ICM

(IBIS Interconnect Modeling Specification)

Version 1.0

Ratified September 2003

```
==============================================================================
==============================================================================
IBIS Interconnect Modeling Specification (ICM) Version 1.0 (Sept. 12, 2003)

ICM is a standard for behavioral descriptions of interconnect electrical
characteristics.
```

```
==============================================================================
==============================================================================

                      T A B L E   O F   C O N T E N T S

==============================================================================
==============================================================================
```

==

Section 1

GENERAL INTRODUCTION

==

This document contains a specification for general purpose 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.

Section 2

STATEMENT OF INTENT

This document is intended to establish a standard method for representing interconnect modeling data for transfer between interconnect design and simulation tools. The text describes a consistent format that can be parsed by software, allowing simulation tool and interconnect vendors to create models compatible with their own products.

One goal of this specification is to incorporate the IBIS format and data representation while allowing future growth to include more complex model structures and modeling methods. This will be accomplished through revisions of the base specification to include the addition of new keywords, subparameters and the like.

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

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

===

Section 3

GENERAL SYNTAX RULES AND GUIDELINES

===

1) Only ASCII characters, as defined in ANSI Standard X3.4-1986, may be used in an IBIS Interconnect Model (ICM) 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. The use of tab characters is legal, but they should be avoided as much as possible. This is to eliminate possible complications that might arise in situations when tab characters are automatically converted to multiple spaces by text editing, file transferring and similar software. In cases like that, lines might become longer than 120 characters, which is illegal in ICM files.

2) Except for keywords (phrases enclosed by a set of square brackets []), the content of an 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 four types: either a text string, a file name, a numeric value 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 after a keyword on the same line, may extend over multiple lines (i.e., line feeds and/or carriage returns may be part of a text block), 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 ICM 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 ICM 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 ICM 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 ICM 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 ICM 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.

13) Unless explicitly overridden by the subparameter description, a single numeric argument is separated from its associated subparameter by an equals sign (=); white space around the equals sign is optional. A set of numeric values (such as typ, min and max values) is separated from its associated subparameter by white space, without an equals sign. Symbolic (text) arguments are separated from their associated subparameter by one or more white spaces.

14) Note that the text here, in many cases, assumes that the model data describes a connector. However, other types of interconnect can be modeled using the ICM format; in these cases, the words "pin" and "port" in the text below can be interpreted to refer to the locations where the interconnect is accessed for probing, measurement or analysis.

```
=============================================================================
```

Section 4

KEYWORD TREE DIAGRAM

```
=============================================================================
```

```
/-- Start of File
|   -------------
|
|--/--[Begin Header]
|  |  |--[ICM Ver]
|  |  |--[File Name]
|  |  |--[File Rev]
|  |  |--[Date]
|  |  |--[Source]
|  |  |--[Notes]
|  |  |--[Disclaimer]
|  |  |--[Copyright]
|  |  |--[Support]
|  |  |--[Redistribution]
|  |  |--[Redistribution Text]
|  \--[End Header]
|
|--[Comment Char]*
|
|--/--[Begin ICM Family]                  | (see 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
|  |  |  |--[Nodal Path Description]
|  |  |  |     Model_nodemap
|  |  |  |     N_section
|  |  |  |--[ICM Swath Parameters]        | (optional, for swaths)
|  |  |  |     Left_edge
|  |  |  |     Right_edge
|  |  |  |     Top_edge
|  |  |  |     Bottom_edge
|  |  |  |--[ICM Swath Pin Numbers]       | (optional, for swaths)
|  |  \--[End ICM Model]
|  |
|  |--[ICM Pin Map]                       | Pins to Physical
|  |     Pin_order
|  |     Num_of_columns
|  |     Num_of_rows
|  |     Pin_list
|  |--[ICM Node Map]                      | Nodes to Physical
|  \--[End ICM Family]
|
|
|--/--[Begin ICM Section]                 | (see Chapter 8)
|  |  |--[Derivation Method]
|  |  |--[Resistance Matrix]
|  |  |    /-- [Bandwidth]
|  |  |    \-- [Row]
|  |  |--[Inductance Matrix]
|  |  |    /-- [Bandwidth]
|  |  |    \-- [Row]
|  |  |--[Conductance Matrix]
|  |  |    /-- [Bandwidth]
|  |  |    \-- [Row]
|  |  |--[Capacitance Matrix]
|  |  |    /-- [Bandwidth]
|  |  |    \-- [Row]
|  |  |
|  |  |--[ICM S-parameter]
|  |  |     File_name
|  |  |     Port_assignment
|  |  |
|  \--[End ICM Section]
|
|
\--[End]
```

* [Comment Char] is a global keyword and may be placed anywhere after the [ICM Ver] keyword and before the [End] keyword.

```
==============================================================================
```

