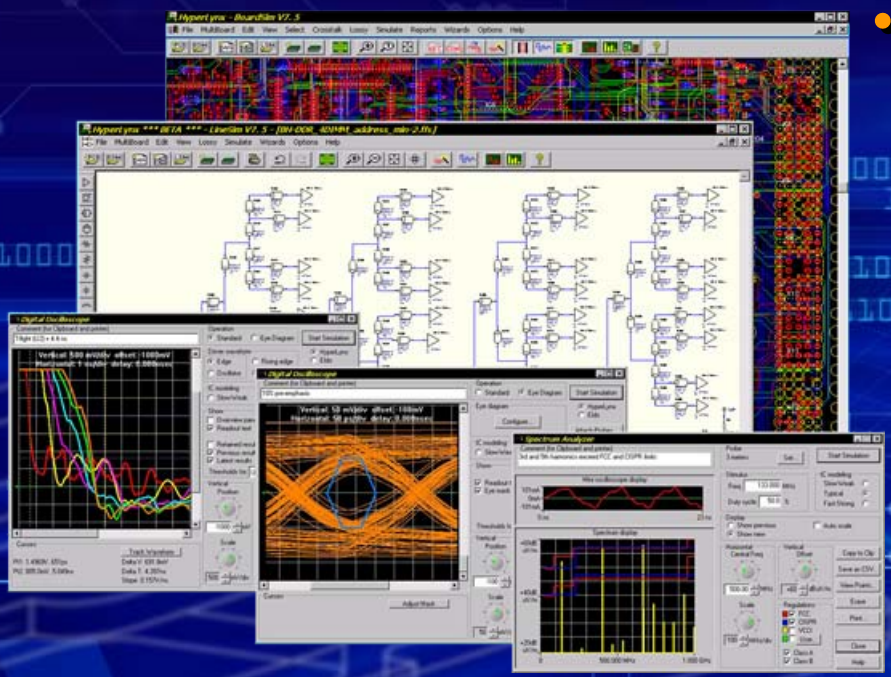
Forward looking trends in SERDES modeling

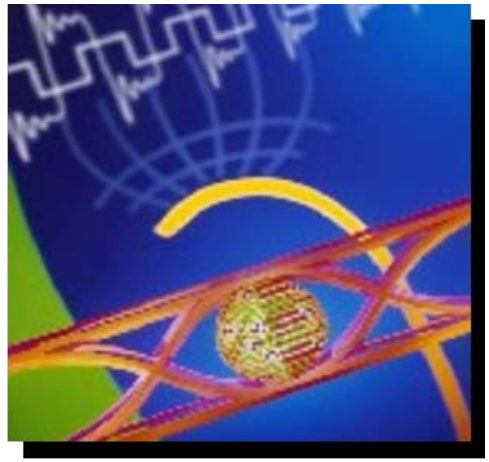


Eric Rongere
Stephane Rousseau

Mentor Graphics®

AGENDA

Forward looking trends in SERDES modeling



1. Current Status & Modeling Options
2. AMS Case Studies: SERDES and DDR2
3. Looking Forward

Current Status

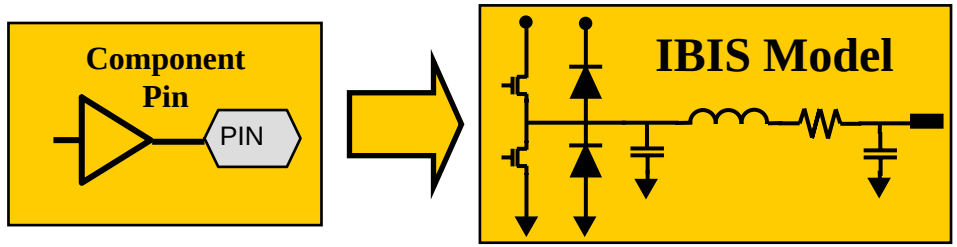
- Traditional IBIS still adds value for modeling behavior, but lacks the ability to adequately characteristics of devices used in state of the art communication channels:
  - Drivers with pre-compensation
  - Receivers with input slew rate sensitivity
  - Phase-locked loop clock and data recovery
  - Simple and adaptive equalization
  - Multi-level signaling
  - Multiple receiver thresholds (e.g, DDR2/3)

- We have additional challenges with measurement ... E.g.,
  - Receivers with slew-rate sensitivity
  - Self-clocked data streams
- Though it's still the first choice where it is sufficiently accurate, IBIS no longer supports the 99% criteria
  - We need a new "lowest common denominator" solution for these needs

