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

Last Updated: 08-26-2001, AP

NEW ADDITIONS

Include "Swath Mapping" for SWATH MATRIX ONLY… Relate "physical location to
diagonal (I.e ROW) matrix order.

Updated Tree and Table of Contents

Deleted
Keyword: [Cn Col of Pins]
Keyword: [Cn Row of Pins]

ISSUES TO ADDRESS

List as a part of the introduction of the specification

Explicitly List the three cases and the associated keywords that go along with.
Specify Full Matrix
Specify Full Matrix and a swath
Specify a partial range to describe a connector family.

AP Action Item, Would this be covered by "Examples"??

* The [File Rev] keywords requires a Text String. Wouldn't it be better to
require specific types of field formats, so that this information could be
processed automatically?

* Still need to address "What information needs to stay with the specific
connector model…" especially   if the connector model is to be eventually
included in a larger file.

* Dual purpose use of the [Source] keyword. It is used to record the
manufacturer's name, and the place and method from whence the model was
created. This mixes two pieces of information that are both useful. I would
propose another keyword [Part Manufacturer] which can be used to contain the
name of the manufacturer of the part. This will make it very easy to search
for manufacturers in a database of connector models.

[Source] then becomes the source of the information for extraction, which might
also be some person or organization other than the connector manufacturer. (For
example, SiQual or Mentor.)

===
                    Specification Revision level 0.976
===
Table of Contents
===

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

==

 1.0 GENERAL INTRODUCTION

==

 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 connector keywords and usage rules.

==

 2.0 STATEMENT OF INTENT

==

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

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

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

 This template is meant to contain a complete description of the
 elements on an entire connector or family of connectors.

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

==

 3.0 GENERAL SYNTAX RULES AND GUIDELINES

==

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

characters and line termination sequences.
NOTE: the use of TAB characters is strongly discouraged.

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

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

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

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

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

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

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

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

9) Valid scaling factors are:
T = tera   k = kilo   n = nano
G = giga   m = milli   p = pico
M = mega   u = micro   f = femto
When no scaling factors are specified, the appropriate base units are assumed.
(These are volts, amperes, ohms, farads, henries, and seconds.) The parser
looks at only one alphabetic character after a numerical entry; therefore it is
enough to use only the prefixes to scale the parameters. However, for clarity,
it is allowed to use full abbreviations for the units, (e.g., pF, nH, mA,
mOhm). In addition, scientific notation IS allowed (e.g., 1.2345e-12).

10) All temperatures are represented in degrees Celsius.

11) Multiple definitions of a specific symbol are illegal. Specifically,

multiple definitions (either redundant or alternate) of an object type of
the same object class are illegal. Refer to section 8 for more information
on object classes and object types.

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

13) The following words are reserved words and must not be used for
any other purposes in the document:
    PWRGND- reserved model name, used with power or ground pins,
    POWER - reserved model name, used with power supply pins,
    GND   - reserved model name, used with ground pins,
    RET   - reserved word indicating either power or ground used
              as a high-frequency return path.
    NC   - reserved model name, used with no-connect pins,
    NA   - used where data not available.

===

 4.0 KEYWORD TREE DIAGRAM

===

/-- Start of File
| ----------------
| /-- [Begin Header]
| | |-- [Comment Char]
| | |-- [IBIS Cn Model Ver]
| | |-- [File Name]
| | |-- [File Rev]
| | |-- [Date]
| | |-- [Source]
| | |-- [Notes]
| | |-- [Disclaimer]
| | |-- [Copyright]
| | |-- [Support]
| | |-- [Redistribution]
| | |-- [Redistribution Text]
| \-- [End Header]
|
| /-- [Begin Cn Model Family] (Section 7)
| | |-- [Manufacturer]
| | |-- [Cn Family Description]
| |
| | -- [Cn Model List]
| |
| | /-- [Begin Cn Model]
| | | Cn_Model_Type
| | |
| | | -- [Path Description]
| | | Model_PinMap
| | | Cn_Fork, Cn_EndFork
| | | Cn_Section
| | |
| | |
| | | -- [Cn Model Description]
| | | -- [Cn Number of Conductors]
| | | -- [Cn Col Swath]
| | | -- [Cn Row Swath]

```
|  |     | -- {Swath Pin Numbers]
|  |     |
|  |     | -- /-- [Begin Cn Auto Map]
|  |     |    |-- [Cn Col Range]
|  |     |    |-- [Cn Row Range]
|  |     |    \-- [End Cn Auto Map]
|  |     |
|  |     | -- [Cn Pin Map]  |  Pins to Physical
|  |     | -- [Cn Phy Map]  |  Physical to Matrix (Physical = Col/Row location)
|  |     |
|  |     \-- [End Cn Model]
|  |
|  |
|  \-- [End Cn Model Family]
|
|
|  /-- [Begin Cn Section] | (Section 8)  Define Matrices
|  |-- [Derivation Method]
|  |-- [Resistance Matrix]
|  |         /-- [Bandwidth]
|  |         \-- [Row]
|  |-- [Inductance Matrix]
|  |          /-- [Bandwidth]
|  |          \-- [Row]
|  |-- [Capacitance Matrix]
|  |          /-- [Bandwidth]
|  |          \-- [Row]
|  |
|  \-- [End Cn Section]
|
|
\-- [End]
```

==

5.0 HEADER KEYWORDS

==

```
     Keyword: [Begin Header]
    Required: Yes
    Argument: None
 Description: This keyword denotes the beginning of the file header section.
 Usage Rules: [Begin Header] must be the first keyword in any ICM file.
              This keyword may be proceeded by lines of general text and/or
              comments which should be treated as comments by the parser.
------------------------------------------------------------------------------
[Begin Header]
```

==
```
     Keyword: [IBIS Cn Model Ver]
    Required: Yes
    Argument: Text String
 Description: This keyword allows electronic parsers to immediately determine
              that this file contains an IBIS connector 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: MUST BE THE FIRST KEYWORD in the file after {Begin Header].

------------------------------------------------------------------------------
```

```
[IBIS Cn Model Ver]  1.0


===============================================================================
    Keyword: [File Name]
   Required: Yes
   Argument: Text String (file name)
Description:  Specifies the name of the IBIS Connector Model (ICM) file.
Usage Rules:  The purpose of this keyword is to document a single filename
              the file uses even if the file is transferred from one
              computer to another.  The file name must conform to the rules
              given in section 4, "General Syntax Rules and Guidelines".
-------------------------------------------------------------------------------
[File Name]     iconm_hdi_202.icm


===============================================================================
    Keyword: [File Rev]
   Required: Yes
   Argument: Text String
 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     connector and/or file in development
              1.x     pre-manufacturing connector and/or model
              2.x     mature connector and or model, no more changes likely

-------------------------------------------------------------------------------
[File Rev]    1.0                       Used for .icm file versions


===============================================================================
    Keyword: [Date]
   Required: No
   Argument: Text String (Date)
 Description: Date this file was last modified.
 Usage Rules: This keyword is provided to insure the last changed date for
              this file is not lost if the file is transmitted between
              computer systems. The format is fixed.  The pattern to be used
              is "Month Day, Year".  Month is fully written out.  Day uses two
              digits.  Year uses four digits.  See example.

 ------------------------------------------------------------------------------
[Date]          July 1, 2000        | The latest file revision date


===============================================================================
    Keyword: [Source]
   Required:  No
   Argument:  Text Block
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 who
              created the model.


              NOTE: It is recommended that the argument to the [Source]
              keyword be limited to a maximum of 24 lines of text.
-------------------------------------------------------------------------------
[Source] Put originator and the source of information here. For example:
        Results from field simulation
```

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

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

              The [Notes] keyword can only be used once.

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


===============================================================================
     Keyword: [Disclaimer]
    Required: No
    Argument: Text Block
 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.

