**BUFFER ISSUE RESOLUTION DOCUMENT (BIRD)**

**BIRD NUMBER:** TBD Draft 4

**ISSUE TITLE:** File Naming Rules

**REQUESTOR:**  Walter Katz, Signal Integrity Software, Inc;

**DATE SUBMITTED:** November 2016

**DATE REVISED:**

**DATE ACCEPTED:**

**DEFINITION OF THE ISSUE:**

IBIS AMI modeling, and Interconnect Modeling may result in a large number of supporting files. This BIRD replaces the restriction that all files referenced by the .ibs file be restricted to the directory containing the IBIS file, to files referenced by the .ibs file shall be either in the same directory as the .ibs file or in a directory structure below the directory containing the .ibs file.

In addition, this BIRD relaxes other constraints on file names such as allowing both upper and lower case letters and allowing longer file names.

The IBIS specification must meet these requirements:

|  |  |
| --- | --- |
| Requirement | Notes |
| 1. Enable file names that are up to 256 characters in length.
 | IBIS in general will need to increase its rule on the number of characters in each line. |
| 1. Enable files to be defined and located in a Hierarchical structure either in the same directory of the IBIS file or in a directory directly below it.
 | AMI supporting files, and Interconnect Model supporting files can be organized in a directory structure that makes it simple for IC vendors to organize and deliver their IBIS models to their customers. |
| 1. Enable file names with both lower and upper case characters,
 | IBIS should allow similar file naming rules as modern day tools and specifications. |
| 1. Enable file names do define a directory
 |  |
|  |  |
|  |  |

**SUMMARY OF PROPOSED CHANGES:**

For review purposes, the proposed changes are summarized as follows:

|  |  |  |
| --- | --- | --- |
| Specification Item | New/Modified/Other | Notes |
| Change the file naming rules in paragraph 3 and the IBIS File line length limits in Paragraph 4 of the GENERAL SYNTAX RULES AND GUIDELINES, |  |  |

**PROPOSED CHANGES:**

Replace the following two paragraphs on page 9 of the IBIS specification

1. To facilitate portability between operating systems, file names used in a .ibs file must only have lower case characters. File names should have a basename of no more than forty (40) characters followed by a period (“.”), followed by a file name extension of no more than three characters. The file name and extension must use characters from the 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 \_ ^ $ ~ ! # % & - { } ) ( @ ‘ `

The file name and extension are recommended to be lower case on systems that support such names.

1. 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.

To:

3.   File names should be no more than two hundred and fifty six (256) characters. The file name 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

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 \_ ^ $ ~ ! # % & - { } ) ( @ ‘ ` . /

Note that files names in the Windows operating system are case insensitive and case preserving , while file names in Linux operating systems are case sensitive. When a [computer](https://en.wikipedia.org/wiki/Computer) [file system](https://en.wikipedia.org/wiki/File_system) stores file names, the computer may keep or discard [case](https://en.wikipedia.org/wiki/Letter_case) information. When the case is stored, it is called **case preservation**.File names in IBIS files should be case sensitive so that the IBIS file and the files it references will work properly on both Windows and Linux operating systems. The “/” character is used to delineate directories in a file name. The character sequence “..” is not permitted. The first character of a file name shall not be “/”. The last character of a file name shall not be “/”. The EDA tool is responsible for making any operating system-specific adjustments (for example, replacing forward slashes "/" with backslashes "\") if necessary.

The characters after the last “.”  are considered the file name extension. There are places in this document that specify the file name extension for specific files types (e.g. .ibs, .pkg, and .ami). These extensions are case sensitive.

“file name” shall mean the name of the file, including the full path relative to the directory containing the file that references the “file name”.  A “file name” may also be a directory.

