parameter matrices. Multiple sections within a path description may be connected in any arbitrary manner by using the usual node naming techniques. Please note that the swath features cannot be used with this keyword.

Usage Rules: There must be one and only one path description keyword between a [Begin ICM Model] and [End ICM Model] keyword pair. The [Nodal Path Description] and [Tree Path Description] keywords are mutually exclusive and only one of them can be used between a [Begin ICM Model] and [End ICM Model] keyword pair. However, these keywords may appear more than once and may be used together within one file.

Following the [Nodal Path Description] keyword line are the subparameters that describe the topology and electrical characteristics of the pin to pin connections through the interconnect.

Path description sections are terminated by the occurrence of the next keyword.

The [Nodal Path Description] keyword line is followed by at least one Model_NodeMap and one N_section subparameters which describe the pin to node mapping, and the topology of the interconnect. The order in which the Model_NodeMap and N_section subparameters are listed in a path description is irrelevant. However, for visual clarity it is a good practice to follow the physical layout of the interconnect with the order in which they are listed. The electrical properties of an interconnect section are described in the matrices under the [Begin ICM Section] keyword which are referenced by the N_section subparameter(s), discussed below.

Connections between the nodes of the various N_sections, branching or T-ing are achieved using the usual node naming techniques.

Detailed descriptions of these subparameters follow below.

N_section

The N_section subparameter is used to describe the connectivity and electrical characteristics of an interconnect segment in the path description.

A path description must have at least one occurrence of the required subparameter N_section, but can have as many N_sections as needed to adequately describe an interconnect. Each N_section subparameter must start on a separate line and must end on the line on which the closing parentheses are found. (This allows long node lists to span over multiple lines without causing parsing problems).

The subparameter name N_section is followed by three arguments, separated by at least one white space.

The first argument is a list of nodes and must be enclosed in parentheses. The length of each node name cannot exceed 20 characters. The node names within the parentheses are separated by at least one white space. The allowable

characters for node names are a-z, A-Z, 0-9, and the underscore character "_".

There must be an even number of nodes in a node list when using an RLGC matrix to describe a section. There will be half as many rows in an RLGC matrix as there are nodes. When an N_section is described by S-parameter matrices an odd number of nodes is allowed in the node list.

The order of the node names in the first half of the node list must match the order of the rows in the corresponding RLGC matrix data, or the order of the node names in the node list must be the same as the order of the rows in the corresponding S-parameter matrix.

A port for the S-parameter data is defined between a node and the ideal ground.

Matching node names inside a path description section are assumed to be connected with an ideal short. Nodes having the same names in different [Nodal Path Description]s are assumed to be not connected.

The second argument contains a single numeric value which allows a matrix section to be re-used as a scaled value. This argument can either be a multiplier factor determining how many times the section should be repeated in series (cascaded), or defines what its length is. The value of the second argument may be any positive value greater than zero. If a section is intended to be repeated (cascaded) a certain amount of times, the syntax of this argument is:

Mult = value

where "value" must be an integer number.

If this argument contains the length of a section, its syntax is:

Len = value

where "value" is a floating point decimal real number. (Note that spaces around the "=" sign are not required).

Mult and Len are separate arguments. Mult and Len are not allowed to be used in the same section.

Some sections within a particular path description can use M while others use Len, if so desired. If "Mult=" is used the [Derivation Method] keyword for the corresponding [ICM Section] must have a value of "Lumped" else it is an error. If "Len=" is used the [Derivation Method] keyword for the corresponding [ICM Section] must have a value of "Distributed" else it is an error.

The scale factor "Len" is applied to each parameter of all matrices of a section resulting in scaled RLGC matrices. When S-parameters are used to describe a section, the usage of scale factor "Len" is not allowed.

The third argument contains the name of the N_section which must match the name of a subsequent [Begin ICM Section] keyword. The electrical characteristics of sections are

described in the form of RLGC or S-parameter matrices under the [Begin ICM Section] keyword(s). A [Begin ICM Section] keyword may be referenced by more than one N_section subparameter even if they are in different path descriptions.

Model_NodeMap

The required subparameter Model_NodeMap is used to reference the [Node Map] keyword which provides a mapping between the actual data book pin (or signal) names and the node names used in the path description.

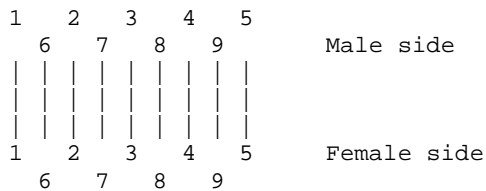
A path description must have at least one occurrence of the required subparameter Model_NodeMap, but can have as many Model_NodeMap subparameters as needed to adequately describe the interconnect. The number of Model_NodeMap subparameters used in a path description would be typically the same as the number of ends or sides in a connector or interconnect. Each Model_NodeMap subparameter must be on a single line.

The Model_NodeMap subparameter is followed by one argument, which is the name of a node map. The argument must match the name of a node map defined by a subsequent [ICM Node Map] keyword. The argument and the subparameter must be separated by at least one white space.

Please note that the following examples can be easily applied to connectors as well as cable assemblies, packages and printed circuit boards.

Example 1:

A simple model for a DB9 male to female adapter with a straight through wiring modeled as one section. The pin numbering increases left to right when facing the male side, and right to left when facing the female side. Note that this connector could have been also described by the [Path Description] keyword due to its regularity and the one to one mapping.



```

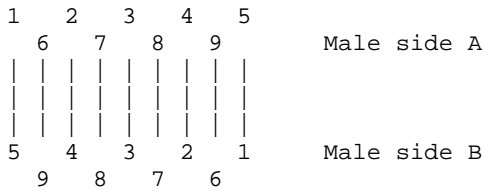
[Begin ICM Model] DB9M_F
  ICM_Model_Type MLM
[Nodal Path Description]
  Model_NodeMap DB9_Male_side
  N_section (M1 M2 M3 M4 M5 M6 M7 M8 M9
            F1 F2 F3 F4 F5 F6 F7 F8 F9) Mult=1 RLGC_matrix_9x9
  Model_NodeMap DB9_Female_side
[End ICM Model]

```

Example 2:

This is a model of the same adapter as in example 1, except that both sides are male which results in a crossed pinout. Note that this connector could have been also described by the [Tree Path Description] keyword due to its

regularity and the one to one mapping.