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


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

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

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


===============================================================================
     Keyword: [Support]
    Required:  No
    Argument:  Text Block (URL path name)
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
```

8

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.

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

```
============================================================================
```
      Keyword: [Redistribution]
     Required: Yes
     Argument: Text String (Yes, No, Specific)
  Description: Indicates to EDA tool companies and model users who may use and
               redistributed 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.

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

```
============================================================================
```
      Keyword: [Redistribution Text]
     Required: Yes, ONLY if [Redistribution] value is "Specific", otherwise it
               is optional
     Argument: Text block
  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.

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

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

```
=============================================================================
```
      Keyword: [End Header]
     Required: Yes
     Argument: None
  Description: Marks the end of an IBIS Connector header section.
  Usage Rules: [End Header] must be the last keyword in any IBIS Connector
               header section.

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


================================================================================

6.0 GLOBAL KEYWORDS

================================================================================

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

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

              If used, the [Comment Char] keyword can only appear after the
              Begin Header] and [IBIS-X Ver] keywords. A file may contain
              multiple [Comment Char] keywords
--------------------------------------------------------------------------------
[Comment Char]  |_char


================================================================================

7.0 CONNECTOR MODEL FAMILY KEYWORDS

================================================================================

Keywords: [Begin Cn Model Family]
    Required: Yes
    Argument: Text String
 Description: This keyword marks the beginning of the connector family
              description section.  A connector "family" consists of one or
              more similar connectors.  Note that each individual connector
              in a family will have a separate connector model.
 Usage Rules: Following the keyword is the name of the connector family
              Only one [Begin Cn Model Family]/[End Cn Model Family]
              keyword is allowed per file.

              Connector family descriptions are terminated by the occurrence
              of the [End Cn Model Family] keyword.

--------------------------------------------------------------------------------
[Begin Cn Model Family] None

================================================================================
     Keyword: [Manufacturer]
    Required: Yes
    Argument: Text String
 Description: This keyword gives the name of the manufacturer of the
              connector family being described in this file.
 Usage Rules: Following the keyword is the manufactures name.  Blank
              characters are allowed.
```

```
              NOTE: It is highly recommended that each model maker
                    uses the connector manufacture name in
                    all .icm files.
-------------------------------------------------------------------------------
[Manufacturer]  XYZ Corp.



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

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

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



===============================================================================
     Keywords: [Cn Model List]
     Required: Yes
     Argument: None
  Description: This keyword lists the model name, mating conditions, minimum
               slew time and picture link for each individual connector model
               described in this file.
  Usage Rules: 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 connector
               model contained in the file.  The model name must match one of
               the model names given as an argument to the [Begin Cn 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 connector are mated together
               for the model.
               UnMated_Side_A: A model of non-mated connector. Typically,
                               this would be the connector half that is
                               attached to a baseboard
               UnMated_Side_B: A model of non-mated connector. Typically,
                               this would be the connector half that is
                               attached to a daughter card, cable, or
                               edgecard.

               Note that using two unmated models in series DOES NOT provide a
               correct model of a mated connector.

               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 connector. Rather, this term is
used to define a usage limit of each model.

Note: It is strongly recommended that a 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 connector. The image file shall be
either a .jpg or .txt (ASCII) formatted picture. The connector
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 connector model file or stored in the same
compressed file with the connector to insure they files
remained 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 a connector.

Suggested .jpg image attributes:
  size = 160 horizontal by 120 vertical (pixels)
  color = 24 bit
  Image quality = 85%
Larger or smaller images may be used but are not
recommended. A standard image size will allow EDA tools to
provide uniform look and feel for images from all vendors.
The size was chosen to be a small disk file size and be
useful as either a thumb-nail in a model selection window
or scaled up to give a larger picture. The 160x120 was
chosen as a perfect 4:1 scale in both axis of a 640x480
picture.

```
|-----------------------------------------------------------------------------
|
[Begin Cn 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
```

==
    Keywords: [Begin Cn Model]
    Required: Yes
    Argument: Text String (Model name)
 Description: This keyword marks the beginning of each individual connector
             model. A connector model describes the electrical
             characteristics of the interconnect path between a pin or pins

on one side of the connector 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 a matrix
of L,R and C simulation data. Sections are generally cascaded
in series, but the Cn_fork and Cn_endfork sub-parameters allows
sections to branch off the main interconnect path as a stub or
tee.

Sub-params: Cn_Model_Type, Cn_GSR
Usage Rules: Following the keyword is a single argument specifying the
             name of this particular connector model. The name must be one
             of the model names listed under the 'model name' column of the
             preceding [Cn Model List] keyword. Note that for every model
             name listed by the [Cn Model List] keyword there must be a
             corresponding model as defined by a [Begin Cn Model]/[End Cn
             Model] keyword pair.

             Model Sub-Parameters:

             Cn_Model_Type

             Following the keyword line is a required sub-parameter
             Cn_Model_Type. This sub-parameter details the conditions and
             usage assumptions under which the connector electrical (L,R,C)
             data was gathered.

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

             The sub-parameter and its value are separated by a single
             Whitespace. There can be as many Cn_model_Type subparameters as
             required to represent a connectors, but each must be on a single
             line.

             A Cn_Model_Type sub-parameter value of SLM_Quiescent indicates
             that the model data was gathered with all other neighboring
             conductors non-switching and terminated by a reference impedance
             that approximates the characteristic impedance of the
             connector. A value of SLM_EvenMode indicates that all other
             neighboring conductors are transitioning at the same time and in
             the same direction as the conductor being modeled (i.e. under
             even mode switching conditions). A value of SLM_OddMode
             indicates that all other neighboring conductors are transitioning
             at the same time but in the opposite direction as the conductor
             being modeled (i.e. under odd mode switching conditions). A
             value of SLM_General indicates that the model data was gathered
             with all other neighboring conductors non-switching and the
             ground to signal ratio is specified by Cn_GSR 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 R/L/C matrix must contain only a single diagonal
             term. SLMs can be used to evaluate for Prop Delay, bandwidth,
             and impedance where ideal ground can be assumed. SLMs should not
             be used when lossy return paths contributions are required for
             simulations.

             SLM_Quiescent, SLM_EvenMode and SLM_OddMode have a defined
             reference or return path. The connection pins used as the return
             path should be specified as "GND" in the matrix. SLM_General
             does not have specified Return Path given in the matrix.

A Cn_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. MLM's should not be used in simulations
with perfect ground planes on both sides of the connector model.

Example: Cn_Model_type SLM_Quiescent

Cn_GSR

This subparameter is only valid for Cn_Model_Type sub-parameter
with a SLM_General value. If the Cn_Model_Type sub-parameter is
equal to SLM_General, then this Cn_GSR sub-parameter is
required. This sub-parameter is used to reference the general
Signal to Ground ratio used to generate the SLM_General model.

===
      Keyword: [Path Description]
     Required: Yes
     Argument: None
  Description: This keyword allows the user to describe the connection
                between cascaded matrices and the names of the ports.
   Sub-params: Model_PinMap, Cn_Section Cn_Fork, Cn_EndFork
  Usage Rules: Each pin can be divided into one or more cascaded "matrix
                sections". The Fork and Endfork subparameters allow the path to
                branch to multiple nodes. Cn_Section references each section
                name. Model_PinMap references the proper pin name to matrix
                specific matrix location.

Model_PinMap

Following the keyword line is a required sub-parameter
Cn_Model_Type. This sub-parameter defines the pin map names to
be used with connector model.

The sub-parameter and its value are separated by a single
Whitespace. There can be as many Model_PinMap subparameters as
required to represent a connectors, but each must be on a single
line.

Must be located between [Begin Cn Model] and [End Cn Model].

This field is used to name a port of each connector terminal.

Under some conditions, all Model_PinMap designations may have the
same name. However, two different names are required to address
crossover pin mapping.

Any Model_PinMap name must match the names given in the [Cn Pin
Map].

Model_PinMap can be used to define port at the end of any
Cn_fork or end of the overall connector model.

Example: Model_PinMap Baseboard_side
         Model_PinMap DaughterCard_side

14

Cn_Section

Following the keyword line is a required sub-parameter
Cn_Section. This sub-parameter defines the name and location of
a model section. Basically, this is used to define a straight
through section of a connector model. Matrix sections are
interconnected, scaled and may be re-used to form a connector
model.

Arguments "multiplier" and "section name" must follow the
Cn_Section parameter. The multiplier is a scale factor. The
"section name" is the name of the section as specified under the
[Begin Cn Section] keyword.

The multiplier may be used to scale any named section. The
multiplier may be any positive value greater than zero. The
multiplier allows a matrix section to be re-used as a scaled
value. The scale factor is applied to each parameter of all
matrices resulting in scaled R,L, and C matrices.

Cn_Section may be used as many time as required to describe the
connector. Each scale factor and section name MUST be separated
by at least one space.

Named sections are NOT specified in per-unit-length. They are
specified as unit-less total R,L,C values for the given section
Cn_Section must appear after the keyword [Begin Cn Model] and
before the keyword [End Cn Model].

Example: Cn_Section 1.0 SectionA

Cn_Fork, Cn_EndFork

Following the keyword line is a sub-parameter is used if there
are forks in the connector being modeled.

This keyword is used to define a branch current path in connector
model. Thus allowing greater than two ports per any connector
pin.

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

Any number of Cn_Fork statements may be used, however only one
keyword is allowed per line. Cn_Fork must appear after the
keyword [Begin Cn Model] and before the keyword [End Cn Model].

Example: Cn_Fork
          Cn_EndFork

==

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

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

