Three Facets of IBIS: Interface, Behavior and Measurement

December 2005

Ian C Dodd
Architect, High Speed Tools
SDD Division

Henry Li
Technical Marketing Engineer
Mentor Graphics (Shanghai)
Electronic Technology Co Ltd

Mentor Graphics®

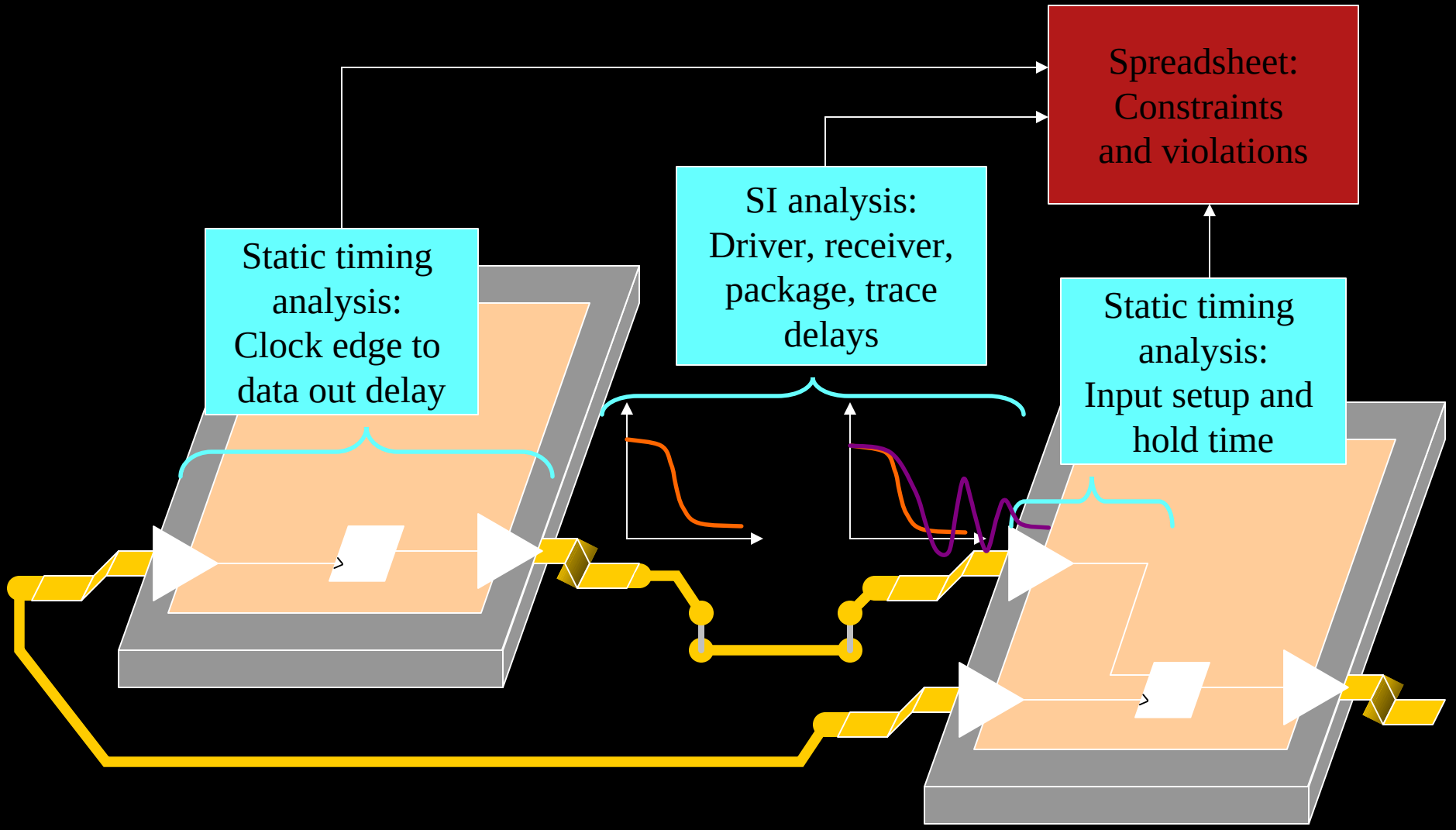
What are the Objectives of the SI Engineer?

- Maximize performance
  - Highest possible clock speed
- Ensure reliable operation
  - Meet timing constraints
  - Keep overshoot below levels that will cause device malfunction or destruction.
  - Maintain adequate switching margins
- Minimize per unit costs
  - Minimize the number of components (including terminators)
  - Use lowest cost PCB materials

How can these objectives be met?

