

IBIS Modeling for High Speed Designs

**Dr. Lynne Green
Green Streak Programs
April 2004**

**lgreen22@mindspring.com
<http://www.greenstreakprograms.com>**

Overview

- Morning:
 - IBIS Specification and IBIS Models
 - Validation Methodology
 - Common Problems and Solutions
- Afternoon:
 - High-speed modeling techniques
 - Differential buffer models
 - Package models

Acknowledgements

- IBIS training materials
 - Arpad Muranyi, Intel Corp.
 - JEDEX 2003 IBIS Workshop, Lynne Green
(then at Cadence Design Systems)
- IBIS Summit Presenters
 - Luca Giacotto, Ecole Doctorale EEATS
 - Arpad Muranyi, Intel
 - Jim Bell, SiQual
 - Barry Katz and Doug Burns, SiSoft

Links

- IBIS Web Site

- Home Page <http://www.eigroup.org/ibis/ibis.htm>
- Specifications <http://www.eigroup.org/ibis/specs.htm>

Also Cookbook and BIRDs

- Parser <http://www.eda.org/pub/ibis/ibischk3/>
- IBIS Summit papers <http://www.eigroup.org/ibis/articles.htm>

Also Training Materials

- Quality Checklist <http://www.sissoft.com/ibis-quality/checklist/>

- FREE Model Review Service

Email Reflectors

- ibis-users and ibis

ibis-request@eda.org

- SI-list

si-list-request@freelists.org

- Send email with “subscribe” in the subject line

AM Overview

- IBIS Specification
- IBIS File Structure
- IBIS Component
- IBIS Models
- AMS Models
- IBIS Model Creation
- Validation Methodology
- Identifying Problems and Solutions

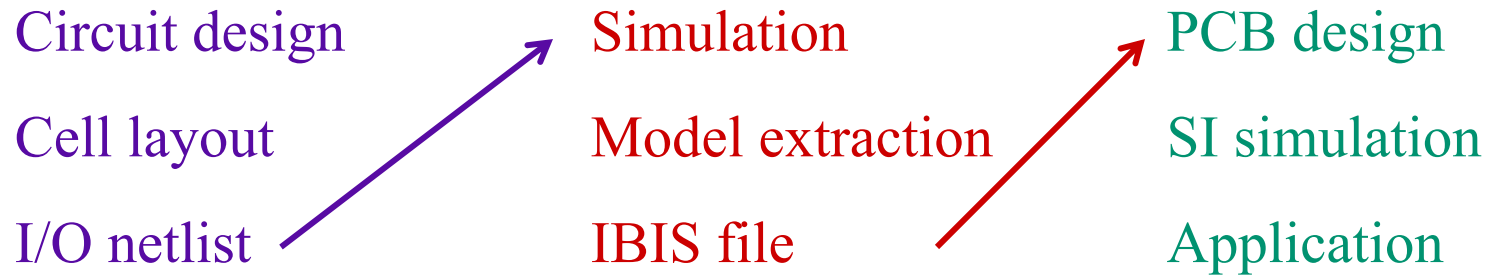
The Cost of Modeling Is Relative Compared to cost of systems failure



Images Courtesy NASA/JPL-Caltech

Creation and Use of IBIS Models

In a typical design flow



Information gets lost each time data is thrown “over the fence”. One common item is the diode resistance (RS in SPICE).

IBIS: I/O Buffer Interface Specification

- Support fast signal integrity simulation
 - Reflections and delays
 - Overshoot and ringing
- Provide for portable model data
 - I/O buffers, series elements, terminators, packages
 - SPICE models are not portable
- Protect intellectual property
 - Protect circuit and process IP
 - Models can be built from test-bench data

EIA/ANSI 656-A

I/O Buffer Interface Specification

- Version 1.0 in 1993
 - Intel, Cadence Design Systems, HyperLynx, Quad Design
- Version 4.1 in 2004
 - Added support for SPICE and AMS models
 - About 30 member companies (Model makers, users, EDA)
- ICM (IBIS Interconnect Model) spec in 2003
- Publicly available parsers
 - IBIS 4.0 and ICM 1.0 parsers available
 - IBIS 4.1 parser coming

IBIS Model Quality

What companies deliver

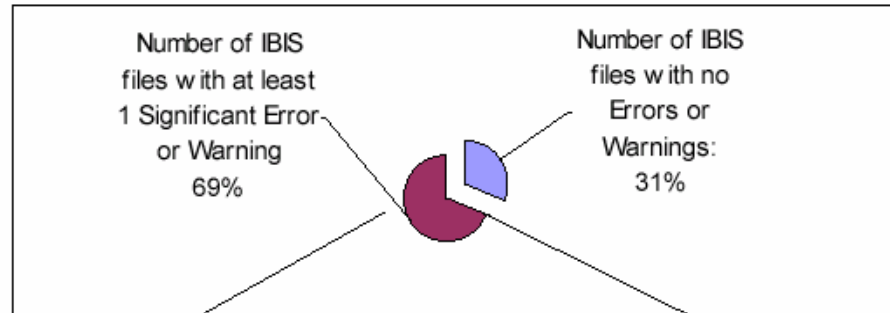
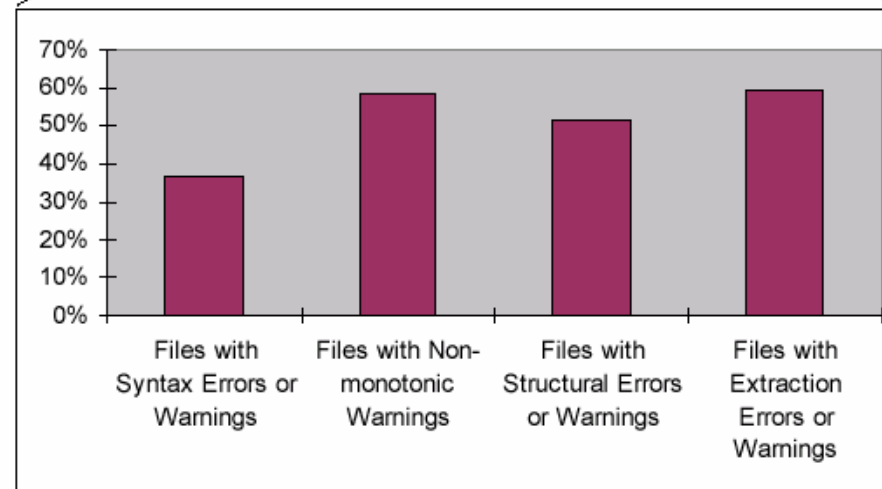


Figure 3-2: Ratio of Bad IBIS Files to Good IBIS files



“A Critique of IBIS Models Available for Download on the Web”, SiQual (IBIS Summit, 2002)

Creating IBIS Models

- s2ibis2 and s2ibis3 (usually customized)
- IBISCenter
- Cadence's Model Integrity
- Various other tools available
- Text editors (cut/paste values into columns)
- Scripts (custom flow)

The IBIS Specification

- Syntax
 - Keywords, sub-parameters, tables
- Data interpretation
 - I-V tables for pullup and pulldown
 - I-V tables for power and ground clamps
 - V-t tables
- Typ/Min/Max ordering
 - Different from datasheets
 - Package: use Typ, Min value, Max value
 - Models: use Typ, Slow/Weak, Strong/Fast

The IBIS Specification

Typ, Min, Max

- Min Corner
 - Weakest current
 - Slowest edge rates
 - Lowest voltage
- Typ Corner = Nominal
- Max Corner
 - Strongest current
 - Fastest edge rates
 - Highest voltage

The IBIS Specification

Typ, Min, Max

- Typ = Nominal voltage, temperature, process
- CMOS
 - Min @ min voltage, max temperature, and slow process
 - Max @ max voltage, min temperature, and fast process
- Bipolar
 - Min @ min voltage, min temperature, and slow process
 - Max @ max voltage, max temperature, and fast process

Important note: Temperature is die temperature, not ambient.
Important in setting up SPICE simulations.

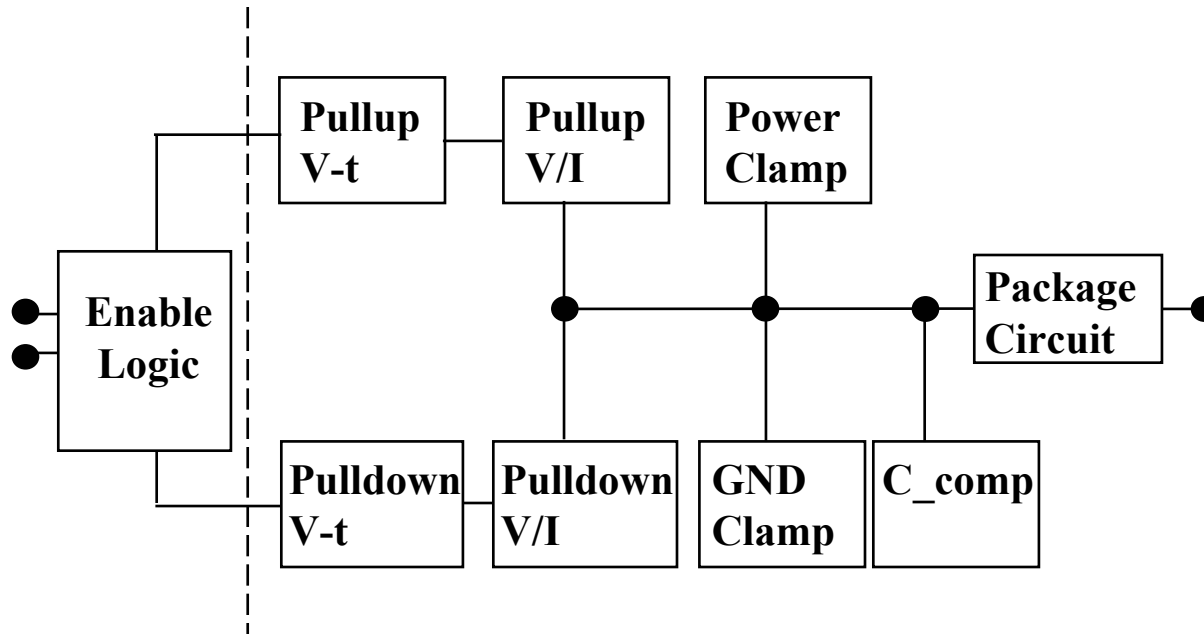
IBIS Syntax

- Keywords
 - Enclosed in []
 - Use “ ” or “_”
 - Case insensitive
- Sub-parameters
- Names
 - Components, pins
 - Signals, models
 - Layout tool naming
- Comment character

```
[Component]      XYZ
[Manufacturer]    Nobody
[Package]
| variable      typ      min      max
R_pkg           0.10     0.05     NA
L_pkg           1.80n    1.0n     3n
C_pkg           0.50p    NA       1p
[Pin]
|pin_name      signal_name  model_name
1              trans1       demo1
B2             GND          GND
C1             VCC1         POWER
D2             NC           NC
```

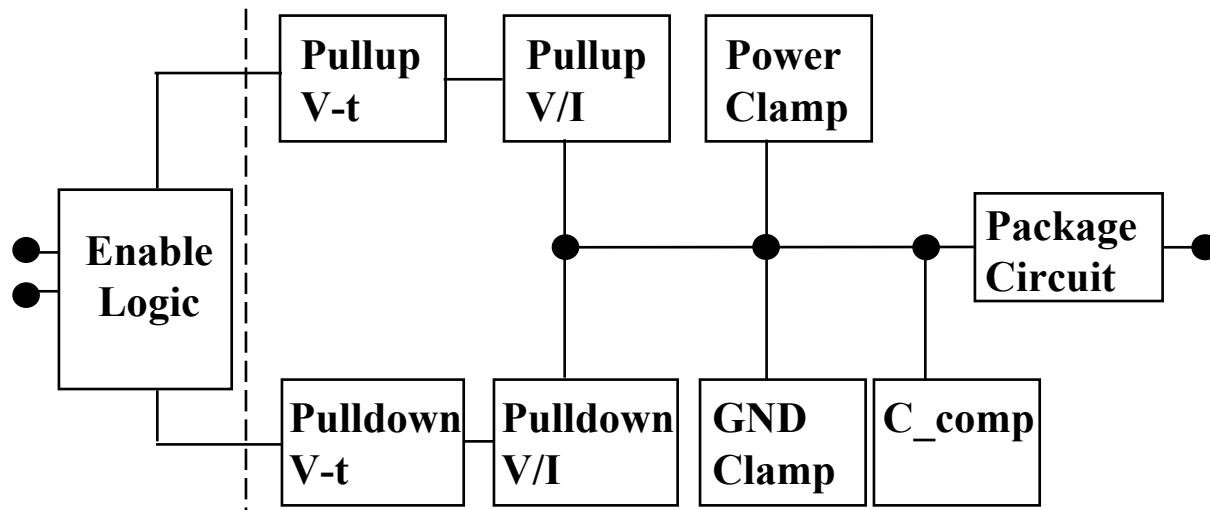

IBIS Data Interpretation

- $I_{dn} = [GND\ clamp] + [Power\ clamp] + [Pulldown]$
- $I_{up} = [GND\ clamp] + [Power\ clamp] + [Pullup]$
- $I_{rcvr} = [GND\ clamp] + [Power\ clamp]$



IBIS Data Interpretation

- Output transitions (dV/dt)
- V-t tables preferred
- [Ramp] values are used before simulation
- [Driver Schedule] for multiple output stages



IBIS Data Interpretation

- Output transitions under loading conditions
- Multiple V-t tables are allowed
- Often get good accuracy with only one set

<http://www.ntu.edu.sg/home/ehntan/glsvlsi.zip>

<http://www.sigrity.com/papers/ectc96/DOectc96ibis.htm>

IBIS File Structure

- Header
- Comments
- Component data
 - One or more components
 - Pin, signal, model, package
 - Diff pin pairs, etc.
- Model data
 - One or more models

IBIS File Structure Header

- IBIS version
 - Highest version supported
 - First keyword in file
- File properties
- Date and file revision
- Legal disclaimer
- Copyright
- Documentation

```
[IBIS Ver] 4.0
[File Name] mine.ibs
[File Rev] 0.0
[Date] April 1, 2004
[Source] Dummy data.
[Notes] This model does not
represent any part from any
vendor.
[Disclaimer] Demo model.
[Copyright] LGreen
```

IBIS File Structure

Header

- Disclaimer
 - Similar to what appears on the datasheet
 - Model is not guaranteed for any specific use ...
 - Subject to change without notice ...
- File revision
 - 0.x: silicon and file in development
 - 1.x: pre-silicon file data from silicon model only
 - 2.x: file correlated to actual silicon measurements
 - 3.x: mature product, no more changes likely

IBIS File Structure Components

- [Component]
 - Can have more than one per file
- [Manufacturer]
- [Package]
- [Pin]
 - Every pin on physical package
 - Optional package parasitics
 - Case-sensitive
 - Match pin case to layout

```
[Component]      XYZ2
[Manufacturer]   Nobody
|
[Package]
| variable typ      min      max
R_pkg           100m    NA      NA
L_pkg           6n      NA      NA
C_pkg           1.5p    NA      NA
|
[Pin] signal_name  model_name
1      io1         demo1
2      io2         demo1
B2     Vcc         POWER
C3     Gnd         GND
A10    unused      NC
```

IBIS File Structure Components

- [Diff Pin] pairs

```
[Diff Pin]  inv_pin  vdiff  tdelay_typ  tdelay_min  tdelay_max
1            2      0.150V  -1ns        0ns         -2ns
```

- [Series Pin Mapping] pairs

```
[Series Pin Mapping]  pin_2  model_name  function_table_group
1                    2      CBTSeries  1
[Series Switch Groups]
| Function Group States
On 1 2 /
```

- [Model Selector]

```
[Model Selector]  Progbuffer1
OUT_2             2mA buffer without slew rate control
OUT_4S            4mA buffer with slew rate control
```


IBIS File Structure

I/O Buffer Models

- Model used by one or more pins
- May be used by more than one component
- Model name is unique within the IBIS file
- 17 pre-defined model types
 - Input, Output, I/O, 3-state, Open_sink, I/O_open_sink, Open_source, I/O_open_source, Input_ECL, Output_ECL, I/O_ECL, 3-state_ECL
 - Series, Series_switch, Terminator
- IBIS 4.1 allows external model files
 - SPICE 3f5, VHDL-AMS, Verilog-AMS

IBIS File Structure

I/O Buffer Models

• Model header	[Model]	demo1		
	Model_type	I/O		
	Polarity	Non-Inverting		
• Voltage keyword(s)	Enable	Active-High		
	Vinl =	0.80		
	Vinh =	1.75		
• Input values	Cref =	2p		
– Logic levels	Vmeas =	1.7		
	variable	typ	min	max
• Output values	C_comp	3p	NA	NA
	[Temperature Range]	50	125	0
– Polarity	[Voltage Range]	3.3	3.0	3.6
	[Pullup Reference]	3.3	3.0	3.6
– Enable	[Pulldown Reference]	0.0	0.0	0.0
	[POWER Clamp Reference]	3.3	3.0	3.6
– Standard load	[GND Clamp Reference]	0.0	0.0	0.0
	[Ramp]			
– [Ramp]	dV/dt_r	1.20/0.9n	0.96/1.5n	1.46/0.7n
	dV/dt_f	1.60/0.9n	1.38/1.4n	1.78/0.7n
	R_load =	50		

IBIS File Structure

I/O Buffer Models

	Power_ Clamp	GND_ Clamp	Pullup	Pulldown	Vinh, Vinl	Vmeas
Input	Opt*	Opt*	No	No	Req	No
Output	Opt	Opt	Req	Req	No	Opt!
I/O	Opt*	Opt*	Req	Req	Req	Opt!
3-State	Opt*	Opt*	Req	Req	Opt!	Opt!
Open_Sink	Opt*	Opt*	No	Req	No	Opt!
I/O_Open_Sink	Opt*	Opt*	No	Req	Req	Opt!
Open_Source	Opt*	Opt*	Req	No	No	Opt!
IO_Open_Source	Opt*	Opt*	Req	No	Req	Opt!

