ANALOG DEVICES

AN-715
APPLICATION NOTE

One Technology Way • P.O. Box 9106 • Norwood, MA 02062-9106 • Tel: 781/329-4700 • Fax: 781/326-8703 • www.analog.com

A First Approach to IBIS Models: What They Are and How They Are Generated

by Mercedes Casamayor

INTRODUCTION

Saving time and reducing costs are key factors when designing systems. Modeling provides system designers with a useful tool to simulate the design before prototyping. This is the case in high speed systems where signal integrity simulations are performed to analyze the circuit behavior under different conditions in the transmission lines, and to prevent and detect typical situations, such as overshoot, undershoot, mismatched impedance, and others at an earlier stage. However, the availability of models for digital ICs is very scarce. When semiconductor vendors are asked for their SPICE models, they are reluctant to provide them since these models can contain proprietary process and circuit information.

This issue has been resolved with the adoption of IBIS (Input/Output Buffer Information Specification), also known as ANSI/EIA-656, a new standard for modeling that is becoming more and more popular among system designers.

WHAT IS IBIS?

IBIS is a behavioral model that describes the electrical characteristics of the digital inputs and outputs of a device through V/I and V/T data without disclosing any proprietary information. IBIS models do not correspond to the conventional idea of a model that system designers are used to, such as a schematic symbol or polynomial expression, among others. An IBIS model consists of tabular data made up of current and voltage values in the output and input pins, as well as the voltage and time relationship at the output pins under rising or falling switching conditions. This tabulated data represents the behavior of the device.

IBIS models are intended to be used for signal integrity analysis on systems boards. These models allow system designers to simulate and therefore foresee fundamental signal integrity concerns in the transmission line that connects different devices. Potential problems that can be analyzed by means of the simulations include the degree of energy reflected back to the driver from the wave that reaches the receiver due to mismatched impedance in the line; crosstalk; ground and power bounce; overshoot; undershoot; and line termination analysis, among others.

IBIS is an accurate model since it takes into account nonlinear aspects of the I/O structures, the ESD structures, and the package parasitics. It has several advantages over other traditional models such as SPICE. Thus, for example, the simulation time can be up to 25 times less, and IBIS does not have the nonconvergence problem SPICE does. In addition, IBIS can be run on any industry-wide platform since most Electronic Design Automation (EDA) vendors support the IBIS specification.

IBIS HISTORY

IBIS was developed by Intel® Corporation in the early 1990s. IBIS version 1.0 was issued in June 1993 and the IBIS Open Forum was created.

The IBIS Open Forum comprises EDA vendors, computer manufacturers, semiconductor vendors, universities, and end-users. It proposes updates and reviews, revises standards, and organizes summits. It promotes IBIS models and provides useful documentation and tools on the IBIS website. In 1995, the IBIS Open Forum teamed up with the Electronic Industries Alliance (EIA).

Several IBIS versions have been published. The first version described CMOS circuits and TTL I/O buffers. Each version adds and supports new capabilities, technologies, and device types. All versions are compatible with one another. IBIS version 4.0 was ratified in July 2002 by the IBIS Open Forum, but it is not yet an ANSI/EIA standard.

HOW TO GENERATE AN IBIS MODEL

IBIS models can be obtained by gathering data in simulations, or from bench measurements. If the former method is chosen, SPICE can be used to run the simulations and collect the V/I and V/T data for each of the input/output buffers. This allows process corner data to be included in the models. Then, using one of the SPICE-to-IBIS conversion programs available from the IBIS website, the IBIS model can be generated from SPICE.

The models can be generated for three different corner conditions: typical, minimum, and maximum. In a typical model, the data will be obtained for nominal supply voltage, nominal temperature, and nominal process parameters; in a minimum model, the data will be obtained with the lowest supply voltage, high temperature, and weak process parameters; and for a maximum model, the conditions will be the highest supply voltage, low temperature, and strong process parameters.