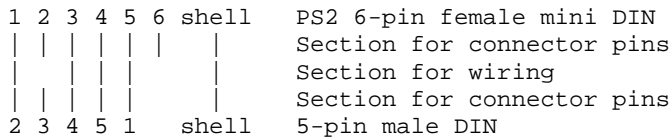


```

[Begin ICM Model] DB9M_M
  ICM_Model_Type MLM
[Nodal Path Description]
  Model_NodeMap DB9_side_A
  N_section (A1 A2 A3 A4 A5 A6 A7 A8 A9
             B5 B4 B3 B2 B1 B9 B8 B7 B6) Mult=1 RLGC_matrix_9x9
  Model_NodeMap DB9_side_B
[End ICM Model]
  
```

Example 3:

This is a model of a PS2 6-pin female mini DIN to 5-pin male DIN adapter. The model consist of three sections to provide a more detailed model. The drawing of this adapter was found on the following web site:
<http://www.grand-tek.com/SPEC-7/GM06F-R05MM0-04.pdf>

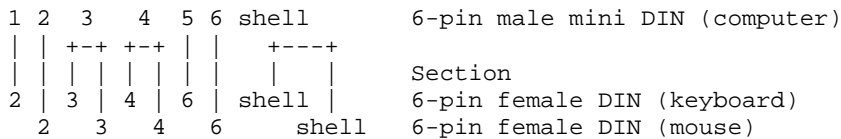


```

[Begin ICM Model] PS2_6_DIN_5
  ICM_Model_Type MLM
[Nodal Path Description]
  Model_NodeMap PS2_6_DIN_5_PS2_side
  N_section ( PS1 PS2 PS3 PS4 PS5 PS6 PS7
             n11 n12 n13 n14 n15 n16 n17) Mult=1 RLGC_matrix_7x7
  N_section ( n11 n13 n14 n15 n17
             n21 n23 n24 n25 n27) Mult=1 RLGC_matrix_5x5
  N_section ( n21 n22 n23 n24 n25 n27
             DIN2 DIN3 DIN4 DIN5 DIN1 DIN7) Mult=1 RLGC_matrix_6x6
  Model_NodeMap PS2_6_DIN_5_DIN_side
[End ICM Model]
  
```

Example 4:

This is a model of a PS2 keyboard Mouse Y-splitter adapter. The model consist of only one section for simplicity, but includes connections between the shells (not shown on the drawing of the web site). The specifications for this adapter were found on the following web site:
<http://www.hardwarebook.net/adapter/userinput/ps2keyboardgateway.html>



```

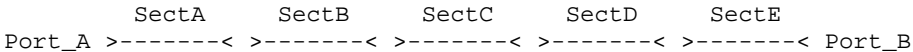
[Begin ICM Model] PS2_splitter
  ICM_Model_Type MLM
  
```



```
[Nodal Path Description]
Model_NodeMap PS2_splitter_Computer_side
N_section (C1 C3 C4 C5 Cs C2 C3 C4 C6 Cs
           K2 K3 K4 K6 Ks M2 M3 M4 M6 Ms) Mult=1 RLGC_matrix_10x10
Model_NodeMap PS2_splitter_Keyboard_side
Model_NodeMap PS2_splitter_Mouse_side
[End ICM Model]
```

Example 5:

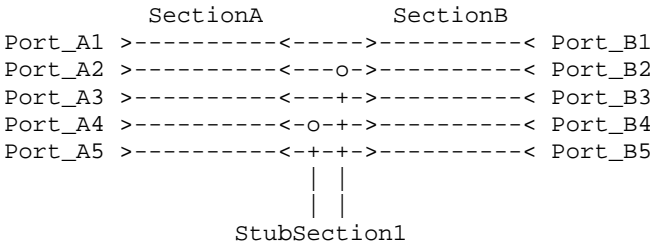
2 ports per pin, 5 sections straight through. Note that this interconnect could have been also described by the [Tree Path Description] keyword due to its regularity and the one to one mapping.



```
[Begin ICM Model] MyModelExample5
ICM_Model_Type SLM_Quiescent
[Nodal Path Description]
Model_NodeMap Port_A
N_section (A1 A2 A3 A4 11 12 13 14) Len=1.0 SectA
N_section (11 12 13 14 21 22 23 24) Len=1.0 SectB
N_section (21 22 23 24 31 32 33 34) Len=1.0 SectC
N_section (31 32 33 34 41 42 43 44) Len=1.0 SectD
N_section (41 42 43 44 B1 B2 B3 B4) Len=1.0 SectE
Model_NodeMap Port_B
[End ICM Model]
```

Example 6:

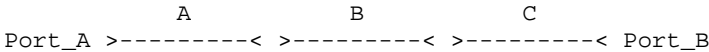
2 ports per pin, 2 series sections with one stub that taps into line_2 and line_4.



```
[Begin ICM Model] MyModelExample6
ICM_Model_Type MLM
[Nodal Path Description]
N_section (A1 A2 A3 A4 A5 11 12 13 14 15) Len=1.0 SectionA
N_section (12 14 x2 x4 ) Len=1.0 StubSection1
N_section (11 12 13 14 15 B1 B2 B3 B4 B5) Len=1.0 SectionB
[End ICM Model]
```

Example 7:

2 ports per pin, 2 stub sections, one stub matrix is re-used, and another stub is repeated.



```

|stub1      |stub4
|stub2      |stub4
|stub1
|Port_C

```

```

[Begin ICM Model] MyModelExample7
  ICM_Model_Type MLM
[Nodal Path Description]
  N_section (A1 A2 A3 A4 A5 11 12 13 14 15) Len=1.0 A
  N_section (11 12 13 14 15 s1 s2 s3 s4 s5) Len=1.0 stub1
  N_section (s1 s2 s3 s4 s5 t1 t2 t3 t4 t5) Len=1.0 stub2
  N_section (t1 t2 t3 t4 t5 C1 C2 C3 C4 C5) Len=1.0 stub1
  N_section (11 12 13 14 15 21 22 23 24 25) Len=1.0 B
  N_section (21 22 23 24 25 x1 x2 x3 x4 x5) Mult=2 stub4
  N_section (21 22 23 24 25 B1 B2 B3 B4 B5) Len=1.0 C
[End_ICM_Model]

```

=====

Keyword: [ICM Swath Parameters]

Required: No, unless an interconnect is described by a swath.

Sub-Params: LeftEdge, RightEdge, TopEdge, BottomEdge

Used By: [Begin ICM Model]

Uses: [ICM Swath Pin Numbers]

Description: Describes a smaller matrix which can be used to represent interconnects of equal or larger size. To visualize this, consider the swath matrix being a smaller version of a connector. The edges of the matrix represent the same edge effects that exists in the full size connector. The center area of the matrix represents the same effects as the center of the full size connector.

It is the responsibility of the EDA tool to use the matrix and the required keyword information as a "mini" interconnect that is centered about the electrical paths of interest and/or to expand this information into a larger sized swath up to the full sized interconnect itself. For example a 2x8 matrix might be expanded to represent a connector that is 2x100.

Usage Rules: Only rectangular interconnects without missing conductor may be represented by a swath matrix. If the interconnect is non-rectangular or the pin field has missing pins then the matrix must describe the full interconnect and this keyword cannot be used. This keyword **MUST** be present if the number of rows in the matrix does not equal the number of pin names as given by the [ICM Pin Map] keyword used for this model.

This keyword must appear between the [Begin ICM Model]/[End ICM Model] keyword pair. This keyword is only required when using only a part of the matrix or when expanding the matrix.

MUST be followed by 4 required subparameters:

- 1) LeftEdge: Columns in the swath matrix used to represent left edge
- 2) RightEdge: Columns in the swath matrix used to represent right edge
- 3) TopEdge: Rows in the swath matrix used to represent top edge
- 4) BottomEdge: Rows in the swath matrix used to represent bottom edge

These subparameters use the physical representation of the pins to define the minimum coupling bandwidth.

The model maker decides on the minimum number of columns (for a column swath) or rows (for a row swath) that should be used to define the banded matrix, then assigns values to the appropriate subparameter that will, by subtraction of the columns or rows, create the appropriate maximum matrix size for the swath.

The basic goal is to define how far the edge effects extend into the coupling fields.