! Required for software timing checks.

* Should not be omitted unless the corresponding input clamping and leakage currents are 0 Amp.

IBIS File Structure

I/O Buffer Models

- I-V tables

	[Pullup]	[Pulldown]	[Power Clamp]	[GND Clamp]
– [Pullup]	– voltage	I (typ)	I (min)	I (max)
	–3.6	–1.0e-1	–8.1e-2	–1.2e-1
– [Pulldown]	–1.0	–7.3e-2	–5.1e-2	–7.7e-2
	–0.4	–3.5e-2	–2.8e-2	–4.1e-2
– [Power Clamp]	–0.1	–9.3e-3	–7.6e-3	–1.0e-2
	0.0	0.0	0.0	0.0
– [GND Clamp]	0.1	9.1e-3	7.5e-3	1.0e-2
	0.6	5.1e-2	4.0e-2	5.7e-2
	1.0	7.6e-2	6.1e-2	8.9e-2
• Column ordering	2.0	1.1e-1	8.4e-2	1.4e-1
	2.5	1.2e-1	8.5e-2	1.5e-1
• Monotonicity	3.0	1.2e-1	8.6e-2	1.6e-1
	3.3	1.2e-1	8.7e-2	1.7e-1
	3.6	1.2e-1	8.7e-2	1.8e-1
	6.6	1.2e-1	8.7e-2	1.8e-1

IBIS File Structure

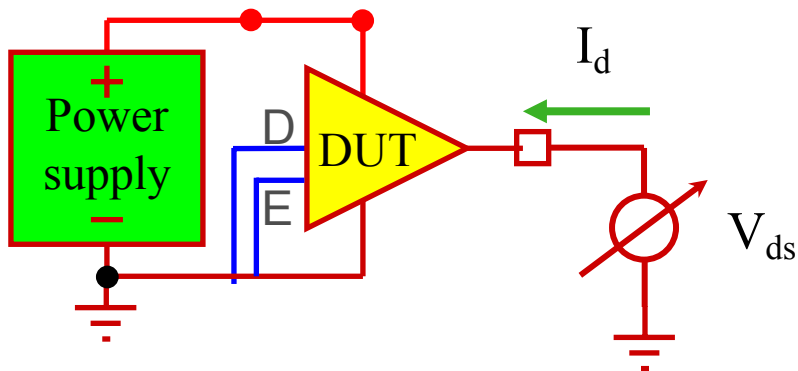
I/O Buffer Models

- Monotonicity
 - Total current
- Simulator convergence
 - 0A at reference voltage
 - Range of $-V_{cc}$ to $2V_{cc}$
 - Best points selection
- Physical operation
 - Feedback effects
 - Driver scheduling

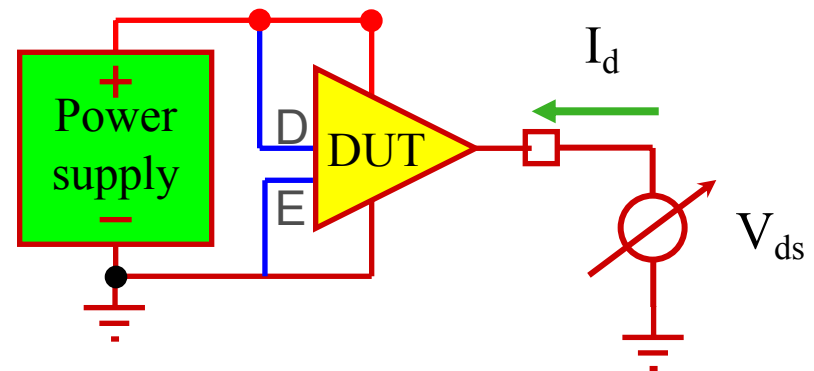
Generating I-V Table Data

- I-V tables
 - Range is $-V_{cc}$ to $+2V_{cc}$
 - Setting DC voltages (Typ/Min/Max)

Pulldown + GND clamp



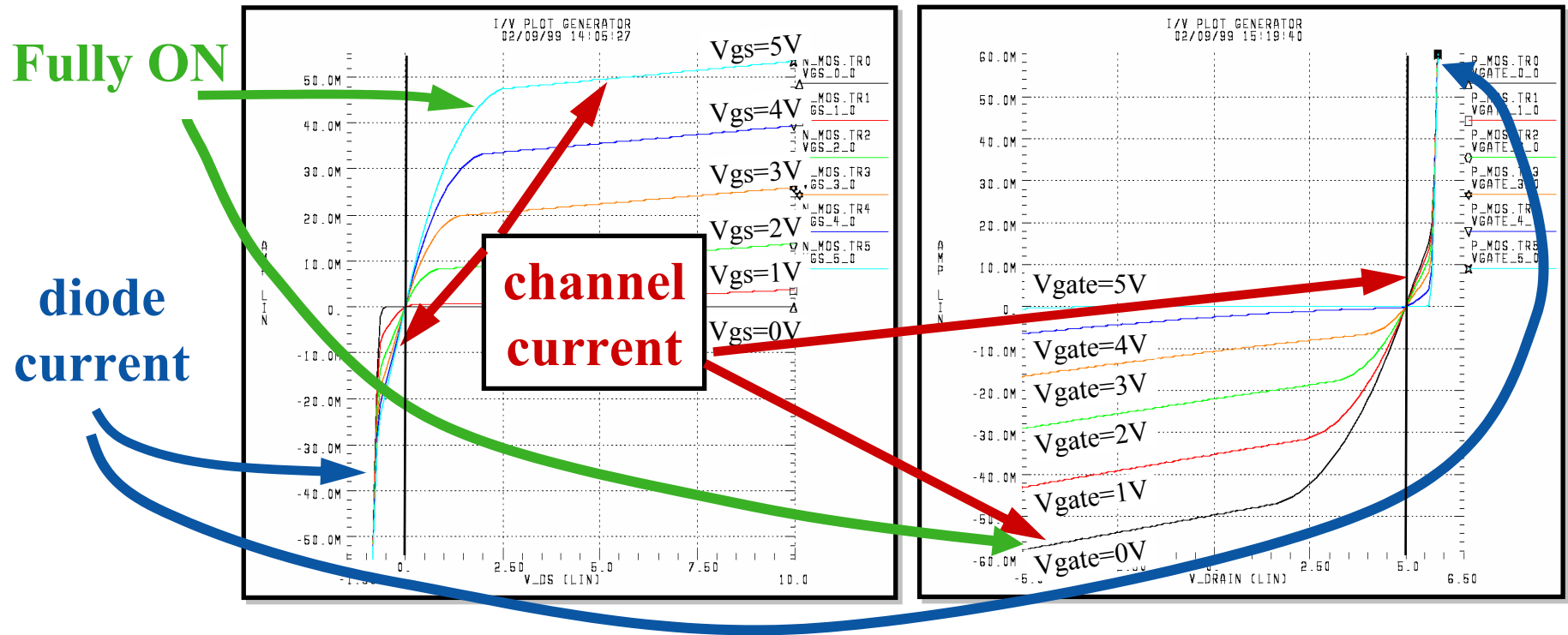
Pullup + Power clamp



Note: Currents are considered positive when their direction is into the component.

Generating I-V Table Data

- Separating the drive and clamp currents

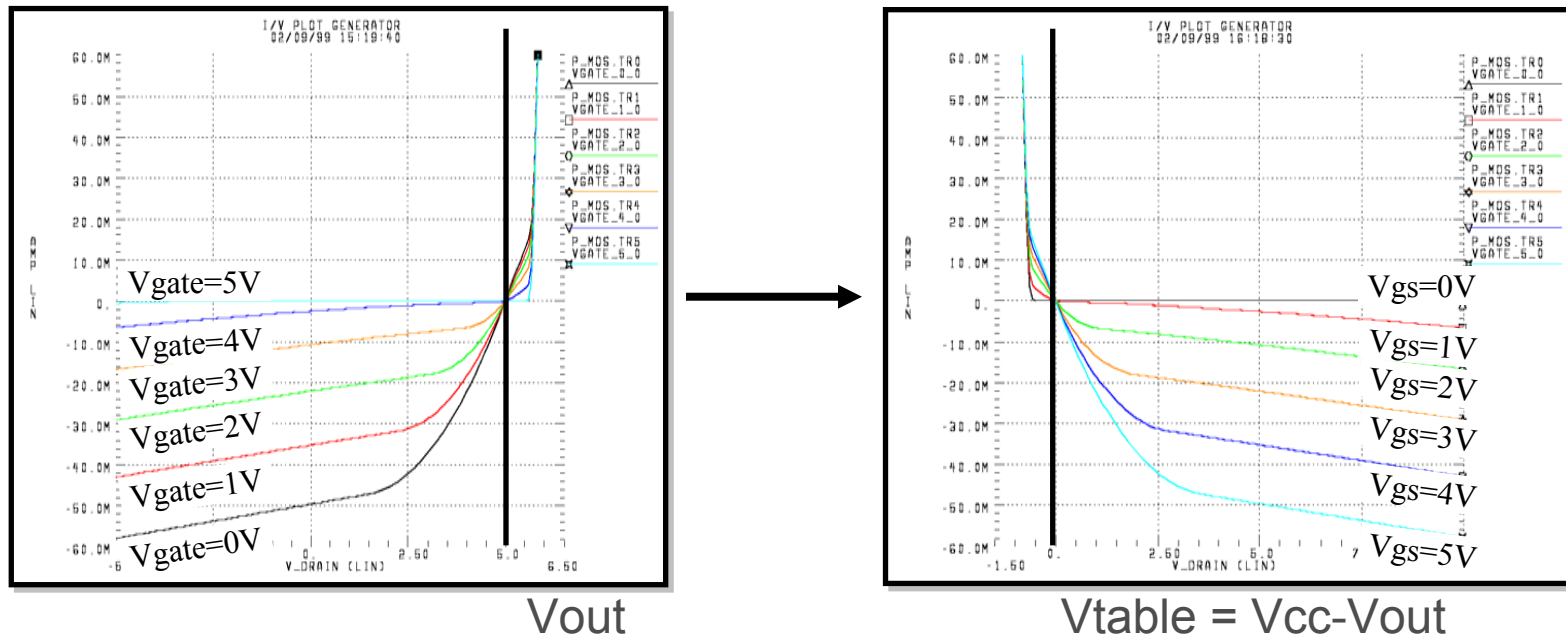


Courtesy of Arpad Muranyi, Intel.

Generating I-V Table Data

Extracting I-V tables

- I-V table
- Separating into Clamp and Drive tables
- Changing reference for Pullup and Power clamp tables



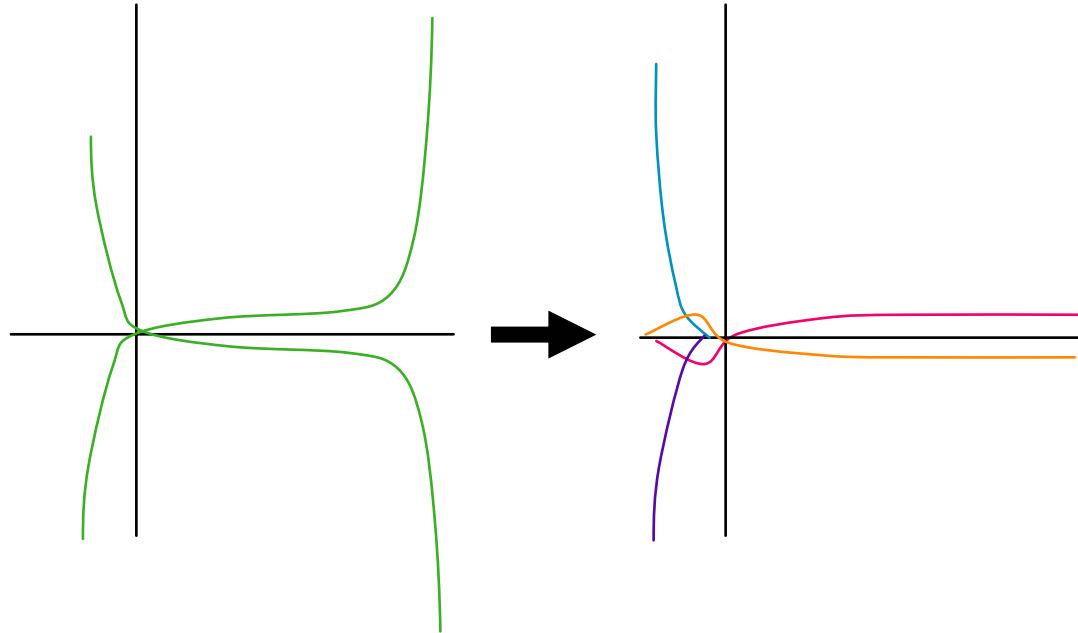
Courtesy of Arpad Muranyi, Intel.

Generating I-V Table Data

Separating I-V tables

- Separate **total** currents into:

- **Pulldown**
- **Pullup**
- **Power clamp**
- **GND clamp**



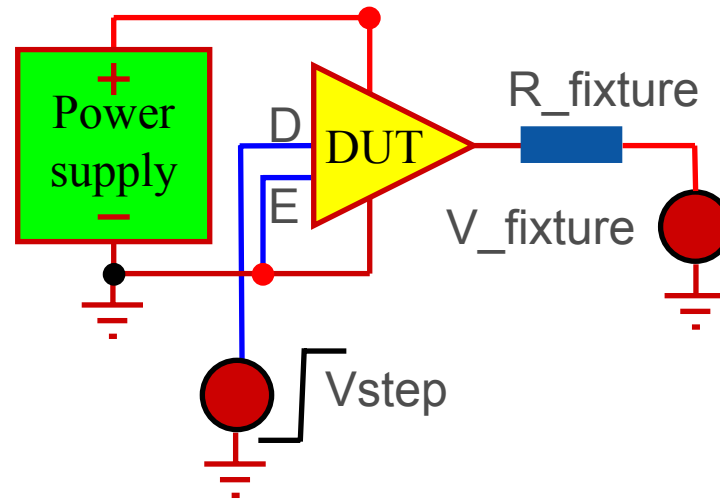
- Make Pullup and Power clamp Vcc-referenced

Generating V-t Table Data

- V-t tables
 - Stop time \Rightarrow steady-state voltage reached
 - Time step $< 0.10 * T_{\text{edge}}$
 - Data step (Core edge rate) (Typ/Min/Max)
 - Specified load (such as 50 Ohms)
 - Resistive load. No ringing!
- Start and end times
 - Start: signal from core into buffer toggles
 - End: Just enough for Min signal to settle

Generating V-t Table Data

- Set of four V-t tables
 - Data rising, $V_{\text{fixture}}=0$
 - Data rising, $V_{\text{fixture}}=V_{\text{cc}}$
 - Data falling, $V_{\text{fixture}}=0$
 - Data falling, $V_{\text{fixture}}=V_{\text{cc}}$
- Output crosses through V_{meas}



Courtesy of Arpad Muranyi, Intel.

Generating V-t Table Data

Leading time in tables

[Rising Waveform]

Time	V(typ)
------	--------

0.00s	25mV
-------	------

0.20ns	35mV
--------	------

[Falling Waveform]

Time	V(typ)
------	--------

0.00s	325mV
-------	-------

0.20ns	322mV
--------	-------

[Rising Waveform]

Time	V(typ)
------	--------

10.00s	25mV
--------	------

10.20ns	35mV
---------	------

[Falling Waveform]

Time	V(typ)
------	--------

10.00s	325mV
--------	-------

10.20ns	322mV
---------	-------

[Rising Waveform]

Time	V(typ)
------	--------

5.00s	25mV
-------	------

5.20ns	35mV
--------	------

[Falling Waveform]

Time	V(typ)
------	--------

55.00s	325mV
--------	-------

55.20ns	322mV
---------	-------

IBIS 3.2 – Tables could both start at 5ns, or start at different times; tool dependent.

IBIS 4.0 – Tables have the same start time.