```
[Begin Cn Model] MyModelExample1
  Cn_Model_Type SLM_Quiescent

  Cn_Section 1.0 Diagonal_matrix1
  .
  .
  .
[End Cn Model] MyExample1
```

Example 2: (2 ports per pin, 5 sections straight through)
```
        SectA        SectB       SectC       SectD        SectE
Port1 >--------< >--------< >-------< >--------< >----------< Port2
[Begin Cn Model] MyModelExample2
Cn_Model_Type SLM_Quiescent

[Path Description]
Model_PinMap = MyModelPinMapA
Cn_Section 1.0 SectA
Cn_Section 1.0 SectB
Cn_Section 1.0 SectC
Cn_Section 1.0 SectD
Cn_Section 1.0 SectE
Model_PinMap MyModelPinMapB
  .
  .
  .
[End Cn Model]
```

```
Example 3: (2 ports per pin, 2 series sections with one stub)
        SectionA          SectionB
Port1 >------------< >--------------< Port2
                |
          StubSection1

[Begin Cn Model] MyModelExample3
Cn_Model_Type MLM

Model_PinMap MyModelPinMapA
Cn_Section 1.0 SectionA
Cn_Fork
Cn_Section 1.0 StubSection1
Cn_EndFork
Cn_Section 1.0 SectionB
Model_PinMap MyModelPinMapB
  .
  .
  .
[End_Cn_Model]
```

Example 4: (2 ports per pin, 2 stub sections, one stub matrix is re-used)
```
          A             B            C
Port1 >----------< >-----------< >-----------< Port2
                |stub1       |stub4
                |stub2
                |stub3
```

```
[Begin Cn Model] MyModelExample4
Cn_Model_Type MLM

Model_PinMap MyModelPinMapA
Cn_Section 1.0 A
Cn_Fork
Cn_Section 1.0 stub1
Cn_Section 1.0 stub2
Cn_Section 1.0 stub3
Cn_EndFork
Cn_Section 1.0 B
Cn_Fork
Cn_Section 1.0 stub4
End_Fork
Cn_Section 1.0 C
Model_PinMap MyModelPinMapB

  .
  .
  .
[End_Cn_Model]
```

Example 5: (2 ports per pin, 3 stubs)
```
        A         B         C          D
Port1 >------< >------< >------< >-------< Port2
          |Stub1    |Stub2    |Stub1
                    |Stub3
                    |Stub4
```

```
[Begin Cn Model] MyModel4
Cn_Model_Type MLM


Model_PinMap MyModelPinMapA
Cn_Section 1.0 A
Cn_Fork
Cn_Section 1.0 Stub1
Cn_Endfork
Cn_Section 1.0 B
Cn_Fork
Cn_Section 1.0 Stub2
Cn_Section 1.0 Stub3
Cn_Section 1.0 Stub4
Cn_Endfork
Cn_Section 1.0 C
Cn_Fork
Cn_Section 1.0 Stub1
Snd_CnFork
Cn_Section 1.0 D
Model_PinMap MyModelPinMapB
  .
  .
  .
[End Cn Model]
```

Example 6: (3 ports per pin, 2 stubs)
```
        A         B         C          D
Port1 >------< >------< >------< >-------< Port 2
          |Stub1    |Stub2    |Stub1
                    |Stub3
                    |Stub4
```
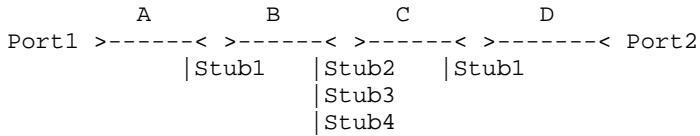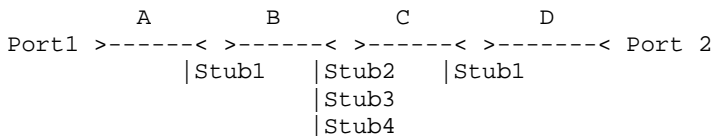
```
                          Port3

[Begin Cn Model] MyModelExample6
Cn_Model_Type MLM


Model_PinMap MyModelPinMapA
Cn_Section 1.0 A
Cn_Fork
Cn_Section 1.0 Stub1
Endfork
Cn_Section 1.0 B
Cn_Fork
Cn_Section 1.0 Stub2
Cn_Section 1.0 Stub3
Cn_Section 1.0 Stub4
Model_PinMap MyModelPinMapC
Endfork
Cn_Section 1.0 C
Cn_Fork
Cn_Section 1.0 Stub1
End_CnFork
Cn_Section 1.0 D
Model_PinMap MyModelPinMapB
  .
  .
  .
[End Cn Model]
```

Example 7: (3 ports per pin, 5 sections straight through)
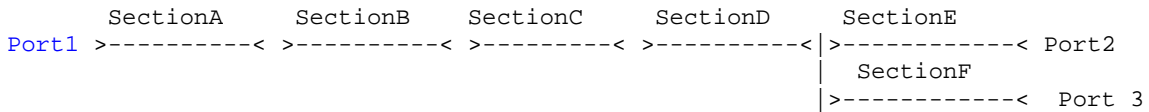
```
        SectionA       SectionB    SectionC      SectionD     SectionE
Port1 >----------< >----------< >---------< >----------<|>------------< Port2
                                                        |   SectionF
                                                        |>------------<  Port 3

[Begin Cn Model] MyModelExample7
  Cn_Model_Type SLM_Quiescent

  Model_PinMap = MyModelPinMapA
  Cn_Section 1.0 SectionA
  Cn_Section 1.0 SectionB
  Cn_Section 1.0 SectionC
  Cn_Section 1.0 SectionD
  Cn_Fork
  Cn_Section  1.0 Cn_SectionF
  Model_PinMap MyModelPinMapB
  Cn_EndFork
  Cn_Section 1.0 SectionE
  Model_PinMap MyModelPinMapC



  .
  .
  .
[End Cn Model]




==============================================================================
    Keywords:  [Cn Model Description]
```

18

```
     Required:  No
     Argument:  Text Block
  Description:  Provides a concise yet easily human-readable description of
               this connector model

  Usage Rules:  This keyword may appear multiple times in the file however
               it may appear only once between each pair of
               [Begin_Cn_Model]/[End_Cn_Model] keywords.