None of the values of these numbers may exceed the number of rows or column of pins that are defined by the matrices of a given model. The swath is applicable to any model supported by the specification.

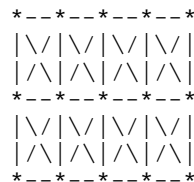
At this time, there must be at least two subparameters with a value not equal to zero.

Please see the example below to see how these are used with the keyword.

The edge effects describe the edge rows or columns that do not contain all of the coupling effects (from both sides) that the centermost rows or columns. This information is useful for "swath expansion" when determining what larger size to use.

When the swath represents a SLM or else documents all of the pins in a physical interconnect, the off diagonal entries are 0 by convention. If the swath describes a section with fewer columns, but the same number of rows as the physical interconnect, then the row edge effects are 0 by convention. Similarly, if the swath describes a section with fewer rows, but the same number of columns as the physical interconnect, then the column edge effects are 0 by convention.

In a fully coupled swath where all pins couple to all others, all of the rows or columns except for the one designated as the centermost row or column can have edge simplification. So the number of rows or columns can extend to all but the center row or column. On the other hand, a coupling pattern that extends over a few columns may result in a smaller number of edges designated. Below is an example of a 3 row by 5 column swath, where the lines in a rectangular connector represent the electrical coupling:

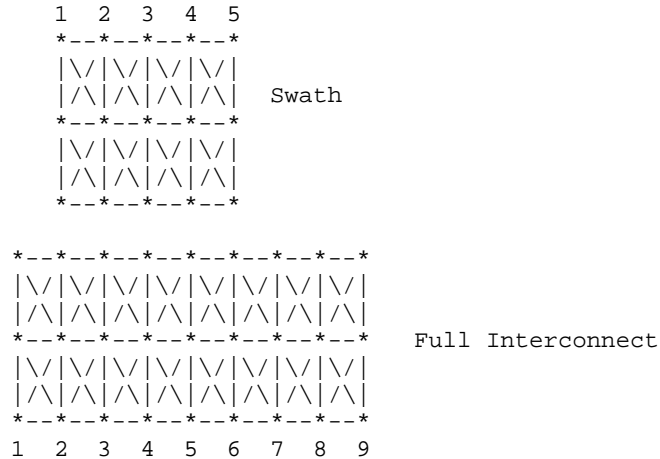


Notice that columns 2, 3, and 4 have the maximum amount of coupling from each side. In this case the edge effect columns 1 and 5 do not have coupling from both sides.

Application of Swath Matrices:

For the purposes of this discussion a 3 row by 5 column swath is used and the full sized interconnect is 3 rows by 9 columns. The swath matrix data can be organized by [ICM Swath Parameters]. Three methods can be used to apply the Swath matrix information:

- (1) Centering the Swath around the pins of interest



The Swath above is centered about column 4. The paths of interest including those of columns 3 and 5 are also described by swath matrix data that has coupling from both sides. However, because of edge effects, columns 2 and 6 are less accurately approximated.

Centering is a way to map the swath matrix pins directly into the corresponding pins of the actual interconnect.

When the signals of interest are at the left or right edges, the swath would normally be positioned just to the edges. For example, column 1 of the swath matrix would line up with column 1 of the interconnect.

- (2) Expansion and Centering: Expanding the Swath matrix into a larger sized Swath matrix and centering it about the paths of interest.

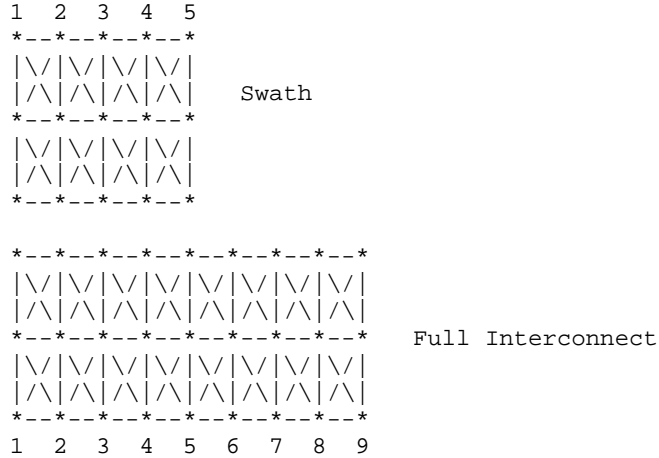
This approach is similar to centering. However, the signal pins of interest might have more columns than the non-edge section of the swath. One solution is to expand the swath to a larger swath and then use the larger swath for centering. This is a subset of expanding the swath into the full sized interconnect.

A larger swath might be constructed to span the number of columns of interest plus the edge effect columns to assure that the most accurate coupling information from both sides is used for the columns of interest.

- (3) Expansion to Full Sized Interconnect: Expanding the swath matrix to the full sized interconnect matrix.

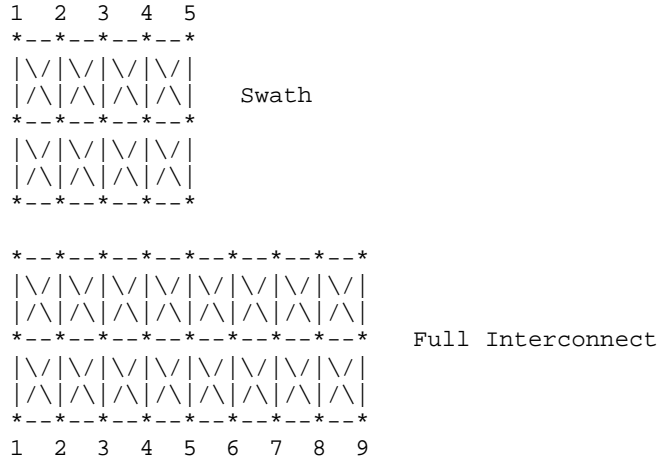
There are three steps to this procedure (a, b, and c below). This is illustrated using the 3 row, 5 column swath and showing how it is expanded into a 3 row, 9 column full sized interconnect.

(a) Copy the left column pattern



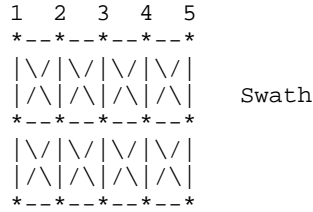
The matrix data for swath columns 1 and 2 (including forward references) copied into the new full interconnect matrix.

(b) Repeat the center pattern,

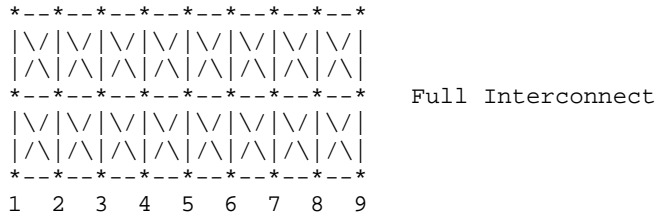


Starting at the center of the swath, the coupling pattern of just the center column (column 3 in this example) is copied for the interconnect matrix. In this case, the coupling pattern extends forward to the next column. The pattern is copied by mapping the swath pin numbers into the interconnect pin numbers and copying the center column forward referenced data. In this example, the operation starts at the full interconnect column 3 and ends upon completion at the full interconnect column 7.