Model Creation

Interpreting V-t tables

- What about those “flat” times
- V-t is not changing, but table contains data
- Actual internal buffer delays
- Different tools do different things!

```
[Rising Waveform]
| Time          V(typ)
0.0ns           25mV
2.0ns           25mV
2.20ns          2mV
...
5.00ns          34uV
20.00ns         34uV
```

Model Creation

Ways to Obtain C_{comp}

- Remember the goals
 - Signal integrity (reflections, crosstalk)
 - Timing (PCB delays)
- Things to include in C_{comp}
 - Metal capacitances
 - Silicon junction capacitances
- There is no one C_{comp} value!
 - Might want to manually adjust value

Model Creation

Ways to Obtain C_{comp}

- Time domain (large signal) effects
 - Edge rate
 - Voltage step values

Frequency (small signal) effects

- Frequency (small signal)
- DC bias voltage sensitivity
- As seen at die pad

Model Creation

Ways to Obtain C_comp

- Step response $I = C \, dV/dt$
 - Using PCB edge rates and voltages
- Tuned tank resonance $\omega = 1/\sqrt{C_{\text{comp}} * L_{\text{ext}}}$
 - Voltage sensitivity
 - Narrowband frequency sweep
- Zout pole frequency $\omega = 1/RC$
 - Single bias
 - Wideband frequency sweep

IBIS File Structure

Recent additions to IBIS

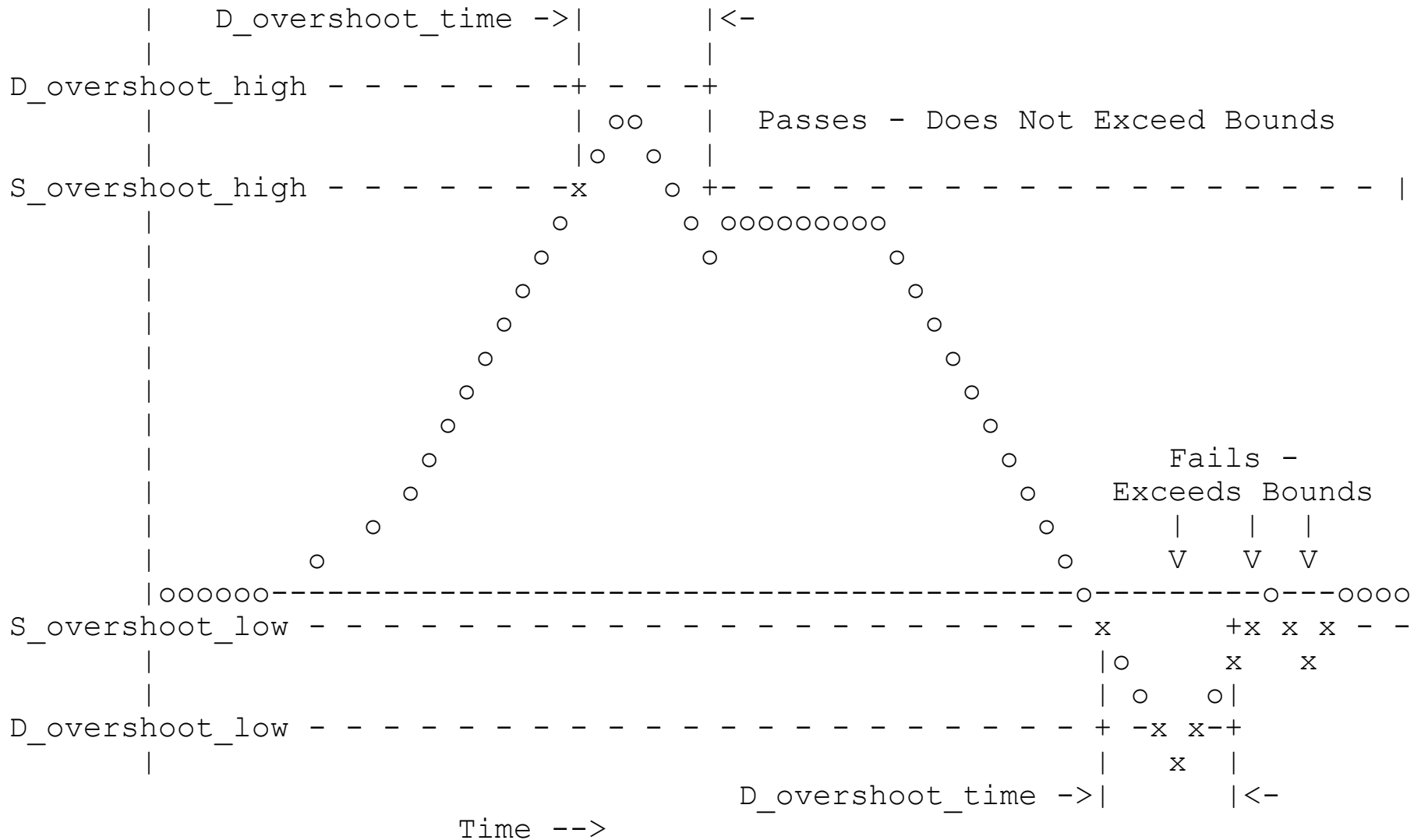
- IBIS 4.0 added
 - [Receiver Thresholds] : Input threshold parameters
 - [External Reference] : DC voltage (for pseudo-differential)
 - [Add Submodel] : Adds Fall_back
- IBIS 4.1 added
 - External models and external circuits
 - SPICE 3f5, VHDL-AMS, Verilog-AMS

IBIS File Structure

I/O Buffer Parameters

- [Model Spec] : Timing sub-params
 - Vinh, Vinl
 - Vinh+, Vinh-, Vinl+, Vinl-
 - S_overshoot_high, S_overshoot_low
D_overshoot_high, D_overshoot_low, D_overshoot_time
Pulse_high, Pulse_low, Pulse_time
 - Vmeas, Vref, Cref, Rref
 - Cref_rising, Cref_falling, Rref_rising, Rref_falling
Vref_rising, Vref_falling, Vmeas_rising, Vmeas_falling
- These override [Model] sub-params
 - Vinh, Vinl, Vmeas, Vref, Cref, Rref

Static and Dynamic Checks



IBIS File Structure

Series and Series Switch Models

- Calling a Series model for the component
- Pins must be in [Pin] list

```
[Series Pin Mapping]  pin_2  model_name  function_table_group
|
2      3      CBTSeries      1  | Four independent groups
5      6      CBTSeries      2
9      8      CBTSeries      3
12     11     CBTSeries      4
|
32     33     Fixed_series      | No group needed
|
[Series Switch Groups]      | Function Group States
On 1 2 3 /      | Default = ON
Off 4 /      | Default = OFF
```

IBIS File Structure

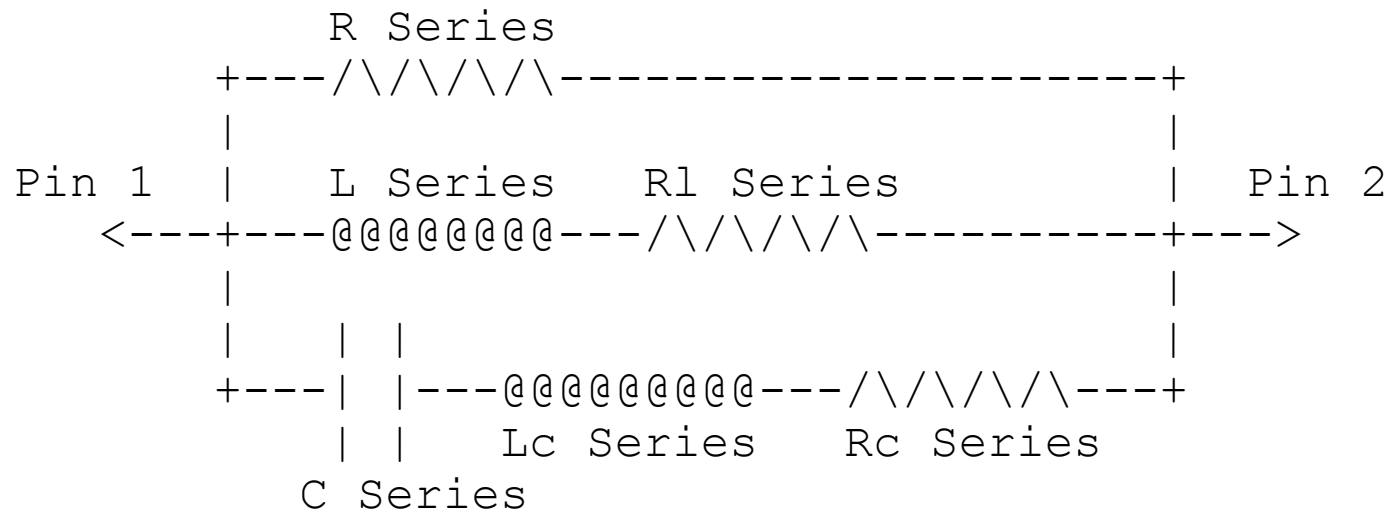
Series model types

- Series and Series_switch models
- Series model always On
- Series-switch state set at simulation time
 - [On], [Off]
 - Used with each Series_switch model type
- Connect two pads on same component (by pin name)

IBIS File Structure

Series model types

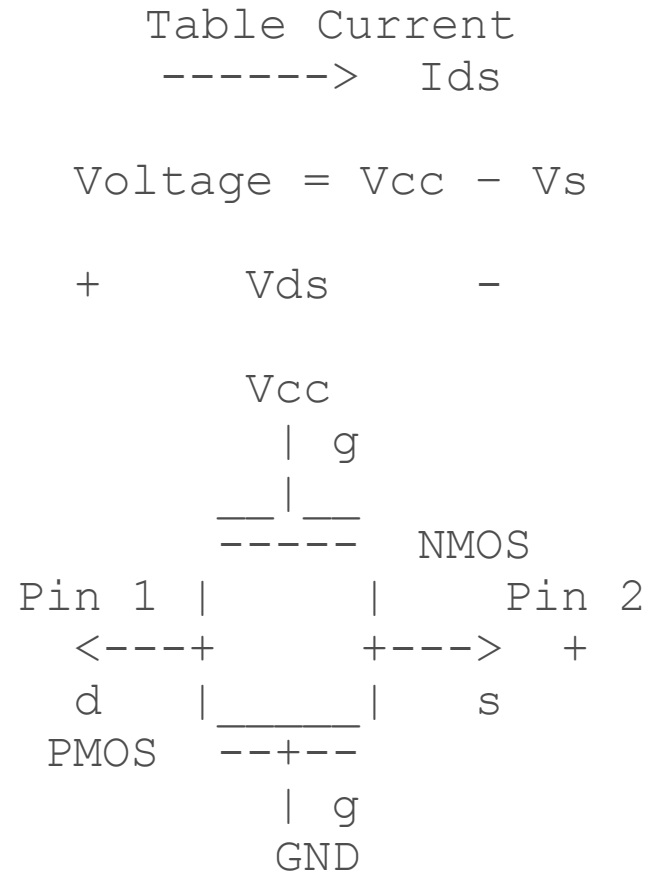
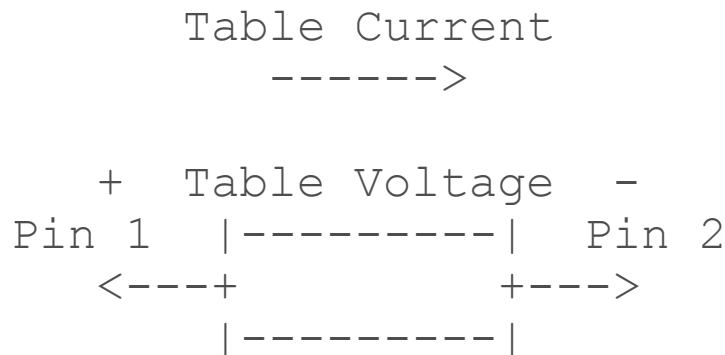
- Fixed value series components
 - [R Series]
 - [L Series], [Rl Series]
 - [C Series], [Lc Series], [Rc Series]



IBIS File Structure

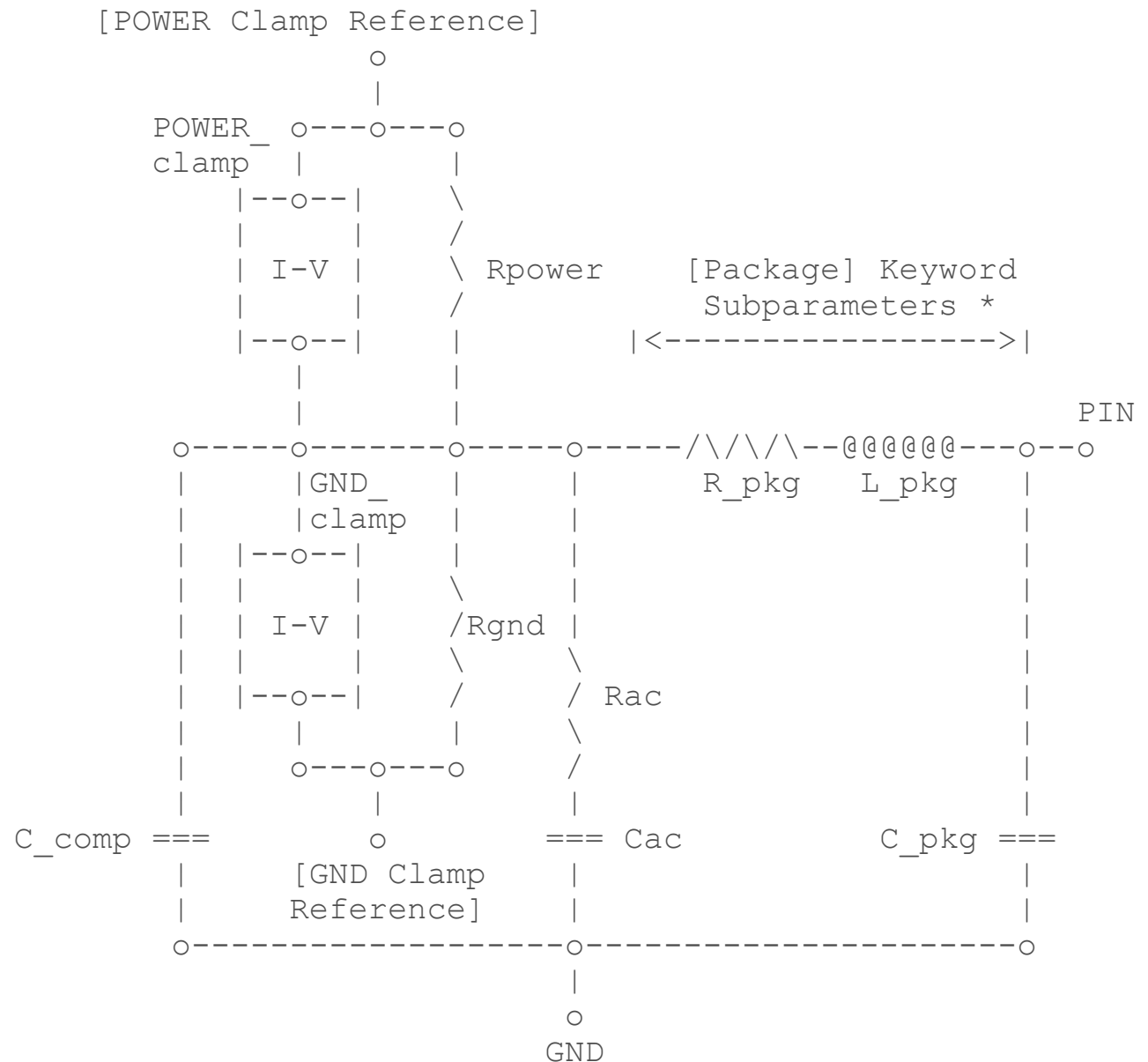
Series model types

- [Series Current]
 - One I-V table
- [Series MOSFET]
 - Multiple I-V tables
 - One table for each V_{gs}



Terminators

- [Rgnd]
- [Rpower]
- [Rac]
- [Cac]



Advanced Buffer Models

- SPICE 3f5
- VHDL-AMS
- Verilog-AMS
- External Circuit

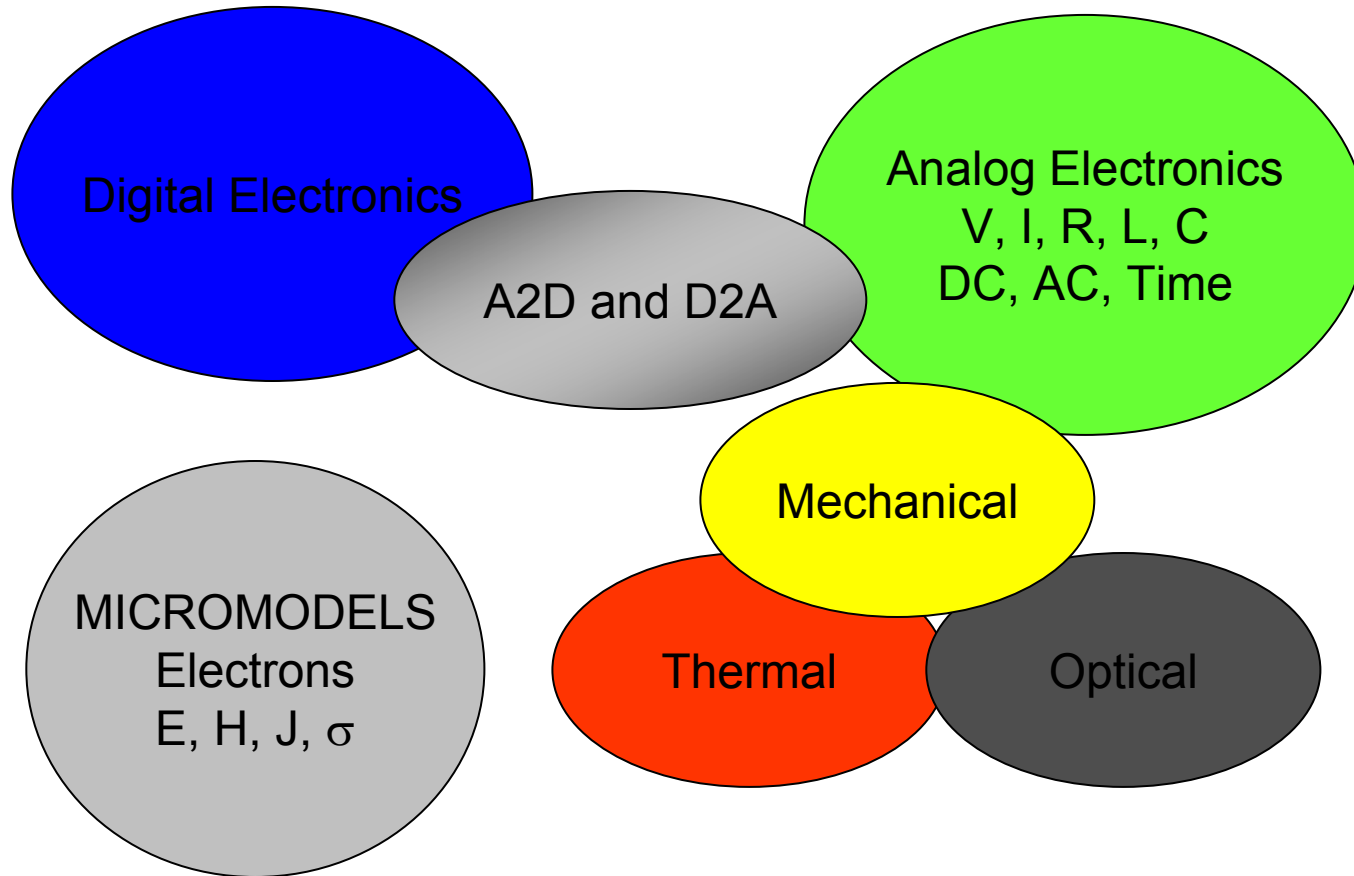
```
[Model] ExBufferSPICE
Model_type I/O
Vinh = 2.0
Vinl = 0.8
[Voltage Range] 3.3 3.0 3.6
[Ramp]
dV/dt_r      1.57/0.36n 1.44/0.57n 1.73/0.28n
dV/dt_f      1.57/0.35n 1.46/0.44n 1.68/0.28n
|
[External Model]
Language SPICE
| Corner corner_name file_name circuit_name (.subckt name)
Corner Typ      buffer_typ.spi buffer_io_typ
Corner Min      buffer_min.spi buffer_io_min
Corner Max      buffer_max.spi buffer_io_max
|
| Ports List of port names (in same order as in SPICE)
Ports A_signal my_drive my_enable my_receive my_ref
Ports A_puref A_pdref A_pcref A_gcref A_extref
|
| D_to_A d_port port1 port2 vlow vhigh trise tfall corner_name
D_to_A D_drive my_drive my_ref 0.0 3.3 0.5n 0.3n Typ
D_to_A D_enable my_enable A_gcref 0.0 3.3 0.5n 0.3n Typ
|
| A_to_D d_port port1 port2 vlow vhigh corner_name
A_to_D D_receive my_receive my_ref 0.8 2.0 Typ
|
[End External Model]
```