Example file names defined inside of IBIS file xyz.ibs:

xyz/ami/Tx/Tx.ami

xyz/ami/Tx/Tx.dll

xyz/interconnect/xyz\_uncoupled/xyz\_uncoupled.ims

xyz\_coupled xyz/interconnect/xyz\_coupled/xyz\_coupled.ims

Example file names defined inside of file xyz/interconnect/xyz\_uncoupled/xyz\_uncoupled.ims

File\_TS snp/DQ1.s2p

If xyz.ibs is in the directory //Libraries/IBIS, then these files and directories will be in the following absolute directories:

IBIS

xyz.ibs

xyz

IBIS/xyz

ami

interconnect

IBIS/xyz/ami

Tx.ami

Tx.dll

IBIS/xyz/interconnect

xyz\_uncoupled

xyz\_coupled

IBIS/xyz/interconnect/xyz\_uncoupled

xyx\_uncoupled.ims

snp

IBIS/xyz/interconnect/xyz\_uncoupled/snp

DQ1.s2p

IBIS/xyz/interconnect/xyz\_coupled

xyx\_coupled.ims

Note that when a “file name” is referenced inside of a “source file” the location of that “file name” shall always be relative to the location of the “source file”.

The space character remains an illegal file name character. When an IBIS file is delivered all supporting files shall either be in the same directory as the IBIS file, or in directories below this directory. Absolute files names (e.g. that begin with // or C:) are not permitted. Users and EDA tools may choose to move supporting files into other directories that are not located below the location of the “source file”. These files may be put on a tool specific search path or have added symbolic links. The IBIS parser may report such non-standard file names as either errors or warnings.

Furthermore, lower-case file\_name entries are recommended to avoid possible conflicts with file naming conventions under different operating systems. Case differences between otherwise identical file name entries should be avoided.

1. A line of the file may have at most 1024 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.

Change to  *Keyword:* **[File Name]**

**Currently**

*Keyword:* **[File Name]**

*Required:* Yes

*Description:* Specifies the name of the .ibs file.

*Usage Rules:* The file name must conform to the rules in paragraph 3 of Section 3, "GENERAL SYNTAX RULES AND GUIDELINES". In addition, the file name must use the extension “.ibs”, “.pkg”, or “.ebd”. The file name must be the actual name of the file.

Change to

*Keyword:* **[File Name]**

*Required:* Yes

*Description:* Specifies the name of the .ibs file.

*Usage Rules:* The file name must conform to the rules in paragraph 3 of Section 3, "GENERAL SYNTAX RULES AND GUIDELINES". In addition

* The file name must use the extension “.ibs”, “.pkg”, or “.ebd”.
* The file name must be the actual name of the file.
* There may not be a “/” in the file name.

Change to  *Parameter:* **DLL\_ID**

From

*Parameter:* **DLL\_ID**

*Required:* No, and illegal before AMI\_Version 6.0

*Direction:* Rx, Tx

*Descriptors*:

Usage: In

Type: String

Format: Value

Default: <string literal>

Description:<string>

*Definition:* The EDA tool is responsible for recognizing this parameter name and replacing the value declared in the .ami file with a string that contains a unique alphanumeric identifier. The algorithmic model is responsible for using DLL\_ID as the base name for any data files that the model creates, either for use as temporary storage or for recording output data. The use of DLL\_ID helps guarantee that multiple instances of the same model (or different models from the same vendor) do not mix up data as a result of collisions between temporary or permanent file names.

To

*Parameter:* **DLL\_ID**

*Required:* No, and illegal before AMI\_Version 6.0

*Direction:* Rx, Tx

*Descriptors*:

Usage: In

Type: String

Format: Value

Default: <string literal>

Description:<string>

*Definition:* The EDA tool is responsible for recognizing this parameter name and replacing the value declared in the .ami file with a string that must conform to the rules in paragraph 3 of Section 3, "GENERAL SYNTAX RULES AND GUIDELINES. The algorithmic model is responsible for using DLL\_ID as the base name for any data files that the model creates, either for use as temporary storage or for recording output data. The use of DLL\_ID helps guarantee that multiple instances of the same model (or different models from the same vendor) do not mix up data as a result of collisions between temporary or permanent file names.

**BACKGROUND INFORMATION/HISTORY:**