Issues Combining Buffer and Interconnect Model Formats

Michael Mirmak
Intel Corporation
Chair, EIA IBIS Open Forum
michael.mirmak@intel.com

IBIS Summit
Beijing, China
September 11, 2007
Legal Disclaimer

THIS DOCUMENT AND RELATED MATERIALS AND INFORMATION ARE PROVIDED "AS IS" WITH NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION, OR SAMPLE. INTEL ASSUMES NO RESPONSIBILITY FOR ANY ERRORS CONTAINED IN THIS DOCUMENT AND HAS NO LIABILITIES OR OBLIGATIONS FOR ANY DAMAGES ARISING FROM OR IN CONNECTION WITH THE USE OF THIS DOCUMENT.

Performance tests and ratings are measured using specific computer systems and/or components and reflect the approximate performance as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance.

Intel may make changes to specifications, product descriptions, dates and plans at any time, without notice.

Copyright © 2007, Intel Corporation. All rights reserved.
An Additional Disclaimer

–The following information is presented as the opinion of one person at Intel. This presentation does **not** necessarily represent Intel policy, commitments or preferences.

–This is **not** presented on behalf of the IBIS Open Forum and does **not** represent the official IBIS Open Forum direction.
Agenda

– The System Viewpoint
– Buffer Model Formats
– Package and Interconnect Model Formats
– Combining Components into a System
  – Summary of Today’s Industry Options
– Steps Toward a Unified Solution
The Number of Model Formats is Increasing!

- IC designers can model **buffers** in many ways
  - Proprietary SPICEs (transistor or behavioral)
  - Traditional IBIS (3.2/4.0): table-based
  - Verilog-A (for analog-only simulation)
  - Verilog-AMS/VHDL-AMS (for mixed-signal)
  - Multi-lingual IBIS 4.1/4.2 (IBIS+*-AMS, IBIS+Verilog-A, IBIS+SPICE)

- There are just as many ways to model **packages & interconnects**
  - Proprietary SPICEs (lumped-element, RLGC)
  - Traditional IBIS (3.2/4.0) for packages
  - S-parameters (Touchstone*, CITI*)
  - ICM (RLGC or Touchstone*)

- ... and many ways to **combine** these models in a system
  - Both system and IC vendors simulate system designs
  - System sims involve all of the above plus other components

How well do all these options work together?
SI Models - The System View

Buffer (Device) Models

Board Trace (Interconnect) Models

Package Trace (Interconnect) Models

SerDes Example

... plus the language or format that connects them all

*Other names and brands may be claimed as the property of others
Buffer Models

• “Traditional” table-based IBIS 3.2/4.0 has some advantages
  – Standard and enjoys widespread support
  – Fast (behavioral) and protects IP
  – Supports integration with layout (pin and signal information)
  – Free syntax checking (IBIS Golden Parser)
  – Reasonably simple to learn

• Format has increasing issues
  – Best for single-ended, time-domain design
  – Frequency-domain support is weak (e.g., C_comp as buffer capacitance)
  – Also weak for SerDes: equalization, frequency-dependence, jitter
“SPICE” vs. “AMS” vs. Algorithmic Models

- Several options exist to improve buffer model support

**SPICE**
- Can be used behaviorally
- Familiar to most engineers
- Versions implemented in most EDA tools
  - Not standard
  - A format, not a language
  - Analog-only
  - Not suited to algorithmic modeling
  - Still data-driven

**“AMS”**
- VHDL-AMS, Verilog-AMS
- Mixed-signal (digital + analog)
- Also Verilog-A (analog-only)
- Industry standard
- Languages, not just formats
- More familiar to IC designers than SI/system designers
- Expensive to implement for lower-cost EDA tools
- Languages are not easily cross-ported
- HDLs, not general purpose mathematical languages

IBIS 4.2 supports Verilog-AMS, VHDL-AMS, Verilog-A and (Berkeley) SPICE; IBIS-ATM working on SerDes algorithmic models

*Other names and brands may be claimed as the property of others*
Package modeling through IBIS 4.2

- **[Package] keyword**
  - Required for each component
  - Contains corner values for \( R_{\text{pkg}} \), \( L_{\text{pkg}} \) and \( C_{\text{pkg}} \)

- **[Pin] keyword**
  - Each pin can have a distinct \( R_{\text{pin}} \), \( L_{\text{pin}} \) or \( C_{\text{pin}} \) value
  - Only a single value can be used (no corners)

- **[Package Model] keyword** (also Electrical Board Description)
  - Can reference an external file or data within the same .IBS file
  - Two methods available currently in IBIS
    - Coupled, single-matrix RLC description
      (full, banded, and sparse matrix formats available)
    - Uncoupled multi-section description using RLC, length

