****
****

```
Buffer Issue Resolution Document  (BIRD)
BIRD ID#:       {TBD}
ISSUE TITLE:    Clarification of Usage Out, InOut and Info for IBIS AMI
REQUESTOR:      Arpad Muranyi, Mentor Graphics
DATE SUBMITTED:
DATE REVISED:   May 16, 2011
DATE ACCEPTED BY IBIS OPEN FORUM:
```

****
****

STATEMENT OF THE ISSUE:

The IBIS 5.0 specification defines the AMI_parameters_out argument for both the AMI_Init and AMI_GetWave functions, but fails to mention in (Usage Out) and (Usage InOut) parameter descriptions which function's argument will return them. This can lead to situations where EDA tool vendors and AMI model authors implement features with different assumptions which can result in unsuccessful or incorrect simulations.

In addition, the IBIS 5.0 specification allows Model_Specific parameters to be combined with (Usage Out/InOut/Info) parameters which can result in undocumented model features which cannot be supported in a standardized way by the EDA vendors in their simulators. This can undermine the promise of portability and interoperability of the IBIS specification.

****

STATEMENT OF THE RESOLVED SPECIFICATIONS:

On pg. 140 insert after these lines:

```
|   Usage: (required for model specific parameters)
|     In     Parameter is required Input to executable
|     Out    Parameter is Output only from executable
|     Info   Information for user or EDA platform
|     InOut  Required Input to executable.  Executable may return different
|            value.
```

the following lines:

```
| Note:  Algorithmic models may return (Usage Out) or (Usage InOut)
| parameters in the AMI_parameters_out argument of either the AMI_Init
| or the AMI_GetWave function, or both.  Reserved parameters are described
| by the specification in detail so that the model writer and the EDA tool
| vendor have enough information about which function’s AMI_parameters_out
| argument returns any given parameter, what the meaning and format of the
| parameter is and how the EDA tool is expected to process it.
|
| Algorithmic models or .ami parameter files are only allowed to influence
| simulation results or inputs through reserved parameters of (Usage Out),
| (Usage InOut) or (Usage Info) whose features and requirements are fully
| described in this specification.  Model_Specific parameters of (Usage
| Out), (Usage InOut) or (Usage Info) must not be used for purposes which
| may alter past, present or future simulation results, or simulation
| inputs in any way.
```

```
|
| On the other hand, EDA tool vendors are strongly encouraged to collect
| the information from Model_Specific parameters of (Usage Out), (Usage
| InOut) and/or (Usage Info) in a database and make it available to the
| user.  Since each call to the AMI_GetWave function may return a unique
| value in the AMI_parameters_out argument, EDA tool vendors are
| encouraged to build a database in which the parameters and values
| obtained from each AMI_GetWave call are organized in rows and columns
| so that the user can view them conveniently.  Information from the
| .ami parameter file or information returned through the
| AMI_parameters_out argument of the AMI_Init function may be treated
| as the first data point(s) of a data series returned by the
| AMI_parameters_out argument of the AMI_GetWave function for matching
| parameter names.  While EDA vendors may use techniques that best suit
| their product, it is expected that (Type String) parameters will be
| displayed in text display dialog(s) or editor(s), and numeric and
| (Type Boolean) parameters will be post processed or displayed in
| plotting utilities.
|
```

****

ANALYSIS PATH/DATA THAT LED TO SPECIFICATION:

A careful examination of the various Usage types and the associated rules for Reserved and Model_Specific parameters was carried out in discussions in the ATM Task Group. The resolution above reflects the decisions made in these conversations.

****

ANY OTHER BACKGROUND INFORMATION:

The issue was raised that the process of adopting new features is very slow in IBIS and there was a need to be able to make use of such unintended capabilities on a temporary bases. However, the majority of the participants felt that new capabilities can be implemented in tools and models outside the specification until the specification incorporates the new features.

Suggestions on how to make the specification more flexible were also mentioned in these discussions, but will not be captured here due to the complexity of the topic.

****