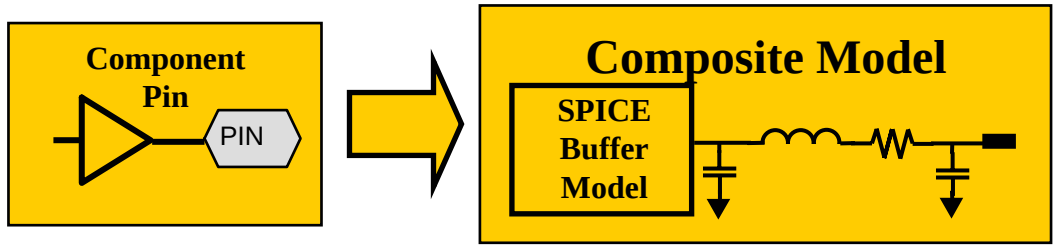
IBIS 4.2 Multi-lingual Extensions

Addressing Most or All of the Limitations

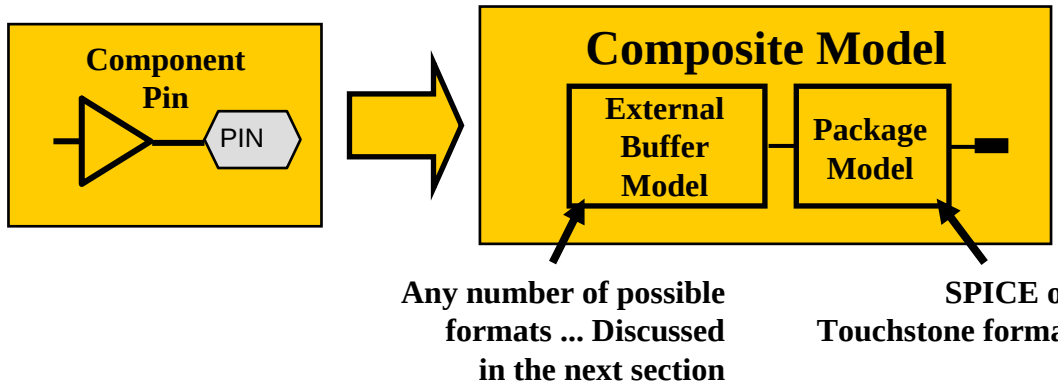
- **IBIS only**



- **IBIS Parasitics with external SPICE model (e.g., Eldo or HSPICE)**



- **IBIS model with external SPICE circuit and external VHDL-AMS model**

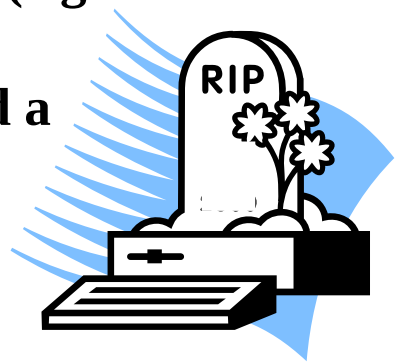


Option 1: Transistor-Level SPICE

- **Pros**
  - IC manufacturers can use the internal models that were developed from the Silicon
- **Cons**
  - Encryption is required to hide vendor IP; Each EDA vendor has their own encryption package and proprietary foibles in their transistor primitives
  - Simulation is slow compared to IBIS or other options
  - Complicated models are prone to hidden problems and instability
  - Implementing flexible measurements in the model (e.g., dynamic overshoot) is too difficult
  - To solve the overall problem we would have to add a comprehensive measurement language to IBIS

Option 2: Macro-models

- **Pros**
  - Will work with any EDA vendor that has AMS, or is willing to add the (macro) building blocks into their SPICE-like simulator
  - Faster simulation than transistor level models
- **Cons**
  - The current building blocks have only been proven in their support of IBIS 3.2
  - It is likely that new building blocks will soon be needed
  - Implementing flexible measurements in the model (e.g. the new dynamic overshoot) is too difficult
  - To solve the overall problem we would have to add a comprehensive measurement language to IBIS



Option 3: Analog-Only AMS*

- **Pros**
  - Most EDA vendors have access to analog-only AMS
  - Stepping stone to full AMS
  - Relatively easy for the model to do its own measurements
    - Can be output as text messages ready for the spreadsheet
    - Augment in IBIS with a simple list of column headings to add to the spreadsheet
  - Fast simulation
- **Cons**
  - The languages are complex and no vendor has every feature supported
    - Could be alleviated by agreeing to a preferred language subset
  - Implementing digital features—e.g., extended history in an equalizer requires jumping through hoops

* Analog “AMS” generally differs from “A” in its support for discontinuous functions

Option 4: AMS

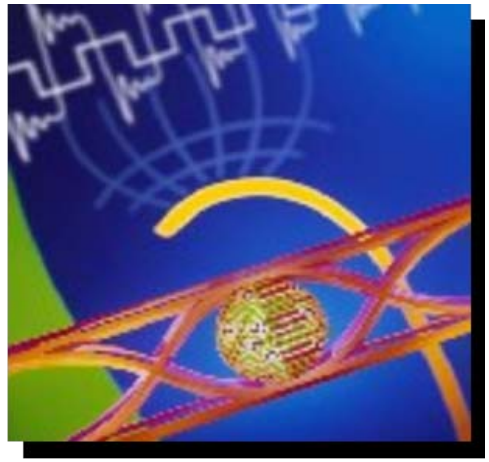
- **Pros**
  - VHDL-AMS and Verilog-AMS are international standards
  - The most flexible solution, compared to Options 1-3
  - Easy for the model to do its own measurements
    - Can be output as text messages ready for the spreadsheet
    - Augment in IBIS with a simple list of column headings to add to the spreadsheet
  - Fast simulation
    - Models are compiled to machine code just like built in primitives
    - Digital content is handled in event driven kernel
- **Cons**
  - Only one EDA vendor currently offers AMS in their SI tools
  - The languages are complex and no vendor has every feature supported
    - Could be alleviated by agreeing to a preferred language subset

Option 5: Algorithmic Modeling

- **Description**
  - Vendor-neutral, C-based models that would be owned, developed and compiled by IP vendors
  - Based on a yet-to-be-defined API/spec from the IBIS committee
- **Pros**
  - Measurements would likely be supported
  - EDA Vendor neutral
  - Encryption would hide IP
  - Fast simulation
    - Models are compiled to machine code as a DLL or executable
- **Cons**
  - No EDA vendor currently offers this support
  - While easy for some vendors—because they're already creating these models—many vendors would require a good bit of work to create these models