Advanced Buffer Models

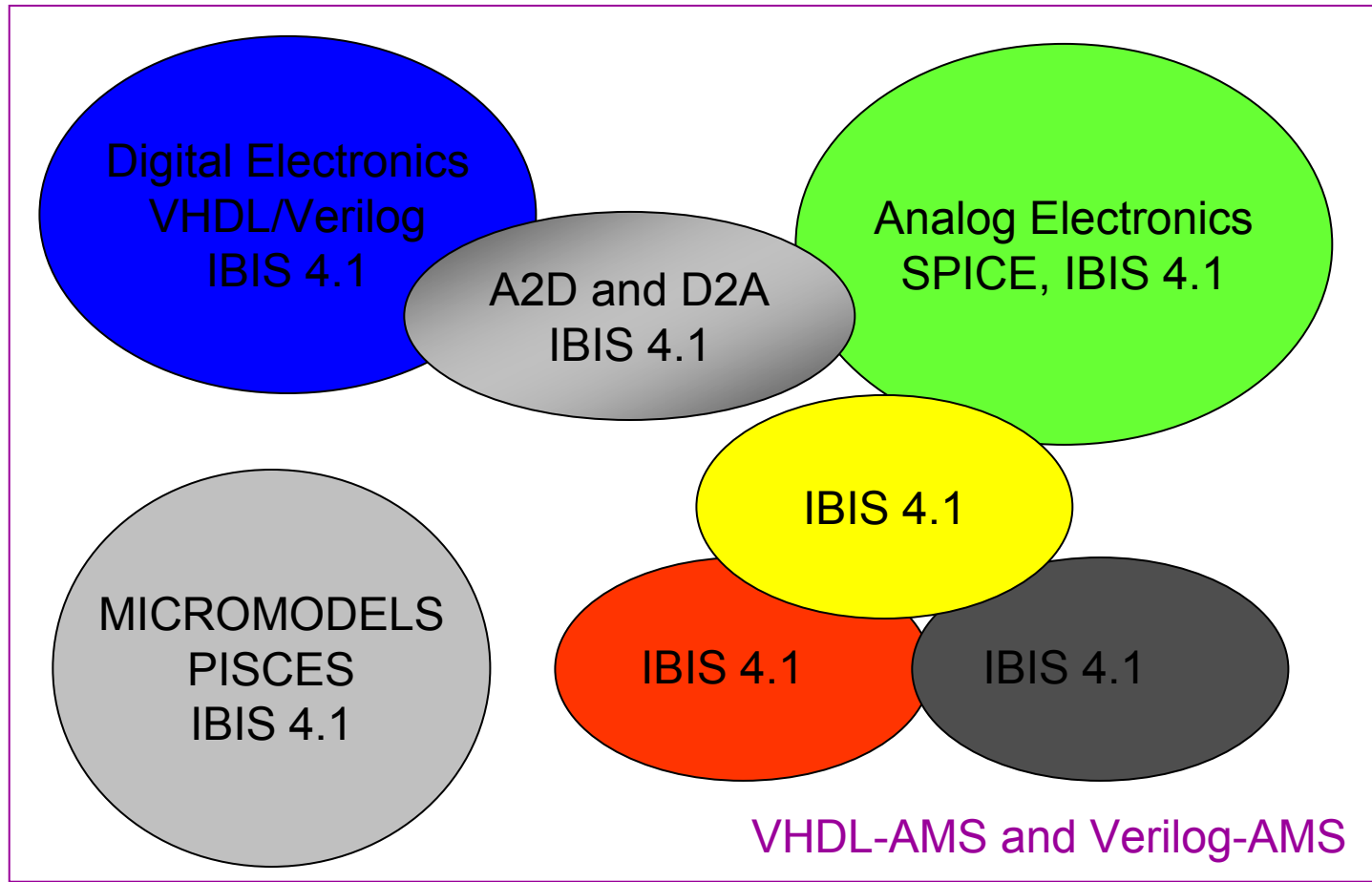
- SPICE 3f5
- VHDL-AMS
- Verilog-AMS
- External Circuit

```
[Model] Ext_VHDL_Diff_Buff
Model_type I/O_diff
Rref_diff = 100
|
|          typ    min    max
[Voltage Range] 3.3    3.0    3.6
|
[Ramp]
dV/dt_r      1.57/0.36n  1.44/0.57n  1.73/0.28n
dV/dt_f      1.57/0.35n  1.46/0.44n  1.68/0.28n
|
[External Model]
Language VHDL-AMS
| Corner corner_name file_name      circuit_name entity(architecture)
Corner  Typ          diffio_typ.vhd  buffer(diff_io_typ)
Corner  Min          diffio_min.vhd  buffer(diff_io_min)
Corner  Max          diffio_max.vhd  buffer(diff_io_max)
|
| Parameters List of parameters
Parameters delay rate pre-emphasis
|
| Ports List of port names (in same order as in VHDL-AMS)
Ports A_signal_pos A_signal_neg D_receive D_drive D_enable
Ports A_puref A_pdref A_pceref A_gceref
|
[End External Model]
```

The System Design Perspective



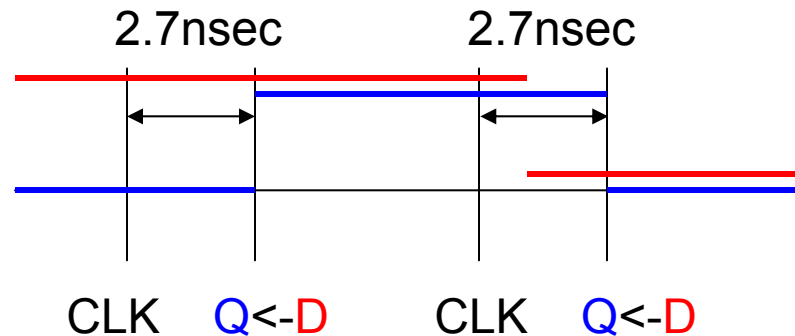
The System Design Perspective



Event-Driven Domain

- All changes occur instantaneously
- Time of event is pre-determined
- Nothing happens between events

When CLK toggles
Wait 2.7nsec
Copy D into Q
Loop;



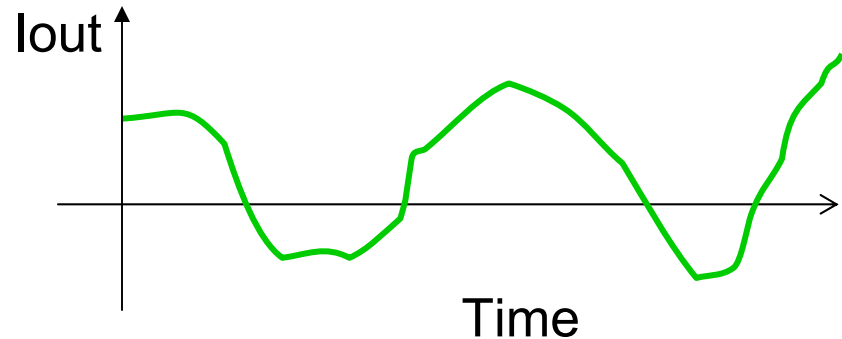
Analog Domain

- Changes occur continuously
- KVL and KCL solved at each time step
- Derivatives and integrals used

$$I_{dc} = I_S * (\exp(V_j / V_t) - 1)$$

$$I_{ac} = C_j * \frac{dV_j}{dt}$$

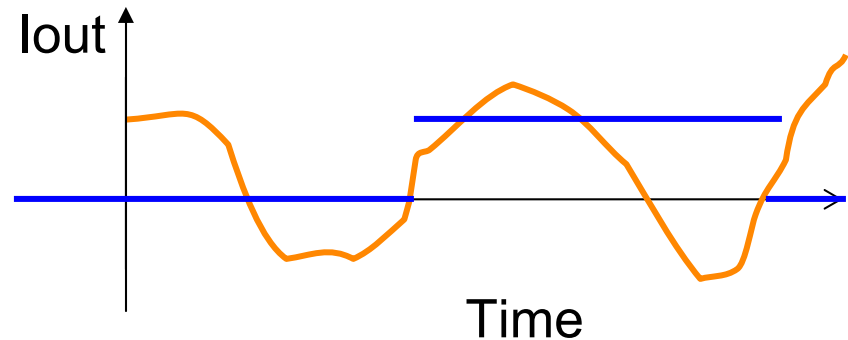
$$I_{out} = I_{dc} + I_{ac}$$



Interactions Between Domains

- Changes in one affect the other
- Analog section must converge
 - Before and after interaction

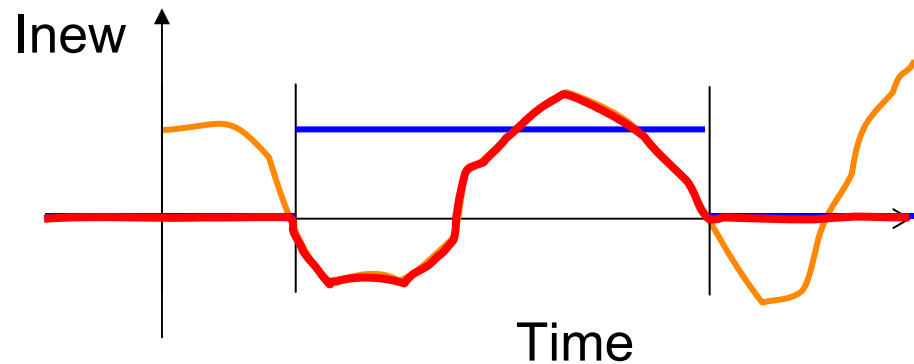
When **lout** crosses 0 rising
Toggle **Q**
Loop;



Interactions Between Domains

- Changes in one affect the other
- Analog section must converge
 - Before and after interaction

$$I_{\text{new}} = d2a(Q) * I_{\text{out}}$$



Example: Ideal LED Model

Such as an indicator light

```
module led(p, n, ps);  
    inout p, n;    output ps;    electrical p, n, ps;    real I_in;  
    // ideal led current and linear optical power  
    analog begin  
        // diode current  
        I_in = exp(V(p,n)/$vt) - 1.0;  
        I(p,n) <+ I_in;  
        // optical output power at 0.5mW/mA  
        V(ps) <+ 0.5*I_in;  
    end  
endmodule
```

Example: Optical Fiber Model

- Analog section of fiber model

```
analog begin
```

```
    // power transfer function
```

```
    // units conversion, since alpha is power loss in dB/km
```

```
    loss = NA*NA * exp(-ln(10)*alpha*length*1e-3/10);
```

```
    p_out <+ V(p_in) * loss;
```

```
    // pulse spreading
```

```
    V(pout) <+ laplace_zp(p_out, {1}, {-D_sigmaLambda*length,0});
```

```
    // Input current does not matter, so use 0.
```

```
    I(p_in) <+ 0;
```

```
end
```

Addressing Common Problems

A Validation Methodology

- Parse to check syntax (ibischk3, ibischk4)
- Examine parameters
- View tables graphically
- Other data checks
- Simulate
- Release for design use
- Close the loop

AMS Validation Methodology

- Verify pin order and parameters
- Otherwise the same as before:
 - Visually check parameters
 - Other data checks
 - Simulate
 - Release for design use
 - Close the loop

Validation Methodology in Design Flow

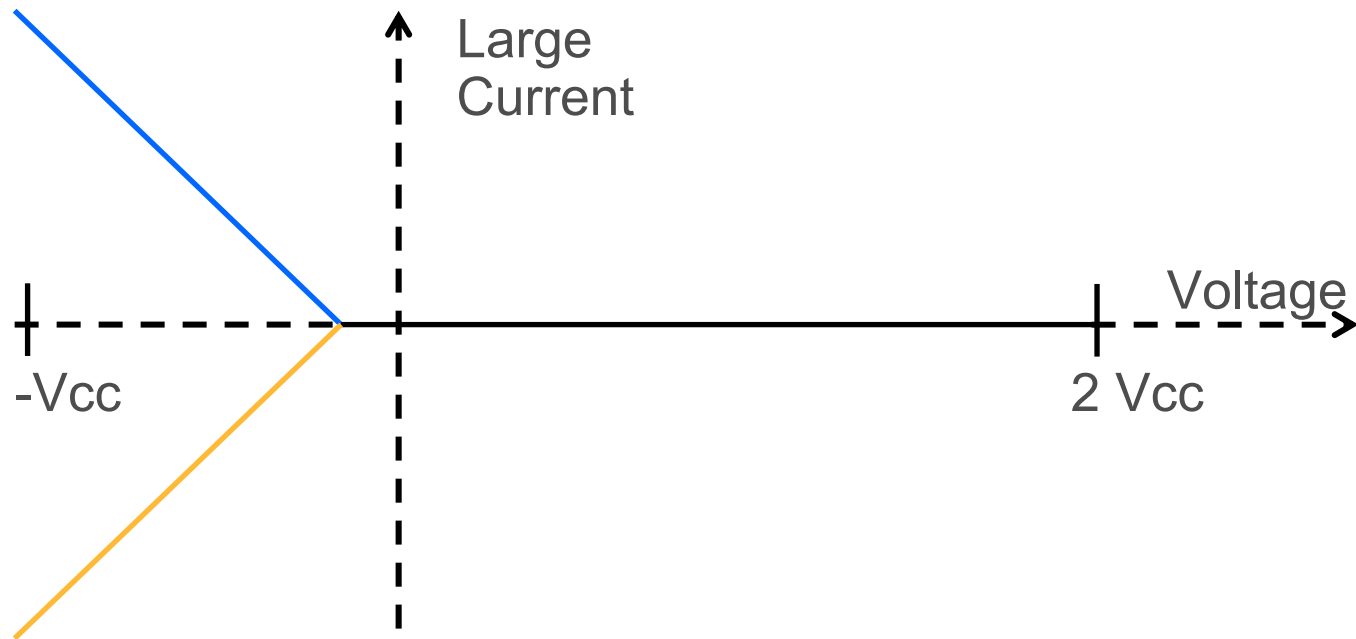
- During buffer design
 - I/O operation compared to I/O interface requirements
 - First IBIS model for test board designers
- Final buffer design
 - IBIS models generated and validated
 - IBIS and package models released
- Silicon back from fab
 - IBIS models validated against hardware
 - Buffer and package models updated

Validation Methodology in Design Flow

- Understanding what is expected
 - Simple CMOS buffers
 - More complex buffers
- Viewing data
 - Visual checks
 - Graphical checks
- Reporting problems
 - And fixing them whenever possible