------------------------------------------------------------------------------
[Begin Cn Model Description]
High Density 0.1 center square pin with PCB effects
```

==

```
      Keyword:  [Cn Number of Conductors]
     Required:  If [Begin Cn Pin Map] is present, this keyword is required, if
               [Begin Cn Pin Map] is not present, this keyword cannot be used.
     Argument:  Numeric Value
  Description:  Documents the number of current carrying conductors for the
               physical connector as defined by the matrix.
  Usage Rules:  A single positive numeric value from 1 to 100,000 is
               recommended. Larger values shall create warnings by the parser.
               This number also represents the maximum swath size for a given
               connector model.

               This keyword MUST BE USED BETWEEN KEYWORDS
               [Begin Cn Model] and [End Cn Model].

Fixed sized connector example:
------------------------------------------------------------------------------
[Cn Number of Conductors]  8
```

==
```
     Keywords:  [Cn Col Swath]; [Cn Row Swath]
     Required:  No, except if [Define Cn Auto Map] is used or if the number
               of matrix entries does not match the number defined by
               [Cn Number of Conductors] for [Cn Pin Map].
     Argument:  Numeric Values
  Description:  Describes a smaller matrix which can be used to
               represent connectors of equal or larger size.  To visualize
               this, consider the swath matrix being a smaller version of the
               connector.  The edges of the matrix represent the same edge
               effects that exists in the full size connector.  The center area
               of the matrix represents the same effects as the center of the
               full size connector.

               It is the responsibility of the EDA tool to use the matrix and
               the required keyword information as a "mini" connector that is
               centered about the electrical paths of interest and/or to
               expand this information into a larger sized swath up to the
               full sized connector itself.

               For example a 2x8 matrix might be expanded to represent a
               connector that is 2x100.  The "swath" can also be combined with
               the [Begin Cn Auto Map] keyword to create multiple connector
               models from a single description.  For example a 2x10 matrix
               might be used to represent connectors ranging from 2x10 to
               2x200 with a single descriptive model.

  Usage Rules:  Only one [Cn Column Swath] or [Cn Row Swath]
```

shall be used per connector model.

MUST BE USED between keywords [Begin Cn Model] and [End Cn Model].

MUST BE specified after [Cn Number of Conductors], [Cn Rows of Pins], and [Cn Columns of Pins] have been specified.

MUST be followed by 6 numbers in order as follows:
  1) Rows in the swath matrix
  2) Columns in the swath matrix
  3) LEFT EDGE: Columns in the swath matrix used to represent
              left edge
  4) RIGHT EDGE: Columns in the swath matrix used to represent
               right edge
  5) TOP EDGE: Rows in the swath matrix used to represent top
             edge
  6) BOTTOM EDGE: Rows in the swath matrix used to represent
                bottom edge

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

Full sized swath: If the [Cn Number of Conductors] value is the same as the number of matrix pin entries for [Begin Cn Section] then neither [Cn Column Swath] nor [Cn Row Swath] are used. The matrices must be ordered exactly as the pins listed under [Begin Cn Pin Map].

Smaller sized swath: The swath matrix must contain a set of pins documenting a complete rectangular section organized by rows and columns. [Cn Row Swath] describe the pins ordered by rows (starting with the top row), and [Cn Column Swath] describe the same information ordered by columns (starting with the left column from a "side A" perspective).

For example, consider a swath that documents rows A and B and columns 1, 2, and 3 of a six pin rectangular [Cn Column Swath] describe pins ordered by columns (for example, A1, B1, A2, B2, A3, B3), and [Cn Row Swath] defines pins ordered by rows (for example, A1, A2, A3, B1, B2, and B3).

When a smaller sized swath is defined for a fixed pin map as defined by [Begin Cn Pin Map], the pin order must be grouped in the same order (by rows or by columns) as the given by EITHER [Cn Row Swath] or [Cn Column Swath]

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

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

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

```
    *--*--*--*--*
    |\/|\/|\/|\/|
    |/\|/\|/\|/\|
    *--*--*--*--*    [Cn Column Swath] 3 5 1 1 0 0
    |\/|\/|\/|\/|
    |/\|/\|/\|/\|
    *--*--*--*--*
```
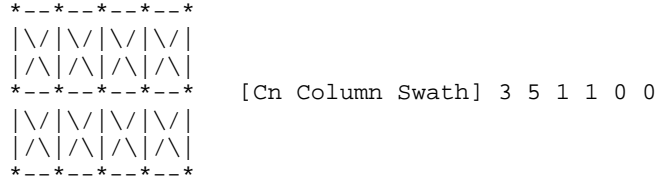
Notice that columns 2, 3, and 4 have the maximum amount of
coupling from each side. In this case the edge effect
columns 1 and 5 do not have coupling from both sides.

 Application of Swath Matrices:

    For the purposes of this discussion a 3 row by 5 column
    swath is used and the full sized connector is 3 rows by 9
    columns.   The swath matrix data can be organized by
    [Cn Column Swath] or [Cn Row Swath] pin ordering convention.
    However, the column expansion algorithms discussed later can
    be implemented using the [Cn Column Swath] format.

    Three methods can be used to apply the Swath matrix
    information:

    (1) Centering the Swath around the pins of interest

```
    1  2  3  4  5
    *--*--*--*--*
    |\/|\/|\/|\/|
    |/\|/\|/\|/\|    Swath:
    *--*--*--*--*    [Cn Column Swath] 3 5 1 1 0 0
    |\/|\/|\/|\/|
    |/\|/\|/\|/\|
    *--*--*--*--*


    *--*--*--*--*--*--*--*--*
    |\/|\/|\/|\/|\/|\/|\/|\/|
    |/\|/\|/\|/\|/\|/\|/\|/\|
    *--*--*--*--*--*--*--*--*    Full Connector
    |\/|\/|\/|\/|\/|\/|\/|\/|
    |/\|/\|/\|/\|/\|/\|/\|/\|
    *--*--*--*--*--*--*--*--*
    1  2  3  4  5  6  7  8  9
```
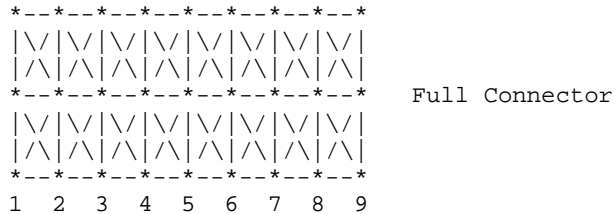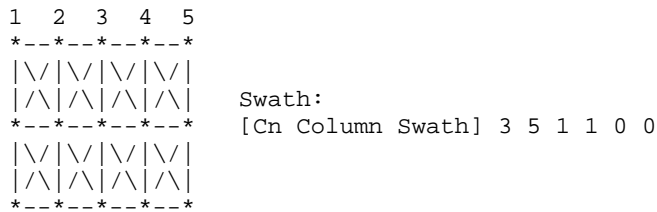
    The Swath above is centered about column 4. The paths of
    interest including those of columns 3 and 5 are also
    described by swath matrix data that has coupling from both
    sides. However, because of edge effects, columns 2 and 6
    are less accurately approximated.

The centering algorithm is a way to map the swath matrix
pins directly into the corresponding pins of the actual
connector.

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

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

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

    Some EDA tools work with the a larger swath, and other
    tools work with full sized connectors. 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 Connector: Expanding the swath
    matrix to the full sized connector matrix. the Full-sized
    matrix

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

    (a) Copy the left column pattern

    1 2 3 4 5
    ----*--*--*
    |\/|\/|\/|\/|
    |/\|/\|/\|/\| Swath:
    ----*--*--*   [Cn Column Swath] 3 5 1 1 0 0
    |\/|\/|\/|\/|
    |/\|/\|/\|/\|
    ----*--*--*

     ----*--*--*--*--*--*
     |\/|\/|\/|\/|\/|\/|\/|\/|
     |/\|/\|/\|/\|/\|/\|/\|/\|
     ----*--*--*--*--*--*   Full Connector
     |\/|\/|\/|\/|\/|\/|\/|\/|
     |/\|/\|/\|/\|/\|/\|/\|/\|
     ----*--*--*--*--*--*
     1 2 3 4 5 6 7 8 9
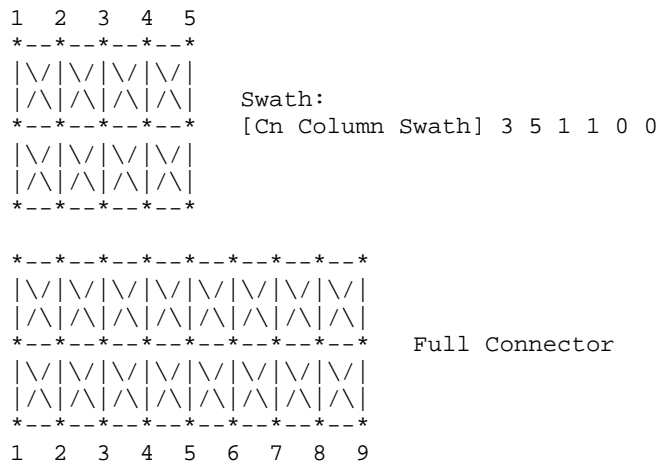
    The matrix date for swath columns 1 and 2 (including forward
    references) copied into the new full connector matrix.

    (b) repeat the center pattern,