- By investing in accurate timing and SI analysis so that safety margins can be controlled
  - Accurate static or dynamic timing models for the internal paths of active devices
  - Accurate SI models for drivers, loads and passive devices
  - Accurate models for the interconnect including the associated dielectrics
  - Using fully automated analysis tools so all vulnerable nets can be analyzed

Basic timing analysis



Part One:
Summary of IBIS to version 3.2

Highlights of IBIS 3.2 Behavior

- IBIS 3.2 assumes fixed topologies for drivers, inputs and terminators
- Support process range: Typ, Min & Max
- Drivers and Inputs
  - Two steady state Current versus Voltage (IV) tables representing power and/or ground clamps.
  - Capacitance
- Drivers
  - Nominal edge rates
  - Two additional steady state Current versus Voltage (IV) tables representing pull-up and pull-down transistors.
  - Optional Dynamic Voltage versus Time (VT) tables
    - Load conditions must be specified for each table

Highlights of IBIS 3.2 Behavior (continued)

- Series and Parallel Terminators
  - Resistance, Capacitance and Inductance
- Package Description
  - Assumes each external pin connects to one internal (die) pin
  - Default Package
    - Assumes no coupling: simple resistance, capacitance and inductance
  - Package Model
    - Allows multiple sections, each with coupled or uncoupled: resistance, capacitance and inductance

IBIS 3.2 Interconnect

- Provides mapping of the pins on a physical part to the models providing electrical behavior.
- Allows EDA tools to automatically build complete simulation models for nets on your PCB
  - Less error prone than creating simulation netlists manually
  - Allows for simulation coverage of the majority of nets on your PCB.

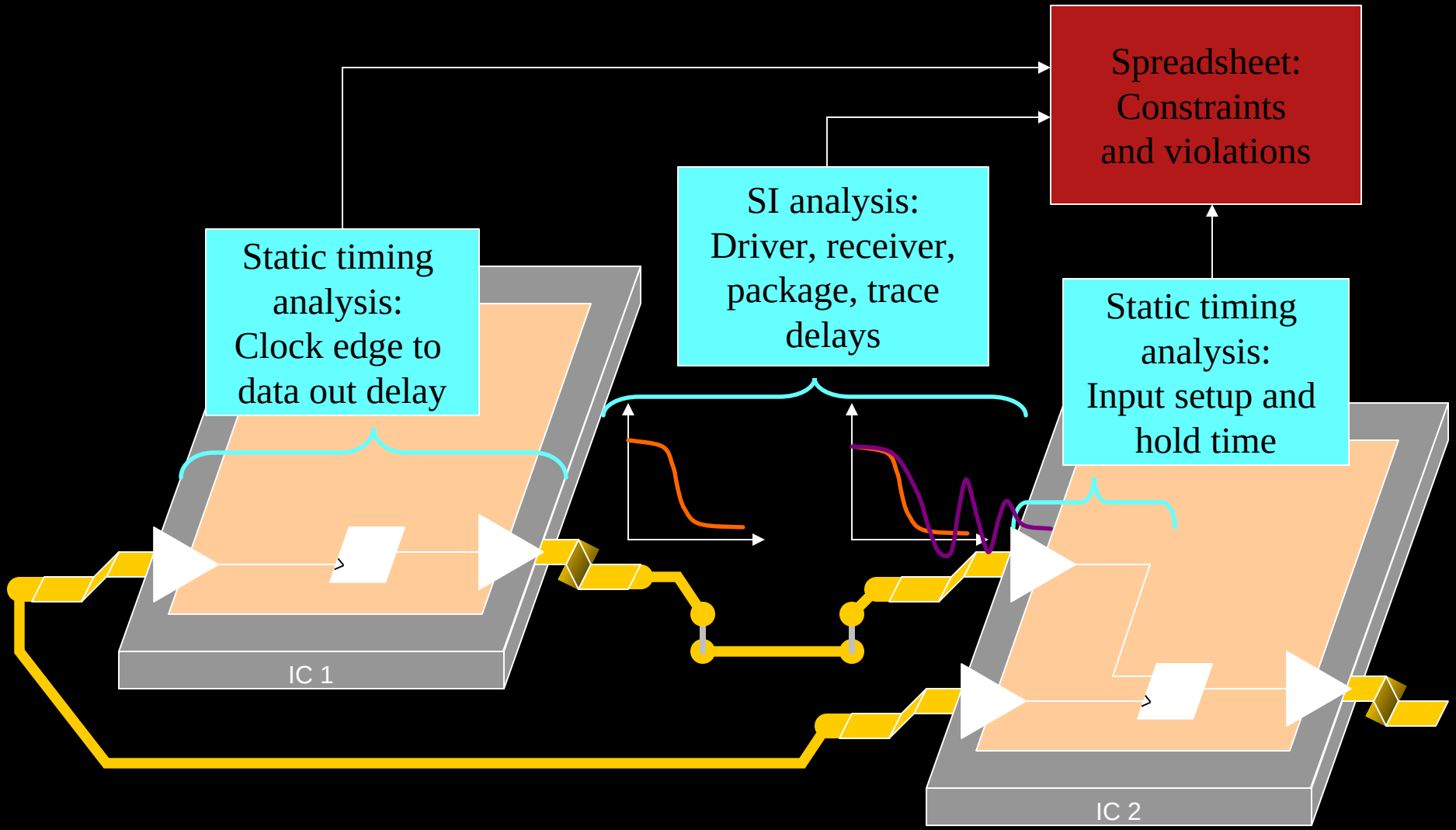
Highlights of IBIS 3.2 Interconnect

- **IBIS Pin Section**
  - **Assigns driver, load and parallel terminator models to pins**
- **Series Pin Mapping Section**
  - **Assigns series terminators (and switches) to pins**
- **Diff Pin Section**
  - **Identifies differential pin pairs & adds differential delay info**
- **Electrical Board Description (EBD)**
  - **Limited description of a PCB containing IBIS components**
  - **Does not support modeling of trace to trace coupling.**