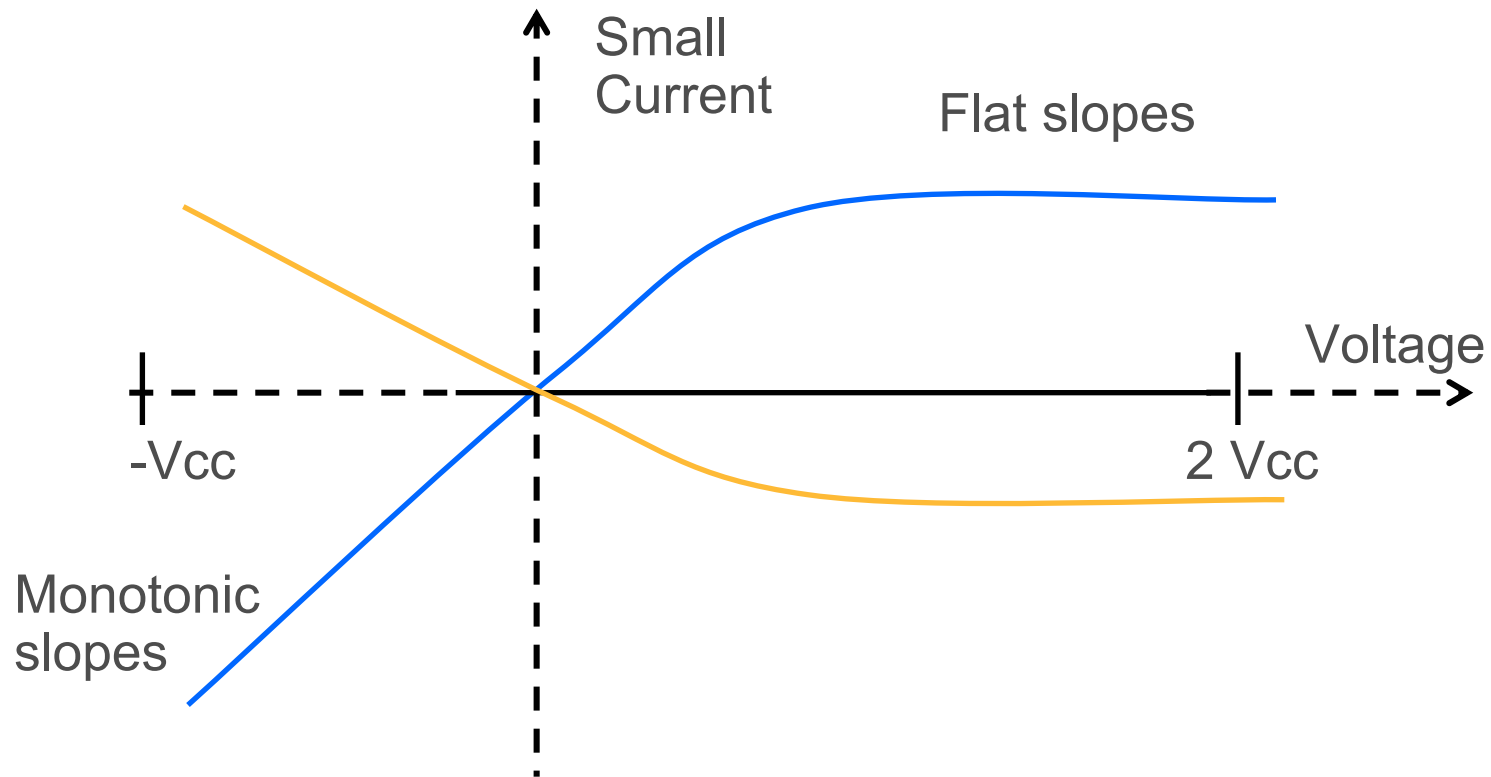
IBIS Tables: Graphical View

[Power Clamp] and [GND Clamp]



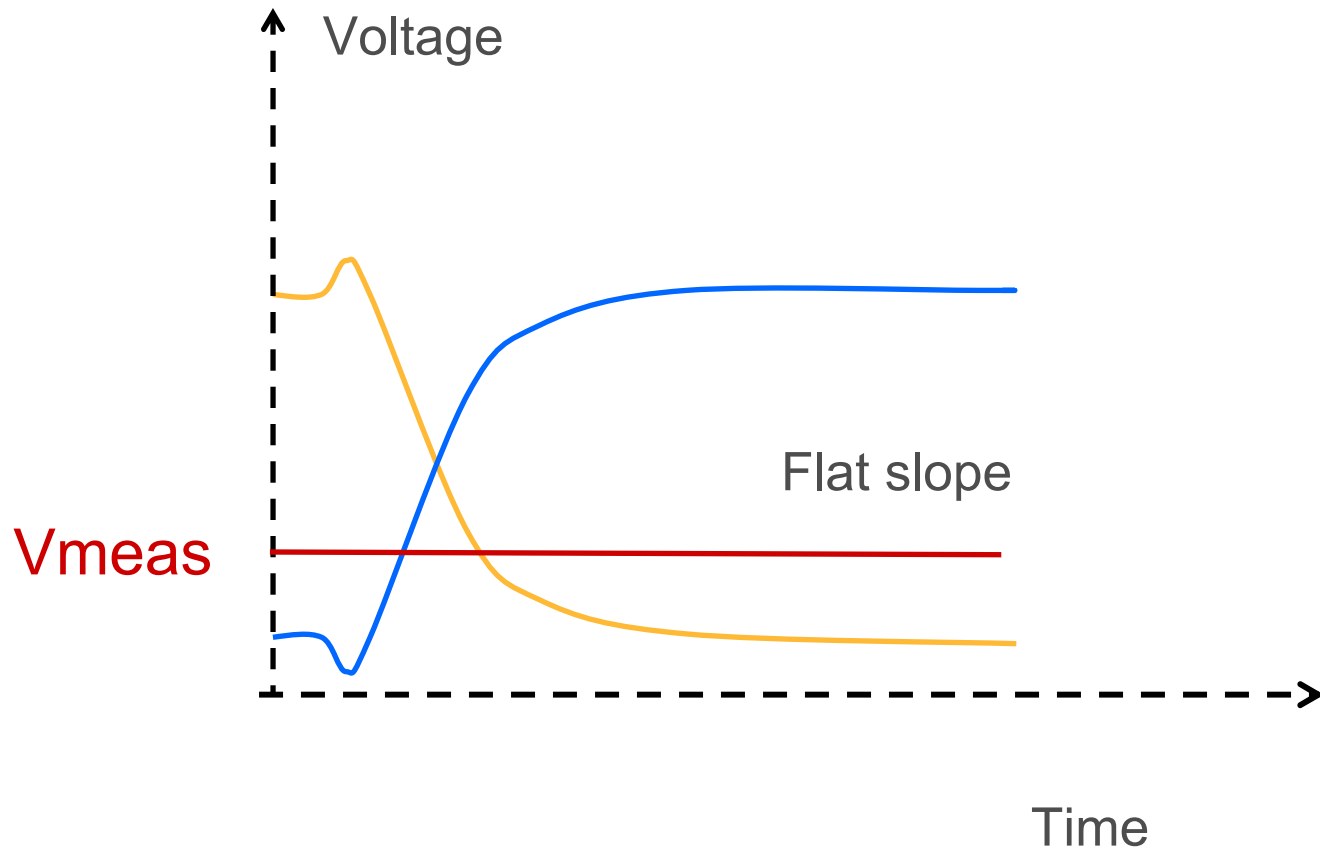
IBIS Tables: Graphical View

[Pullup] and [Pulldown]



IBIS Tables: Graphical View

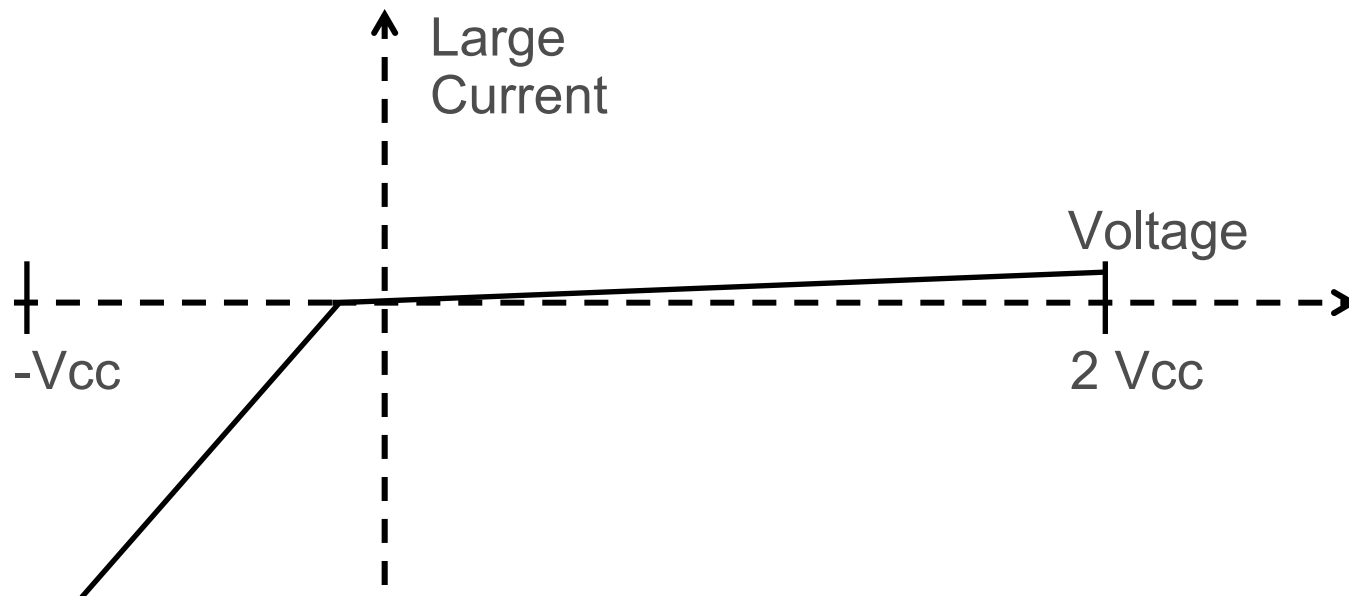
[Rising Waveform] and [Falling Waveform]



Common Problems

Power and GND clamp tables

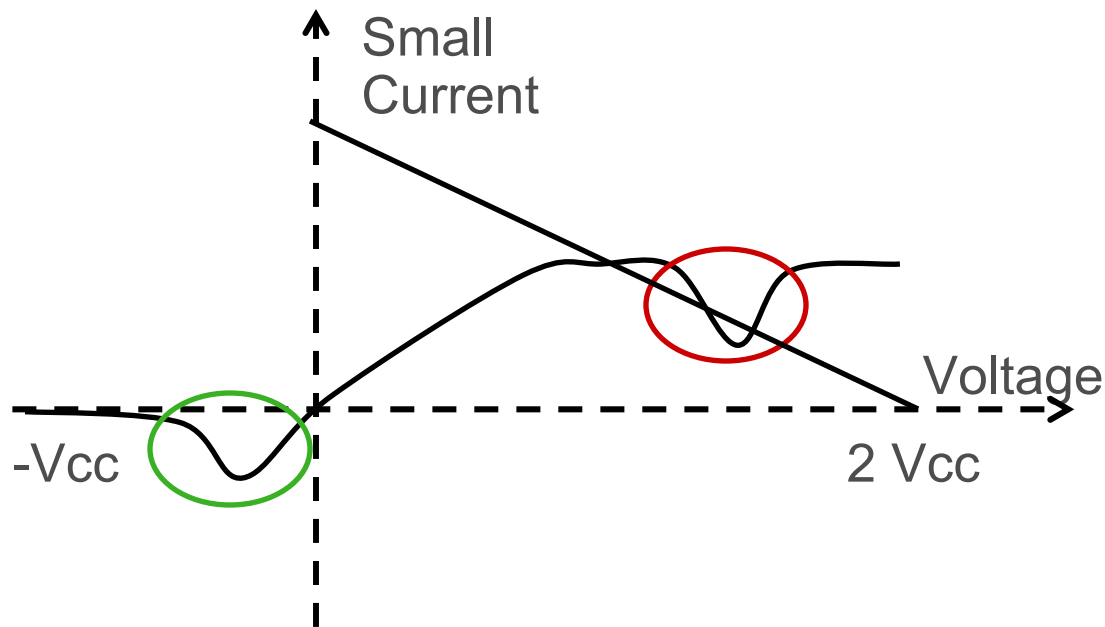
- Little or no power clamp current
- Current $> 100\text{A}$ (even $1\text{e}18$ Amps!)
- Table does not cover $-V_{cc}$ to $+2V_{cc}$



Common Problems

Pullup and Pulldown tables

- Non-monotonic tables can lead to DC convergence problems
- Tables do not cover $-V_{cc}$ to $+2V_{cc}$



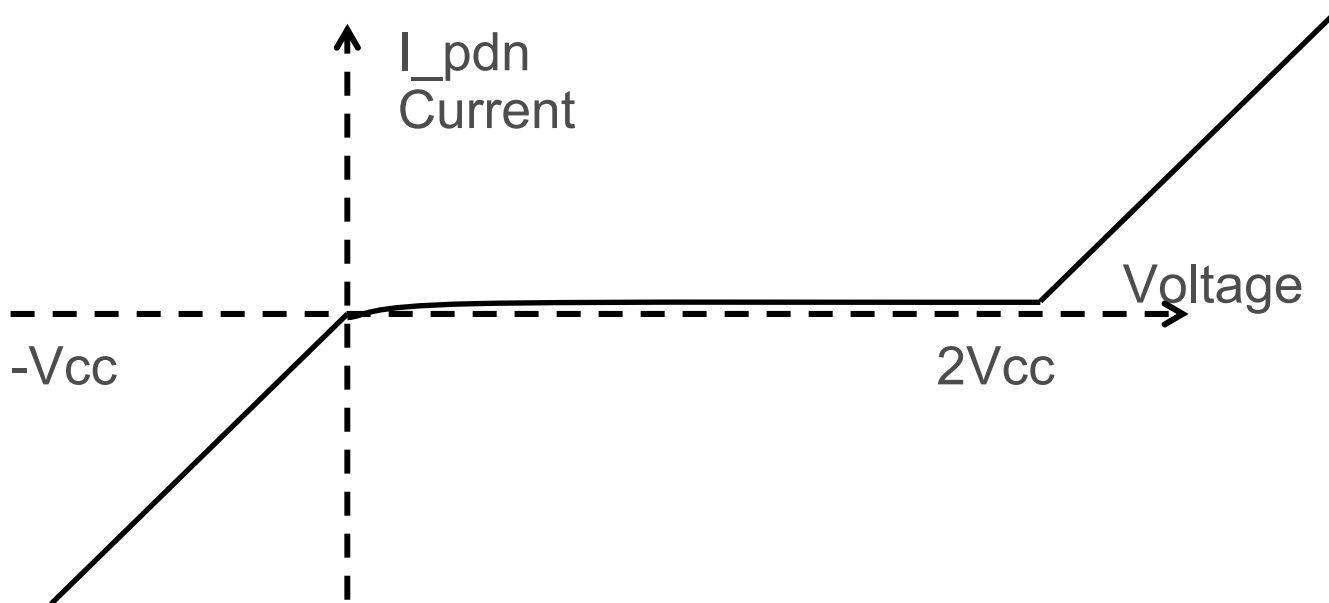
Common Problems

Pullup and Pulldown tables

- Double counting of clamp currents
- Incorrect subtraction of clamp currents

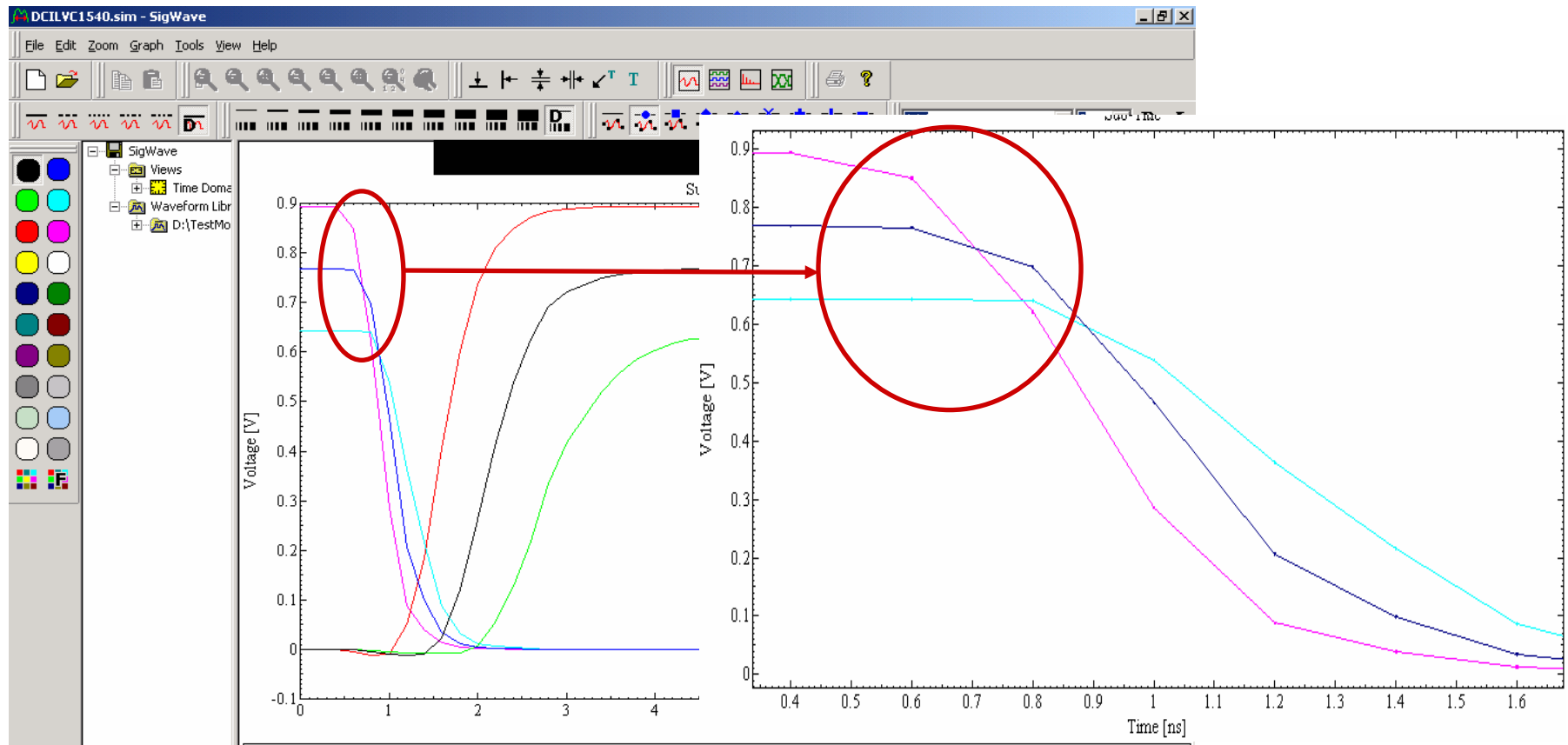
$$I_{\text{pdn}} = [\text{GND clamp}] + [\text{Power clamp}] + [\text{Pulldown}]$$

$$I_{\text{pdu}} = [\text{GND clamp}] + [\text{Power clamp}] + [\text{Pulldown}]$$