Section 5

HEADER KEYWORDS

```
==============================================================================

     Keyword: [Begin Header]
    Required: Yes
        Uses: [ICM Ver], [File Name], [File Rev], [Date], [Source], [Notes],
              [Disclaimer], [Copyright], [Support], [Redistribution],
              [Redistribution Text], [End Header]
 Description: This keyword denotes the beginning of the file header section.
 Usage Rules: The [Begin Header] keyword may appear only once per ICM file
              and must be the first keyword in the file.  This keyword may be
              preceded by lines of general text and/or comments which should
              be treated as comments by the parser.
------------------------------------------------------------------------------
[Begin Header]

==============================================================================
     Keyword: [ICM 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: Only one [ICM Ver] keyword is allowed per ICM file, which must
              appear directly after the [Begin Header] keyword.

------------------------------------------------------------------------------
[ICM Ver]  1.0

==============================================================================
     Keyword: [File Name]
    Required: Yes
    Argument: Text String (file name)
     Used by: [Begin Header]
 Description: Specifies the name of the 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 [Date] keyword argument is limited to a
              maximum of 40 characters, and the month should be spelled out
              for clarity.

              The [Date] keyword may appear only once between the [Begin
              Header]/[End Header] keyword pair.
 |----------------------------------------------------------------------------
[Date]        September 12, 2003         | 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.

              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.
              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.

              The [Copyright] keyword may appear only once between the [Begin
              Header]/[End Header] keyword pair.
|-----------------------------------------------------------------------------
[Copyright]   Copyright 2003, 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 ICM header section
 Usage Rules: Only one [End Header] keyword is allowed per file, which must
              be the last keyword in any ICM header section.
------------------------------------------------------------------------------
[End Header]
```

Section 6

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] may appear anywhere in a file after the
              [ICM Ver] keyword, except within a text block (table of
              numerical values, etc.).  A file may contain multiple
              [Comment Char] keywords.
-----------------------------------------------------------------------------
[Comment Char]  |_char
```

```
=============================================================================
```

Section 7

INTERCONNECT MODEL FAMILY KEYWORDS

```
=============================================================================
     Keyword: [Begin ICM Family]
    Required: Yes
    Argument: Text String
        Uses: [Manufacturer], [ICM Family Description], [ICM Model List],
              [Begin ICM Model], [End ICM Model], [End ICM 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 Family] keyword is allowed per file, and
              this keyword may appear only after the [End Header] keyword.
              White space, as described in Section 3.0 above, is permitted in
              the family name.

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

=============================================================================
     Keyword: [Manufacturer]
    Required: Yes
    Argument: Text String
     Used By: [Begin ICM 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.

              Only one [Manufacturer] keyword is allowed per file, and this
              keyword may appear only between the [Begin ICM Family] and
              [ICM Model List] keywords.
-----------------------------------------------------------------------------
[Manufacturer]  XYZ Incorporated

=============================================================================
     Keyword: [ICM Family Description]
    Required: Yes
    Argument: Text Block
     Used By: [Begin ICM Family]
 Description: Provides a human-readable description of this interconnect
              family.
 Usage Rules: Only one [ICM Family Description] keyword is allowed per file,
              and this keyword may appear only between the
              [Begin ICM Model Family] and [ICM Model List] 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 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 Family]/[End ICM Family] keyword pair, and must be
              placed before the [Begin ICM Model] keyword.

              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 package 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, edgecard,
                              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.

```
|----------------------------------------------------------------------------
|
[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 Family]

Uses: [ICM Model Description], [Tree Path Description], [Nodal Path Description], [ICM Swath Parameters], [ICM Swath Pin Numbers], [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] keyword may appear only between the [Begin ICM Family]/[End ICM Family] keyword pair, and must be placed after the [ICM Model List] 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_even_mode", "SLM_odd_mode", "MLM", and "S-parameter".

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.

"SLM_even_mode" indicates that all other neighboring conductors are transitioning at the same time and with the same polarity (i.e. low-to-high or high-to-low) as the conductor being modeled (i.e. under even mode switching conditions). "SLM_odd_mode" indicates that all other neighboring conductors are transitioning at the same time as but with opposite polarity to 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, the signal-to-ground 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 row must contain only a diagonal term. SLMs can be used to evaluate for propagation delay, bandwidth, and impedance where ideal ground can be assumed. SLMs should not be used when lossy return path contributions are required for simulations.

"SLM_quiescent", "SLM_even_mode" and "SLM_odd_mode" 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_matix 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 data is contained in a secondary file in "Touchstone" format (see [ICM S-Parameter] below).

Note that, for the purposes of this document, multi-line models do not have a specified return path. Single-line models always exclude coupling and have some predefined return path implied in the model. Single-line model S-parameter descriptions are not permitted under this specification.

Example: ICM_model_type SLM_quiescent

SGR

If the ICM_model_type subparameter value is SLM_general, then the 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 by an non-negative, non-zero integer number of signal pins (1, 2, etc.), followed by a colon (:), followed by a non-negative, non-zero integer number of ground pins. 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. Similarly, if the data for a 22 pin connector was taken with 7 of the pins grounded, the signal to ground ratio would be listed as 22:7.

White space, as described in Section 3.0 above, is not permitted between the SGR integers and the separating colon (:).

Both the numerator and the denominator values should be as small as possible without inaccurately representing the signal-to-ground ratio.

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 "Touchstone" model file (see [ICM S-Parameter] below).

=============================================================================
     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: The [ICM Model Description] 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 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.
Usage Rules:  A [Tree Path Description] is used when all of the sections in a
              given interconnect have one-to-one mapping.  In other words,
              each pin is electrically connected to at least one other pin of
              the interconnect through a conductor.  Topologies where physical
              crossing of one conductor over or under another occurs shall not
              be described using [Tree Path Description].

              The following example is considered legal under a
              [Tree Path Description]:


                Pin1 >--------------------------------< Pin2

                Pin3 >--------------------------------< Pin4


              The following example is considered illegal under a
              [Tree Path Description]:


                Pin1 >------------  ------------------< Pin2
                                  \/
                                  /\
                Pin3 >------------  ------------------< Pin4
```