No coupling for multi-section packages, no AC loss

*Not suitable for today’s fastest designs!*

*Other names and brands may be claimed as the property of others*
BGA Package Structure

• A typical application

• BGA (Ball Grid Array) Package structure
  - Bond wire (usually gold): connects die pad to traces
  - Package trace: lossy routing path in package FR4
  - Via: connects trace to solder ball
  - Solder ball: attach point for device to PCB
  - Plating par: lossy stub left from separation of packages

Can we describe this today using EBD, IBIS or PKG? Not with adequate loss and coupling.
**SPICE and ICM**

- Many vendors use SPICE to address IBIS package shortcomings
  - Most SPICEs support lumped or distributed RLGC
  - Proprietary! One model set may not work across multiple tools
- ICM (IBIS Interconnect Modeling specification) available but new
  - Standard, with industry support picking up
  - A separate specification – no direct connection to IBIS yet
  - Not intended to support series components (e.g., series capacitance)

---

**Diagram:**

- Described by
  - [Begin ICM Model]
  - (path description)
  - [End ICM Model]

- Described by
  - [Begin ICM Section]
  - (RLGC or S-params)
  - [End ICM Section]
**Package & Interconnect Formats**

**S-Parameters**
- Standard (2+ formats!)
- Frequency-dependent
- No pin or other connection information

**IBIS (.PKG)**
- Per-pin & global lumped data
- Distributed single-line data
- Single-segment matrices
- Industry standard

**RLGC**
- Usually proprietary
- Most are frequency-dependent
- Can describe lumped or distributed circuit
- May not be supported across multiple tools

**ICM 1.1**
- Industry standard (ANSI)
- Freq.-dependent RLGC matrices
  - *Loss, coupling, length variation*
- Link to Touchstone* S-parameter files
- Supports images, edge rate information
- Modular pinlist, electrical data

*Other names and brands may be claimed as the property of others*
Combining Models – A Dilemma

Full System Description

**Standard Format**
Same as buffer, interconnect...

**EDA tool specific**

**Standard Format**
Different from buffer, interconnect...

Can one format support all elements of the system?

Do you create a new description for each tool? How many tools?

How many formats do you have to learn? How many do your tools support?

*Other names and brands may be claimed as the property of others*
SPICE as a Topology Format

- Some IC vendors want to provide complete system “decks”
- Using proprietary SPICEs is popular, but faces several issues
  - Inconsistency in format
    - *SPICE is not standard! Elements and features vary across SPICEs*
    - *Some analysis functions are proprietary (e.g., S-parameters)*
  - Lack of expandability
    - *Users cannot define new elements or analyses, only new subcircuits*
  - Inconsistency in results
    - *With different engines, the same model can give different results*
  - Integration with layout is not automatic

*All are barriers to using SPICE to represent systems across tools*
# Models in Combination

<table>
<thead>
<tr>
<th>Model Type</th>
<th>Buffer</th>
<th>Interconnect (Board, Package, etc.)</th>
<th>“System Assembly”</th>
</tr>
</thead>
<tbody>
<tr>
<td>Behavioral SPICE</td>
<td></td>
<td>Which SPICE? Frequency-dependent interconnect? Features compatible across several tools?</td>
<td></td>
</tr>
<tr>
<td>Traditional IBIS</td>
<td></td>
<td>Weak C_comp, no core-side information...</td>
<td>No distributed, frequency-dependent interconnect</td>
</tr>
<tr>
<td>IBIS 4.2 + Berkeley SPICE</td>
<td>✔️</td>
<td>No distributed, frequency-dependent interconnect</td>
<td>not intended</td>
</tr>
<tr>
<td>IBIS 4.2 + *-AMS</td>
<td>✔️</td>
<td>Need custom transmission line libraries</td>
<td>not intended</td>
</tr>
<tr>
<td>ICM 1.1</td>
<td>☒️</td>
<td>not intended</td>
<td>No active models, limited series components</td>
</tr>
<tr>
<td>Verilog-A</td>
<td>✔️</td>
<td>Need custom transmission line libraries</td>
<td>✔️</td>
</tr>
</tbody>
</table>

*Other names and brands may be claimed as the property of others*
Steps Toward a Unified Solution

• Options exist for a unified buffer+interconnect+system format
  – e.g., Verilog-A can be treated as a SPICE-like standard analog language
  – Standard transmission line libraries would be required to support
    interconnects under Verilog-A

• The IBIS Open Forum is addressing several of these issues
  – IBIS-ATM is adding algorithmic model support for SerDes
  – IBIS-ICM links are being developed for a future version of IBIS
  – A standard SPICE superset syntax has been proposed

• Market forces will determine the most popular combination
  – e.g., IBIS 4.2 + ICM 1.1 + Verilog-A might be possible, but it might not
    be least expensive or easiest to learn

Make your preferences known!
*Other names and brands may be claimed as the property of others