Common Problems

Not enough points in transition region

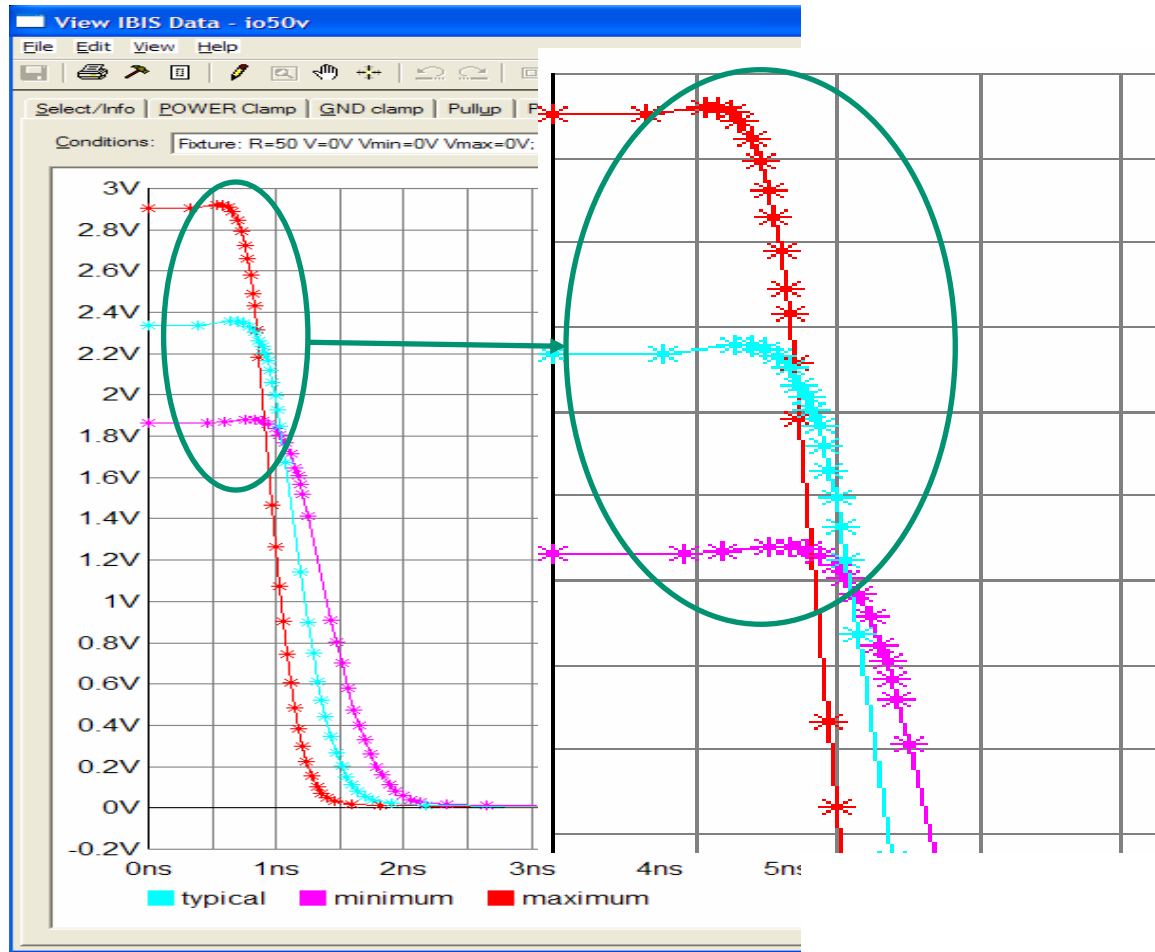


Courtesy of Cadence Design Systems

JEDEX 2004 IBIS Workshop

Common Problems

Use a “Best points” algorithm



Courtesy of Mentor Graphics/HyperLynx

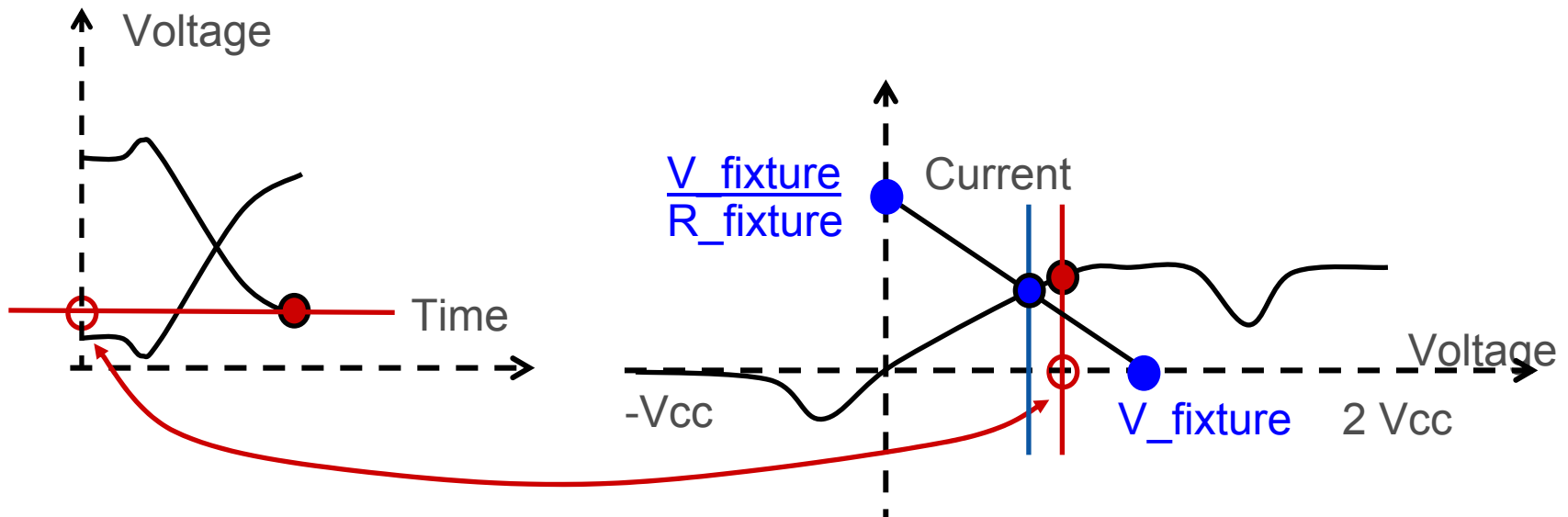
Common Problems

Rising and Falling V-t tables

- AC endpoint (DC point does not match I-V load line)
- One loadline point for $V_{\text{fixture}}=V_{\text{cc}}$, one for $V_{\text{fixture}}=\text{GND}$.

$$I_{\text{fixture}} = \frac{V - V_{\text{fixture}}}{R_{\text{fixture}}}$$

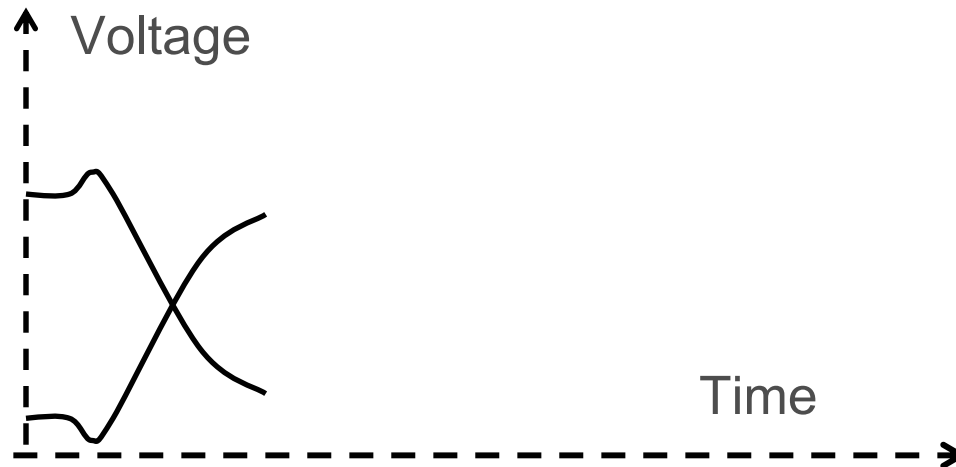
V at loadline with R_{fixture}
 $V @ I_{\text{fixture}}$



Common Problems

Rising and Falling V-t tables

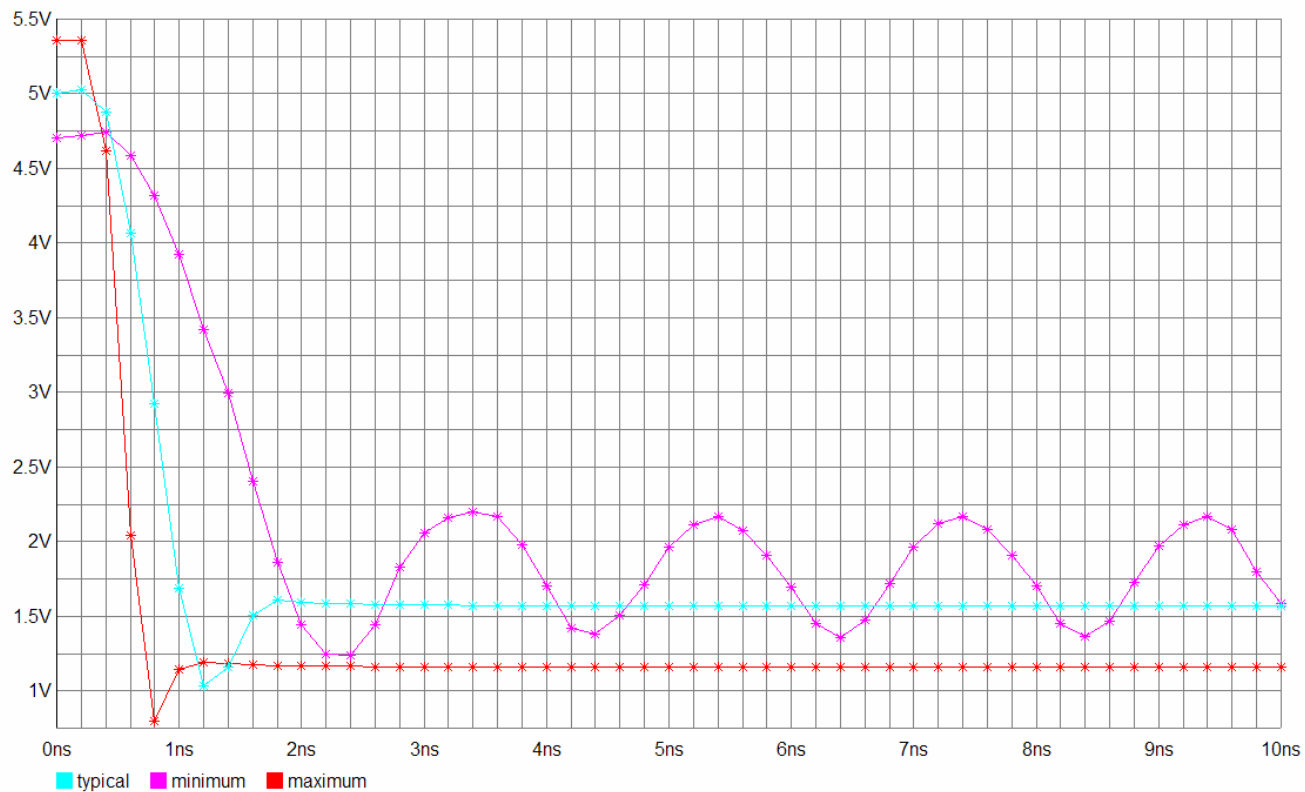
- End slope not flat
- Tables not starting at same time
- Excessively long V-t tables
- Not enough points in the transition region
- Output does not cross V_{meas}



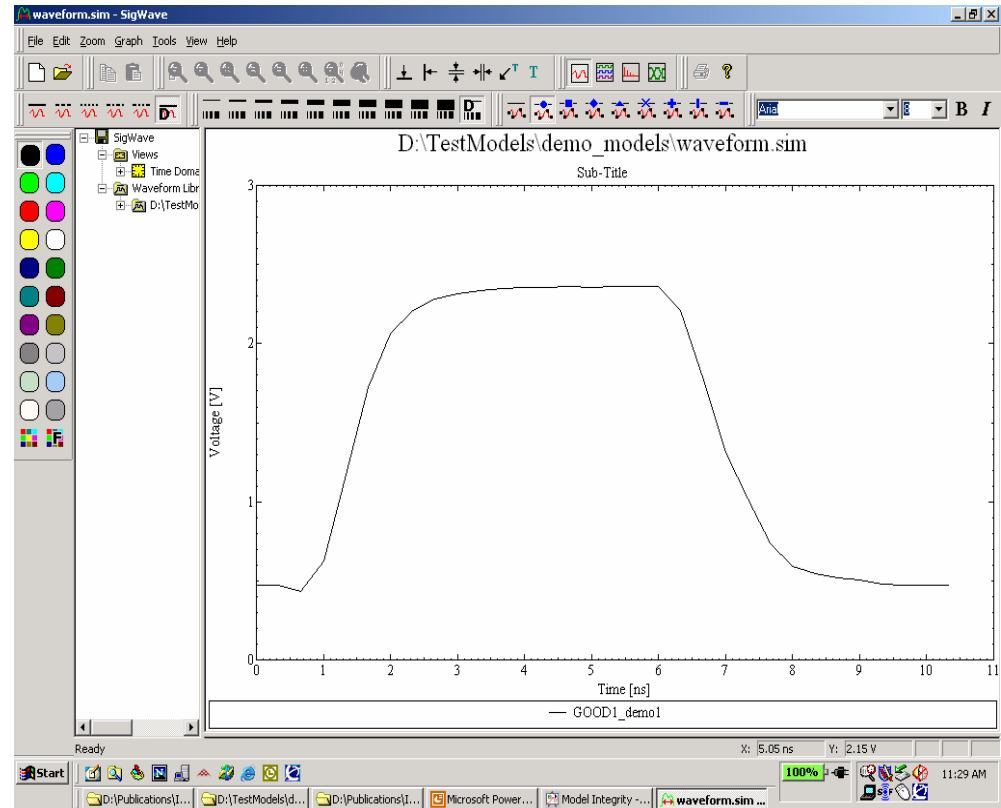
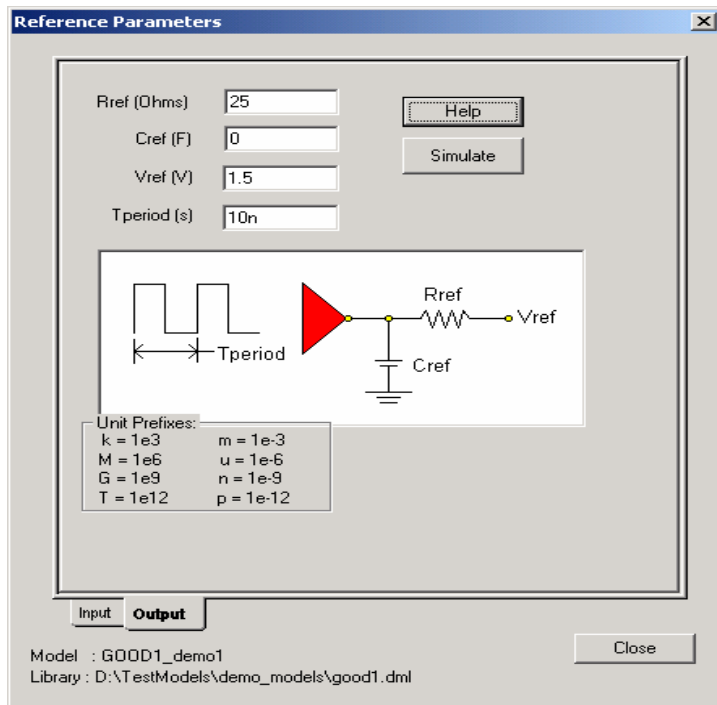
Common Problems

Rising and Falling V-t tables

- Signal fails to settle (not simple Rload)



Simulation Validation with your EDA tool!



Addressing Common Problems

A Validation Methodology

- Identifying problems quickly takes experience
- Identifying problems easier with good tools
- Fixing problems requires judgment calls
 - When in doubt, talk to the model maker!