```
 1  2  3  4  5
 *--*--*--*--*
 |\/|\/|\/|\/|
 |/\|/\|/\|/\|        Swath:
 *--*--*--*--*        [Cn Column Swath] 3 5 1 1 0 0
 |\/|\/|\/|\/|
 |/\|/\|/\|/\|
 *--*--*--*--*


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

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

(c) Copy the right column patterns

```
         1  2  3  4  5
         *--*--*--*--*
         |\/|\/|\/|\/|
         |/\|/\|/\|/\|        Swath:
         *--*--*--*--*        [Cn Column Swath] 3 5 1 1 0 0
         |\/|\/|\/|\/|
         |/\|/\|/\|/\|
         *--*--*--*--*


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

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

A similar process could be created for expanding the number
of rows. Some details are left to the EDA tool to
implement. For example, if the swath contains an even

```
                  number of columns, then either the left of center or the
                  right of center column may be used for center section
                  expansion.



     ------------------------------------------------------------------------------
                        SwathRows  SwathCols  LeftEdge  RightEdge  TopEdge  BottomEdge
       [Cn Column Swath]     2          8         1         1         0         0
```

==
Keywords: [Swath Pin Numbers]
    Required: No, except if [Define Cn Auto Map] is used or if the number
                of matrix entries does not match the number defined by
                [Cn Number of Conductors] for [Cn Pin Map].
   Arguments: Swath row name, physical row number, physical column number
 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 [Swath Pin Numbers] keyword is followed by a list of
                entries that apply for all of the sections for which the
                [Path Description] references.

                The number of entries in the list is equal to the product of
                the rows and columns of the [Cn Col Swath] or [Cn Row Swath]
                keywords, whichever is used.

                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 [Define Cn 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 [Cn Col Swath] or [Cn Row
                Swath] keywords. Similarly, the physical column number
                consists of an integer value from 1 the number of columns
                documented in the [Cn Col Swath] or [Cn Row Swath] keywords.

                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 connector 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 [Define Cn 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 connector swath, but
                issuing its results as separate sections. Furthermore, the
                row names used in teahc

                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

```
                    connector.

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

          A B C
          D E F

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

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

     This keyword defines the [Row] names if the Diagonal_matrix format is chosen,
     and it also is used as check on the [Row] name contents for the other formats
     in the same manner that [Pin Numbers] is used in IBIS Version 2.1 and 3.2.

     One issue is whether this needs to be used with each [Define Cn Section]
     keyword (supporting unique ordering with each section) and/or to be
     more closely associated with the matrix data for each section.


     ===============================================================================
      *************** For Variable size connectors ONLY **************************
         Keywords:  [Begin Cn Auto Map]
         Required:  If the [Begin Cn Pin Map] keyword is NOT present than this
                    keyword is required else it shall not be used.
         Argument:  None
      Description:  Using the [Begin Cn Auto Map] keyword indicates that this
                    connector is variable size and will be automatically increased
                    or decrease in size by an EDA tool using the algorithm defined
                    below.

      Usage Rules:  The keyword [Begin Cn Auto Map] replaces the
                    {Begin Cn Pin Map] keywords.
                    If the keyword [Begin Cn Auto Map] is used than the connector
                    model must be defined using the [Begin Cn Swath] keyword.

                    MUST BE USED after keywords [Begin Cn Model] and [End Cn Model].

                    This keyword allows an EDA tool to create a pin map
                    automatically for a connector by first determining the total
                    number of pins required is a particular schematic/PCB layout or
                    user interface, then execute the given algorithm until the
                    desired number of pins is created.

                    The Auto Map section contains a simple program function that
                    provides the algorithm for generating the signal and pin
                    names. The function must be written in a simplified form of the
                    'C' programming language.

                    The function may include:
                       - The built in variables ROW and COLUMN the result variables
                          SIGNAL_NAME and PIN_NAME
                       - Integer variables, simply created on usage.
```

```
                          - Integer constants e.g 55
                          - String constants enclosed by double quotes e.g. "example"
                          - Character constants enclosed in forward single quotes e.g.
                            'a'
                          - The mathematical operators + - * / %
                          - Boolean operators > < >= <= == AND OR
                          - Statements terminated by a semicolon
                          - The formatting function stringf(<template string>, <list
                            of variables>) the conditional statements if
                            (<expression>) { <statement> } else { <statement>}

                    The formating template string may include the following
                    substitution sequences:
                        %d   decimal number
                        %c   character
                        %s   string
                        %%   a percent character


                    Motes:
                    Variable map creation should only be used for connectors that
                    are non-industry standard pin outs.  For example an ISA,
                    SCSI, or PCI bus connector should not use the
                    [Begin Cn Auto Map] keyword as the signal names would not
                    match generally defined signal names.  It may be desired to
                    create multiple models for some connectors for example: The
                    model for a square pin connector might have two industry
                    standard pin outs like 34 pin floppy and 50 pin SCSI and also
                    a generic 8 to 100 pin model.  The Generic version could use
                    the variable size model and the 34 and 50 pin connectors
                    would use fixed size models.

                    Signal names are automatically generated and may not be
                    useful for identifying nets.  The EDA tools should give
                    priority to identifying connectivity with pin names if this
                    keyword is used.




 Example one:
--------------------------------------------------------------------------------
row_letter = 'a' + ROW - 1;
SIGNAL_NAME = stringf("Signal_%c%d", row_letter, COLUMN);
PIN_NAME = stringf ("%c%d", row_letter, COLUMN);


| This function results in pin and signal names as follows:

| Pin Name    Signal Name
|    a1           Signal_a1
|    a2           Signal_a2
|         ....
|    b1           Signal_b1
|         ....