For such a structure, a [Nodal Path Description] (see below) would be more appropriate. Alternately, the pin names can be rearranged to preserve the conductor-to-pin mapping while avoiding physical crossing of conductors, as shown in the following example.

```
  Pin1 >---------------------------------< Pin4

  Pin3 >---------------------------------< Pin2
```

These "straight-through" connections allow mapping of conductors to RLGC matrix data by reference to the pins at the endpoints of the individual conductors (the first conductor is represented by the first row of RLGC matrix data, the second conductor by the second row of matrix data, and so on). 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, all RLGC matrixes must have the same number of diagonal terms, regardless whether they are swath or full matrices. Further, all diagonal terms must be in the same order.

S-parameter data may not be used in combination with the [Tree Path Description] keyword.

The [Tree Path Description] 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.

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.

The [Tree Path Description] keyword line is followed by at least one Model_pinmap and one Section subparameters which describe the pin-to-conductor mapping, and the topology of the interconnect. The electrical properties of an interconnect section are described in the matrices under the [Begin ICM Section] keyword which are referenced by the Section subparameter(s), discussed below. Branches or "T" connections are indicated by the "Fork" and "Endfork" subparameters. Tree path descriptions are terminated with 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 can also be 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 sub-parameter itself are separated by whitespace. There can be as many Model_pinmap subparameters as required, 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.

Section

Following the opening Model_pinmap subparameter is one or more instances of the required subparameter Section. This sub-parameter defines which 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 multiplier factor specifies how many times the section should be repeated in series (cascaded) by the EDA tool (the node names therefore represent the ends of the cascaded group of sections). The value of the 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 any positive number (see number format rules in Section 3.0). 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 section must have a value of "Lumped" else it is an error. If "Len=" is used the [Derivation Method] keyword for the corresponding 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.

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 fork/stub while Endfork marks the end of the fork/stub. These subparameters do not have any arguments.

At a fork, a path split similar in shape to the letter "T" is created. The matrix sections constituting the "T" are connected, including the first of any sections listed between the Fork and Endfork subparameters. The sections (zero or more) between the Fork and Endfork subparameters are connected together as in any other path description. If no pin map name is placed between a Fork and Endfork block, then the block 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

```
        SectA
Port1 >--------< Port2

[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 conductor, 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 conductor, 2 series sections with one stub

        SectionA          SectionB
Port1 >------------< >--------------< Port2
                    |
             StubSection1

[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 conductor with 2 stub sections; one stub matrix is
             re-used, and another stub is repeated

           A            B            C
Port1 >----------< >-----------< >-----------< Port2
                  |stub1        |stub4
                  |stub2        |stub4
                  |stub1

[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 conductor, 3 stubs

         A        B        C         D
Port1 >------< >------< >------< >-------< Port2
              |Stub1   |Stub2   |Stub1
                       |Stub3
                       |Stub4

[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 conductor, 2 stubs

         A        B        C         D
Port1 >------< >------< >------< >-------< Port 2
              |Stub1   |Stub2   |Stub1
                       |Stub3
                       |Stub4
                       |Port3

[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 conductor, 5 sections straight through

       SectionA     SectionB    SectionC     SectionD     SectionE
Port1 >----------< >----------< >---------< >----------<|>------------< Port2
                                                        |  SectionF
                                                        |>------------< Port3

[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. Also note that the use of both RLGC and S-parameter matrix data for sections within the same [Nodal Path Description] is prohibited.

Usage Rules: The [Nodal Path Description] 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.

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 (see example below).

```
[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]
```

In the example above, nodes A1 through A9 must map to rows 1 through 9 respectively of the [Resistance Matrix], [Capacitance Matrix], etc. within the section model "RLGC_matrix_9x9".

Nodes are explicitly mapped to S-parameter matrices through the [ICM S-parameter] keyword (see below).

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 "multiplier" argument contains a single numeric value which allows a matrix section to be re-used as a scaled value. This multiplier factor specifies how many times the section should be repeated in series (cascaded) by the EDA tool (the node names therefore represent the ends of the cascaded group of sections). The value of the 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 any positive number (see number format rules in Section 3.0). 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 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 [ICM 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 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 [Tree Path Description] keyword due to its regularity and the one to one mapping.

```
  1   2   3   4   5
    6   7   8   9       Male side
  | | | | | | | | |
  | | | | | | | | |
  | | | | | | | | |
  1   2   3   4   5     Female side
    6   7   8   9
```

```
[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.

```
  1   2   3   4   5
    6   7   8   9       Male side A
  | | | | | | | | |
  | | | | | | | | |
  | | | | | | | | |
  5   4   3   2   1     Male side B
    9   8   7   6
```

```
[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 6-pin female mini DIN to 5-pin male DIN adapter. The model consist of three sections.