Highlights of IBIS 3.2 Measurement

- Overshoot limits
  - If exceeded device may malfunction or be destroyed
  - Static and dynamic
- Input switching thresholds
  - Logic low
    - Vinl (or with hysteresis Vinl+ and Vinl-)
  - Logic high
    - Vinh (or with hysteresis Vinl+ and Vinl-)
- Timing Reference Voltage (Vmeas)
  - Transition voltage on driver pin used when measuring static timing delays e.g. “clock edge to data out”
  - Associated with a reference load
    - Vref, Rref and Cref

Basic timing analysis



Putting it all together with IBIS 3.2

- Simple synchronous example
  - First consider the data path
- Simulate the data path for the rising and the falling edges
  - Find the time at which the input pin on U2 leaves the initial logic state (raw minimum delay)
  - Find the time at which the input pin on U2 becomes stable at the final logic state (raw maximum delay)
- Simulate the driver connected to the timing reference load
  - Find the time to Vmeas
- Normalize the raw delays by subtracting the time to Vmeas
  - Removes any initial delay in VT table and partial edge time already included in the “clock edge to data out” delay.

Including Clock Skew

- Determine Clock Period
  - T_{clock}
- Simulate Data Path
  - Find Normalized Data Path Delays
    - TD_{data_min}, TD_{data_max} for rising and falling edges
- Simulate Clock Paths
  - Clock Driver to Clock Input on IC1
    - TD_{clock1_min}, TD_{clock1_max} for active edge
  - Clock Driver to Clock Input on IC2
    - TD_{clock2_min}, TD_{clock2_max} for active edge
- Get Internal Path Delay and Minimum Setup and Hold constraints from Databook
  - TD_{path}, T_{setup_min}, T_{hold_min}

Including Clock Skew (continued)

Worst Case Analysis

- Calculate the actual setup and hold times for the rising edge
- Compare the results to the timing constraints from the databook
- Repeat for the falling edge

$$T_{setup} = T_{clock} - TD_{data\_max} + TD_{clock2\_min} - TD_{clock2\_max}$$

$$T_{hold} = T_{clock} - TD_{data\_min} + TD_{clock2\_max} - TD_{clock2\_min}$$

EDA tools that have integration between timing and SI analysis commonly have spreadsheets that automate these calculations and highlight timing violations

Part Two:
Beyond IBIS Version 3.2

Technology Changes

- Data rates on PCB have dramatically increased.
- PCB trace lengths have not shrunk in proportion
  - clock periods are often shorter than the interconnect delay
  - Source synchronous design must be used either with a separate clock signal or with an embedded clock
- Power budgets have decreased
  - Signaling voltages have been accordingly decreased
- PCB materials have not dramatically improved
  - Dielectric losses become the predominant circuit effect
  - Differential signaling is required to overcome attenuation
  - Pre-compensation or equalization is becoming common to compensate for wavefront distortion

Highlights of Post IBIS 3.2 Behavior

- ICM provides multi-stage coupled lossy transmission line or S-parameter modeling for connectors
  - Presently considering allowing ICM package descriptions so we can have consistent S-parameter support
- Multi-lingual extensions support behavior modeled in SPICE, VHDL-AMS and Verilog-AMS.
  - Almost no limit to the complexity of behavior that can be modeled
  - Depending on the simulator, model types can be mixed across the PCB and even on individual nets
- Power Integrity extensions

Highlights of Post IBIS 3.2 Interconnect

- **Multi-lingual extensions provide flexible connectivity between SPICE, VHDL-AMS or Verilog-AMS models and the external component pins**
  - **Allows EDA tools to automatically build complete simulation models for nets on your PCB including the multi-lingual models**

Existing Post IBIS 3.2 Measurement

- No enhancements to IBIS built in measurements since version 3.2
- Multi-lingual input models can determine when a transition has occurred
  - External models and circuits for inputs have a standard terminal which can be monitored by EDA tools
- Multi-lingual models can do extensive analysis and use (external circuit) terminals to present the results of analysis
  - No conventions as to how an EDA tool should interpret signals on these terminals
  - Cannot presently be used for automated analysis or to populate constraint spreadsheets

Recommendation for New IBIS Measurements

- We want to validate a high percentage of nets on PCBs
  - Requires automated measurement and constraint validation
  - IBIS is needed to provide the interface and measurement functions even when SPICE or AMS descriptions are used for behavior
- Measurement enhancements
  - Support measurements specified for any IBIS external pin or internal terminal
  - Provide voltage or current thresholds
  - Provide logic and equations that the EDA tool should use to determine values to be reported
  - Provide limits for reported values