Each of these conditions leads to typical, slow, and fast models. A fast model is created by considering the highest current values with the fast transition time and the minimum package characteristics. On the other hand, the lowest current values with a slow transition time and maximum package values will produce a slow model.

If the data is obtained from lab measurements, then the model will be dependent on the characteristics of the device. If the device were a nominal device, a typical model would be obtained.

Once the data is collected, it is put into a file formatted in human-readable, ASCII text. The Golden Parser, also known as ibischk3, is used to check that the syntax and structure of the IBIS file follow the standard. As a last step, the designer should validate the model by correlating the simulation results with actual silicon measurements.

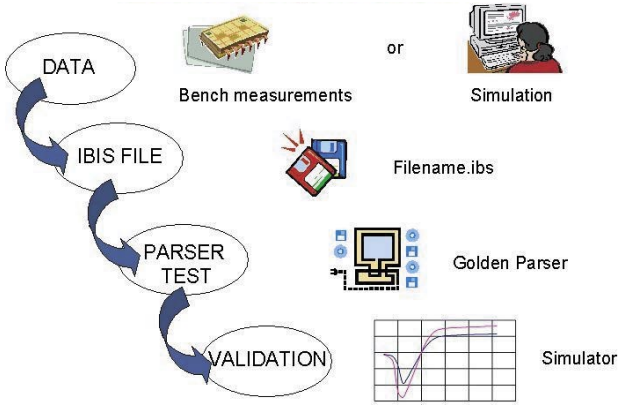


Figure 1. IBIS Model Generation Flow

REQUIRED DATA

The IBIS specification supports several types of inputs and outputs that can be modeled three-state, open collector, open drain, I/O, and ECL, for example. The first step is to identify the different types of inputs and outputs on the device and determine how many buffer designs are present. It should be noted that one model can be used to represent more than one input or output in an IBIS file. However, separate models are required if the C_Comp and package parameters are different.

THREE-STATE OUTPUT

Figure 2 shows the structure for a three-state output; the model can be viewed as a driver. It consists of a PMOS transistor and a NMOS transistor, two diodes for ESD protection, the die capacitance, and the package parasitics.

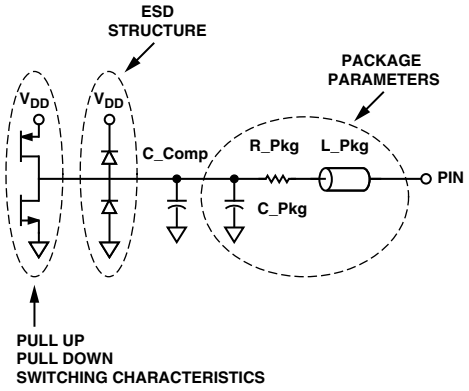


Figure 2. Three-State Output Buffer

The output model is characterized by the following dc electrical data, ac or switching data, and parameters:

1. Pull-Up and Pull-Down Curves
2. Power and GND Clamp Curves
3. Ramp Rate
4. Rising and Falling Waveforms
5. C_Comp
6. Package Parameters

Pull-Up and Pull-Down Curves

The pull-up and pull-down data define the drive strength of the device. These curves are obtained by characterizing the two transistors in the output. The pull-up data describes the I/V behavior when the output is in a logic high state (PMOS transistor on). On the contrary, the pull-down data shows the dc electrical characteristics when the output is in a logic low state (NMOS transistor on).

Data needs to be acquired from $-V_{DD}$ to $2 \times V_{DD}$. Even though this voltage range exceeds the maximum absolute ratings that semiconductor vendors would indicate in the specifications of the devices, this range covers the region where undershoot, overshoot, and reflections in the transmission line could happen. Therefore, drivers and receivers need to be modeled using this voltage range.