```
    1 2 3 4 5 6 shell   PS2 6-pin female mini DIN
    | | | | | |    |     Section for connector pins
    |   | | |      |     Section for wiring
    | | | | |      |     Section for connector pins
    2 3 4 5 1   shell   5-pin male DIN
```

```
[Begin ICM Model] PS2_6_DIN_5
ICM_model_type MLM
| A drawing of this adapter can be found on the following web site:
| http://www.grand-tek.com/SPEC-7/GM06F-R05MM0-04.pdf
|
[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 consists of only one section for simplicity, but includes connections between the shells (not shown on the drawing).

```
    1 2  3   4  5 6 shell       6-pin male mini DIN (computer)
    | | +-+ +-+ | |   +---+
    | | | | | | | |   |   |     Section
    2 | 3 | 4 | 6 | shell |     6-pin female DIN (keyboard)
      2   3   4   6     shell   6-pin female DIN (mouse)
```

```
[Begin ICM Model] PS2_splitter
ICM_model_type MLM
| The specifications for this adapter were found on the following web site:
| http://www.hardwarebook.net/adapter/userinput/ps2keyboardygateway.html
[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.

```
         SectA     SectB     SectC     SectD     SectE
Port_A >-------< >-------< >-------< >-------< >-------< Port_B
```

```
[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.

          SectionA         SectionB
Port_A1 >----------<----->----------< Port_B1
Port_A2 >----------<---o->----------< Port_B2
Port_A3 >----------<---+->----------< Port_B3
Port_A4 >----------<-o-+->----------< Port_B4
Port_A5 >----------<-+-+->----------< Port_B5
                     | |
                     | |
                StubSection1

[Begin ICM Model] MyModelExample6
ICM_model_type MLM
[Nodal Path Description]
Model_nodemap Port_A
  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
Model_nodemap Port_B
[End ICM Model]
-----------------------------------------------------------------------------
Example 7:

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

            A           B           C
Port_A >---------< >---------< >---------< Port_B
                  |stub1      |stub4
                  |stub2      |stub4
                  |stub1
                  |Port_C

[Begin ICM Model] MyModelExample7
ICM_model_type MLM
[Nodal Path Description]
Model_nodemap Port_A
  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
Model_nodemap Port_C
  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
Model_nodemap Port_B
[End ICM Model]
```

===

Keyword: [ICM Swath Parameters]

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

Sub-Params: Left_edge, Right_edge, Top_edge, Bottom_edge

Used By: [Begin ICM Model]
Uses: [ICM Swath Pin Numbers]
Description: Describes a smaller matrix which will be expanded by EDA tools to represent interconnects of equal or larger size.
Usage Rules: The [ICM Swath Parameters] keyword is required when the coupling matrix that describes the interconnect has been compacted to represent only the edges and center of the full connector. This compacted set of data is referred to as a 'swath matrix'. A swath matrix can be expanded to represent a larger interconnect. To visualize this, consider the swath matrix as being a smaller version of a connector. The edges of the swath matrix duplicate the coupling effects that exist along the edges of the full size connector. Similarly, the center area of the swath matrix duplicate the coupling effects present in the center of the full size connector.

Only rectangular interconnects without missing conductors may be represented by a swath matrix. If the interconnect is non-rectangular or pins within the pin field are missing then the supplied matrix data (under the [Resistance Matrix], etc. keywords) must completely describe the interconnect and the [ICM Swath Parameters] 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.

The [ICM Swath Parameters] 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.

Edge Effect subparameters:

Edge effects are defined as non-symmetrical coupling between pins due to their proximity to the edge of the connector. Four subparameters describe how far into the body of the connector these edge effects extend. All four subparameters are required and may appear in any order.

1) Left_edge: Number of columns in the swath matrix used to represent left edge coupling effects
2) Right_edge: Number of columns in the swath matrix used to represent right edge coupling effects
3) Top_edge: Number of rows in the swath matrix used to represent top edge coupling effects
4) Bottom_edge: Number of rows in the swath matrix used to represent bottom edge coupling effects

Each subparameter is followed by a single integer argument. If the swath matrix has fewer columns but the same number of rows as the physical interconnect then the row edge effect terms are 0. Similarly, if the swath describes a section with fewer rows but the same number of columns as the physical interconnect then the column edge effect terms are 0. Note however that at least two subparameters must contain non-zero values. None of the argument values may exceed the sum of the number of rows or columns present in the corresponding model's matrix data.

Expansion of Swath Matrices:

Some explanation of edge effects is needed before swath expansion techniques can be clearly understood. Several illustrative examples are given in the discussion below.

Consider pin 'C' near the center of a connector with uniform pin spacing. Because the pin field surrounding pin 'C' is symmetric about pin 'C', the electromagnetic field surrounding pin 'C' will also be symmetric. Therefore, the coupling between pin 'C' and any two equidistant pins will be identical. For example, a pin in the first column to the right of pin 'C' will have the same coupling to pin 'C' as the corresponding pin in the first column to the left. Likewise, pins at equal distances above and below pin 'C' will have identical coupling to pin 'C'.

Consider next pin 'B' near the edge of the connector. Because the pin field is no longer symmetrical around pin 'B', the coupling to all nearby pins will not be same. For example, if pin 'B' is close to the right edge of the connector it may more strongly couple to the pins on its left. This is the edge effect. As pins towards the center of the pin field are considered the edge effects become diminished until the coupling is again symmetrical about the pin.

The model maker must use the available information on the extent of the edge effects when specifying the minimum number of columns (for a column swath) or rows (for a row swath) to be used to define interconnect swath. The model maker then assigns those values to the corresponding subparameters for the swath.

Shown below is a 3 row by 5 column swath, as viewed from one end. The conductors are represented by asterisks (*) and connecting lines (/, | and - characters) represent the electrical coupling.