Addressing Common Problems

A Validation Methodology

- Parse to check syntax (ibischk3, ibischk4)
- Examine parameters
- View tables graphically
- Other data checks
- Simulate
- Release for design use
- Close the loop

Addressing Common Problems

A Validation Methodology

- Cadence IBIS model validation webinars

<http://www.cadence.com/webinars/webinars.aspx?xml=Modeling>

<http://www.cadence.com/webinars/webinars.aspx?xml=Modeling2>

- Other links with IBIS papers

<http://www.teraspeed.com>

<http://www.specctraquest.com>, <http://www.allegrosi.com>

<http://www.eigroup.org/ibis/articles.htm>

PM Overview

- High-speed modeling techniques
- Differential buffer models
- On-die terminations
- Interconnect Models

High-speed Modeling Techniques

- Datasheet information
 - Pin and model assignment
 - Operating parameters: V_{ih} , V_{il} , V_{meas} , etc.
 - Single-ended and differential pins
- Buffer characteristics
 - Transistor-level simulations (HSPICE, Spectre, Eldo)
 - Test bench measurements
 - Programmable buffer options

High-speed Modeling Techniques

Pin/Model Assignment

[Pin]

Pin name	Model Name
D1	IO_1 Single-ended I/O
DD1	IO_1 Differential I/O, non-inverting
DD2	IO_1 Differential I/O, inverting
4	In1 Diff input, non-inverting
5	In1 Diff input, inverting
6	In1 Single-ended input
9	GND Ground pin #1
10	GND Ground pin #2
11	POWER Power Pin #1
12	POWER Power Pin #2

High-speed Modeling Techniques

Pin Relationships

[Diff Pin]	inv_pin	vdiff	tdelay_typ	tdelay_min	tdelay_max
DD1	DD2	NA	-1.0ns	0ns	0.1ns
4	5	150mV	NA	NA	NA

[Series Pin Mapping]	pin_2	model_name	function_table_group
4	5	Rser1	1 Series Resistor, always ON
DD1	DD2	MOS1	1 Series Resistor, two values

[Series Switch Groups]	Function Group States
On 1	

High-speed Modeling Techniques

Pin Association for SSN

| For SSN analysis

[Pin Mapping]	pdn_ref	pup_ref	gnd_cl_ref	power_cl_ref
DD1	9	11	9	11
DD2	9	11	9	11
D1	9	11	9	11
4	10	12	10	12
5	10	12	10	12
6	10	12	10	12

High-speed Modeling Techniques

Dual-voltage buffers

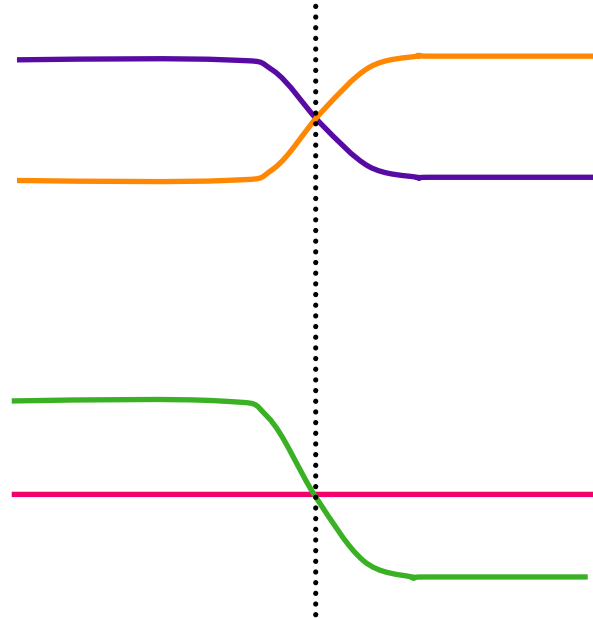
- Dual-voltage I/O operation
 - Do V_{cc_core} and V_{cc} track?
 - Where is level shifting done?
- SPICE: 8-corner vs. 16-corner vs. 32-corner
 - Factor of 2 for each independent variable
 - V_{cc} , Temp, $N_process$, $P_process$, V_{cc_core} , ...
 - Which combinations are “min” and “max”?
- Use those “min” and “max” in Spice-to-IBIS

Overview

- High-speed modeling techniques
- Differential buffer models
- On-die terminations
- Interconnect Models

Differential Signals

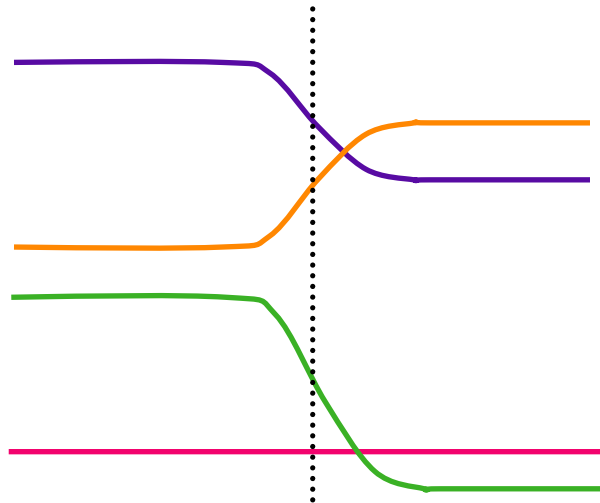
- Non-inverting pin signal
- Inverting pin signal
- Differential signal
- Common mode signal



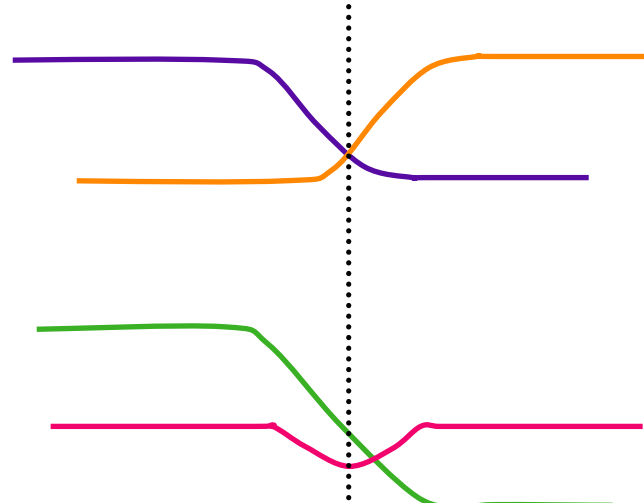
Differential Signals

Timing Relationships

- It does not matter what causes the shift
 - Anything that shifts signal in time or voltage
 - Driver skew, routing skew, crosstalk, etc.



Voltage shift

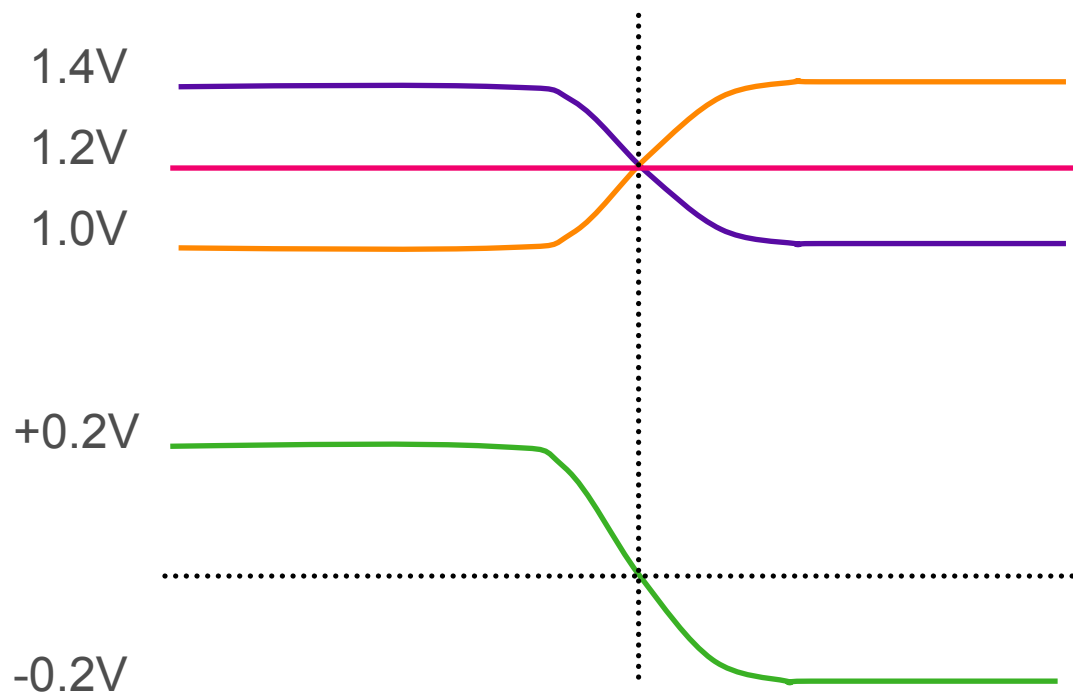


Time shift

Differential Signals

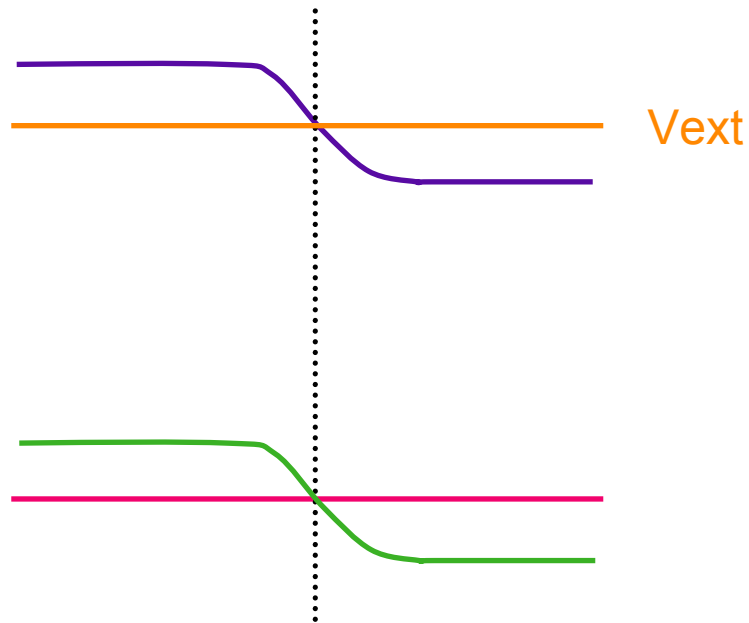
LVDS Example

- Ideal LVDS switching



Pseudo-Differential Signals

- Single Buffer
- Single trace to route
- Receiver referenced to Vext
- More sensitive to crosstalk
- Sensitive to bounce in Vext



Paired-differential Signals

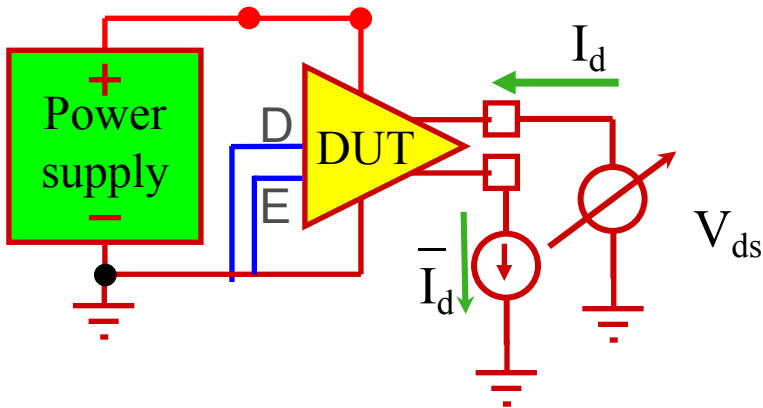
- Inverting and non-inverting buffer pair
- Better than single-ended differential
 - Reduces crosstalk sensitivity
- Data is inverted
- IBIS assumes independent buffers
 - Independent voltages and currents
 - Slew rise/fall not required to match
 - Put driver slew in V-t tables or in tdelay, but not both

Model Creation

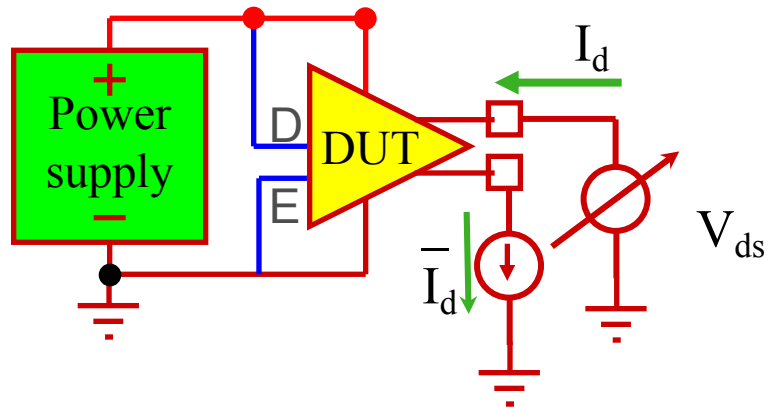
Differential I-V tables

- No package when generating buffer tables
- Differential driver
 - Complementary current

Pulldown + GND clamp + Power Clamp



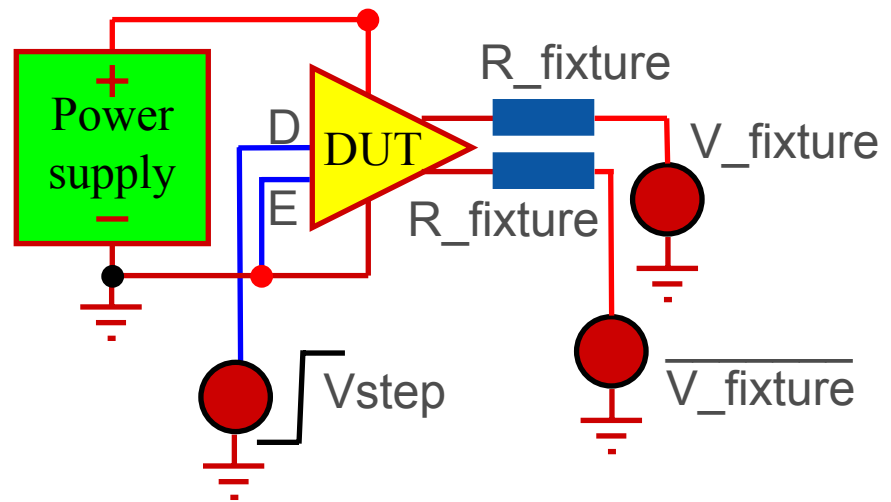
Pullup + GND Clamp + Power clamp



Model Creation

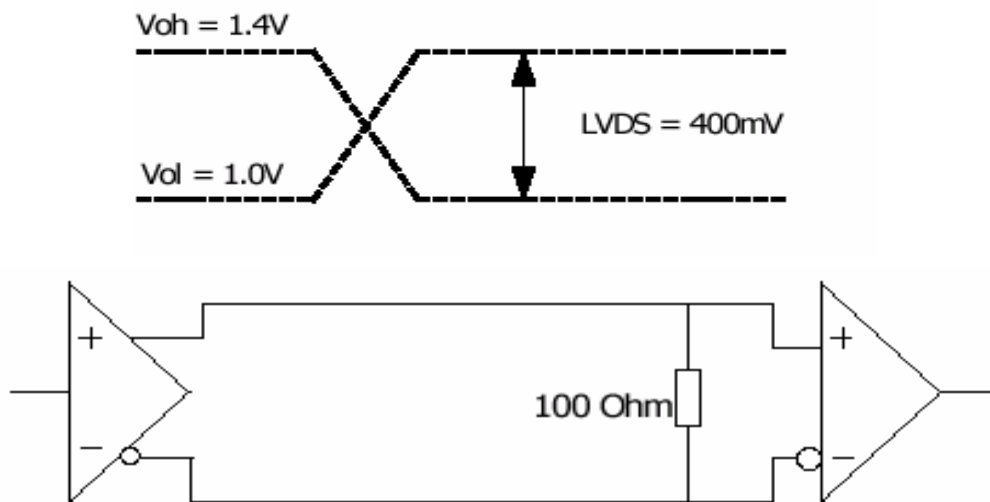
Obtaining V-t tables

- V_{step} at core edge rate
- Without terminators
 - If represented separately in the IBIS model
- Four tables
 - $D=\text{Low}$, $V_{\text{fixture}}=V_{\text{low}}$
 - $D=\text{Low}$, $V_{\text{fixture}}=V_{\text{high}}$
 - $D=\text{Hi}$, $V_{\text{fixture}}=V_{\text{low}}$
 - $D=\text{Hi}$, $V_{\text{fixture}}=V_{\text{high}}$



LVDS Model Example

- Ideal LVDS operation
- 400 mV differential mode
- 1.2 V common mode
- 100 Ω termination



LVDS Model Example

V-t tables

- V-t tables needed for DC operating point
- Timing between rising and falling edges
- Timing relative to core data signal
- Many ways to include differential terminator
 - Termination current in I-V tables
 - Termination current using [Series Current]
 - Termination current using [Series MOSFET]
 - Termination in package model

LVDS Model Example

- LVDS IBIS Models @ 1.25GHz

- Douglas Burns, SiSoft
- Used for next two slides
- Intentional time offset in plots

IBIS Summit <http://www.eda.org/pub/ibis/summits/jun02>

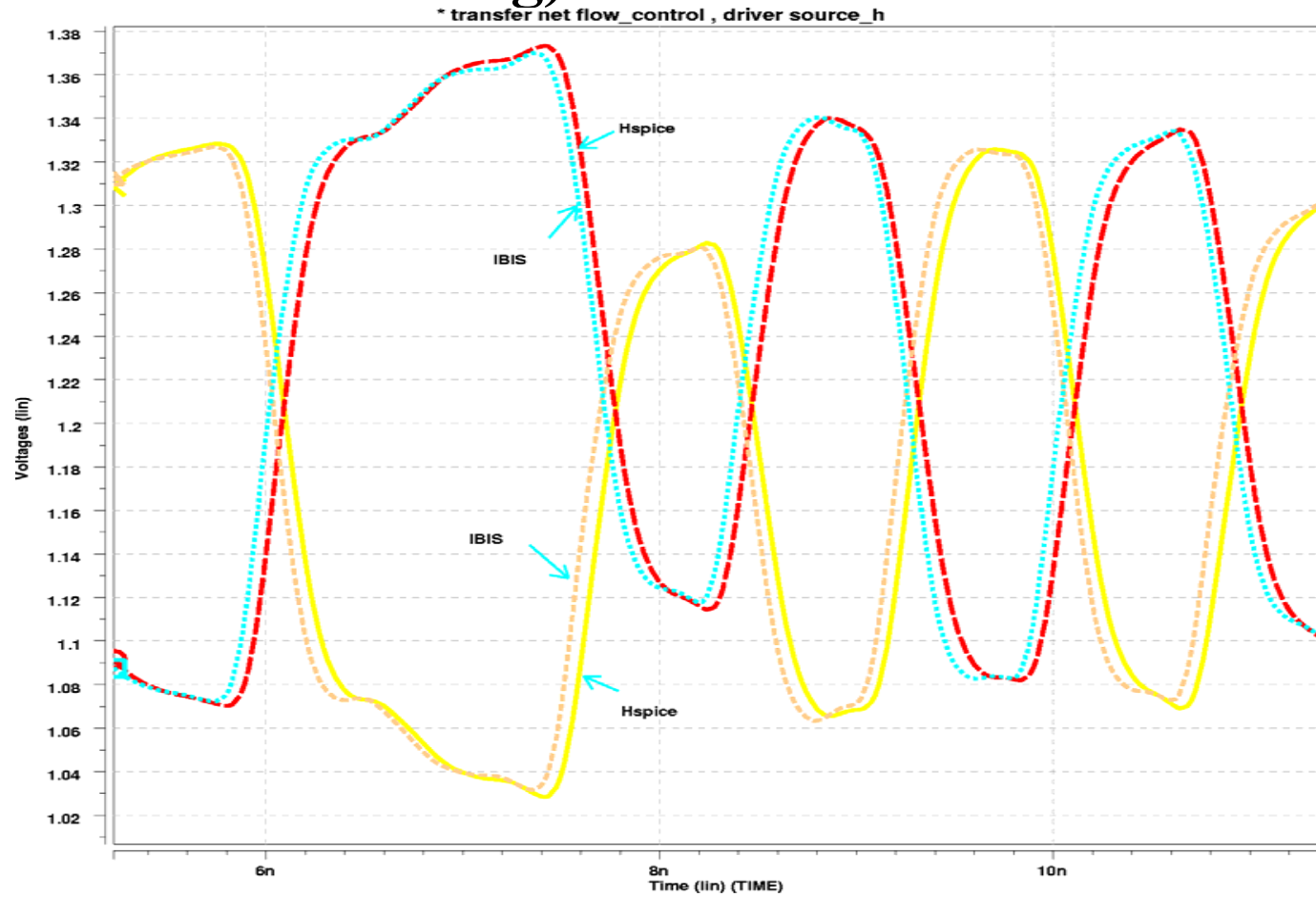
burns.zip: LVDS IBIS Models @ 1.25GHz (.ppt)

burns.pdf: Douglas Burns, Steven Coe, and Kevin Fisher,
Signal Integrity Software (SiSoft)

