The 3S Proposal: A SPICE Superset Specification for Behavioral Modeling

Michael Mirmak
Intel Corporation

June 5, 2007

Legal Disclaimer

An Additional Disclaimer

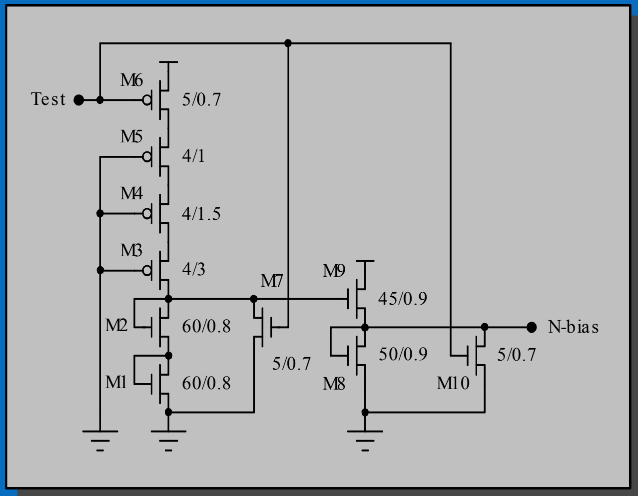
- The following information is presented as the opinion of one person at Intel. This presentation does not necessarily represent Intel policy, commitments or preferences.

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

Agenda

- The What and Why of SPICE
- Analog Behavioral Modeling Today
- Pros/Cons of SPICE in General
- SPICE Compatibility
  - Elements as a Case Study
- Outline of a Behavioral SPICE Specification
  - What it must include
  - What it should exclude
- What a Standard SPICE Wouldn't Address
  - Alternatives
- Summary

SPICE Review: A Tool and Modeling Method

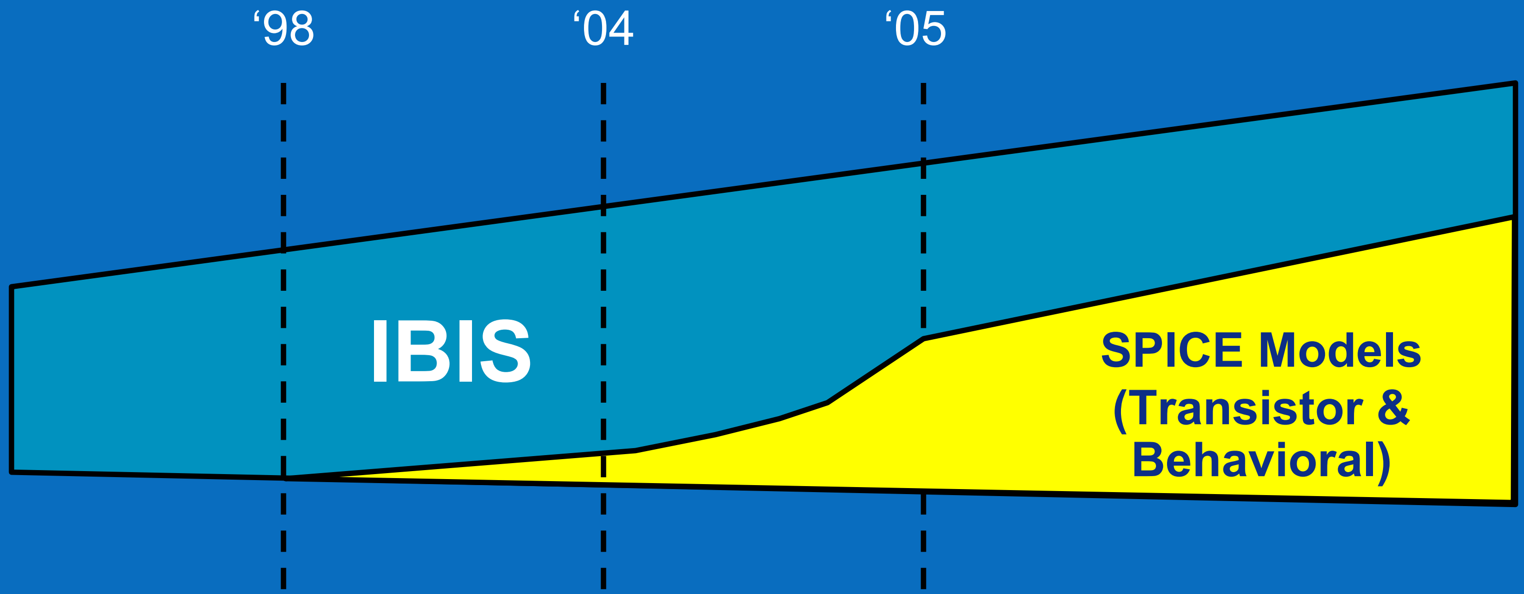



```
************************************************
.MODEL NMOS NMOS
+ (LEVEL=3          UO=400.0         VTO=1.00
+  TPG=1            TOX=15E-9        NSUB=1.00E17
+  VMAX=200.0E3     RSH=50           XJ=100.0E-9
+  LD=120.0E-9      DELTA=20.0E-3    THETA=0.10
+  ETA=10.0E-3      KAPPA=20.0E-18   PB=0.40
+  CGSO=2.00E-10    CGDO=2.00E-10    CJ=0.30E-3
+  CJSW=0.20E-9     MJ=350.0E-3      MJSW=200.0E-3)
************************************************
```