```
        A  B  C  D  E
        *--*--*--*--*
        |\/|\/|\/|\/|
        |/\|/\|/\|/\|
        *--*--*--*--*
        |\/|\/|\/|\/|
        |/\|/\|/\|/\|
        *--*--*--*--*
```

Notice that columns B, C, and D have the maximum amount of coupling to conductors on each side. However, columns A and E are only coupled to conductors on one sides.

For the purposes of the expansion methods discussed below, a 3 row by 5 column swath and a 3 row by 9 column full-sized interconnect are used.

Application of Swath Matrices:

Three methods can be used to expand a swath matrix into a full-sized interconnect. Note that these methods are provided for informational purposes only; while EDA tools are encouraged to implement one or more of the approaches described, they are free to implement others not listed here:

(1) Centering the Swath around the pins of interest
(2) Expansion and Centering: Expanding the Swath matrix into a larger sized Swath matrix and centering it about the paths of interest.
(3) Expansion to Full Sized Interconnect: Expanding the swath matrix to the full sized interconnect matrix.

These methods are described in detail below.

(1) Centering the Swath around the pins of interest

```
           A  B  C  D  E
           *--*--*--*--*
           |\/|\/|\/|\/|
           |/\|/\|/\|/\|  Swath
           *--*--*--*--*
           |\/|\/|\/|\/|
           |/\|/\|/\|/\|
           *--*--*--*--*

        *--*--*--*--*--*--*--*--*
        |\/|\/|\/|\/|\/|\/|\/|\/|
        |/\|/\|/\|/\|/\|/\|/\|/\|
        *--*--*--*--*--*--*--*--*   Full Interconnect
        |\/|\/|\/|\/|\/|\/|\/|\/|
        |/\|/\|/\|/\|/\|/\|/\|/\|
        *--*--*--*--*--*--*--*--*
        1  2  3  4  5  6  7  8  9
```

Centering is a way to map the swath matrix pins directly into the corresponding pins of the actual interconnect. For example, the swath above is centered about column 4 of the full sized interconnect. 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 by the corresponding swath columns (A & E, respectively).

Note that the swath data is not expanded under this method; the swath simply represents a small portion of the larger interconnect. EDA tools should not infer data regarding the other columns in the full interconnect in this case.

When the signals of interest are at the left or right edges, the swath would normally be positioned to "cover" those edges. For example, column A of the swath matrix may be "lined up" with column 1 of the interconnect.

However, columns 6 through 9 of the full interconnect would not be described at all, nor should their behavior be inferred by EDA tools using this swath data.

(2) Expansion and Centering: Expanding the swath matrix into a larger swath matrix, centering both swaths about the paths of interest.

Note that 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:

There are three steps to this procedure (a, b, and c below).

(a) Copy the left columns specified by the edge effect subparameters

```
A  B  C  D  E
*--*--*--*--*
|\/|\/|\/|\/|
|/\|/\|/\|/\|   Swath
*--*--*--*--*
|\/|\/|\/|\/|
|/\|/\|/\|/\|
*--*--*--*--*

*--*--*--*--*--*--*--*--*
|\/|\/|\/|\/|\/|\/|\/|\/|
|/\|/\|/\|/\|/\|/\|/\|/\|
*--*--*--*--*--*--*--*--*   Full Interconnect
|\/|\/|\/|\/|\/|\/|\/|\/|
|/\|/\|/\|/\|/\|/\|/\|/\|
*--*--*--*--*--*--*--*--*
1  2  3  4  5  6  7  8  9
A  B                        Mapping of swath data
```

In this case, columns A and B of the swath data become columns 1 and 2 of the expanded interconnect.

(b) Repeat the center pattern

```
A  B  C  D  E
*--*--*--*--*
|\/|\/|\/|\/|
|/\|/\|/\|/\|   Swath
*--*--*--*--*
|\/|\/|\/|\/|
|/\|/\|/\|/\|
*--*--*--*--*

*--*--*--*--*--*--*--*--*
|\/|\/|\/|\/|\/|\/|\/|\/|
|/\|/\|/\|/\|/\|/\|/\|/\|
*--*--*--*--*--*--*--*--*   Full Interconnect
|\/|\/|\/|\/|\/|\/|\/|\/|
|/\|/\|/\|/\|/\|/\|/\|/\|
*--*--*--*--*--*--*--*--*
1  2  3  4  5  6  7  8  9
A  B  C  C  C  C  C         Mapping of the swath data
```

Starting at the center of the swath, the coupling pattern of just the center of the swath (column C in this example) is copied into the full interconnect matrix being constructed by the EDA tool. In this case, the data for the conductors in column C, lacking edge effects, may be used to represent several conductors near the center of the expanded matrix.

(c) Copy the right columns as specified by the edge effect subparameters

```
            A  B  C  D  E
            *--*--*--*--*
            |\/|\/|\/|\/|
            |/\|/\|/\|/\|   Swath
            *--*--*--*--*
            |\/|\/|\/|\/|
            |/\|/\|/\|/\|
            *--*--*--*--*