AGENDA

Forward looking trends in SERDES modeling

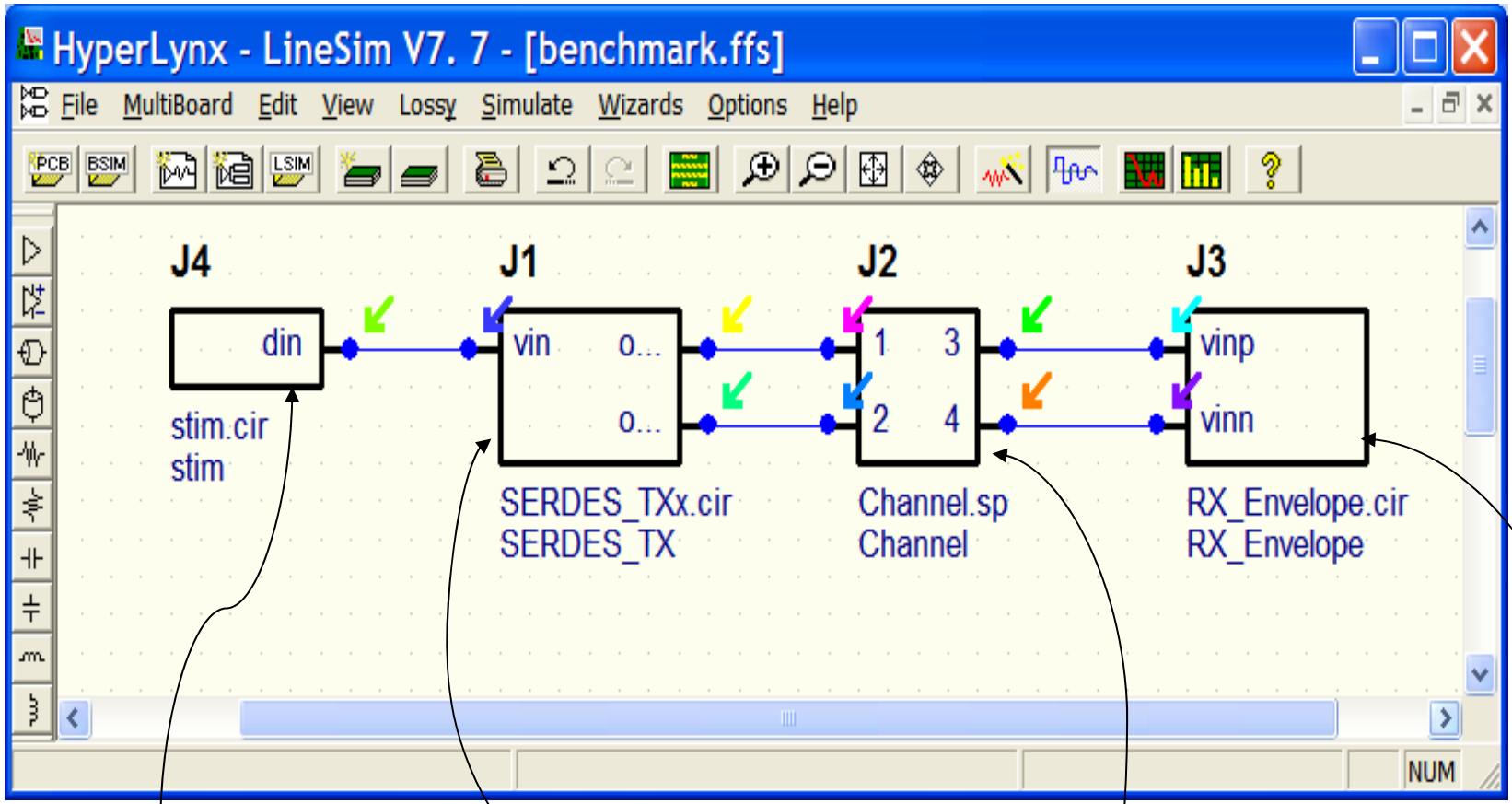


1. Current Status & Modeling Options
2. **AMS Case Studies: SERDES and DDR2**
3. Looking Forward

AMS Case Study 1

Full non-linear analysis of a SERDES channel

- **Objectives:**
  - **VHDL-AMS Driver with non-linear drive characteristics and pre-compensation**
  - **Employ a realistic S-Parameter model for packages, two connectors and a backplane**
  - **VHDL-AMS receiver model with built-in (AMS) envelope recorder**
  - **Simulate to 10 million data bits using a custom data pattern on an average single-processor notebook computer running Microsoft Windows and an appropriate simulation time-step for accurate results**



AMS PRBS Generator

AMS SERDES TX

S-Parameter TX Package, Stub, Connector, Backplane, Connector, Stub, RX Package

AMS RX and Envelope Generator

The VHDL-AMS SERDES Transmitter Model

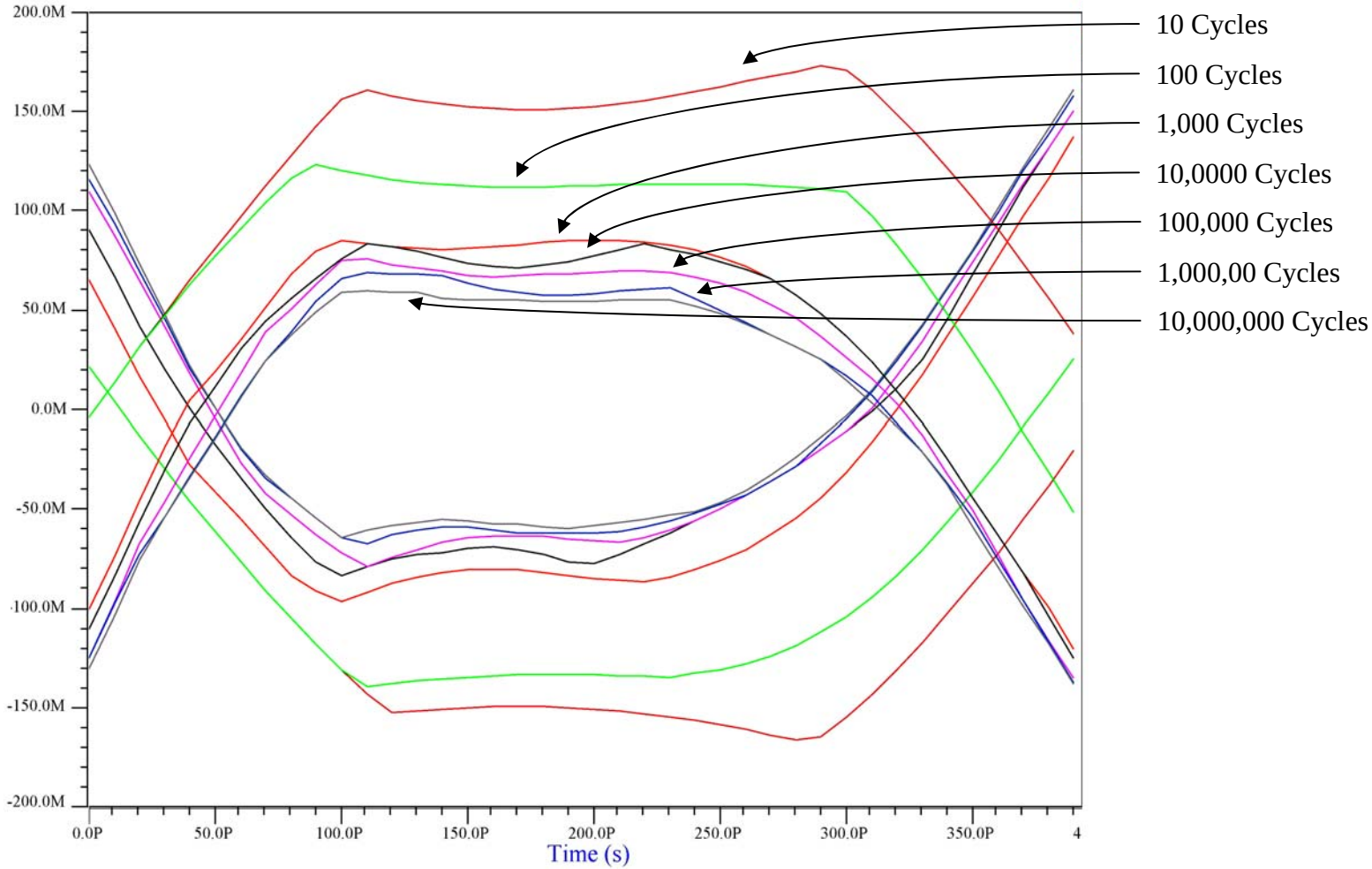
```
begin
  -- output the proper current based on the state of signal din,
  -- and values of constants Ipe and Imain
  if domain = quiescent_domain use  – if DC then
    itxp == Ipe/2.0; itxn == Ipe/2.0;     -- set both outputs to half
  elsif din='1' and din'delayed(bit) = '0' use
    itxp == Ipe; itxn == 0.0;             -- first pulse (txp positive)
  elsif din='1' and din'delayed(bit) = '1' use
    itxp == Imain; itxn == Ipe-Imain; -- normal pulse (txp positive)
  elsif din='0' and din'delayed(bit) = '1' use
    itxp == 0.0; itxn == Ipe;             -- first pulse (txn positive)
  elsif din='0' and din'delayed(bit) = '0' use
    itxp == Ipe-Imain; itxn == Imain; -- normal pulse (txn positive)
  end use;
  break on din, din'delayed(bit) ;   -- deal with the discontinuities

  -- P and N-side C_comp, R_term, Vdd
  i_r_term_p == (vtxp - Vdd)/R_term; i_c_comp_p == c_comp * vtxp'dot;
  i_r_term_n == (vtxn - Vdd)/R_term; i_c_comp_n == c_comp * vtxn'dot;

end architecture;
```