Example one can be condensed to:

SIGNAL_NAME = stringf("Signal_%c%d", ROW + 'a' - 1, COLUMN);
PIN_NAME = stringf ("%c%d", ROW + 'a' - 1, COLUMN);


Example Two (with conditional)
```

```
if ((ROW < 5) OR (ROW > 7))
{
        SIGNAL_NAME = stringf("Signal_%c%d", ROW + 'a' - 1, COLUMN);
PIN_NAME = stringf ("%c%d", ROW + 'a' - 1, COLUMN);
}
else
{
        SIGNAL_NAME = "NA";
        PIN_NAME = "NA";
}
```

   -signal names may also be defined by either algorithm or using one of the
    fixed reserved names:
        PWRGND- reserved model name, used with power or ground pins,
        POWER - reserved model name, used with power supply pins,
        RET   - reserved word indicating either power or ground used
        GND   - reserved model name, used with ground pins,
        NC   - reserved model name, used with no-connect pins,
        NA   - used where data not available.

==
    Keywords: [Cn Col Range]
    Required: If the [Begin Cn Auto Map] keyword is present
              than this keyword is required else it shall not be used.
    Argument: Numeric Values
 Description: Used to define the minimum and maximum number of columns of
              pins for the connector represented by this model. The simulator
              will not expand or reduce the matrix beyond the limits defined
              by this parameter.
 Usage Rules: [Cn Col Range] Must be followed by two numbers. The first
              number relates to the minimum number of conductor columns.
              The second number is the maximum number of conductor columns

              Usage Context:
                  MUST BE USED after keywords [Begin Cn Auto Map] and
                  [End Cn Auto Map].

Example: This example is of a connector that may have 9 to 25 columns
----
[Cn Col Range] 9 25

==
    Keywords: [Cn Row Range]
    Required: If the [Begin Cn Auto Map] keyword is present
              than this keyword is required else it shall not be used.
    Argument: Numeric Value
 Description: Used to define the minimum and maximum number of rows of pins
              for the connector represented by this model. The simulator will
              not expand or reduce the matrix beyond the limits defined by
              this parameter.
 Usage Rules: [Cn Row Range] Must be followed by two numbers. The first
              number relates to the minimum number of conductor columns. The
              second number is the maximum number of conductor columns

              Usage Context: MUST BE USED after keywords [Begin Cn Auto Map]
                       and [End Cn Auto Map].

Example: This example is of a connector that may have 2 to 5 rows
     [Cn Row Range] 2 5

==
    Keywords: [End Cn Auto Map]
    Required: If the [Begin Cn Pin Map] keyword is NOT present than this
               keyword is required else it shall not be used.
    Argument: None
 Description: Provides end to [Begin Cn Auto Map]

==
    Keywords: [End Cn Model]
    Required: Yes
    Argument: Text Block
 Description: Used to define the end of the connector model

==
    Keywords: [Cn Pin Map]
    Required: If the [Begin Cn Auto Map] keyword is NOT present
               than this keyword is required else it shall not be used.
    Argument: Text String

UPDATE??

Description: Describes the index into the matrix for which a given pin
              applies and the reference to the physical map.

  Sub-Params: Pin, ModelPinMapName
 Usage Rules: The ModelPinMapName fields MUST appear after
               the [Cn Pin Map]. This name is used by the [Begin Cn Model]
               keyword to reference a connector pin map.

               More than one [Begin Cn Model] record may reference the same
               pin map.

               The keyword NA indicates that there is not a pin in the
               location (e.g. PGA socket with corner pin missing)
               An end of line character indicates the end of the
               information for the row

               There shall be one line for each pin in this connector
               model.

               The 'Pin" name allows EDA tools to align pin names in the model
               with pin names in the net list.

               The "Signal" name can be based on application but recommended if
               applicable.

               It is acceptable to have more than one pin map in a file.
               Therefore the keywords [Cn Pin Map] may appear as a pair
               more than once in a file.

               This keyword pair MUST be placed outside the keyword pair
               [Begin Cn Model] / [End Cn Model] keywords to allow it
               to be reused.

               There must be a [Cn Pin Map] that is correct for each port.

                        POWER, RET, GND, and PWRGND
                        Used to indicate that this pin should be
                        connected to either a decoupled power source (POWER),
                        a ground node (RET or GND), or either a power or
                        ground pin (PWRGND). The keyword RET is available to
                        indicate it is a return pin and could be either power
                        or ground.

[Cn Pin Map] Baseboard_side
|Pin   ModelPinMapName
  A1   GND
  A2   D0
  A3   CLK
  A4   D1
  B1   STROBE1
  B2   A0
  B3   POWER
  B4   PWRGND

[Cn Pin Map] Daughtecard_side
|Pin   ModelPinMapName
  A1   GND
  A2   D0
  A3   CLK
  A4   D1
  B1   STROBE1
  B2   A0
  B3   POWER
  B4   PWRGND

==
Keywords: [End Cn Model]
     Required: Yes
     Argument: None
 Description: Defines the end of the connector models for the connector family.

[End Cn Model]

==
Keywords: [End Cn Model Family]
     Required: Yes
     Argument: None
 Description: Defines the end of the connector model family.
 Usage Rules: These commands wrap the entire model in this file.

[End Cn Model Family]

==

8.0 MATRIX KEYWORDS

==
     Keyword: [Begin Cn Section]
     Required: No
     Argument: Text String
 Description: Indicates the beginning and the name of a set of R,L,C matrix

```
              data.
 Usage Rules:  Each matrix used for given connector model MUST have exactly
               the same number conductors.

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

               If the swath method is not used the matrix MUST have exactly
               the same number of conductors as the connector has pins.

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

               MUST be placed outside the context of keywords
               [Begin Cn Model] and [End Cn Model] as sections may be
               reused by more than one connector model.

                MUST be placed inside the context of the family

               Every pin in the SLM matrix is a separate matrix value.

               The following keywords are optional:
               [Resistance Matrix], [Bandwidth], [Row]

               Must include [Derivation Method]

For an example of [Begin Cn Section] see matrix examples that follow.
--------------------------------------------------------------------------
=
==========================================================================
            Derivation Method = xxxxx
                  Required:  Yes
                  Argument:  Lumped or Distributed
               Description:  Describes the type of model that should be
                             enerated from the Matrices.
               Usage Rules:  Must be between [Begin Cn Section] and [End Cn
                             Section]

                             The [Derivation Method] keyword is required for
                             each section.

                              "Lumped" or "Distributed"
                             For LUMPED models any single, any pair, or all
                             of the following keywords could be used:
                             [Inductance Matrix], [Capacitance Matrix] ,
                             [Resistance Matrix]

                             For DISTRIBUTED models BOTH the [Inductance
                             Matrix] and [Capacitance Matrix] keywords are
                             required.  The [Resistance Matrix] keyword is
                             optional.

                             All three matrices within the same section
                             share the same derivation method

                             If ModelType is SLM then simulators should use
                             this data to create a non-coupled fully
                             distributed transmission line model.
                             It may be also be used to create a multi-
```

section ladder RLC-LCR model as is often used
                              with SPICE simulators.

                              If a ladder network is substituted for
                              a fully distributed transmission line model the
                              simulator vendor should insure enough sections
                              are provided to insure the bandwidth specified
                              by the Min_Slew_Time of this model is achieved.
                              The symmetric RLC-LCR method should be used as
                              opposed to the non-symmetric RLC-RLC method if
                              a ladder network is used.

                       [Derivation Method] LUMPED

===
    Keywords: [Resistance Matrix], [Inductance Matrix], [Capacitance Matrix],
               [Conductance Matrix]
    Required: Yes, If a simple single line model is defined then the
               following matrices are required:
                       case 1: [Resistance Matrix] only (Not to be used with
                                distributed models
                                       OR
                       case 2: [Inductance Matrix] and [Capacitance Matrix] only
                               for which the [Resistance Matrix] is assumed to
                               be zero ohms
                                       OR
                       case 3: [Resistance Matrix] and [Inductance Matrix]
                               and [Capacitance Matrix]

                        NOTE: In case 1,2,3 the matrices must be Diagonal_matrix.

               If a coupled model is defined then the following matrices are
               required:
                       case 4: [Inductance Matrix] and [Capacitance Matrix] only
                               for which the [Resistance Matrix] is assumed to be
                               zero ohms
                                       OR
                       case 5: [Resistance Matrix] and [Inductance Matrix]
                               and [Capacitance Matrix]

                       case 6: [Resistance Matrix] and [Inductance Matrix]
                               and [Capacitance Matrix] and [Conductance Matrix]

               NOTE: In case 4,5, and 6 the matrices shall be any combination of
                      matrix types (Diagonal, Banded, Sparse, Full).

     Argument: Matrix Type
 Description: The sub-parameters mark the beginning of a matrix, and
               specify how the matrix data is formatted.
 Usage Rules: For each matrix keyword, use only one of the sub-parameters.

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

               MUST BE USED between keywords [Begin Cn Section] and [End Cn
               Section].

               If the model type in SLM, then only the diagonal matrix is
               allowed.