*--*--*--*--*--*--*--*--*
|\/|\/|\/|\/|\/|\/|\/|\/|
|/\|/\|/\|/\|/\|/\|/\|/\|
*--*--*--*--*--*--*--*--*   Full Interconnect
|\/|\/|\/|\/|\/|\/|\/|\/|
|/\|/\|/\|/\|/\|/\|/\|/\|
*--*--*--*--*--*--*--*--*
1  2  3  4  5  6  7  8  9
A  B  C  C  C  C  C  D  E   Mapping of the swath data
```

The final step is similar to step (a) above, but applied using the right side of the swath. When the swath is positioned against the right edge of the full interconnect as shown, the data for the rightmost swath columns is copied into the rightmost columns of the full interconnect. In this example, the data for swath columns D and E are placed in the full interconnect in columns 8 and 9.

The process is similar for expanding based on rows rather than columns.

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.

While this document presents three possible methods for swath expansion, there is no formal specification or convention in the industry for expanding a swath matrix. Swath expansion is somewhat arbitrary and depends on the number of row or columns in the swath, the physical layout of the connector and tool-specific processing needs. It is strongly recommended that EDA tool documentation be provided to the user explaining the method(s) used for swath matrix expansion.

```
-----------------------------------------------------------------------------
[ICM Swath Parameters]
    Left_edge = 2
    Right_edge = 2
    Top_edge = 0
    Bottom_edge = 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 may appear multiple times in the file, however it may appear only once between each pair of [Begin ICM Model]/[End ICM Model] keywords.

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
physically arranged as shown below:

       C1   C2  C3
R1     A    B   C
R2     D    E   F

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

[ICM 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
Usage Rules: The [End ICM Model] keyword may appear only between the [Begin ICM Family]/[End ICM Family] keyword pair. Every [Begin ICM Model] keyword must have a corresponding [End ICM Model] keyword, which must be the last keyword of an ICM Model section.

```
-----------------------------------------------------------------------------
[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, Pin_list

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.

The [ICM Pin Map] keyword may appear multiple times in the file, however it may appear only between the last [End ICM Model] and before the [End ICM 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.

The 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 Unordered. The subparameter name and its argument are separated by one or more white spaces.

Note that the terms "row" and "column" for [ICM Pin Map] refer to the physical pins or conductors of the interconnect being described, not to the arrangement of matrix data.

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 Unordered 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. Unordered 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 required Pin_list subparameter is a list of the pin names covered by this pin map. The pin list is required. The pin names must be organized as specified by the Pin_order subparameter. All pins on a component must be specified.

Each pin is represented on a separate line by two columns of data. The first column must contain the pin name. The second column gives the data book name for the signal on that pin. If there is no data book name, then an arbitrary name such as SIG may be used.

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 pin and signal names of a six pin connector were listed as follows:

```
A1  AD21
A2  STOP#
A3  AD2
B1  PERR
B2  GNT0#
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.

Blank or white-space signal names are prohibited.

Other notes: More than one [ICM Pin Map] keyword is allowed in an ICM 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.

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

|Examples 1 through 5 below use the connector represented by:
|A1 A2 A3 A4
|B1 B2 B3 B4

[ICM Pin Map] Example1_pinmap
Pin_order Row_ordered
Num_of_columns = 4
Num_of_rows = 2
Pin_list
|Pin Name
A1 AD2
A2 AD5
A3 AD7
A4 GND
B1 STOP#
B2 PAR
B3 PERR
B4 GND

[ICM Pin Map] MyModelPinMapA
Pin_order Column_ordered
Num_of_columns = 4
Num_of_rows = 2
Pin_list
|Pin Name
A4 AD12
B4 AD15
A3 AD7
B3 PERR
A2 AD5
B2 PAR
A1 AD2
B1 STOP#
```

```
[ICM Pin Map] MyModelPinMapB
Pin_order Column_ordered
Num_of_columns = 4
Num_of_rows = 2
Pin_list
|Pin Name
A1 AD2
B1 STOP#
A2 AD5
B2 PAR
A3 AD7
B3 PERR
A4 AD12
B4 AD15

[ICM Pin Map] MyModelPinMapC
Pin_order Column_ordered
Num_of_columns = 4
Num_of_rows = 2
Pin_list
|Pin Name
A1 AD2
B1 STOP#
A2 AD5
B2 PAR
A3 AD7
B3 PERR
A4 AD12
B4 AD15
```

```
=============================================================================
     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.

              The [ICM Node Map] keyword may appear multiple times in the
              file, however it may appear only between the last
              [End ICM Model] and before the [End ICM Family] keywords.

              The [ICM Node Map] line is followed by a three-column table.
              The first column contains the list of pin 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. The third column contains the data book name for the signal on that pin.

Note that the first column (i.e., the pin names) under 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.

Blank or white-space pin and signal names are prohibited.