- "Simulation Program with Integrated Circuit Emphasis"
- Developed by Donald Pederson at UC Berkeley in 1960s
- Not standardized, but the general format is widely recognized
- Berkeley still develops process models (BSIM3, BSIM4, etc.)
- SPICE 3F6 program available from Berkeley
  http://bwrc.eecs.berkeley.edu/Classes/IcBook/SPICE/
- Many commercial SPICE flavors are available
- Most IC vendors have their own flavors, for their own processes
  - *Usually not compatible with any commercial SPICE variant*

The Need for Analog Alternatives to “Old” IBIS

For a time, traditional IBIS (3.2/4.0) was “going out of style”

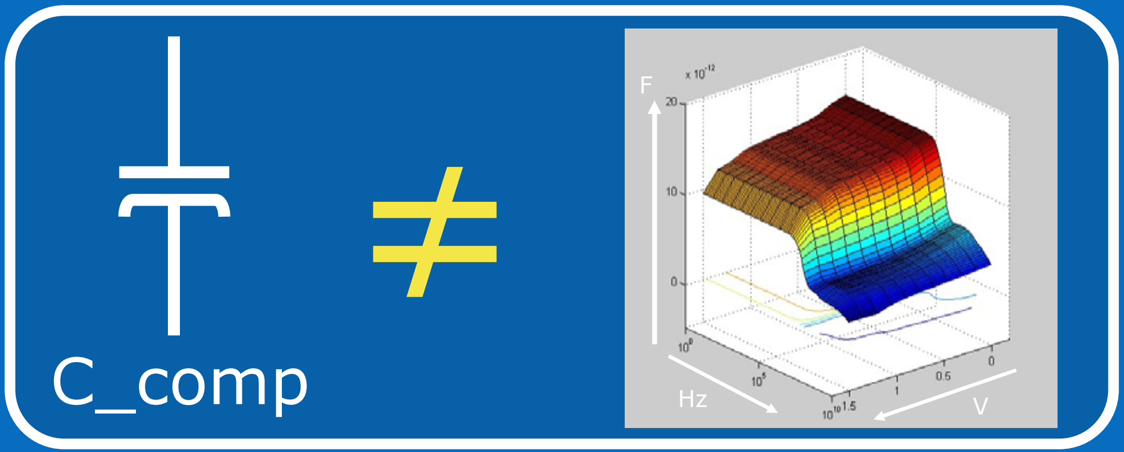


- IBIS is well-tuned to single-ended, simple designs (e.g., CMOS)
- IBIS 3.2/4.0 increasingly hard to use when modeling complex buffers
  - SerDes buffers with multi-tap equalization
  - Complex impedance modeling (frequency- and voltage-dependent C_comp)
- SPICE returned briefly as a popular alternative
  - IBIS Macromodeling Task Group emerged to support IBIS+ behavioral SPICE