Part Three:
IBIS 4.1 Multilingual Example

Typical IBIS 4.1Model

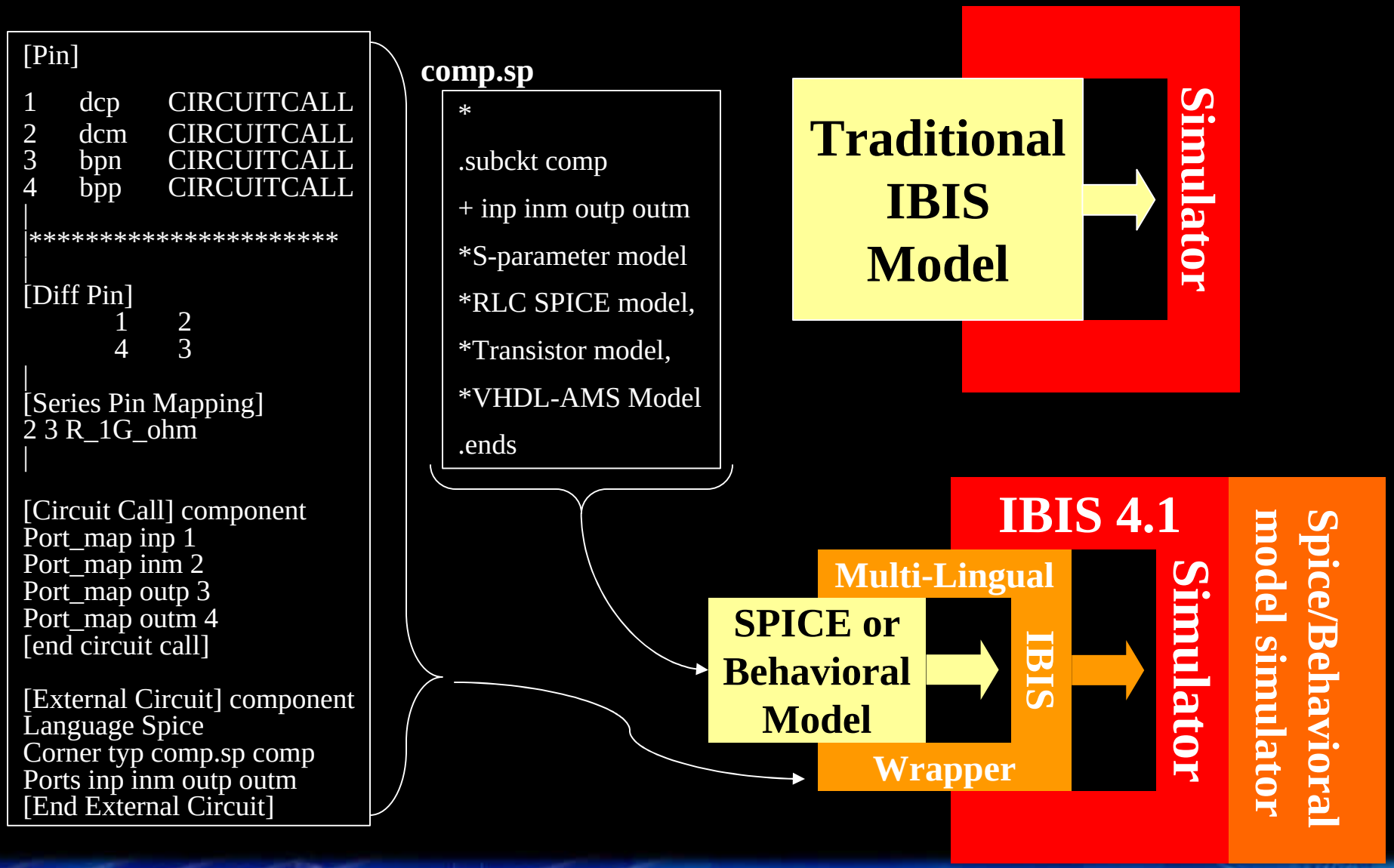
- IBIS 4.1 adds four new keywords
  - [External Model] (new part of [Model])
  - [External Circuit]
  - [Circuit Call], [Node Declarations]
- Spice/Behavioral model is in external file