```
Matrix Type Options:
        Diagonal_matrix, Banded_matrix, Sparse_matrix, or Full_matrix

            Diagonal_matrix
                Required:  No
                Argument:  None
              Description:  The Diagonal_matrix is used to specify uncoupled
                            models.

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

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

                            Specify the Diagonal_matrix one row at a time,
                            starting with row 1 and working up to higher
                            rows.  Diagonal elements are entered one per
                            line. MUST be used between keywords [Begin Cn
                            Section] and [End Cn Section].


            Banded_matrix
                Required:  No
                Argument:  None
              Description:  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.  Entries other than the
                            main diagonal are not required.


            Sparse_matrix
                Required:  No
                Argument:  None
              Description:  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.

                            Symmetry can be exploited to reduce the amount
                            of data by eliminating from the matrix any
                            entries below the main diagonal.
```

An N x N 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.
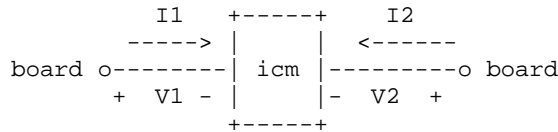
Full_matrix
        Required: No
        Argument: None
     Description: When the Full_matrix format is used, the couplings between every Pair of elements is specified explicitly. Assume that the Matrix has N rows and N columns.

                  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.

Matrix Descriptions
          The resistance, inductance, and capacitance matrices may also be referred to as "RLC matrices" within this specification. When measuring the entries of the RLC matrices, either with laboratory equipment or field-solver software, currents are defined as ENTERING the pins of the package from the board. The corresponding voltage drops are to be measured with the current pointing "in" to the "+" sign and "out" of the "-" sign.

```
                  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], or [Capacitance Matrix] a different matrix type can be used for the data.

Resistance Matrix Description
==
      UNITS: OHMS
 DESCRIPTION: The resistance matrix represents the total DC resistance between the two ends of this section of the connector.

          NOTES:
              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.

              DO NOT USE ohms/meter as this is the total value and includes the length of this section.

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

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

Inductance Matrix Description
===
       UNITS: HENRIES
 DESCRIPTION: Two inductance matrix methods are supported:
             Loop-Inductance Matrix for SLMs ONLY   (non-coupled)
             Partial-Inductance Matrix for MLMs ONLY (coupled)

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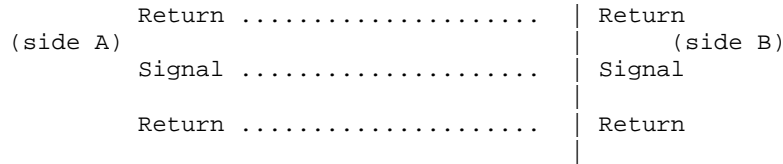   and capacitance matrices.

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

Loop-Inductance Matrix Description (SLMs ONLY):
===
 DESCRIPTION: 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.

                    connector 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 pin with return currents assumed
         returned through all pins defined to be power or ground.

          NOTES (Single-Line Loop-Inductance Matrix):
                 This method must be used if the ModelType is SLM.

                 Loop inductance can be experimentally measured.

                 Loop inductance matrices can be created using field
                 solvers.

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

                 SLM models make have zero values for power or ground
                 R,L,C values which prevents there use in analyzing
                 connector ground bounce effects.

                 Typical uses for SLM model might include for example
                 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 would 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 indicate the ground-signal-ratio has been
violated.

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

All connector pins must be specified with non-zero L
and C values in at least one of the sections. The
simulator will zero out portions of all sections
corresponding to the ground or power pins.

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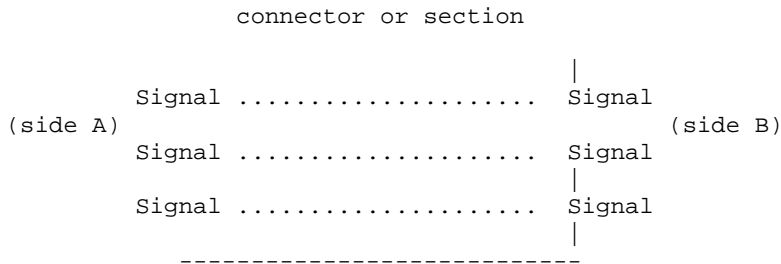signal is long compared to the electrical delay
through the connector.

Partial Self Inductance Matrix Description (MLMs ONLY):
==
 DESCRIPTION: A partial self inductance is an approximation of the loop
               inductance in situations where the current return path cannot be
               explicitly stated in the field extraction stages.
               Partial self inductance can be numerically calculated by
               introducing a current in a circuit element that originates on
               one end and terminates on the other end. There is no physical
               loop per se, instead charges will build up on each side of the
               circuit element. In practice, this situation cannot exist more
               than instantaneously. For the purposes of simulation the

condition does not affect the resultant loop inductance values.
Then from the currents found the magnetic fields and
inductance's can be calculated. In this way, the self and
mutual inductances of any section of a closed circuit 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.

```
              connector or 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 it is required that at least one section
include partial inductance and capacitance matrices.

Capacitance Matrix Description
==

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

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

 USAGE RULES:  Two Maxwell capacitance matrix methods are supported:

              SLM, Single-Line, Loop-capacitance Matrix     (non-coupled)
              MLM, Multi-Line, Capacitance Matrix (coupled)

              If ModelType is SLM then all sections must only contain a single
              capacitance value per line.

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


              Single-Line Capacitance Matrix Description:
              -------------------------------------------
              To define capacitance requires conductors separated by a
              dielectric.  The conductors are those found in a loop
              consisting of a closed path including a signal and all possible
```

return paths. The Maxwell Capacitance matrix relates the total
charges on the conductors to the voltages applied to them.
[Q]=[Cm][V] Where [Cm] is the Maxwell matrix, [Q] is the vector
of total conductor charges, and [V] is a vector of applied
voltages. The negative values of the off-diagonal entries can be
explained on physical grounds: if you put +1 Volt on a single
conductor and 0 Volts on the others, then you induce negative
charges on those other conductors to terminate the field lines
emanating from the conductor at 1 Volt.

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 effected by other nearby
conductors during the field extraction phase. As such it should
not normally be used by itself to determine the capacitance of a
random pair of conductors in a given structure.

Conductance Matrix Description
==
        Units: Siemens (mhos)
 Description: The Conductance maxtrix 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.

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

             Do NOT use Siemens/meter as this is the total value and includes
             the length of this section.

Loss Matrix Description
==
    This release does not support frequency dependent loss matrices.
    A future release is expected to add support for frequency dependent loss
    matrices.

Un-Coupled Models:
==
    Single-Line-Models or SLM models are used for un-coupled models.
    These models are useful for most simulations where coupled effects
    can be ignored safely. These models are most accurate if closely
    spaced ground pins or a ground plane is present.