Results: Simulation to 10 Million Data Cycles

(All simulations completed overnight)

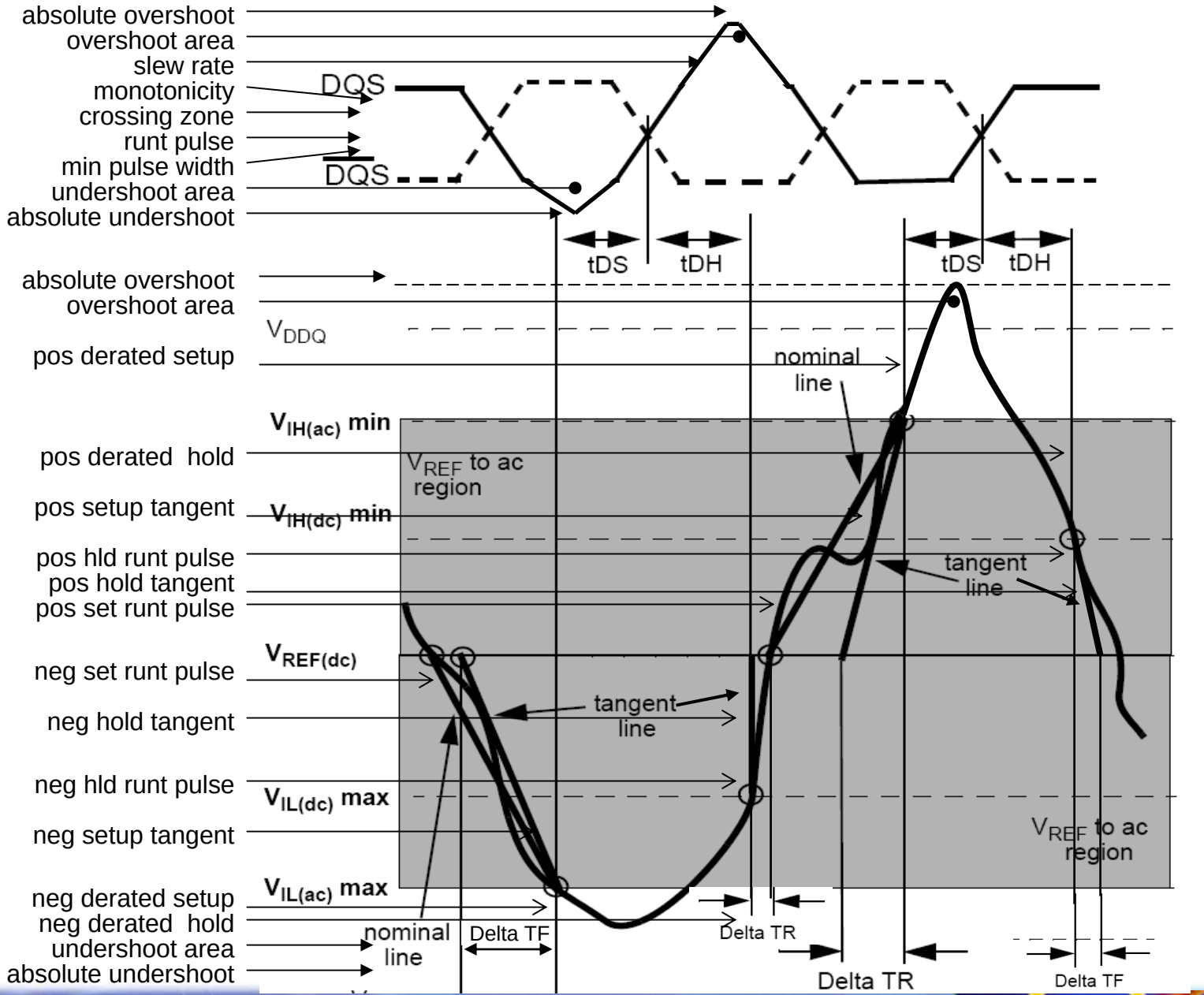


AMS Case Study 2:

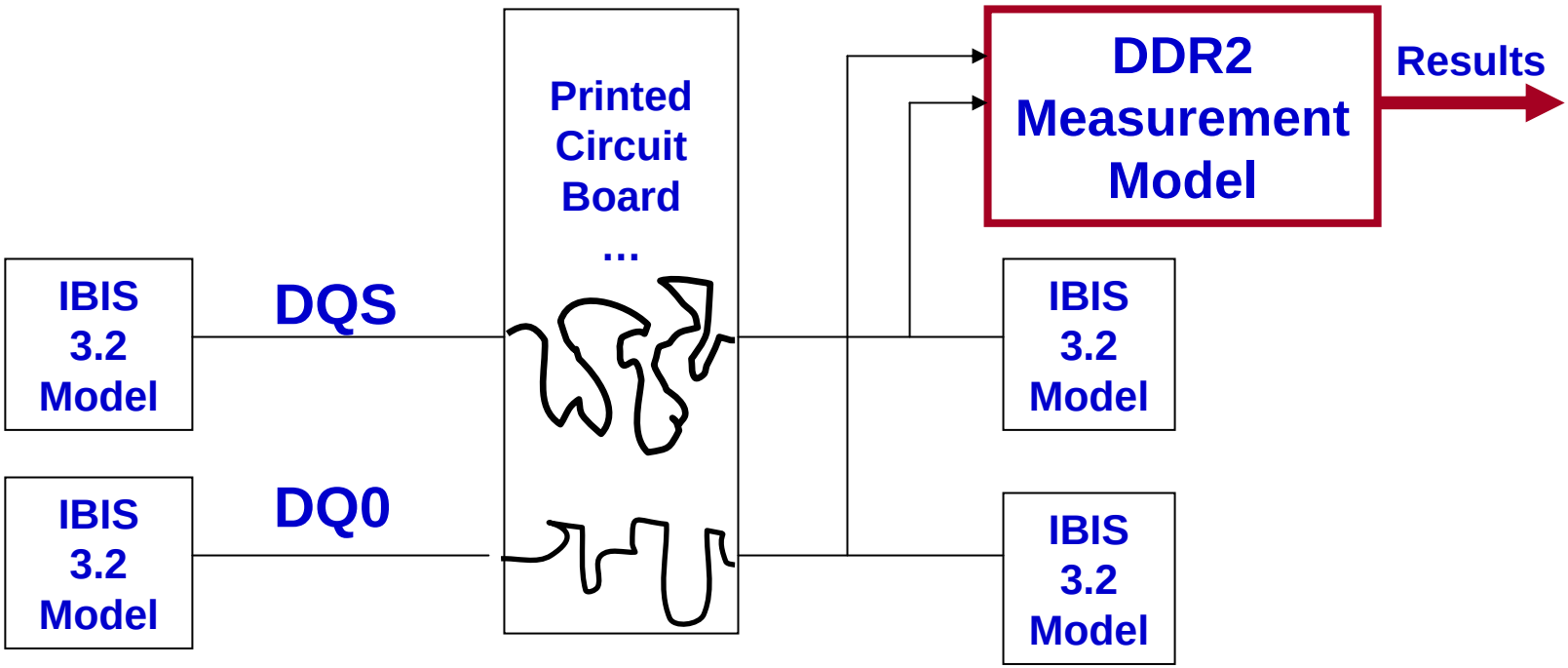
Automated DDR2 Measurements

- **Objectives:**
  - Implement all measurements specified in the DDR2 datasheet in a VHDL-AMS model
  - Utilize off-the-shelf IBIS 3.2 driver and receiver models
  - We want to be able to include DDR2 nets in an automated scan of the whole PCB
  - We do not want to hard code DDR2 specific features into the waveform analysis or results spreadsheet
    - This would be a never ending job and delivered later than typical delivery schedules
- **Issues:**
  - The clock and data receivers are slew-rate sensitive
  - Dynamic overshoot is specified using a VT area