(c) Copy the right column patterns



Swath



Full Interconnect

The final step is similar to Step (a), but concluded using the right side of the swath. When the swath is positioned against the right edge as shown, then the coupling patterns for the swath columns are duplicated for the final columns of the full sized interconnect. So the patterns of swath columns 4 and then 5 are duplicated for the full sized interconnect columns 8 and 9.

A similar process could be created for expanding the number of rows. Some details are left to the EDA tool to implement. For example, if the swath contains an even number of columns, then either the left of center or the right of center column may be used for center section expansion.

```

-----
[ICM Swath Parameters]
LeftEdge = 1
RightEdge = 1
TopEdge = 0
BottomEdge = 0

```

=====

```

Keyword: [ICM Swath Pin Numbers]
Required: Yes, if [ICM Swath Parameters] is present, else it must not be
used.
Used By: [ICM Swath Parameters]
Description: Lists the pins of the matrix in the exact order that the
swath matrix rows are entered and gives the corresponding
physical location of the pin

```

```

Usage Rules: The [ICM Swath Pin Numbers] keyword is followed by a list of
entries that apply for all of the sections as referenced by the
[Tree Path Description] keyword.

```

The number of entries in the list is equal to the product of the rows and columns of the [ICM Swath Pin Numbers] keyword.

The [ICM Swath Pin Numbers] keyword must appear only once between the [Begin ICM Model]/[End ICM Model] keyword pair.

Each line must contain three columns documenting the row

names and physical row and column locations for each entry of the matrices under the corresponding [Begin ICM Section] keyword. The entries consist of [Row] name designated row_name, physical row number designated physical_row, and physical column number designated physical_column. All three columns are required and must be separated by at least one white space.

The physical row number consists of an integer value from 1 to the number of rows documented in the [ICM Swath Pin Numbers] keyword. Similarly, the physical column number consists of an integer value from 1 the number of columns documented in the [ICM Swath Pin Numbers] keyword.

These physical row and column numbers give the relative physical location of each of the swath matrix elements as if they were the pin numbers of the interconnect itself. Only one unique physical row and physical column number pair are allowed in the list. In other words, no two or more row_name entries shall refer to the same physical_row, physical_column pair.

Other Notes: It is required that all of the sections that are referenced by the [Begin ICM Section] keyword have swath matrix entries that are ordered in the same manner. This would be expected from the field solver working on a single interconnect swath, but issuing its results as separate sections.

The EDA tool that processes this keyword has enough information to reorder the swath matrix, if needed, to position its electrical contents among the pins of interest in the actual interconnect.

 Consider a 2 row by 3 column Swath matrix that was extracted for the pins as shown below:

```
A B C
D E F
```

The field extractor may have produced matrices ordered: C E B F A D

```
[Swath Pin Numbers]
| row_name      physical_row  physical_column
| C             1             3
| E             2             2
| B             1             2
| F             2             3
| A             1             1
| D             2             1
```

```
=====
Keyword: [End ICM Model]
Required: Yes
Used By: [Begin ICM Model]
Description: Defines the end of the interconnect models for the
            interconnect family
```

 [End ICM Model]

=====

Keyword: [ICM Pin Map]

Required: No, unless the [Tree Path Description] keyword exists.

Argument: Text String (Model_PinMap name)

Sub-params: Pin_order, Num_of_columns, Num_of_rows

Used By: [Tree Path Description]

Description: This keyword gives the names of the pins on one side of an interconnect, then defines a pin ordering such that pins can be associated with rows in the RLGC or S-parameter section matrixes. This keyword also defines whether the pins are listed in column ordered or row ordered fashion.

Usage Rules: Following the keyword is the name of this pin map. The pin map name must match the pin map name used as an argument to one of the Model_PinMap subparameters in a model's [Tree Path Description] keyword.

This key word should be between the [Begin ICM Model Family] and [End ICM Model Family] keywords.

Following the [ICM Pin Map] line are three subparameters: Pin_order, Num_of_rows, Num_of_columns. Each subparameter is placed on a separate line, and when used they must appear in the above order. Following the subparameters is a list of pin names that are covered by this pin map.

Subparameters and the pin list are described in detail below.

Pin_order

The Pin_order subparameter is required. Legal arguments to the Pin_order subparameter are Row_ordered, Column_ordered or Un-ordered. The subparameter name and its argument are separated by one or more white spaces.

Row_ordered and Column_ordered refers to the order in which the pins in the accompanying pin list are listed. Row_ordered means that all the pins in the top row of an interconnect are listed in order, then the second row of pins is listed, and so on. Likewise, Column_ordered means that all the pins in the first column of an interconnect are listed, then the second column, and so forth.

If the Row_ordered or Column_ordered argument is specified then the Num_of_rows and Num_of_columns subparameters are also required.

If the Un-ordered argument is specified, the pins are not listed in row or column order and the Num_of_rows and Num_of_columns subparameters must not be used. Un-ordered also implies that the interconnect is not swathable.

Num_of_columns

This optional subparameter specifies the number of columns that are in the physical interconnect. The subparameter name is followed by a single positive integer.

Num_of_rows

This optional subparameter specifies the number of rows that

are in the physical interconnect. The subparameter name is followed by a single positive integer.

Note that the pin order and number information are required in order for EDA software to properly position and expand a swath matrix around a given pin.

Pin List

Following the subparameters is a list of the pin names covered by this pin map. The pin list is required. Each pin name is listed on a separate line on the file. The pin names must be organized as specified by the Pin_order subparameter. There is a second column which contains a name for the pin

IMPORTANT NOTE: The intent of the [ICM Pin Map] keyword pin order information is to give an EDA tool enough information to map pin names to the rows of an interconnect's section matrix. In the case of a full-matrix/full-pinout description of an interconnect, it is required that the rows of a section matrix appear in the same row or column order as the pin names. For example, suppose the pins names of a six pin connector were listed as follows:

```
A1 SIG
A2 SIG
A3 SIG
B1 SIG
B2 SIG
B3 GND
```

A six pin connector can be described by a 6x6 matrix. It is required that the first row of the matrix contain the data for pin A1, the second row contain the data for pin A2, and so on.

Other Notes: More than one [ICM Pin Map] keyword is allowed in a file.

This keyword must occur outside the [Begin ICM Model]/[End ICM Model] keyword pair. This is to allow a [ICM Pin Map] to be referenced by multiple interconnect models.

"GND" is a reserved word. The use of "GND" is required for models designated as "SLM_Quiescent", "SLM_EvenMode" and "SLM_OddMode". For SLM_General and MLMs "GND" cannot be used. "GND" is optional for model type s-parameter.

All diagonal values in any of the parametric matrix positions that are referenced as "GND" must contain the value of "0". See the "Matrix Description" section of this document for further information.

The examples below use the connector represented by:

```
A1 A2 A3 A4
B1 B2 B3 B4
```

```
-----
[ICM Pin Map] Baseboard_side
Pin_order = row_ordered
Num_of_columns = 4
Num_of_rows = 2
```

```
|Pin  Type
  A1  SIG
  A2  SIG
  A3  SIG
  A4  SIG
  B1  SIG
  B2  SIG
  B3  SIG
  B4  SIG
```

```
[ICM Pin Map] Baseboard_side2
Pin_order = column_ordered
Num_of_columns = 4
Num_of_rows = 2
```

```
|Pin
  A1  SIG
  B1  SIG
  A2  SIG
  B2  SIG
  A3  SIG
  B3  SIG
  A4  SIG
  B4  SIG
```

```
[ICM Pin Map] Daughtecard_side
Pin_order = Un_ordered
```

```
|Pin
  A1  SIG
  A2  SIG
  A3  SIG
  A4  SIG
  B1  SIG
  B2  SIG
  B3  SIG
  B4  SIG
```

```
=====
```

Keyword: [ICM Node Map]

Required: No, unless the [Nodal Path Description] keyword exists.