Traditional IBIS

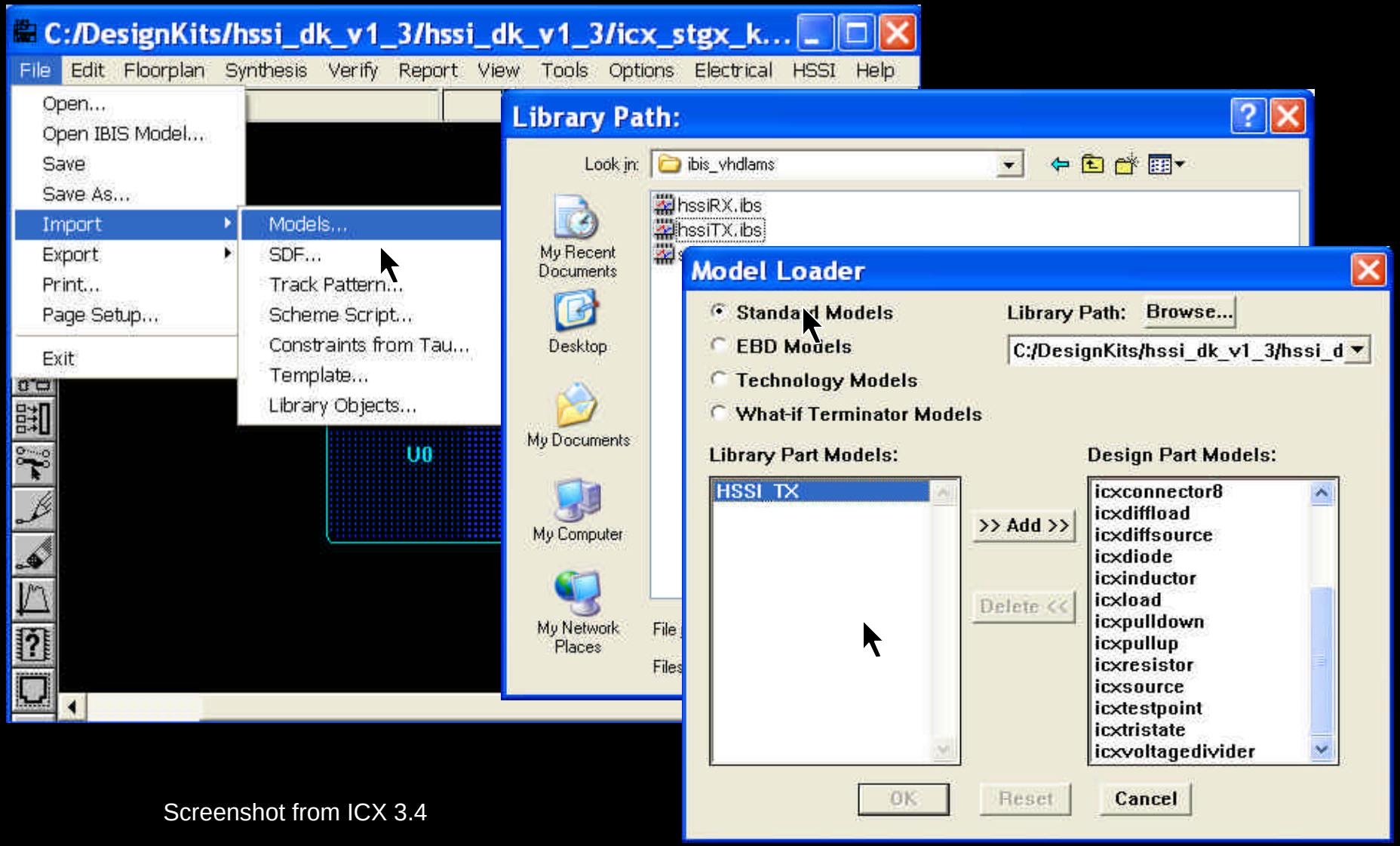
IBIS 4.1 Syntax

```
[Model] ExBufferVHDL
Model_type          Output_diff
Polarity            Non-Inverting
Vmeas = 0.75V
Rref  = 1Mohms
Vref  = 0V
|
[Voltage Range]    3.3      3.0     3.6
|
[External Model]
Language VHDL-AMS
|
| Corner corner_name file_name          circuit_name entity(architecture)
Corner     Typ          buffer_typ.vhd   buffer(buffer_io_typ)
Corner     Min          buffer_min.vhd   buffer(buffer_io_min)
Corner     Max          buffer_max.vhd   buffer(buffer_io_max)
|
| Parameters List of parameters
Parameters delay rate
Parameters preemphasis
|
| Ports List of port names (in same order as in VHDL-AMS)
Ports  D_control A_signal_neg A_signal_pos
|
[End External Model]
```