DDR2 Electrical and Timing Constraints

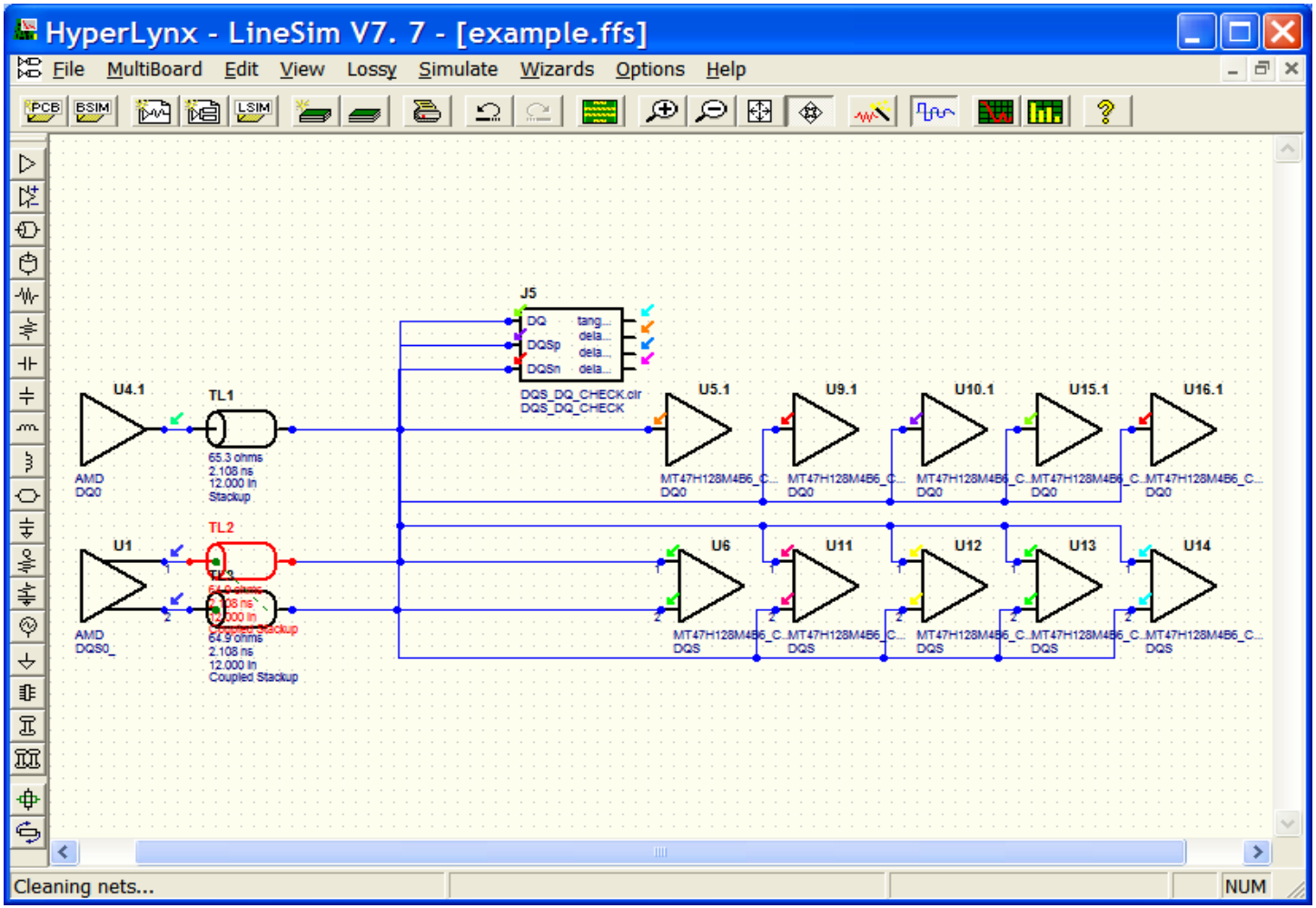


DDR2 IBIS 4.2 Measurement Model



Pre-Layout Analysis

Using the DDR2 IBIS 4.2 Measurement Model

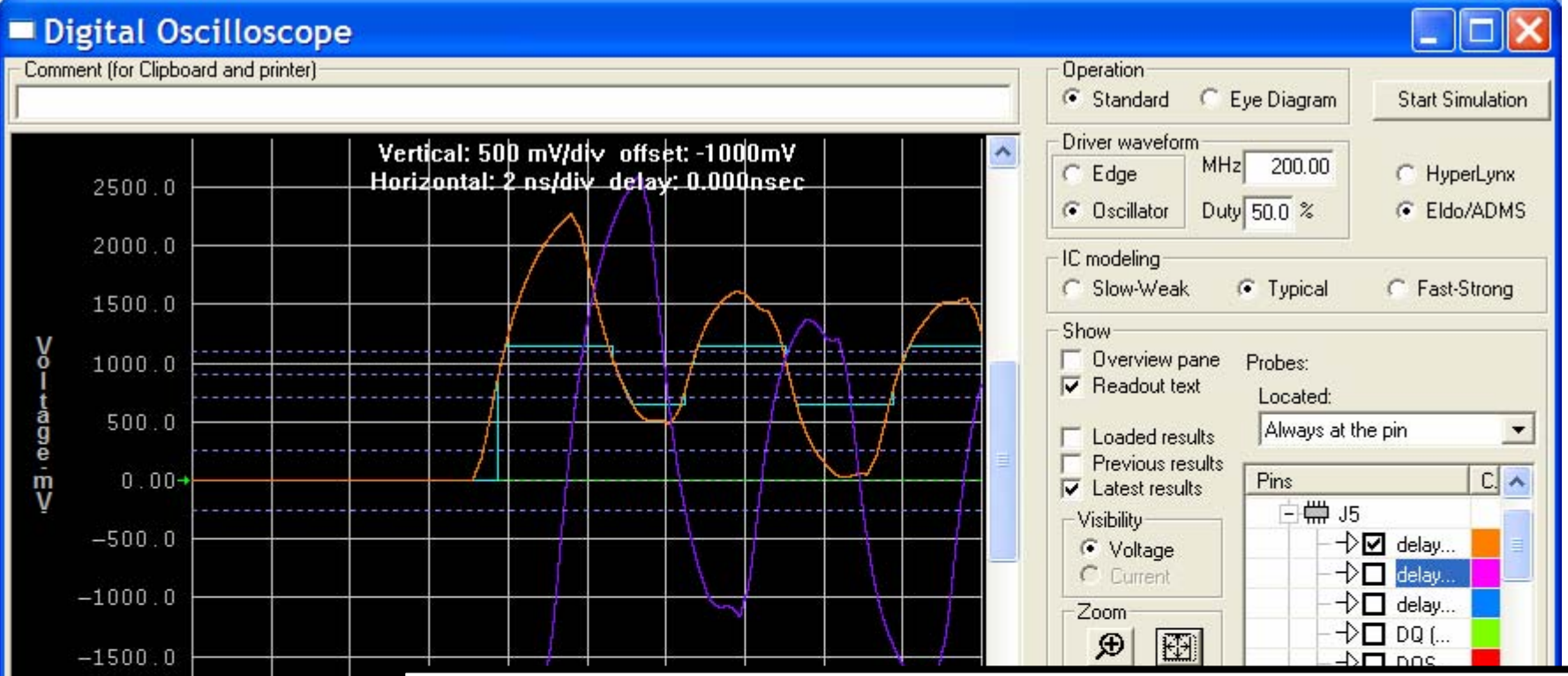


TANGENT MEASUREMENT

- Wait for vref crossing
- Store data points
- Wait for vix_ac cross
- Calculate the slope from each point to the vix_ac crossing point
- Return the maximum slope
- Wait for vix_dc crossing
- Calculate the slope from each subsequent point back to the vix_dc crossing
- Wait for vref crossing
- Return the max slope

```
begin
   ------------------------------------------------------------------------
   -- measure the setup time tangent
   ------------------------------------------------------------------------
   wait until VREFDC; -- wait for a crossing of correct direction
   max_slope:=0.0; data_point_cntr:=0;  setup_crossing <= 0.0*sec;
   while not vix_ac'event loop -- store all the data points until vix_ac crossing
      data_point_v(data_point_cntr) := Vin'reference;
      data_point_t(data_point_cntr) := now;
      wait on vix_ac, ASP; -- wait for next event
      data_point_cntr := data_point_cntr + 1;
   end loop;  -- go on to find the maximum slope
   setup_crossing <= now;
   for i in min_slope to data_point_cntr-1 loop
      slope := (crossing_point_v - data_point_v(i)) /
               (crossing_point_t - data_point_t(i));
      if slope > max_slope then max_slope:=slope; end if;
   end loop;
   setup_slope <= max_slope;
   ------------------------------------------------------------------------
   -- measure the hold tangent
   ------------------------------------------------------------------------
   wait until not vix_dc;  -- wait for opposite crossing of vix_dc
   max_slope := 0.0;
   crossing_point_v := Vin'reference; crossing_point_t:=now;
   -- calculate slope of each point until vix_dc, or max_points
   while not VREFDC'event loop
      wait on VREFDC, ASP ;
      slope := -(Vin'reference - crossing_point_v) /
                (now - crossing_point_t);
      if slope > max_slope then max_slope := slope; end if;
   end loop;
   hold_slope <= max_slope; -- in v/s
end process;
```