Argument: Text String (Model_NodeMap name)

Used By: [Nodal Path Description]

Description: This keyword is used to provide a mapping between the actual data book pin names and the node names used in the path description. This may be useful, for example, when the data book pin names are identical on either side of a connector, or when only a signal name is available for a conductor on package or BCP, because the node names on the two ends of these interconnects must be unique in the path description (to prevent shorting the pins of the two sides). Typically there would be as many Model_NodeMap subparameters in a path description as there are ends or sides.

Usage Rules: Following the keyword is the name of this node map. The node map name must match the node map name used as an argument to one of the Model_NodeMap subparameters in a model's [Nodal Path Description] keyword.

This key word should be between the [Begin ICM Model Family] and [End ICM Model Family] keywords.

The [ICM Node Map] line is followed by a two-column table. The first column contains the list of pin or signal names and the

second column contains the corresponding node names. The node names listed here must match the node names by the N_section subparameter(s) under the [Nodal Path Description] keyword to make a connection.

Note that the first columns (i.e. the pin or signal names) of different [ICM Node Map] keywords may be identical in certain situations. This does not result in an electrical short between those pins or signals even if these [ICM Node Map] keywords are called by the Model_NodeMap subparameters of the same [Nodal Path Description] keyword, because it is the associated node names which make the electrical connections. Also, the node names defined in the [ICM Node Map] keyword are not required to be nodes having a single connection only in the path description.

There is a third column, which contains a name for the pin.

"GND" is a reserved word. The use of "GND" is required for models designated as "SLM_Quiescent", "SLM_EvenMode" and "SLM_OddMode". For SLM_General and MLMs "GND" cannot be used. "GND" is optional for model type s-parameter.

All diagonal values in any of the parametric matrix positions that are referenced as "GND" must contain the value of "0". See the "Matrix Description" section of this document for further information.

Other Notes: More than one [ICM Node Map] keyword is allowed in a file.

This keyword must occur outside the [Begin ICM Model] / [End ICM Model] keyword pair. This is to allow a [ICM Node Map] to be referenced by multiple interconnect models.

[ICM Node Map] sections are terminated by the occurrence of the next keyword.

The following examples are to be used with the examples following the description of the [Nodal Path Description] keyword.

Example 1:

```
[ICM Node Map] DB9_Male_side
| pin    node    name
|  1     M1     SIG
|  2     M2     SIG
|  3     M3     SIG
|  4     M4     SIG
|  5     M5     SIG
|  6     M6     SIG
|  7     M7     SIG
|  8     M8     SIG
|  9     M9     SIG
[ICM Node Map] DB9_Female_side
| pin    node    name
|  1     F1     SIG
|  2     F2     SIG
|  3     F3     SIG
|  4     F4     SIG
|  5     F5     SIG
|  6     F6     SIG
```

7	F7	SIG
8	F8	SIG
9	F9	SIG

Example 2:

```
[ICM Node Map] DB9_side_A
| pin  node
  1    A1    SIG
  2    A2    SIG
  3    A3    SIG
  4    A4    SIG
  5    A5    SIG
  6    A6    SIG
  7    A7    SIG
  8    A8    SIG
  9    A9    SIG
[ICM Node Map] DB9_side_B
| pin  node
  1    B1    SIG
  2    B2    SIG
  3    B3    SIG
  4    B4    SIG
  5    B5    SIG
  6    B6    SIG
  7    B7    SIG
  8    B8    SIG
  9    B9    SIG
```

Example 3:

```
[ICM Node Map] PS2_6_DIN_5_PS2_side
| pin  node
  1    PS1   SIG
  2    PS2   SIG
  3    PS3   SIG
  4    PS4   SIG
  5    PS5   SIG
  6    PS6   SIG
  shell PS7   SIG
[ICM Node Map] PS2_6_DIN_5_DIN_side
| pin  node
  1    DIN1  SIG
  2    DIN2  SIG
  3    DIN3  SIG
  4    DIN4  SIG
  5    DIN5  SIG
  shell DIN7  SIG
```

Example 4:

```
[ICM Node Map] PS2_splitter_Computer_side
| pin  node
  1    C1    SIG
  2    C2    SIG
  3    C3    SIG
  4    C4    SIG
  5    C5    SIG
  6    C6    SIG
  shell Cs   SIG
```

[ICM Node Map] PS2_splitter_Keyboard_side

```
| pin    node
  2     K2     SIG
  3     K3     SIG
  4     K4     SIG
  6     K6     SIG
shell  Ks     SIG
```

[ICM Node Map] PS2_splitter_Mouse_side

```
| pin    node
  2     M2     SIG
  3     M3     SIG
  4     M4     SIG
  6     M6     SIG
shell  Ms     SIG
```

Example 5:

[ICM Node Map] Port_A

```
| pin    node
Port_A1  A1     SIG
Port_A2  A2     SIG
Port_A3  A3     SIG
Port_A4  A4     SIG
```

[ICM Node Map] Port_B

```
| pin    node
Port_B1  B1     SIG
Port_B2  B2     SIG
Port_B3  B3     SIG
Port_B4  B4     SIG
```

Example 6:

[ICM Node Map] Port_A

```
| pin    node
Port_A1  A1     SIG
Port_A2  A2     SIG
Port_A3  A3     SIG
Port_A4  A4     SIG
Port_A5  A5     SIG
```

[ICM Node Map] Port_B

```
| pin    node
Port_B1  B1     SIG
Port_B2  B2     SIG
Port_B3  B3     SIG
Port_B4  B4     SIG
Port_B5  B5     SIG
```

Example 7:

[ICM Node Map] Port_A

```
| pin    node
Port_A1  A1     SIG
Port_A2  A2     SIG
Port_A3  A3     SIG
Port_A4  A4     SIG
Port_A5  A5     SIG
```

[ICM Node Map] Port_B

```
| pin    node
Port_B1  B1     SIG
Port_B2  B2     SIG
```

Port_B3	B3	SIG
Port_B4	B4	SIG
Port_B5	B5	SIG
[ICM Node Map] Port_C		
pin	node	
Port_C1	C1	SIG
Port_C2	C2	SIG
Port_C3	C3	SIG
Port_C4	C4	SIG
Port_C5	C5	SIG

```

=====
Keyword: [End ICM Model Family]
Required: Yes
Used By: [Begin ICM Model Family]
Description: Defines the end of the interconnect model family
Usage Rules: These commands wrap the entire model in this file.

```

```

-----
[End ICM Model Family]

```

8.0 MATRIX KEYWORDS

```

=====
Keyword: [Begin ICM Section]
Required: Yes
Argument: Text String
Used By: [Tree Path Description] or [Nodal Path Description]
Uses: [Derivation Method], [Resistance Matrix], [Inductance Matrix],
      [Conductance Matrix], [Capacitance Matrix], [ICM Sparameter],
      [End ICM Section]
Description: Indicates the beginning and the name of a set of RLGC
            or S-parameter matrix data
Usage Rules: This keyword is followed by the name of a section. The section
            must be used in one of the models in the document.