diagram courtesy T. Westerhoff; used with permission

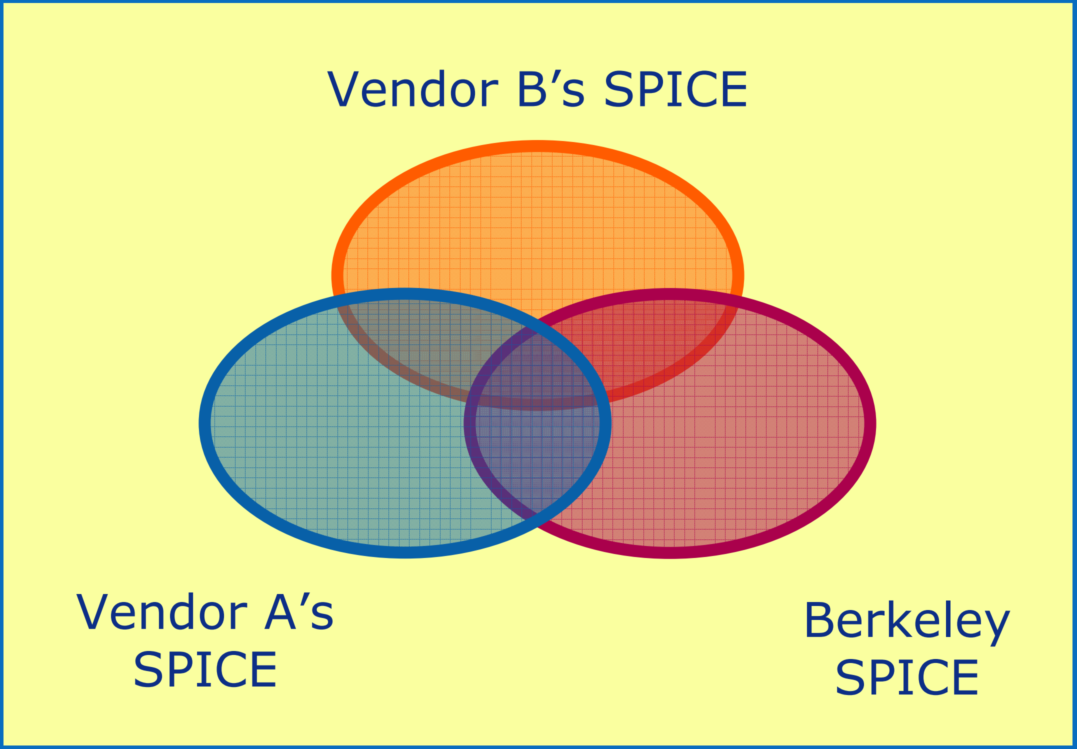
Why Use Behavioral SPICE for Signal Integrity?

- For buffers, address "traditional" (3.2/4.0) IBIS shortcomings above
  - IBIS 4.1/4.2 supports Berkeley SPICE code



- Many vendors use SPICE to address IBIS <u>package</u> shortcomings
  - IBIS package support is clearly inadequate (see previous IBIS Summits)
  - ICM available but new; automated extraction for packages WIP
  - Integration into IBIS still not done
- IC/IP Vendor Needs
  - For complex systems, device/package models are not enough
  - Many customers expect full system <u>decks</u>, for analysis and correlation

Which SPICE?



- In early 2007, IBIS-ATM took up the challenge
  - Asked a major proprietary SPICE vendor to release its manuals to the public
  - This was politely declined

Can we define a standard SPICE superset for behavioral modeling, "3S"?

"SPICE" Pros and Cons

- Something more than transistor models or traditional IBIS is needed
- How best to address advanced modeling and analysis needs?

For

- Can be used behaviorally
- Simple to understand
- Familiar to most engineers
- Versions implemented in most EDA tools
- Fairly flexible: if you can describe it with electrical elements, you can describe it with SPICE

Against

- Not standard
- A format, not a language
- Analog-only
- Not suited to algorithmic modeling (numeric processing instead of V, I analysis)
- Still data-driven

SPICE has a value for behavioral modeling. It is more flexible than table-driven IBIS with wider support at lower cost

Problems of Implementation: Elements

- Some Berkeley SPICE definitions are effectively universal…

Element Prefix	Element Definition
C-element	capacitor
E-element	VCVS
F-element	CCVS
G-element	VCCS
H-element	CCCS
K-element	mututal inductance/transformer
L-element	inductor
R-element	resistor
T-element	lossless transmission line
V-element	voltage source
X-element	subcircuit call

- But evolution from Berkeley SPICE has caused deviations
  - The elements below are completely different under different implementations

Element Prefix	Conflicting Element Definitions	
B-element	Non-linear dependent source	IBIS element
O-element	Lossy transmission line	Opamp
P-element	Semiconductor Resistor	Port element
S-element	Switch element (voltage controlled)	Multiport S-parameter t-line
W-element	Switch element (current controlled)	RLGC transmission line

  - These elements are implemented in some tools but not all

Element Prefix	Element Definition
I-element	Current source
N-element	Lossy transmission line
U-element	Lumped lossy transmission line
Y-element	Macro element
Z-element	Frequency-dependent component

SPICE Problems

- Inconsistency
  - As shown above, even basic elements are inconsistent between SPICEs
  - Shared element functions are often inconsistent (e.g., V sources)
  - Some analysis functions are proprietary (e.g., S-parameter generation)

- Lack of expandability
  - Users cannot define new elements or analyses, only new subcircuits
    - *At least one proprietary SPICE defines a "macro" element, but the macros are under the control of the tool vendor, not the user or author*

- Lack of control
  - In an age of 10 Gbps signals, obtaining inconsistent results for "standard" models is no longer tolerable

These have been the hurdles to widespread SPICE usage for behavioral modeling

What Would A Standard Behavioral SPICE Look Like?