IBIS 4.1 Traditional and Multi-Lingual

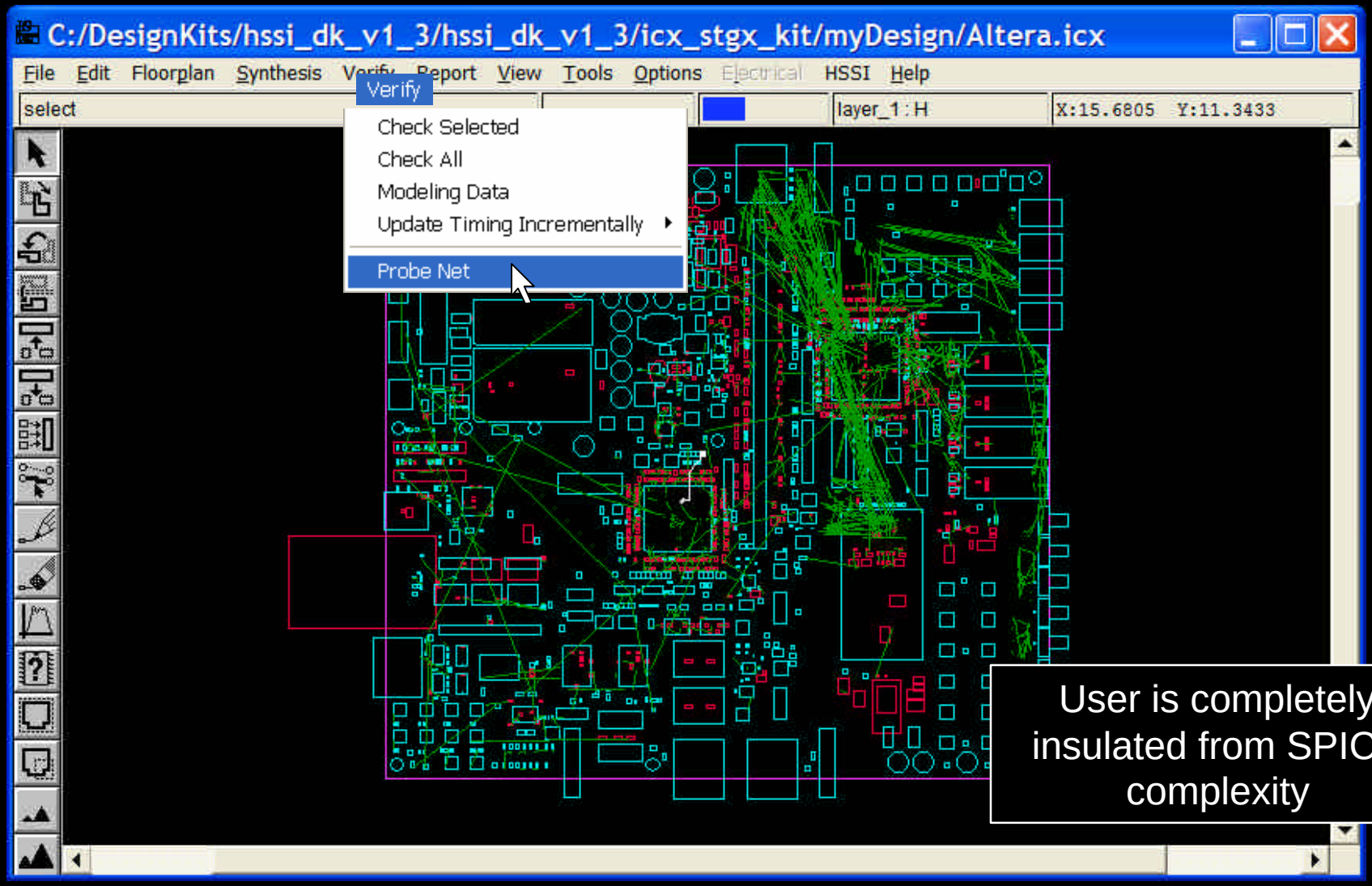


Import like any other IBIS model



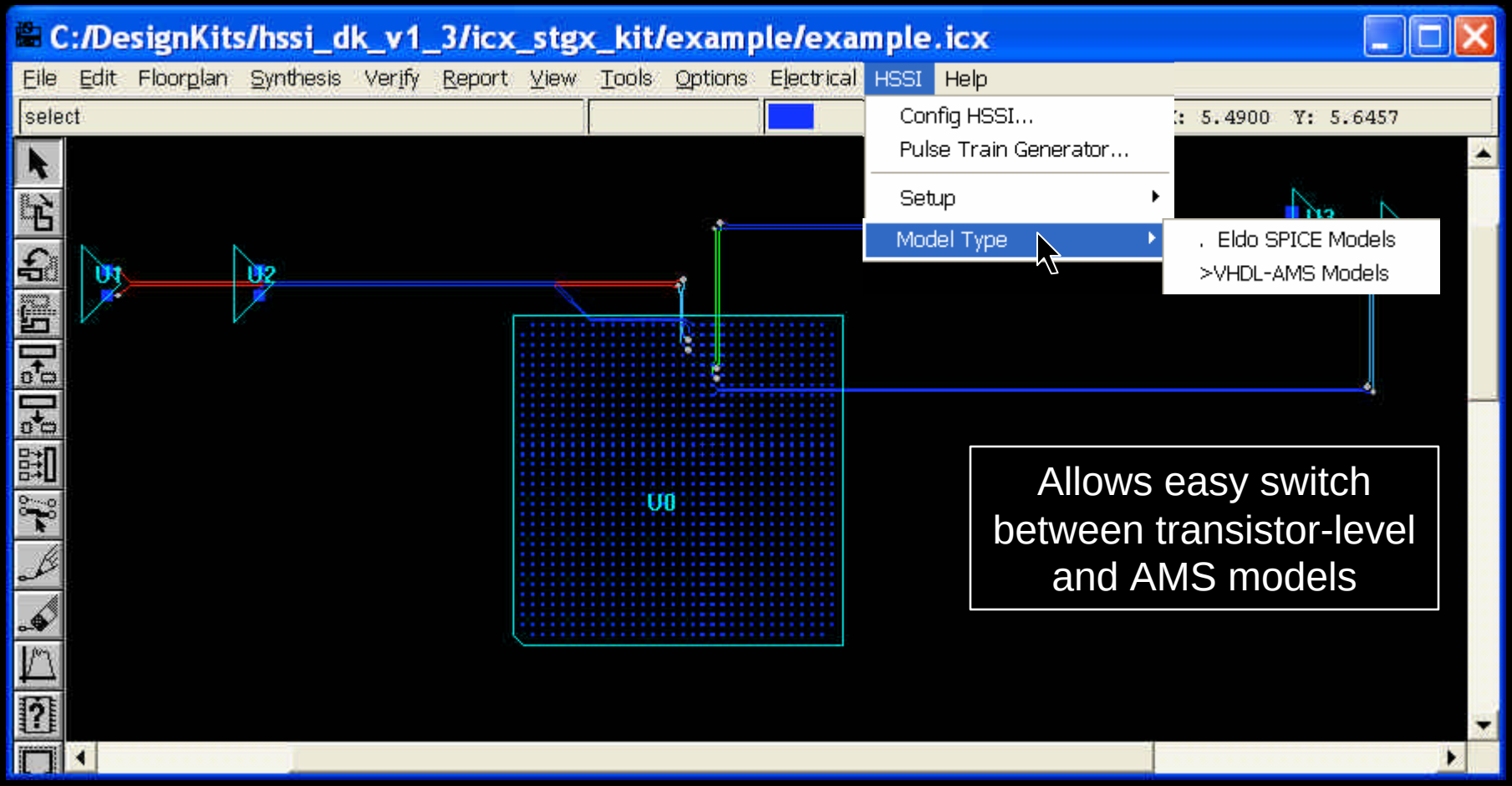
Screenshot from ICX 3.4