```

This keyword should be after the [End ICM Model Family] keyword.

When using [Tree Path Description], every matrix used for given interconnect model MUST have exactly the same number of diagonal terms. When using a [Nodal Path Description], the number of diagonal terms in the matrices may vary.

If the swath method is used the matrix may have fewer conductors than the interconnect model. When the swath method is not used, the matrix MUST have exactly the same number of diagonal terms as the interconnect has conductors.

The matrices used for the RLGC description can be defined using any of the matrix description methods allowed in the specification. The matrix description methods can also be mixed within a given section. For example the resistance matrix might use the Diagonal_matrix and the inductance matrix might use a Full matrix.

MUST be placed outside the context of keywords

[Begin ICM Model Family] and [End ICM Model Family] as sections may be reused by more than one interconnect model.

For an example of [Begin ICM Section] see matrix examples that follow.

```
=====
Keyword: [Derivation Method]
Required: Yes
Argument: Text String (Lumped or Distributed)
Used By: [Begin ICM Section]
Description: This keyword describes how the data in the RLGC or S-parameter
matrixes should be interpreted. This keyword also places rules
on what individual matrixes must be present in a section.
Usage Rules: Following the keyword is one of two arguments: "Lumped" or
"Distributed".
```

If "Lumped" is specified then the RLGC matrix data represents discrete values suitable for creating a circuit of interconnected inductor, capacitor and resistor elements. A section may be replicated, but the values in each matrix are fixed. For sections described using lumped data any single, any pair, or all of the following keywords could be used to describe a section: [Resistance Matrix], [Inductance Matrix], [Conductance Matrix], and [Capacitance Matrix].

If "Distributed" is specified then the RLGC matrix data represents per-unit-length values suitable for creating a transmission line representation of the section. A multiplier value representing the physical length of a section may be applied to the data in the RLGC matrixes. For sections described using distributed data both the [Inductance Matrix] and [Capacitance Matrix] keywords are required. The [Conductance Matrix] and [Resistance Matrix] keywords are optional.

For a section described using S-parameter data the argument to the [Derivation Method] keyword shall be "lumped".

All matrices within the same section must share the same derivation method.

For any SLM lumped model, Resistance and Inductance are series components, Capacitance and Conductance are shunted to the return path.

The assumed circuit topology is LRGC for any derivation method. If a different circuit topology is desired (for example, CLRG) then start the model by creating a section that has only the capacitance element, then follow this by a standard LRG section.

```
-----
[Derivation Method] Lumped
```

```
=====
Keyword: [Resistance Matrix], [Inductance Matrix], [Conductance Matrix],
[Capacitance Matrix]
Required: Yes, at least one
Argument: Text String (Matrix Type)
Used By: [Begin ICM Section]
Description: Each keyword marks the beginning of its respective matrix data
set and also specifies how the matrix data is formatted.
```

Usage Rules: Following each matrix keyword, one of the following arguments is required: `Diagonal_matrix`, `Banded_matrix`, `Sparse_matrix`, or `Full_matrix`. No more than one of each matrix type is allowed per section. If model type is SLM only a diagonal matrix is allowed.

Each [`Resistance Matrix`], [`Inductance Matrix`], [`Conductance Matrix`], and [`Capacitance Matrix`] keyword must appear between the [`Begin ICM Section`]/[`End ICM Section`] keyword pair."

After each of these keywords, insert the matrix data in the appropriate format. (These formats are described in detail below).

If the location in the matrix is defined as a return path ("GND") in the [`ICM Pin Map`] or [`ICM Node Map`], the values associated with the "GND" designation in the matrix diagonal must be 0 for any of the matrices.

Arguments:

`Diagonal_matrix`, `Banded_matrix`, `Sparse_matrix`, or `Full_matrix`

`Diagonal_matrix`

The `Diagonal_matrix` is used to specify uncoupled models.

This subparameter is exactly the same as using the `Banded_matrix` described below with a `Bandwidth=0`. It has the added benefit of requiring half as many lines.

A `Diagonal_matrix` is one whose entries are guaranteed to be zero if they are not on the main diagonal.

Specify the `Diagonal_matrix` one row at a time, starting with row 1 and working up to higher rows. Diagonal elements are entered one per line. Mark each row with the [`Row`] keyword.

`Banded_matrix`

The `Banded_matrix` is used to specify the coupling effects up to "X" pins on either side for a given pin.

A `Banded_matrix` is one whose entries are guaranteed to be zero if they are farther away from the main diagonal than a certain distance, known as the "bandwidth."

The bandwidth for a `Banded_matrix` must be specified using the [`Bandwidth`] keyword.

Specify the banded matrix one row at a time, starting with row 1 and working up to higher rows. Mark each row with the [`Row`] keyword.

Symmetry is exploited to reduce the amount of data by eliminating, from the matrix, any entries below the main diagonal.

Sparse_matrix

A Sparse_matrix is expected to consist mostly of zero-valued entries, except for a few nonzero. Unlike the Banded_matrix, there is no restriction on where the nonzero entries can occur.

An NxN Sparse_matrix is specified one row at a time, starting with row 1 and continuing down to row N. Each new row is marked with [Row] keyword, as in the other matrix formats.

Symmetry is exploited to reduce the amount of data by eliminating, from the matrix, any entries below the main diagonal.

Full_matrix

Description: When the Full_matrix format is used, the coupling between every pair of elements is specified explicitly. Assume that the Matrix has N rows and N columns.

The Full_matrix is specified one row at a time, starting with Row 1 and continuing down to Row N. Each new row is identified with the [Row] keyword.

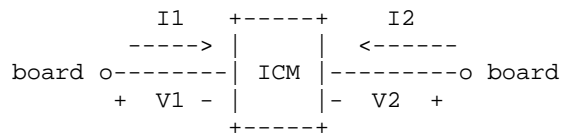
Symmetry is exploited to reduce the amount of data by eliminating, from the matrix, any entries below the main diagonal.

See example "Matrix Example" lower in this document.

Matrix Descriptions

This insert provides additional information regarding the [Resistance Matrix], [Inductance Matrix], [Conductance Matrix], and [Capacitance Matrix] keywords. These will be referred to as "RLGC" matrices.

When measuring the entries of the RLGC matrices, either with laboratory equipment or field solver software, currents are defined as ENTERING the pins of the package from the board. The corresponding voltage drops are to be measured with the current pointing "in" to the "+" sign and "out" of the "-" sign.



It is important to observe this convention in order to get the correct signs for the mutual inductance and resistances.

For each [Resistance Matrix], [Inductance Matrix], [Conductance Matrix] or [Capacitance Matrix] a different matrix type can be used for the data.

Resistance Matrix Description:

The resistance matrix represents the total DC resistance in ohms between the two ends of this section of the interconnect.

The resistance matrix is optional. Simulation tools shall use 0.0 ohms for all diagonal entries and "infinite" for all off diagonal entries if this matrix is not present.

When the Section or N_section subparameters use the "Mult" variable then DO NOT USE ohms/meter as this is the total value and includes the length of this section.

When the Section or N_section subparameters use the "Len" variable then USE ohms/meter.