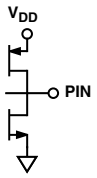


Figure 3. PMOS and NMOS Transistor in the Output

It should be noted that pull-down data is relative to GND, whereas pull-up data is V_{DD} relative since the output current depends on the voltage between the output and V_{DD} pins, and not the voltage between the output and ground pins. Thus, the pull-up data has to be entered in the IBIS file following the expression

$$V_{TABLE} = V_{DD} - V_{OUT}$$

Power and GND Clamp Curves

These curves are generated when the output is in a high impedance state. The GND and power clamp data represent the electrical behavior of the output when the GND clamp and the power clamp diodes are turned on, respectively. The GND clamp is active when the output is below ground, and the power clamp is active when the output is above V_{DD}. Data is taken from $-V_{DD}$ to V_{DD} for the GND clamp curve, and from V_{DD} to $2 \times V_{DD}$ for the power clamp curve. As it is for the pull-up data, the power clamp data needs to be relative to V_{DD}, and therefore the values entered in the file are obtained using the same expression shown above ($V_{TABLE} = V_{DD} - V_{OUT}$).

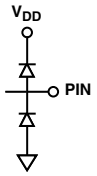


Figure 4. GND and Power Clamp Diodes

The GND and power clamp data need to be subtracted from the pull-up and pull-down data. Otherwise, the simulator takes this into account twice.

Table I. Sweep Voltage Ranges

I/V Characteristics	Range
Pull-Down	$-V_{DD}$ to $2 \times V_{DD}$
Pull-Up	$-V_{DD}$ to $2 \times V_{DD}$
GND Clamp	$-V_{DD}$ to V_{DD}
Power Clamp	$+V_{DD}$ to $2 \times V_{DD}$

Ramp Rate and Switching Waveforms

The ramp rate (dV/dt) describes the transition time when the output is switching from the current logic state to another logic state. It is measured at the 20% and 80% points with a default resistive load of 50 Ω.

The falling and rising waveforms show the time it takes the device to go from a high to low and from a low to high when driving a resistive load connected to ground and V_{DD}. For a standard push/pull CMOS, four different waveforms can be generated: two rising and two falling. In each case, one is with the load connected to V_{DD} and the other with the load connected to GND. However, it is very common to see only two of these waveforms in the model.

The ramp rate and the falling and rising waveforms include the effects of the die capacitance. Therefore, wrong results are generated if a simulator uses the C_Comp value as an additional load on the output. This is "double-counting" the effect of C_Comp.

As with the I/V curves, the effects of the package are not included.

C_Comp

This is the silicon die capacitance and does not account for package capacitance. It is the capacitance seen when looking from the pad back into the buffer. C_Comp is a key parameter, especially for receiver inputs. C_Comp should have a value for each of the different corners, min, typ, and max. The largest value of C_Comp will be under the max corner, and the smallest value will be under the min corner.

Package Parameters

R_Pin, L_Pin, and C_Pin are the electrical characteristics of resistance, inductance, and capacitance for each pin-to-buffer connection. The R_Pkg, L_Pkg, and C_Pkg are the lumped values for the overall package. As for the C_Comp parameter, the largest values are listed as max values and the smallest values are listed as min values.

Input Model

Figure 5 shows the structure for an input; the model can be viewed as a receiver. It consists of the two diodes for ESD protection, the die capacitance, and the package parasitics.

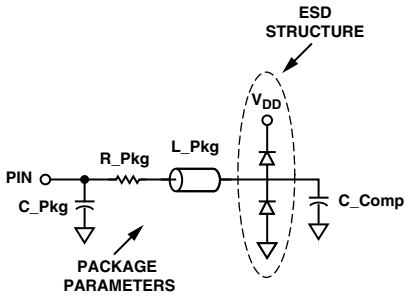


Figure 5. Input Buffer

These elements lead to the V/I curves that characterize an input. In this case, apart from the package parasitics and the C_Comp parameter, the model for an input contains the power and GND clamp data obtained from the ESD diodes. These curves are generated following the same procedure used for an output. The sweep voltage range will be $-V_{DD}$ to V_{DD} for the GND clamp and V_{DD} to $2 \times V_{DD}$ for the power clamp curve. In addition, because the power clamp data is relative to V_{DD}, it needs to be entered in the file as $V_{TABLE} = V_{DD} - V_{IN}$.