Other Notes: More than one [ICM Node Map] keyword is allowed in an ICM 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     SIG1
    2     M2     SIG2
    3     M3     SIG3
    4     M4     SIG4
    5     M5     SIG5
    6     M6     SIG6
    7     M7     SIG7
    8     M8     SIG8
    9     M9     SIG9
[ICM Node Map] DB9_Female_side
|  pin    node   name
    1     F1     SIG1
    2     F2     SIG2
    3     F3     SIG3
    4     F4     SIG4
    5     F5     SIG5
    6     F6     SIG6
    7     F7     SIG7
    8     F8     SIG8
    9     F9     SIG9
```

```
-----------------------------------------------------------------------------
Example 2:

[ICM Node Map] DB9_side_A
|  pin    node   name
    1     A1     SIG1
    2     A2     SIG2
    3     A3     SIG3
    4     A4     SIG4
    5     A5     SIG5
    6     A6     SIG6
    7     A7     SIG7
    8     A8     SIG8
    9     A9     SIG9
[ICM Node Map] DB9_side_B
|  pin    node   name
    1     B1     SIG5
    2     B2     SIG4
    3     B3     SIG3
    4     B4     SIG2
    5     B5     SIG1
    6     B6     SIG9
    7     B7     SIG8
    8     B8     SIG7
    9     B9     SIG6
-----------------------------------------------------------------------------
Example 3:

[ICM Node Map] PS2_6_DIN_5_PS2_side
|  pin    node   name
    1     PS1    SIG1
    2     PS2    NC
    3     PS3    SIG3
    4     PS4    NEG_1_PS2
    5     PS5    SIG5
    6     PS6    NC
    shell PS7    SHELL
[ICM Node Map] PS2_6_DIN_5_DIN_side
|  pin    node   name
    1     DIN1   SIG5
    2     DIN2   SIG1
    3     DIN3   NC
    4     DIN4   SIG3
    5     DIN5   CONTROL1
    shell DIN7   SHELL
```

```
-----------------------------------------------------------------------------
Example 4:

[ICM Node Map] PS2_splitter_Computer_side
|  pin    node   name
    1     C1     SIG1
    2     C2     SIG2
    3     C3     SIG3
    4     C4     SIG4
    5     C5     SIG5
    6     C6     SIG6
    shell Cs     SHELL
[ICM Node Map] PS2_splitter_Keyboard_side
|  pin    node   name
    2     K2     SIG1
    3     K3     SIG3
    4     K4     SIG4
    6     K6     SIG6
    shell Ks     SHELL
[ICM Node Map] PS2_splitter_Mouse_side
|  pin    node   name
    2     M2     SIG2
    3     M3     SIG3
    4     M4     SIG4
    6     M6     SIG6
    shell Ms     SHELL
-----------------------------------------------------------------------------
Example 5:

[ICM Node Map] Port_A
|   pin         node   name
    PORT_A1     A1     POS_0
    PORT_A2     A2     NEG_0
    PORT_A3     A3     POS_1
    PORT_A4     A4     NEG_1
[ICM Node Map] Port_B
|   pin         node   name
    PORT_B1     B1     POS_0
    PORT_B2     B2     NEG_0
    PORT_B3     B3     POS_1
    PORT_B4     B4     NEG_1
-----------------------------------------------------------------------------
Example 6:

[ICM Node Map] Port_A
|   pin         node   name
    PORT_A1     A1     POS_0_A
    PORT_A2     A2     NEG_0_A
    PORT_A3     A3     POS_1_A
    PORT_A4     A4     NEG_1_A
    PORT_A5     A5     REF
[ICM Node Map] Port_B
|   pin         node   name
    PORT_B1     B1     POS_0_B
    PORT_B2     B2     NEG_0_B
    PORT_B3     B3     POS_1_B
    PORT_B4     B4     NEG_1_B
    PORT_B5     B5     REF
```

```
|-----------------------------------------------------------------------------
Example 7:

[ICM Node Map] Port_A
|   pin         node   name
    PORT_A1     A1     POS_0_A
    PORT_A2     A2     NEG_0_A
    PORT_A3     A3     POS_1_A
    PORT_A4     A4     NEG_1_A
    PORT_A5     A5     REF
[ICM Node Map] Port_B
|   pin         node   name
    PORT_B1     B1     POS_0_B
    PORT_B2     B2     NEG_0_B
    PORT_B3     B3     POS_1_B
    PORT_B4     B4     NEG_1_B
    PORT_B5     B5     REF
[ICM Node Map] Port_C
|   pin         node   name
    PORT_C1     C1     POS_0_C
    PORT_C2     C2     NEG_0_C
    PORT_C3     C3     POS_1_C
    PORT_C4     C4     NEG_1_C
    PORT_C5     C5     REF