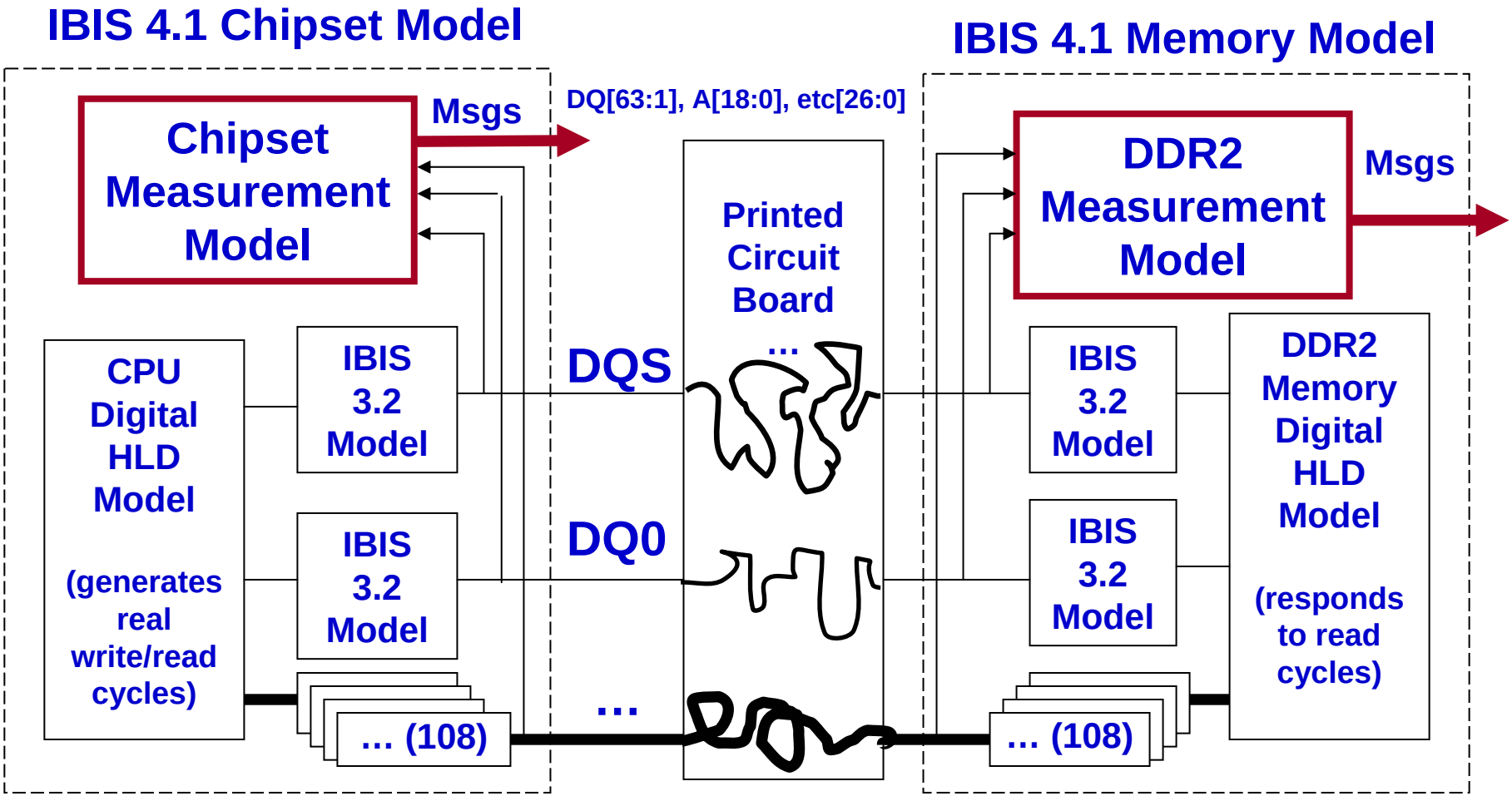
(error and exception handling removed for clarity)



U2 pin 1 Violation - MAXOVERSHOOT or MAXUNDERSHOOT Level Exceeded at time: 2337
U2 pin 1 Violation - MAXOVERSHOOT or MAXUNDERSHOOT Level Exceeded at time: 4790
U2 pin 1 Violation - MAXOVERSHOOT or MAXUNDERSHOOT Level Exceeded at time: 7290
U2 pin 1 Violation - MAXOVERSHOOT or MAXUNDERSHOOT Level Exceeded at time: 9815
U2 pin 1 Violation - MAXOVERSHOOT or MAXUNDERSHOOT Level Exceeded at time: 12294
U2 pin 1 Violation - MAXOVERSHOOT or MAXUNDERSHOOT Level Exceeded at time: 14816
U2 pin 1 Violation - MAXOVERSHOOT or MAXUNDERSHOOT Level Exceeded at time: 17293
U2 pin 1 Violation - MAXOVERSHOOT or MAXUNDERSHOOT Level Exceeded at time: 19816
U2 pin 2 Violation - DQS exceeded VIXACMIN at Differential Crossing at time: 1275
U2 pin 2 Violation - DSQ exceeded VINDCMAX or VINDCMIN at time: 5914
U2 pin 2 Violation - DSQ exceeded VINDCMAX or VINDCMIN at time: 5955
U2 pin 1 Setup Check PASSED. Expected: 0.275 ns. Actual: 1.55 ns at time: 4536
U2 pin 2 Warning: dqs slew value out of range of table.
dqs_slew= 4.33e+9. Max table index value= 4.00e+009 at time: 5799
U2 pin 2 Warning: dq slew value out of range of table.
dq_slew= 3.566e+9. Max table index value= 2.00e+009 at time: 5799
U2 pin 1 Hold Check PASSED, Expected: 0.32 ns. Actual: 1.177 ns. at time: 5799

Model for the Full DDR2 Channel

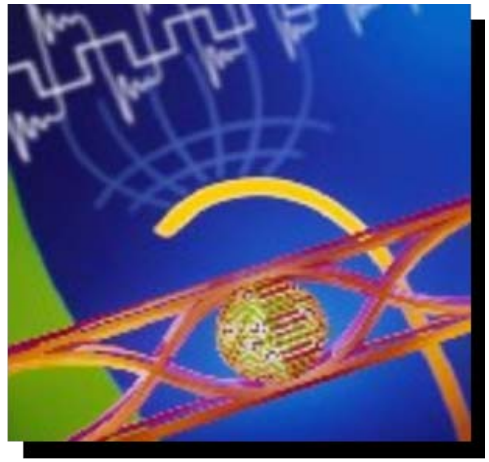
Integrating both Behavior and Measurement *



* Note: This model is not in strictly IBIS 4.2 compliant because it uses an external circuit that references an IBIS 3.1 model