OTHER PARAMETERS

For an output model, there are some parameters that should be included in the file to perform posterior simulations for timing requirements. Those timing test loads and measurement points are the test load capacitive value (C_{REF}), the test load resistive value (R_{REF}), the test load pull-up or pull-down reference voltage (V_{REF}), and the output voltage measurement point (V_{MEAS}). They are the same test loads the semiconductor vendor uses when specifying the propagation delay and/or output switching time of the device.

For an input, the V_{INL} and V_{INH} parameters should be included. These are the input voltage thresholds for the input and can be obtained from the data sheet.

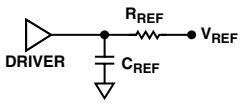


Figure 6. Connections for C_{REF}, R_{REF}, and V_{REF}

WHAT DOES AN IBIS FILE LOOK LIKE?

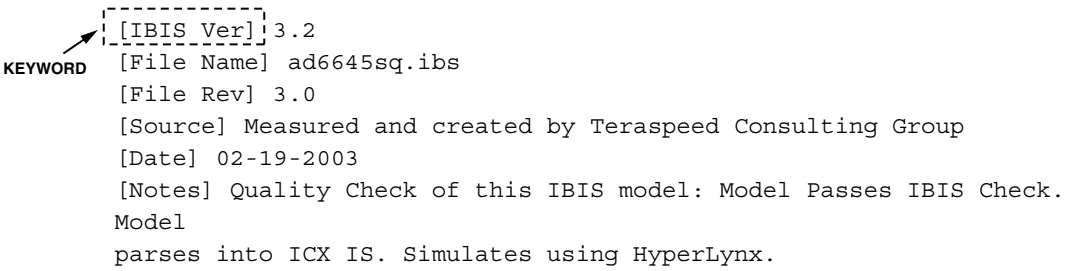
An IBIS file is not an executable file; it is a file which collects all the data that describes the electrical behavior of a device and can be used in a simulator. An IBIS file consists of three main parts.

1. The header or general information about the file, the device, and the company
2. The device name, pinout, and pin-to-buffer mapping
3. I/V and V/T data for each model

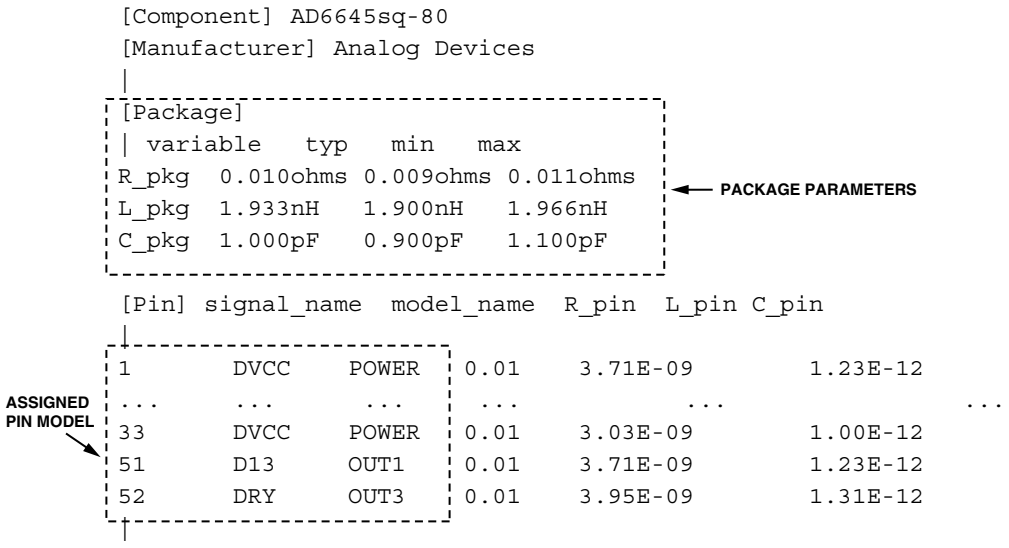
An IBIS file can contain more than one device characterized. In that case, points 2 and 3 would be repeated as many times as devices are included.