    A "Diagonal_matrix" matrix type is the best choice for describing
       all matrices R,L and C for SLM models.
       (see matrix descriptions below).

    Several methods can be used:
    Field solvers with return path pins defined. Only the diagonal
    matrix data is used. The coupling information is ignored.
    The R,L,C matrices should represent the electrical parameters for a
    specific section.

    2D field solver with a specific return path or ground. Only the diagonal

matrix data is needed. No coupling data is present in the final matrix

Closed form round wire inductance equation.

Experimental lab data may be obtained from TDR, VNA or other means

Coupled Models:
==
Multi-Line-Models or MLM models are used for coupled models.

A "Full_matrix" matrix type is the best choice for describing
all matrices R,L and C for MLMs. (see matrix descriptions below)

Several methods can be used:
3D field solver can be used to create partial inductance and
capacitance matrices that allow accurate simulations without
the need to pre-define the ground pins.

Field solvers without specific return path pins. A return plane
   May be specified as a reference only and is placed far field.

Experimental lab data may be obtained from multichannel
measurements that incorporate TDR, VNAs or other means as long as
the resulting inductance matrix contains partial inductance values.

Common to Coupled and Un-Coupled Models
==
One common aspect of all the different formats is that they exploit
the symmetry of the matrices they describe. This means that the
entries below the main diagonal of the matrix are identical to the
corresponding entries above the main diagonal. Therefore, only
roughly one-half of the matrix needs to be described. By convention,
the main diagonal and the UPPER half of the matrix are provided.

In the following text, we use the notation [I, J] to refer to the entry
in row I and column J of the matrix. Note that I and J are allowed
to be alphanumeric strings as well as integers.

The ordering of these strings is determined by the row/column ordering
convention given by the [Cn Row Swath] or [Cn Column Swath] convention,
or else matches exactly the [Cn Pin Map] order when the pin map contains
the same number of pins as the Swath.

In the following text, "Row 1", means the row corresponding to the first
pin.

Also note that the numeric entries of the RLC matrices are standard
IBIS floating point numbers. As such, it is permissible to use
metric "suffix" notation. Thus, an entry of the C matrix could be
given as 1.23e-12 or as 1.23p or 1.23pF.

Each new row is identified with the Row keyword.

==
 Keyword: [Bandwidth]
 Required: Yes, if a Banded_matrix matrix is being defined.
 Argument: Numeric Value
Description: Indicates the size of the width of the coupling matrix….

```
   of the matrix.  This has nothing to to do 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:  MUST be used between keywords [Begin Cn Section] and
                 [End Cn Section].

                 MUST follow any usage of one of the following keyword:
                 Banded_matrix
--------------------------------------------------------------------------
                  See example "Matrix Example" lower in this document.


===========================================================================
      Keyword:  [Row]
     Required:  Yes if a matrix is being defined.
     Argument:  None
  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.  You can break the data up with new-line characters so
                 that this 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.


                 Usage Context:
                     MUST be used between keywords [Begin Cn Section] and
                     [End Cn Section].

                     MUST follow any usage of one of the following keywords:
                     Banded_matrix, Sparse_matrix, Full_matrix
--------------------------------------------------------------------------
 See example "Matrix Example" lower in this document.




===========================================================================
      Keyword:  [End Cn Section]
     Required:  Yes
     Argument:  Text String
  Description:  Indicates the end of [Begin Cn Section]
```

==
 The resistance matrix for this package has no coupling

[Begin Cn 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 Cn Section] ExampleMatrix00

[Begin Cn Section]   ExampleMatrix01

 The resistance matrix for this package does not have coupling
 and is exactly the same as the Diagonal_matrix shown above
 it 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 Cn Section] ExampleMatrix01
```

==

9.0 End Keyword

==

```
    Keyword:  [End]
   Required:  Yes
   Argument:  None
Description:  Specifies end of file
--------------------------------------------------------------------------
[End]
```

==

10.0 ICM EXAMPLES
!!!!!   NEEDS MAJOR REVISION !!!!!
INCLUDE IN DIFFERENT DOCUMENT????

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

[IBIS Cn Model Ver]  1.0
[File Rev]           1.0                1.0 -> design data, no verification
[Date]          July 1, 2000           The latest file revision date

[Begin Cn Model Family]
[Begin Cn Model List]
DB9              Mated           10ns       db9mated.jpg
Header2x5     Mated          500ps    Header100.jpg
Header2x5s    Mated          500ps    Header100.jpg
[End Cn Model List]
```

 EXAMPLE: single SLM DB9 connector
    (NOTE: DATA IS "MAKE BELIEVE")

```
[Begin Cn Model] DB9 PinMapDB9a PinMapDB9b SLM 1:9
[Cn Number of Conductors]  9
[Cn Columns of Pins]      5
[Cn Rows of Pins]         2
Cn_Section           SectionDB9
[End Cn Model] DB9
```

 EXAMPLE: single SLM DB9 connector with swath
    (NOTE: DATA IS "MAKE BELIEVE")

```
[Begin Cn Model] DB25swath PinMapDB25_in PinMapDB25_out SLM 1:25
[Cn Number of Conductors]  25
[Cn Columns of Pins]       13
[Cn Rows of Pins]          2
Cn_Section                            SectionDBSwath
           SwathRows SwathCols RightEdge LeftEdge TopEdge BottomEdge
 [Begin Cn Swath]   2         3         1         1        0         0
 [End Cn Swath]
[End Cn Model] DB25swath
```

Pin maps for DB9

```
[Begin Cn Pin Map] PinMapDB9
|Index Pin    Signal
1         1        CD
2         2        RXD
3         3        TXD
4         4        DTR
5         5        GND
6         6        DSR
7         7        RTS
8         8        CTS
9         9        RI
 [End Cn Pin Map]

[Begin Cn Pin Map] PinMapDB25
|Index Pin    Signal
1         1        CD
2         2        RXD
3         3        TXD
```

```
4            4          DTR
5            5          GND
6            6          DSR
7            7          RTS
8            8          CTS
9            9          RI
10          10          CD
11          11          CD
12          12          RXD
13          13          TXD
14          14          DTR
15          15          GND
16          16          DSR
17          17          RTS
18          18          CTS
19          19          RI
20          20          RI
21          21          CD
22          22          RXD
23          23          TXD
24          24          DTR
25          25          GND
[End Cn Pin Map]
```

SECTION EXAMPLES WITH MATRICES

```
[Begin Cn Section] SectionDB9
[Derivation Method] Distributed
 [Resistance Matrix] Diagonal_matrix
.05
.05
.05
.05
.05
.05
.05
.05
.05
[Inductance Matrix] Diagonal_matrix
1.5nH
1.5nH
1.5nH
1.5nH
1.5nH
1.5nH
1.5nH
1.5nH
1.5nH
[Capacitance Matrix] Diagonal_matrix
0.3pF          |Edge pin has less capacitance
0.5pF
0.5pF
0.5pF
0.3pF          |Edge pin has less capacitance
0.3pF          |Edge pin has less capacitance
0.5pF
0.5pF
0.3pF          |Edge pin has less capacitance
[End Cn Section] SectionDB9

[Begin Cn Section] SectionDBswath
[Resistance Matrix] Diagonal_matrix
.05
```

```
.05
.05
.05
.05
.05
 [Inductance Matrix] Diagonal_matrix
1.5nH
1.5nH
1.5nH
1.5nH
1.5nH
1.5nH
 [Capacitance Matrix] Diagonal_matrix
0.3pF            |Edge pin has less capacitance
0.3pF            |Edge pin has less capacitance
0.5pF
0.5pF
0.3pF            |Edge pin has less capacitance
0.3pF            |Edge pin has less capacitance
 [End Cn Section] SectionDBswath

[End Cn Model Family]


[End]
```