And PROBE, just like any IBIS model



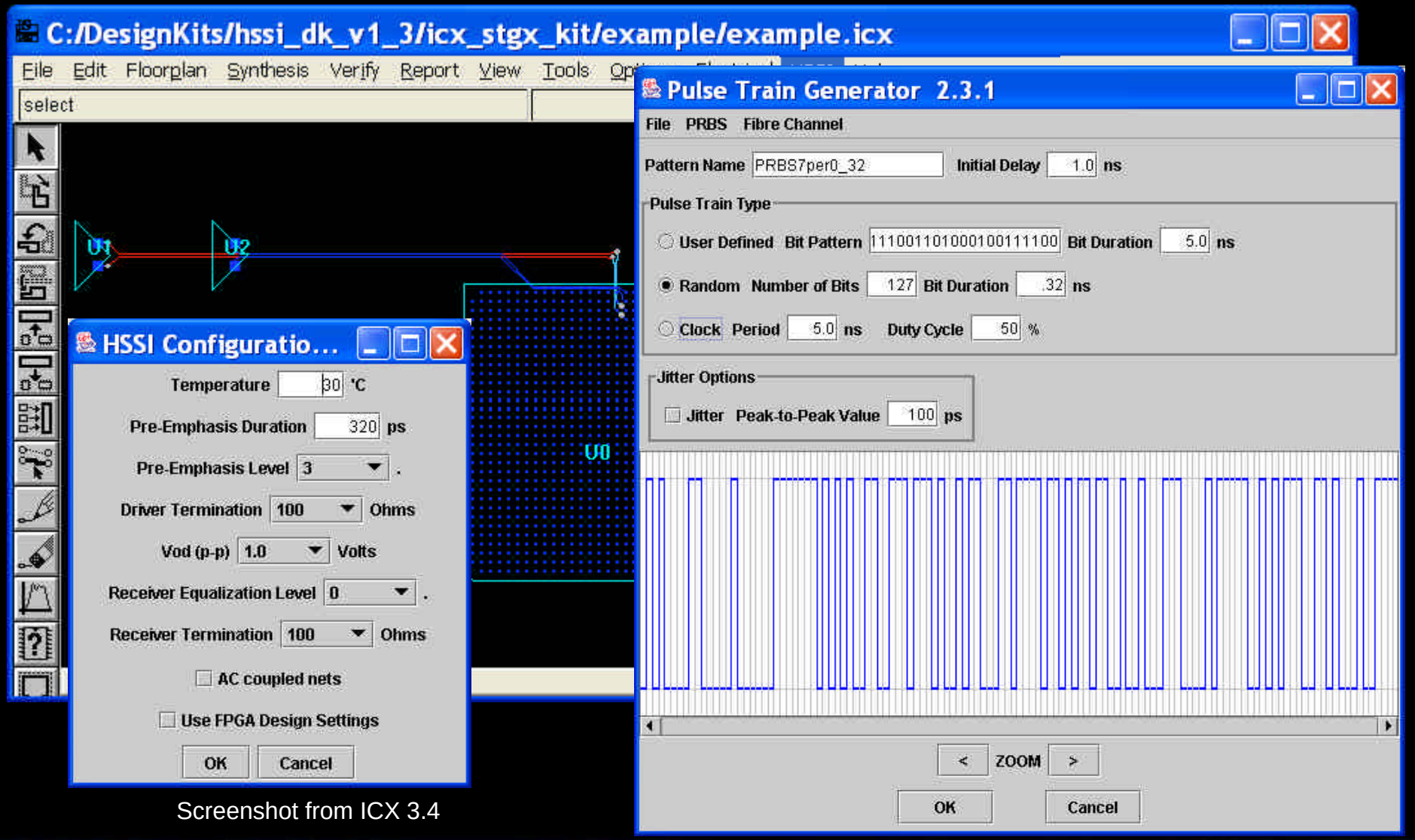
Screenshot from ICX 3.4

IBIS 4.1 Supports BOTH Transistor-level and AMS Models