- Any standard SPICE would have to support the following:
  - "First-letter" name + node + function syntax for elements
    - *Including the truly common elements and formats*
  - Subcircuit syntax and approach
  - "Dot" syntax for functions, parameters and analysis types (e.g., ".OPTIONS")
    - *Common analysis types and functions TBD*
  - Other common structural assumptions
    - *No ordering requirements aside from title and .END*
- What would be excluded from the standard definition?
  - Transistors and other active elements
    - *Process files and "LEVEL" would also be excluded*
    - *Do we need diodes?*
  - Any element name that is inconsistent among proprietary implementations
  - Functions or capabilities outside circuit solving
    - *Field solvers, digital logic functions, links to other languages or tools*

This is readily achievable

What Would A Standard Behavioral SPICE <u>Add</u>?

- Major additions
  - The A-element
    - *Undefined in any SPICE, based on informal survey*
    - *Could be used as a specification-level "catch-all" macro alias*
    - *Allow 3S elements new to some SPICEs to be easily implemented through extended parsing in more sophisticated SPICEs*
    - *Specification would control associated function definitions*

	→	**`Nxxxx n1 n2 …`**	SPICE 1
`Axxxx macroname n1 n2 …`	→	**`Wxxxx n1 n2 …`**	SPICE 2
	→	**`?xxxx n1 n2 …`**	SPICE 3

  - .COMPAT/.UNCOMPAT switch
    - *"Wrapper" for standard behavioral SPICE text*
    - *Netlist-level flag to tool to enforce behavioral SPICE standard rules*
    - *Enables use of 3S code within a proprietary netlist*

These should not represent a significant industry burden

Issues

- A format-only specification may be insufficient
  - Different tools may still interpret data differently
  - Complex SI measurements (DDR2) would still be painful
  - IC/IP vendors want more control over analysis methods (algorithms)
    - *e.g., causality and passivity enforcement on transmission lines*
  - This favors language-based rather than format-based approaches

- Can we truly exclude active devices (transistors)?
  - BSIMx is still highly popular, useful and effectively a standard

- Administrative burdens are considerable
  - Maintenance, including adding new macro functions, would be required
  - Development of a syntax parser would also be needed

Is this still worth doing?
Are there alternatives?

"The Split"

- Both models and tools have to address varying market segments
- IBIS-ATM work shows that model creation and use are growing apart
  - System designers need relatively simple models in inexpensive tools
  - IC designers need more detail on the digital, data-processing side

IC Vendors

- Higher-cost, highly capable tools
- Digital and analog support needed
- Same environment for model development and testing
- Portability less important until export stage

- Lower-cost tools
- Analog focus (no digital support needed)
- Model use, not development
- With multiple suppliers, model portability is key

System Designers

A Better Solution than SPICE?

- Verilog-A is already a SPICE superset
  - Verilog-A is already standard, widely supported (IBIS 4.2!) and available
    - *At least one major SPICE vendor supports Verilog-A today*
  - Supports analysis control <u>without requiring digital language support</u>
    - *IC vendors get control over model interpretation*
  - Addresses both the element naming and function definition problems
    - *Element names are separate from instance names*

```
IBIS_R  #(.Rval(R_val), .Scale(Scale_val))  R1  (Node1, Node2);
```

Anyone recall the IBIS Macromodel library using Verilog-A for SPICE?

- Transmission lines and S-parameters are a significant omission
  - An enhanced Verilog-A could support SI features, system netlists

Is Verilog-A a more compelling <u>analog</u> modeling solution than SPICE?

If We Go Ahead with 3S, Next Steps…

1. Outline and document the basic features, common to most SPICEs
   - Element names and formats
   - Element functions
   - Options and analysis types
   - Other key syntax and netlist structural assumptions
2. Define the new features, unique to 3S
   - The A-element syntax
   - A-element macro names and data formats
   - .COMPAT/.UNCOMPAT usage
3. Define analysis requirements (“the hard part”)
4. Create illustrative test cases
5. Write the relevant specification documents
   - Likely in parallel with the steps above

Delaying work on a unified analog solution means more “lost ground” for IBIS and standards

Leap ahead™