The following section shows the main parts of an IBIS file. The words in brackets are called keywords; some of them are optional and others have to be included.

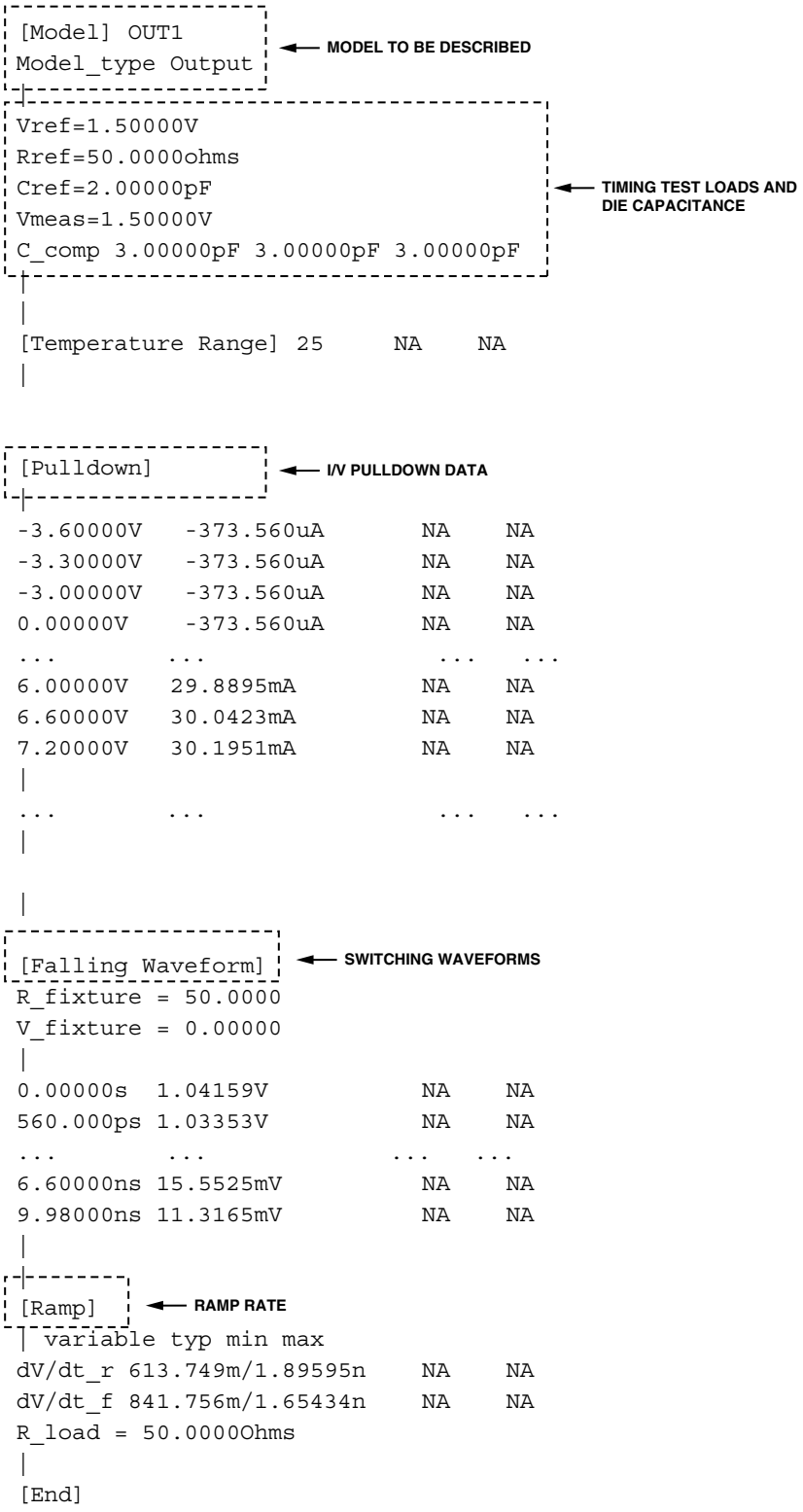
Header and General Information



Component and Pin Information



Model Data


[Model] OUT1
Model_type Output
MODEL TO BE DESCRIBED
Vref=1.50000V
Rref=50.0000ohms
Cref=2.00000pF
Vmeas=1.50000V
C_comp 3.00000pF 3.00000pF 3.00000pF
TIMING TEST LOADS AND DIE CAPACITANCE
[Temperature Range] 25 NA NA
[Pulldown]
I/V PULLDOWN DATA
-3.60000V -373.560uA NA NA
-3.30000V -373.560uA NA NA
-3.00000V -373.560uA NA NA
0.00000V -373.560uA NA NA
... ...
6.00000V 29.8895mA NA NA
6.60000V 30.0423mA NA NA
7.20000V 30.1951mA NA NA
... ...
[Falling Waveform]
SWITCHING WAVEFORMS
R_fixture = 50.0000
V_fixture = 0.00000
0.00000s 1.04159V NA NA
560.000ps 1.03353V NA NA
... ...
6.60000ns 15.5525mV NA NA
9.98000ns 11.3165mV NA NA
[Ramp]
RAMP RATE
variable typ min max
dV/dt_r 613.749m/1.89595n NA NA
dV/dt_f 841.756m/1.65434n NA NA
R_load = 50.0000Ohms
[End]

MODEL VALIDATION

Once the IBIS file has been created, it must be validated. The Golden Parser, also known as ibischk3, is a program that checks to ensure the syntax and structure of the file complies with the IBIS specification. This program is available for free from the IBIS website at http://www.eigroup.org/ibis/tools.htm. Next, the user should make a visual inspection of the I/V and V/T curves generated from the file and make sure that the results are as expected. This can be done using a Visual IBIS Editor from Innoveda, available at the IBIS website at no charge.