AGENDA

Forward looking trends in SERDES modeling

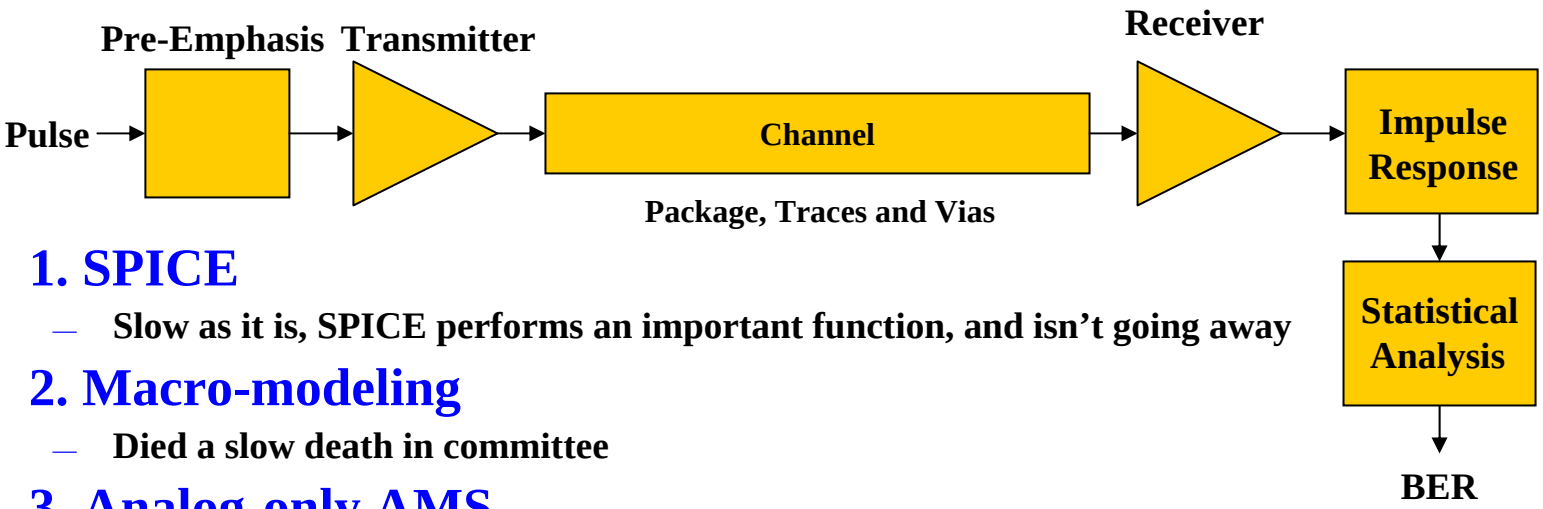


1. Current Status & Modeling Options
2. AMS Case Studies: SERDES and DDR2
3. **Looking Forward**

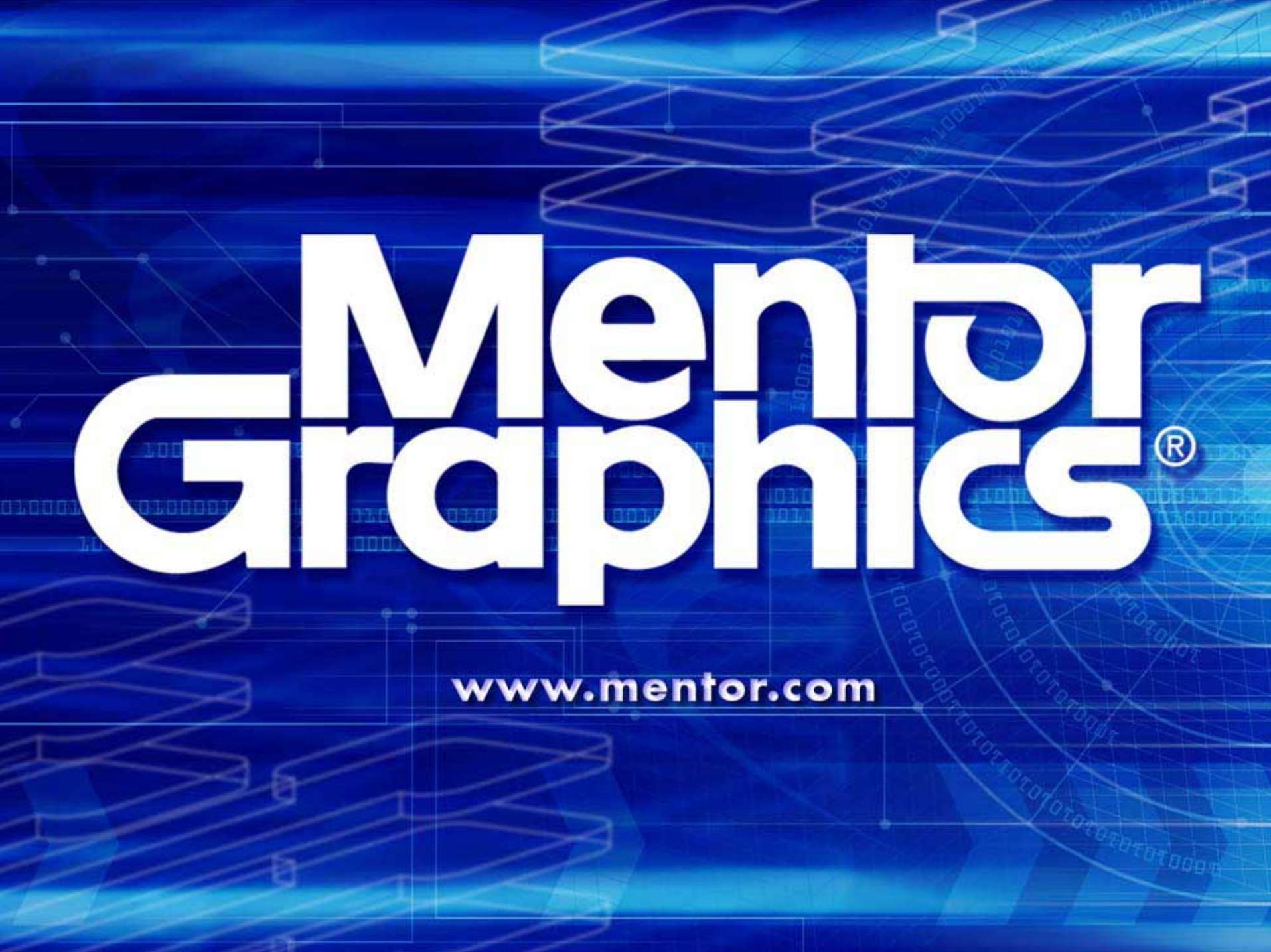
Conclusions

- **IBIS 4.2 Multi-lingual extensions**
  - The model maker and user can best decide whether it is best to create models using the IBIS 4.2 multi-lingual extensions utilizing SPICE or AMS
  - The best solution for the SI Engineer may well be a tool that supports the mixing of both
- **Necessary IBIS enhancements**
  - Create intelligent multi-lingual input models that delay their output signal based on the slew rate
  - Enhance the IBIS measurement facilities to support the new dynamic overshoot constraint
    - Describe how to make the measurement
    - Inform the spreadsheet to add new columns for the data

Summary of SERDES Modeling Options



- 1. SPICE
  - Slow as it is, SPICE performs an important function, and isn't going away
- 2. Macro-modeling
  - Died a slow death in committee
- 3. Analog-only AMS
  - Hasn't gotten much traction, but might represent the middle ground between functionality and vendor neutrality
- 4. AMS
  - Not vendor-neutral, but supported today for 45% of the market
  - Additional models and development tools are needed
- 5. Algorithmic modeling
  - Still in its infancy, but would meet functional requirements and be vendor-neutral
- 6. Statistical modeling (AKA: Linear Channel Analysis or Fast Eyes)
  - Still in it's infancy, but would meet functional requirements and be vendor-neutral


Mentor Graphics®
www.mentor.com