Values should all appear in the diagonal terms of the matrix. For example: value 1,1 is the resistance for pin 1 and value 2,2 is the resistance for pin 2.

A "Diagonal_matrix" matrix type is the best choice for describing the resistance matrix. Diagonal elements may be dominant. Comparably, the off diagonal elements may be negligible thus leading to off diagonal element values of zero leading to the [Diagonal_matrix] format.

Inductance Matrix Description:

Two inductance matrix methods are supported:

Loop-Inductance Matrix for SLMS ONLY (non-coupled)

Partial-Inductance Matrix for MLMS ONLY (coupled)

If ModelType is SLM then all sections must use loop-inductance.

If ModelType is MLM then all sections must use partial-inductance.

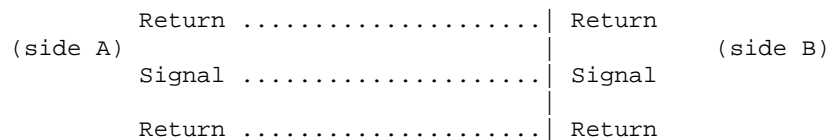
When the Section or N_section subparameters use the "Mult" variable then DO NOT USE henries/meter as this is the total value and includes the length of this section.

When the Section or N_section subparameters use the "Len" variable then USE henries/meter

Loop-Inductance Matrix Description (SLMS ONLY):

To define inductance requires a loop. The loop consists of a closed path including a signal and all possible return paths. The inductance is a coefficient between the emf induced in the loop and the change of magnetic flux through the loop.

interconnect or section



The return path must include all pins that are power or ground. In a loop inductance matrix, the diagonal terms represent the

loop inductance of a specific pin given a specific return path.

The loop inductance method must be used if the ModelType is SLM.

Loop inductance can be experimentally measured.

Loop inductance matrices can be created using field solvers.

An SGR value is normally specified for an SLM model that indicates the ratio of signal pins to return path pins (i.e., power or ground) pins.

Typical uses for SLM_general model might include:
a 2x17 header connector where the bottom row is all grounded. The SGR would be specified in this case as 1:1. A model would be developed with these assumptions made. Then all pins could be filled with average values. To further expand this example see the figure below:

```
Pin out:  
  1 ..... 17  
 18 ..... 34
```

```
Measurement or Field solver connection:  
  Sig Sig Sig ... Sig Sig  
  Gnd Gnd Gnd ... Gnd Gnd
```

```
Measured or Computed values:  
  3nH 4nH 4nH ... 4nH 3nH
```

```
Matrix data:  
  3nH 4nH 4nH ... 4nH 3nH  
  3nH 4nH 4nH ... 4nH 3nH
```

This allows the user to specify any of the following power/ground connections and still use this model:

```
Simulator Connection A:  
  Sig Sig Sig ... Sig Sig  
  Gnd Gnd Gnd ... Gnd Gnd
```

```
Simulator Connection B:  
  Gnd Gnd Gnd ... Gnd Gnd  
  Sig Sig Sig ... Sig Sig
```

```
Simulator Connection C:  
  Sig +5V Sig ... Gnd Sig  
  Gnd Sig Gnd ... Sig +5V
```

Given the following connection a simulation should generate a warning to the user if the indicated ground-signal-ratio has been violated.

```
Simulator Connection D:  
  Sig Sig Sig ... Sig Sig  
  Gnd Sig Sig ... Sig Sig
```

All interconnect conductors must be specified with non-zero L and C values in at least one of the sections. The simulator will zero out portions of all sections

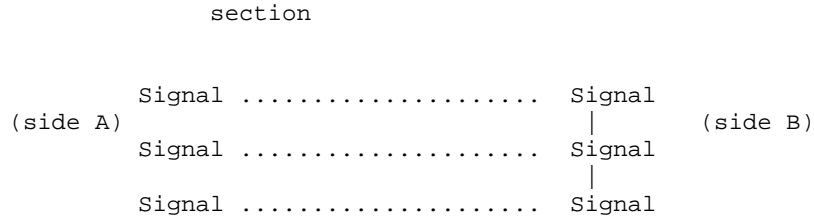
corresponding to the ground or power pins.

SLM models are most useful where coupling analysis is not critical and where the rise time of the signal is long compared to the electrical delay through the interconnect.

Partial Self Inductance Matrix Description (MLMs ONLY):

A partial self inductance is an approximation of the loop inductance in situations where the current return path cannot be explicitly stated in the field extraction stages. Partial self inductance can be numerically calculated by introducing a current in a circuit element that originates on one end and terminates on the other end. There is no physical loop per se, instead charges will build up on each side of the circuit element. In practice, this situation cannot exist more than instantaneously. For the purposes of simulation the condition does not affect the resultant loop inductance values. Then from the currents found the magnetic fields and inductance can be calculated. In this way, the self and mutual inductances of any section can be calculated.

This representation of inductance in terms of partial self and mutual inductance allows the analysis of any combination of current paths. The signal and return current paths can be defined in circuit after the model is established. Assuming that the self inductance is included in the return path, the circuit under simulation can account for the impact on signal-integrity, cross-talk and ground-bounce from currents passing through any combination of pins.



There are many ways to create this information using either 2D or 3D field solvers. It may also be possible to create this information using other methods which are left to the model developer.

For an MLM model partial inductance values are required.

Capacitance Matrix Description

A capacitance is an approximation of the capacitance in situations where the return path cannot be explicitly stated in the field extraction stages.

See "Multi-Line Capacitance Matrix" section above for more information on Multi-Line Matrices.

Two capacitance matrix methods are supported:

SLM, Single-Line, Capacitance Matrix (non-coupled)

MLM, Multi-Line, Maxwell Capacitance Matrix (coupled)

If ModelType is SLM then all sections must only contain a Single Line Capacitance Matrix.

If ModelType is MLM then all sections must use a Maxwell Capacitance as stated below.

Single-Line Capacitance Matrix Description:

To define capacitance requires conductors separated by a dielectric. The conductors are those found in a loop consisting of a closed path including a signal and all possible return paths. The Maxwell Capacitance matrix relates the total charges on the conductors to the voltages applied to them. $[Q]=[C_m][V]$ Where $[C_m]$ is the Maxwell matrix, $[Q]$ is the vector of total conductor charges, and $[V]$ is a vector of applied voltages. The negative values of the off-diagonal entries can be explained on physical grounds: if you put +1 Volt on a single conductor and 0 Volts on the others, then you induce negative charges on those other conductors to terminate the field lines emanating from the conductor at 1 Volt.

Maxwell Capacitance Matrix Description

In the Maxwell Capacitance matrix, the diagonal term is the maximum capacitance that the conductor encounters, meaning the capacitance to ground plus the capacitances to all the other conductors.

In the Maxwell capacitance matrix, the off-diagonal term is the capacitance that the conductor encounters relative to a second conductor. This capacitance value is affected by other nearby conductors during the field extraction phase. It should not normally be used by itself to determine the capacitance of a random pair of conductors in a given structure.

When the Section or N_section subparameters use the "Mult" variable then DO NOT USE farads/meter as this is the total value and includes the length of this section.

When the Section or N_section subparameters use the "Len" variable then USE farads/meter

Conductance Matrix Description:

The Conductance matrix represents the energy transfer between the conductor and its surroundings, including the return path. Conductance includes dielectric and radiation losses, as well as any DC conductance.