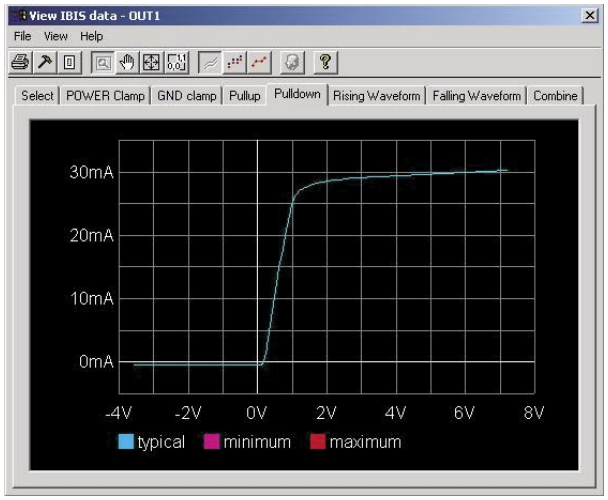


Figure 7. Pull-Down Curve

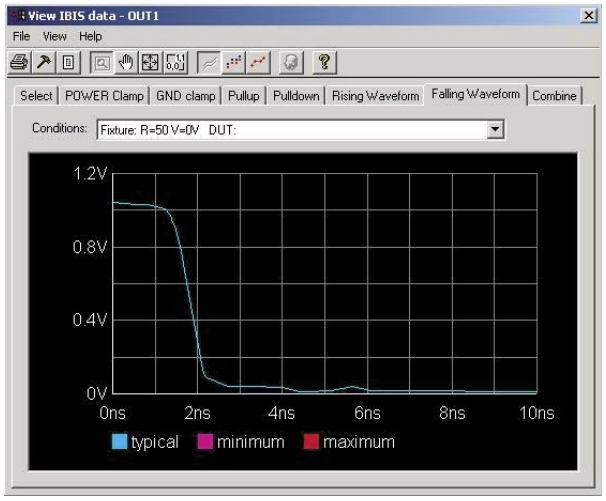


Figure 8. Falling Waveform

Following this, the model should be run with different standard loads in one of the IBIS simulators offered from the different EDA vendors. Some of these vendors include HyperLynx, Cadence, and Avanti Corporation. The results should be compared against a transistor-level reference simulation (SPICE simulation) using the same loads. Finally, the IBIS simulation results should be correlated with actual silicon measurements.

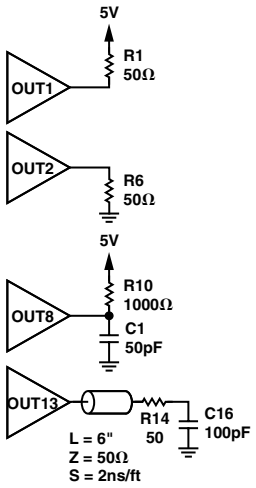


Figure 9. Simulating the Model with Different Loads

Because the quality of the models offered by semiconductor companies varies from vendor to vendor, the IBIS Quality Committee has developed a quality checklist to define different "quality" levels, as shown in Table II. In addition, the IBIS Accuracy Handbook describes methods for correlation with simulation and measurement. The main goal of this is to provide accurate and quality models so users can feel confident they are getting reliable data.

Table II. Quality Levels in the IBIS Quality Checklist

Quality Level	Description
Level 0	Passes ibischk, Golden Parser
Level 1	Complete and correct as defined in checklist documentation
Level 2a	Correlation with simulation
Level 2b	Correlation with measurement
Level 3	All of the above

CONCLUSION

IBIS models seem accurate, easy to generate, and compatible with a wide range of simulation platforms.

From the point of view of a semiconductor vendor, IBIS is a standard specification that has solved the issue of proprietary information surrounding SPICE models. IBIS models can be generated from SPICE models with a SPICE to IBIS conversion. Translators can be found on the IBIS website. From a user point of view, the points outlined above should improve the availability of models. However, there is still work to be done to generate more of these models at a good level of quality.

REFERENCES

IBIS (I/O Buffer Information Specification) *Version 4.0*

I/O Buffer Modeling Cookbook, IBIS Open Forum, Sept. 1997

IBIS Model Process For High-Speed LVDS Interface Products, National Semiconductor Corporation, Nov. 2002

IBIS Models: Background and Usage, Actel Corporation, May 2002

Effective Signal Integrity Analysis using IBIS Models, Syed B. Huq, DesigCon 2000

Introduction to IBIS Modeling of Fiber Optic Transceivers, Mark Chang, Agilent Technologies

Generating Accurate Behavioral Models of I/O Buffers, Thomas Fisher, Texas Instruments

Practical Issues with IBIS Models, Bob Ross, Inteconnectix Unit of Mentor Graphics Corporation

Ease System Simulation with IBIS Device Models, Syed Huq, 1996

IBIS Models for Signal Integrity Applications, Bob Ross, Syed Huq, John Powell, Sept. 1996

IBIS Behavioral Models, Micron, 1996

I/O Buffer modeling spec simplifies simulation for high-speed systems, Derrick Duehren, Will Hobbs, Arpad Muranyi, Robin Rosenbaum, Sept. 1994

AN04780–0–4/04(0)