==============================================================================
     Keyword: [End ICM Family]
    Required: Yes
     Used By: [Begin ICM Family]
 Description: Defines the end of the interconnect model family
 Usage Rules: Every [Begin ICM Family] keyword must have a corresponding
              [End ICM Family] keyword, which must be the last keyword of an
              ICM Model Family section.
------------------------------------------------------------------------------
[End ICM Family]
```

Section 8

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 S-parameter], [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 name must match one of the "Section" or "N_section" names used under the [Tree Path Description] or [Nodal Path Description] elsewhere in the ICM file.

The [Begin ICM Section] keyword may appear multiple times in the file, however it may appear only after the [End ICM 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 for different sections may vary (note that all matrices within a section must have the same number of diagonal terms).

If the swath method is used the matrix may have fewer conductors then 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.

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".

The [Derivation Method] keyword may appear multiple times in the file, however it may appear only once between the [Begin ICM Section]/[End ICM Section] keyword pair, and must be placed before any [*** Matrix] or [ICM S-parameter] keywords.

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 must 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 circuit reference.

The assumed circuit topology is RLGC 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]

Uses: [Row], [Bandwidth]

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 one of the SLM_* types, only a diagonal matrix is allowed.

The [Resistance Matrix], [Inductance Matrix], [Conductance Matrix], and [Capacitance Matrix] keywords may appear multiple times in the file, however they may appear only once between the [Begin ICM Section]/[End ICM Section] keyword pair, and they must be placed after the [Derivation Method] keyword.

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

Users should be aware that matrix data of value 0 represents an ideal short to reference ground and should not be associated with conductors which carry signal information (see below).

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, as no instances of the [Row] keyword are required.

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.

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

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.

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.

```
                   I1   +-----+   I2
                 -----> |     | <-----
                        |     |
         board o--------| ICM |--------o board
               V1       |     |       V2
                        |     |
                        +-----+
```

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/unit length 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/unit length.

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.

Diagonal elements may be dominant. Comparably, the off-diagonal elements may be negligible thus leading to off-diagonal element values of infinite value. Therefore, a "Diagonal_matrix" matrix type is the best choice for describing a resistance matrix.

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 ICM_model_type is SLM_* then all sections must use loop-inductance.

If ICM_model_type 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/unit length 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/unit length.

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


         Return ....................| Return
(side A)                            |           (side B)
         Signal ....................| Signal
                                    |
         Return ....................| Return
```

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 conductor given a specific return path.

The loop inductance method must be used if the ICM_model_type is SLM_*.

Loop inductance can be experimentally measured.

Loop inductance matrices can be created using field solvers.

An SGR value is normally specified for models of type SLM_* that indicates the ratio of signal pins to reference 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.

Models of type SLM_* 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; 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.

```
              section


         Signal .....................  Signal
(side A)                                 |       (side B)
         Signal .....................  Signal
                                         |
         Signal .....................  Signal
```

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.

Off-diagonal terms may be positive real numbers, negative real numbers or zero.

Capacitance Matrix Description

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 ICM_model_type is SLM_* then all sections must only contain a Single Line Capacitance Matrix.

If ICM_model_type 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]=[Cm][V] where [Cm] is the Maxwell Capacitance 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/unit length 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/unit length.

Off-diagonal terms must be negative real numbers or zero.

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/unit length 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/unit length.

Off-diagonal terms may be positive real numbers, negative real numbers or zero.

Loss Matrix Description:

The [ICM S-parameter] 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 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. The
              [Bandwidth] keyword may appear multiple times in the file,
              however it may be used only once per each set of [*** Matrix]
              keywords, and it must be placed before the matrix data is
              given).
------------------------------------------------------------------------------
              See example "Matrix Example" lower in this document.

=============================================================================
     Keyword: [Row]
    Required: Yes
    Argument: Numeric Value
     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 must be placed directly after the [Bandwidth]
              keyword.  The data entries for the row defined by this keyword
              begin on the line below the keyword and may span multiple lines.
------------------------------------------------------------------------------
 See example "Matrix Example" lower in this document.

==============================================================================
     Keyword: [ICM S-parameter]
    Required: No, unless ICM_model_type subparameter of [Begin ICM Model] is
              equal to "S-parameter".
  Sub-Params: File_name, Port_assignment
     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.

              The [ICM S-parameter] keyword may appear multiple times in the
```

file, however it may appear only once between the [Begin ICM Section]/[End ICM Section] keyword pair, and they must be placed after the [Derivation Method] keyword.

File_name

File_name refers to the name of an external file in "Touchstone" format containing S-parameter matrix data for this section. The Touchstone file must be located in the same directory as the ICM file which makes reference to it.

Port_assignment

"Port_assignment" is used to relate the ports in the S-parameter file to the node map for a particular N_Section as defined under the [Nodal Path Description] keyword. The "Port_assignment" table entries must match the node names from the corresponding [Nodal Path Description]. No more and no fewer entries in the table are allowed. Note that [ICM S-parameter] may not be used to describe data for sections in a [Tree Path Description].

There are two columns used in the Port_assignment. The first column should reference the input port name as defined in the Touchstone file that is related to the "Node Name" in the second column. The second column contains a "Node Name" as defined in the node list under the corresponding [Nodal Path Description].

```
------------------------------------------------------------------------------
[ICM S-parameter]
File_name sample.s4p     | any name and extension allowed
Port_assignment
|Port    Node
  1       A1
  2       B1
  3       A2
  4       B2
  5       A3
  6       B3
  7       A4
  8       B4

==============================================================================
```

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 S-parameter].

Usage Rules: Every [Begin ICM Section] keyword must have a corresponding [End ICM Section] keyword, which must be the last keyword of an ICM Section description.

```
-----------------------------------------------------------------------------
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]

[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]
```

===

Section 9

END KEYWORD

===

```
     Keyword: [End]
    Required: Yes
 Description: Specifies end of file.  Any text after the [End] keyword is
              ignored.
 Usage Rules: The [End] keyword marks the end of the interconnect model in an
              ICM file and must be the last ICM keyword in the file.
-----------------------------------------------------------------------------
[End]
```