The conductance matrix is optional. Simulation tools shall use 0.0 Siemens for all entries if this matrix is not present.

When the Section or N_section subparameters use the "Mult" variable then DO NOT USE siemens/meter as this is the total value and includes the length of this section.

When the Section or N_section subparameters use the "Len" variable then USE siemens/meter

Loss Matrix Description:

The [ICM Sparameter] keyword introduced later provides a format for describing losses that vary with frequency.

```
=====
Keyword: [Bandwidth]
Required: Yes, if a Banded_matrix matrix is being defined.
Argument: Numeric Value
Used By: [Resistance Matrix], [Inductance Matrix], [Conductance Matrix],
[Capacitance Matrix]
Description: This keyword indicates the size of the width of the coupling
matrix. This does not have anything in common with electrical
bandwidth. The bandwidth field must be a nonnegative integer.
This is the number of off-diagonal matrix rows for which data is
provided.
Usage Rules: The bandwidth field must be a non-negative integer. This keyword
must occur after the [Resistance Matrix], etc., keywords, and
before the matrix data is given.
-----
See example "Matrix Example" lower in this document.
```

```
=====
Keyword: [Row]
Required: Yes
Argument: Row Number
Used By: [Resistance Matrix], [Inductance Matrix], [Conductance Matrix],
[Capacitance Matrix]
Description: Indicates the beginning of a new row of the matrix
Usage Rules: The argument must be a sequential number beginning at 1
and continuing to the last row for the matrix.

Following a [Row] keyword is a block of numbers that represent
the entries for that row. Suppose that the current row is
number M. Then the first number listed is the diagonal entry,
[M,M]. Following this number are the entries of the upper half
of the matrix that belong to row M: [M, M+1], [M, M+2], ... up
to [M,N].

For even a modest-sized matrix, this data will not all fit on
one line. It is allowed to break the data up with new-line
characters so that the line length limit is observed.

Example: Suppose the matrix has 40 conductors and that we are
currently working on Row 19. There is 1 diagonal entry, plus
40 - 19 = 21 entries in the upper half of the matrix to be
specified, for 22 entries total.

Row 1 always has the most entries, and that each successive row
has one fewer entry than the last; the last row always has just
a single entry.

The [Row] keyword is placed directly after the [Bandwidth]
keyword. Following the keyword, on the next line, is the actual
data for that row.
-----
See example "Matrix Example" lower in this document.
=====
```

Keyword: [ICM Sparameter]

Required: No, unless ICM_Model_Type subparameter of [Begin ICM Model] is equal to "S-parameter".

Sub-Params: File_name, Port_map

Used By: [Begin ICM Section]

Description: Indicates a reference to an external S-parameter file

Usage Rules: An S-parameter matrix can be used in place of an RLGC matrix. One [ICM Sparameter] keywords is allowed for each [Begin ICM Section] / [End ICM Section].

The [ICM Sparameter] keywords is allowed between [Begin ICM Section] / [End ICM Section] keywords.

File_name

File_name is to reference the name of the Touchstone file. It is required that the ICM and the referenced file be in the same directory.

Port_map

"Port_map" is to be used to relate the ports in the S-parameter file to the "fixed" pin map in the ICM file. It is important that the "Port_map" uses the pin names from [ICM PinMap] or [ICM Node Map]. No more or no less entries in the table are allowed.

There are three columns used in the Port_map. The first column is a reference to a "Pin Name" as defined in the [ICM Pin Map] or [ICM Node Map] keyword. The second column should reference the input port name as defined in the Touchstone file that is related to the "Pin Name". The third column should reference the output port name as defined in the Touchstone file that is related to the "Pin Name".

Example:

```
-----  
[ICM Sparameter]  
File_name anyname.any_extension  
Port_map  
|Pin      Port A  Port B  
  A1       1      2  
  A2       3      4  
  A3       5      6  
  A4       7      8  
  B1       9     10  
  B2      11     12  
  B3      13     14  
  B4      15     16
```

=====
Keyword: [End ICM Section]

Required: Yes

Argument: Text String

Used By: [Begin ICM Section]

Description: Indicates the end of [Begin ICM Section]. This keyword is after the [Derivation Method], [Resistance Matrix], [Inductance Matrix], [Conductance Matrix], [Capacitance Matrix], [Bandwidth], [Row], [ICM Sparameter].

Matrix Examples:

=====
The resistance matrix for this package does not have coupling:

```
[Begin ICM Section] ExampleMatrix00
[Derivation Method] Distributed
[Resistance Matrix] Diagonal_matrix
10.0
15.0
15.0
10.0
10.0
15.0
15.0
10.0
[End ICM Section] ExampleMatrix00
```

```
[Begin ICM Section] ExampleMatrix01
```

The resistance matrix for this package does not have coupling and is exactly the same as the Diagonal_matrix shown above and is described using a trivial version of the Banded_matrix as an example only. The Diagonal_matrix should be used for this example to conserve file size.

```
[Resistance Matrix] Banded_matrix
[Bandwidth] 0
[Row] 1
10.0
[Row] 2
15.0
[Row] 3
15.0
[Row] 4
10.0
[Row] 5
10.0
[Row] 6
15.0
[Row] 7
15.0
[Row] 8
10.0
```

The inductance matrix has coupling between pins:

```
[Inductance Matrix] Full_matrix
[Row] 1
3.04859e-07    4.73185e-08    1.3428e-08    6.12191e-09
1.74022e-07    7.35469e-08    2.73201e-08    1.33807e-08
[Row] 2
3.04859e-07    4.73185e-08    1.3428e-08    7.35469e-08
1.74022e-07    7.35469e-08    2.73201e-08
[Row] 3
3.04859e-07    4.73185e-08    2.73201e-08    7.35469e-08
1.74022e-07    7.35469e-08
[Row] 4
3.04859e-07    1.33807e-08    2.73201e-08    7.35469e-08
1.74022e-07
[Row] 5
4.70049e-07    1.43791e-07    5.75805e-08    2.95088e-08
[Row] 6
4.70049e-07    1.43791e-07    5.75805e-08
[Row] 7
```



```
4.70049e-07      1.43791e-07
[Row]      8
4.70049e-07
```

The capacitance matrix has sparse coupling:

```
[Capacitance Matrix] Sparse_matrix
[Row]      1
1          2.48227e-10
2          -1.56651e-11
5          -9.54158e-11
6          -7.15684e-12
[Row]      2
2          2.51798e-10
3          -1.56552e-11
5          -6.85199e-12
6          -9.0486e-11
7          -6.82003e-12
[Row]      3
3          2.51798e-10
4          -1.56651e-11
6          -6.82003e-12
7          -9.0486e-11
8          -6.85199e-12
[Row]      4
4          2.48227e-10
7          -7.15684e-12
8          -9.54158e-11
[Row]      5
5          1.73542e-10
6          -3.38247e-11
[Row]      6
6          1.86833e-10
7          -3.27226e-11
[Row]      7
7          1.86833e-10
8          -3.38247e-11
[Row]      8
8          1.73542e-10
```

```
[End ICM Section] ExampleMatrix01
```

```
=====
```

9.0 End Keyword

```
=====
```

```
Keyword: [End]
Required: Yes
Description: Specifies end of file. Any text after the [End] keyword is
            ignored.
```

```
-----
[End]
```