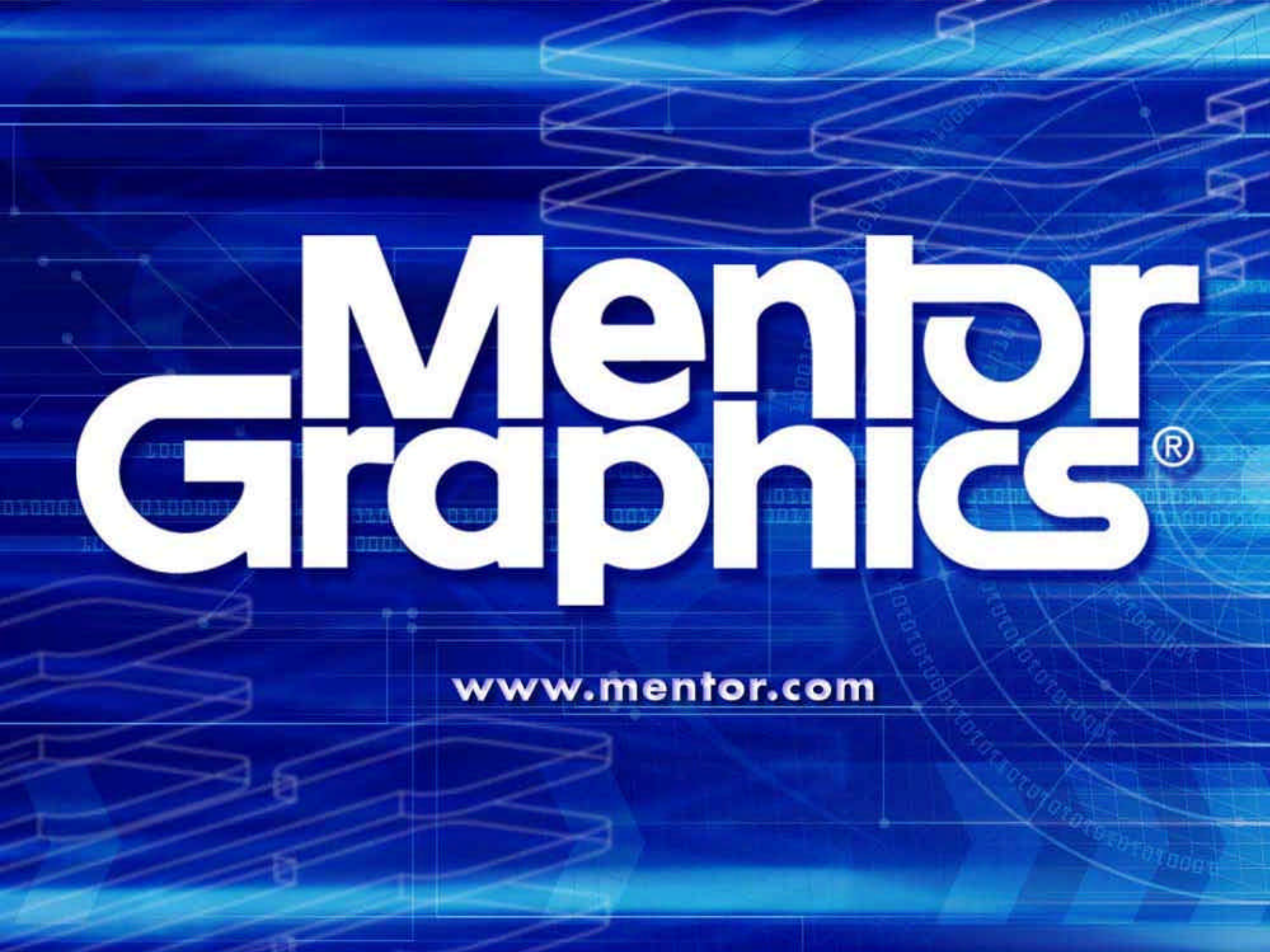


Screenshot from ICX 3.4

IBIS 4.1 “Parameters” Provide for Automation



Screenshot from ICX 3.4


Mentor Graphics®
www.mentor.com