- Capabilities and limitations of IBIS models

Accurate LVDS IBIS Model @ 1.25GHz

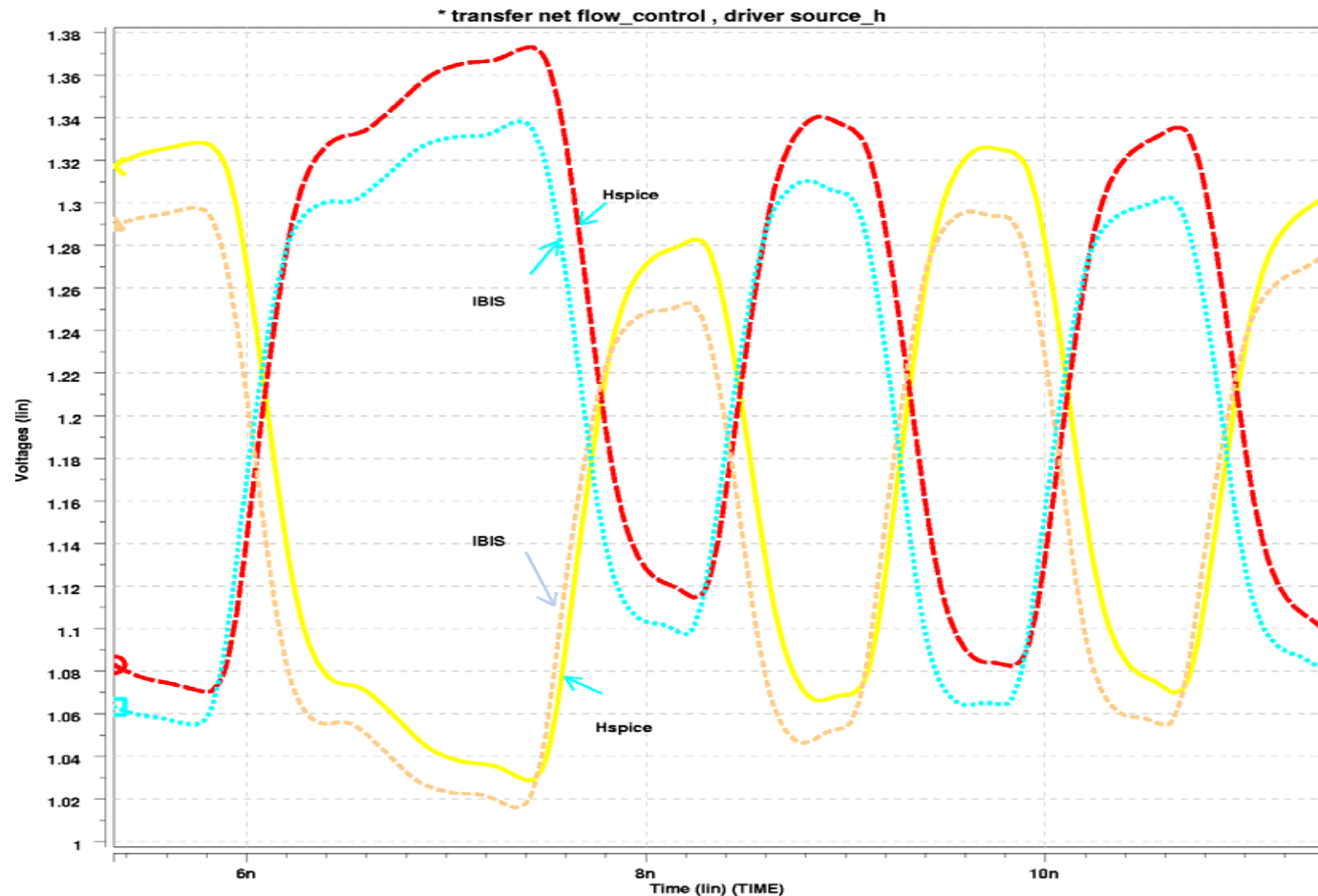
SiSoft presentation (time shift to make it easier to see matching)



SPI4 interface: 1.25GHz, target pad, VDDQ=2.375

Effects of Changing VDDQ

SiSoft presentation (same time shift)



SPI4 interface: 1.25GHz, target pad, VDDQ=2.325,
Model generated w/VDDQ=2.375

LVDS Model Example

- Model made at one V_{common}
 - Not valid if V_{common} changes
- Implications
 - Hard to select Min, Max conditions
 - Submodels for different V_{common}
- Reality is V_{common} changes with bit pattern

IBIS was never designed to handle this!

LVDS Model Example

The IBIS 4.1 solution

- IBIS 4.1 approach
 - SPICE 3f5, VHDL-AMS, Verilog-AMS
- LVDS model in SPICE
 - Requires both circuit and process data
 - Multiple NDAs could be required
- LVDS models in an AMS language
 - Behavioral equations
 - Use of both digital and analog information
 - Such as medium-term value of V_{common}
 - Can include effects not addressed in IBIS 4.0

Overview

- High-speed modeling techniques
- Differential buffer models
- On-die terminations
- Interconnect Models

On-die Terminators

- High-speed differential termination
- Inside the package and bond wire
- Minimizes reflection effects at the receiver
- Fixed or variable resistance

Representing On-die Terminators

Many ways to do this

- Use “terminator” model type
- Use [Series Current]
- Use [Series MOSFET]
- Include current in one of the clamp tables
- Include in a SubModel
 - Termination can be disabled
 - Value can be changed

Representing On-die Terminators Tradeoffs

- Ease of model creation
- Linear or non-linear load
- Actual FET or pass-gate load
- Support in a specific simulator

Representing On-die Terminators

Terminator in buffer method

- Do this within the [Model]

- To insert a resistor to GND

variable	R (typ)	R (min)	R (max)
[Rgnd]	100ohm	80ohm	120ohm

- To insert a series resistor

variable	R (typ)	R (min)	R (max)
[R Series]	8ohm	6ohm	12ohm

Representing On-die Terminators

Series current between pins

- Define series connection for component

```
[Series Pin Mapping] pin_2 model_name function_table_group
4 5 Rser1 | Series Resistor, always ON
```

- Define (linear or non-linear) resistance using I-V table

```
[Series Current]
```

Voltage	I (typ)	I (min)	I (max)
-5.0V	-3900.0m	-3800.0m	-4000.0m
-0.7V	-80.0m	-75.0m	-85.0m
-0.6V	-22.0m	-20.0m	-25.0m
-0.5V	-2.4m	-2.0m	-2.9m
-0.4V	0.0m	0.0m	0.0m
5.0V	0.0m	0.0m	0.0m

Overview

- High-speed modeling techniques
- Differential buffer models
- On-die terminations
- Interconnect Models

Module and Board Models

- EBD model
 - Simple transmission lines (L/R/C per unit length)
 - Lumped R, L, C elements
 - IC pin attachment
- PKG model
 - Adds RLC matrices for coupled lines
- ICM (connector and interconnect) model
 - Supports S-parameters and multi-section RLGC matrices

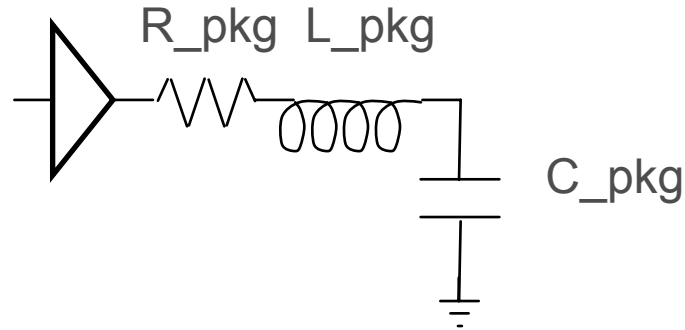
High-Speed Interconnect Models

Tool support

- EBD and PKG models
 - Check with ibischk3 / ibischk4
- IBIS Interconnect (ICM)
 - Check with icmchk1
- [External Circuit]
 - New in IBIS 4.1
- EDA tools connect package to component
 - Pkg Pin == IC Pad == Buffer

High-Speed Interconnect Models

- IBIS lumped package
 - R_{pkg} , L_{pkg} , C_{pkg}
 - Parameters are required
 - Values can be set to 0.0
 - Can define Typ/Min/Max values
- Real packages are transmission lines
 - Impedance
 - Delay



High-Speed Interconnect Models

- Lumped parameters
 - Only valid at slow edge rates
- Use of default parameters
 - Can override on a pin-by-pin basis

```
R_pkg  0.12  0.10  0.15
```

```
L_pkg  2n    1n    3n
```

```
C_pkg  3p    2p    5p
```

```
[Pin]
```

```
! Pin name      Model Name      R_pkg L_pkg C_pkg
```

```
D1           IO_1
```

```
DD1          IO_1          0.090      6n      7p
```

High-Speed Interconnect Models

- Package Override Order
- [Package Model] > [Pin] values > *_pkg

R_pkg 0.12 0.10 0.15

L_pkg 2n 1n 3n

C_pkg 3p 2p 5p

[Pin]

! Pin	name	Model	Name	R_pkg	L_pkg	C_pkg
-------	------	-------	------	-------	-------	-------

D1		IO_1				
----	--	------	--	--	--	--

DD1		IO_1		0.090	6n	7p
-----	--	------	--	-------	----	----

[Package Model] pkg1

High-Speed Interconnect Models

- Package model location
- In same file as IBIS file using [Package Model]
 - *.ebd, *.pkg
- In a separate file
 - File name based on package model type
 - *.ebd, *.pkg, *.icm
 - Must be in same directory
- If files are in different directories
 - Move one of the files

High-Speed Interconnect Models

- Lumped vs. distributed parameters
 - Using same parameters could result in different characteristics
 - *_pkg are lumped parameters by definition
- Distributed parameters
 - Values of parameters change (do not use lumped parameters)
 - Use EBD or Package or Interconnect
 - Valid at both fast and slow edge rates
- Bond wires must be included somewhere
 - Usually in the package model parameters

High-Speed Interconnect Models Comparison

	EBD	PKG	ICM
Lumped	Yes	Yes	Yes
Coupled traces	No	Yes	Yes
Matrices	No	Yes	Yes
S-parameters	No	No	Yes
Y-connects and dangling connects	Yes	No	Yes

High-Speed Interconnect Models

- Characteristics of high-speed packages
 - Lumped model is not adequate
 - Crosstalk between package traces
- EBD models for packages
 - Does not model any coupling
- PKG models for packages
 - Transmission line models
 - RLC matrices for coupling
- ICM models for packages
 - Both matrix and S-parameters (Touchstone format)

High-Speed Interconnect Models

EBD Example

[Path Description] IN0

Pin 1	Edge of Module
Len=0 L=0.2n R=10m /	Lumped Connector trace
Len=2.1 L=2.0n C=12.0p /	Units in inches
Fork	A Tee-connection
Len=40 L=0.15n C=1.2p /	Units in mils
Len=0 L=0.6 R=0.02 /	Bond wire to U0 Pin2
Node U0.2	
Endfork	
Len = 0 C = 0.8p /	Socket to U1 Pin2
Len = 0 L = 3.7n /	
Len = 0 R = 90m /	
Node U1.2	

High-Speed Interconnect Models

EBD Example

[Path Description] IN2

Pin 2

Len = 1.5 L=6.0n C=2.0p / | Trace on module

Len = 0 R=50 / | Series terminator

Len = 0.25 L=6.0n C=2.0p / | Trace between R and package

Node R2.1 | Series resistor pack

Node R2.2

Len = 0.25 L=6.0n C=2.0p / | Trace between R and package

Node U0.4

[Reference Designator Map]

Ref Des	File name	Component name
U0	good1.ibs	nonesuch
U1	good1.ibs	nonesuch
R2	r10k.ibs	A_10K_Pullup

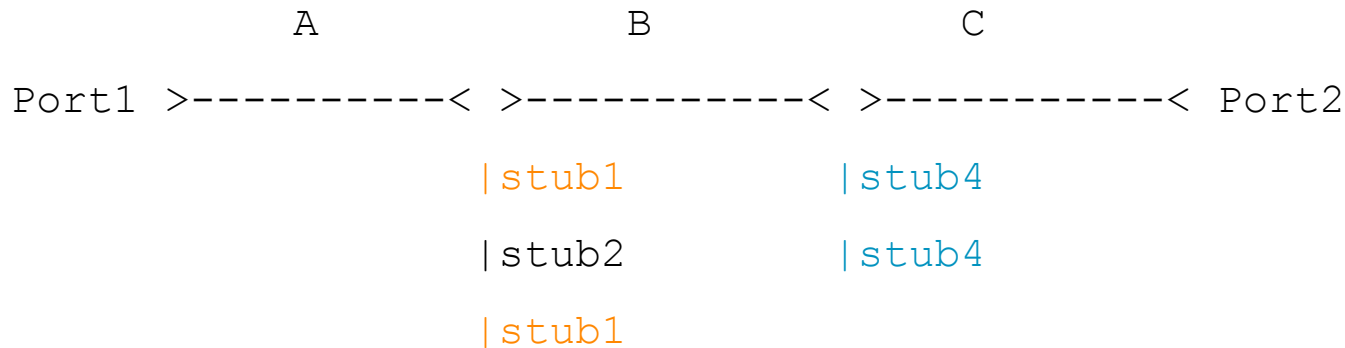
High-Speed Interconnect Models

PKG Example

```
[Inductance Matrix]      Full_matrix
[Row]    1
3.04859e-07  4.73185e-08  1.3428e-08  6.12191e-09
1.74022e-07  7.35469e-08  2.7321e-08  1.33807e-08
[Row]    2
3.04859e-07  4.73185e-08  1.3428e-08  7.35469e-08
1.74022e-07  7.35469e-08  2.7320e-08  1.74022e-07
[Capacitance Matrix]     Sparse_matrix
[Row]    1
1          2.48227e-10
2          -1.56651e-11
[Row]    2
2          2.51798e-10
```

High-Speed Interconnect Models

ICM Model Using [Tree Path Description]



Section Mult=1 A

Fork

Section Mult=1 stub1

Section Mult=1 stub2

Section Mult=1 stub1

Endfork

Section Mult=1 B

Fork

Section Mult=2 stub4

Endfork

High-Speed Interconnect Models

ICM Model Using [Nodal Path Description]

1 2 3	4 5 6	shell	6-pin male mini DIN
(computer)			
+--+	+--+	+---+	
			Section
2 3	4 6	shell	6-pin female DIN
(keyboard)			
2 3	4 6	shell	6-pin female DIN (mouse)

```
[Begin ICM Model] PS2_splitter
ICM_model_type MLM
| as found on http://www.hardwarebook.net/adapters/
| userinput/ ps2keyboardgateway.html
[Nodal Path Description]
Model_nodemap PS2_splitter_Computer_side
    N_section (C1 C3 C4 C5 Cs C2 C3 C4 C6 Cs
                K2 K3 K4 K6 Ks M2 M3 M4 M6 Ms) Mult=1
RLGC_matrix_10x10
Model_nodemap PS2_splitter_Keyboard_side
Model_nodemap PS2_splitter_Mouse_side
```

High-Speed Interconnect Models

ICM Model Using [ICM S-parameter]

- S-parameters in Agilent's Touchstone format
- IBIS Touchstone draft
 - http://www.eda.org/pub/ibis/connector/touchstone_spec11

```
[ICM S-parameter]
File_name sample.s4p | any name and extension
Port_assignment
|Port      Node
  1         A1
  2         B1
  3         A2
  4         B2
```


Summary

- Not all features are supported by all vendors
 - IBIS 4.1 parser is not yet available
 - Support is limited for IBIS 4.1 and ICM
 - Check with your EDA vendor
- IBIS 3.2 and critical IBIS 4.1 features are better supported
 - Model validation
 - SPICE-to-IBIS
 - S-parameters
 - IBIS with SPICE and VHDL/Verilog-AMS

Summary

- Creating a high-speed model is non-trivial
- More than one way to do things
 - On-die termination
 - LVDS simulation
- True differential drivers are hard to model
 - IBIS 4.0 assumes no signal coupling between buffers
 - IBIS 4.1 approach offers greater flexibility
- Package models have become more important
 - Critical designs will move to ICM models over time