An Alternative Signoff Approach - IBIS+AMI Models

Skipper Liang

Graser Annual User Conference 2016

2016.July

cadence®

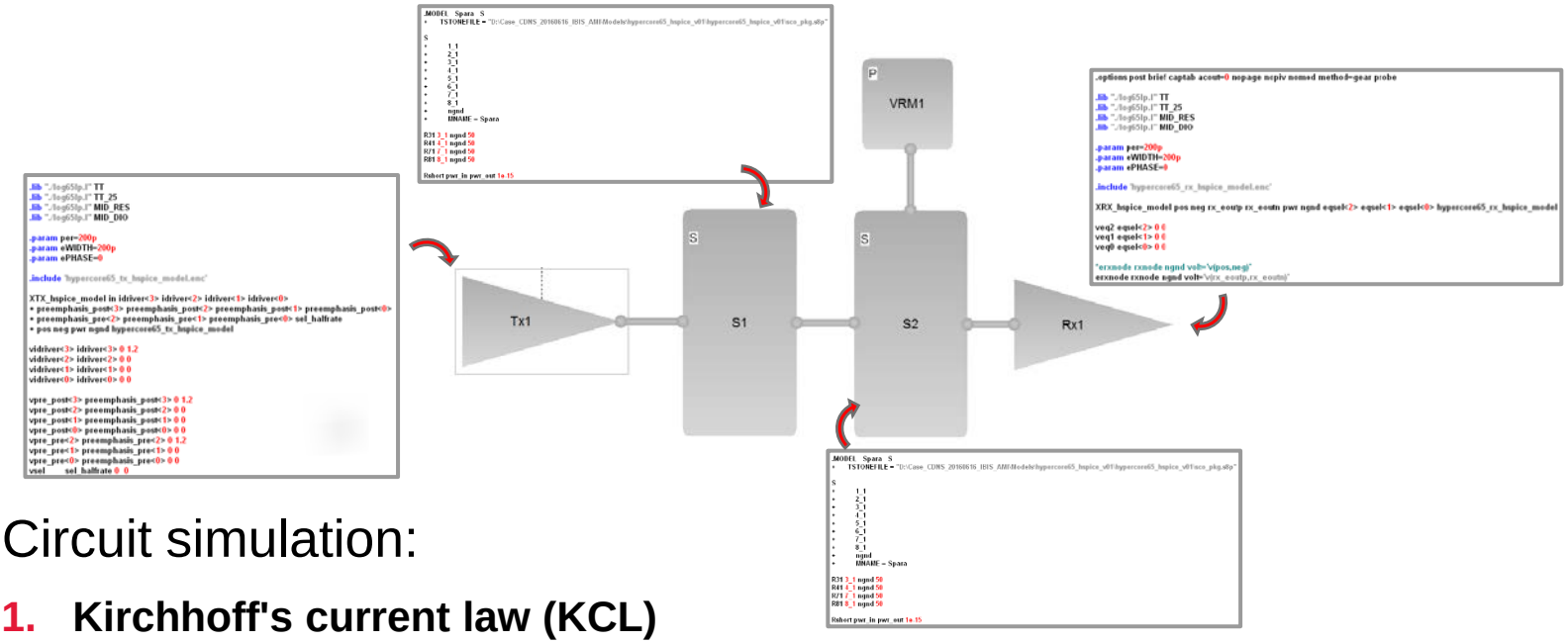
Agenda

- Traditional signoff flow – circuit simulation
- Channel simulation
  - LTI system
  - Channel simulation
- IBIS+AMI model
  - What is IBIS+AMI model
  - And your concerns?
- IBIS+AMI model signoff flow
- Why Cadence?
- A real case
- Conclusion
- More than model generation/validation/simulation…

Agenda

- Traditional signoff flow – circuit simulation
- Channel simulation
  - LTI system
  - Channel Simulation
- IBIS+AMI model
  - What is IBIS+AMI model
  - Algorithmic Simulation?
- IBIS+AMI model signoff flow
- Why Cadence?
- A real case
- Conclusion
- More than model generation/validation/simulation…

Traditional signoff flow –
Using transistor **SPICE netlist** model

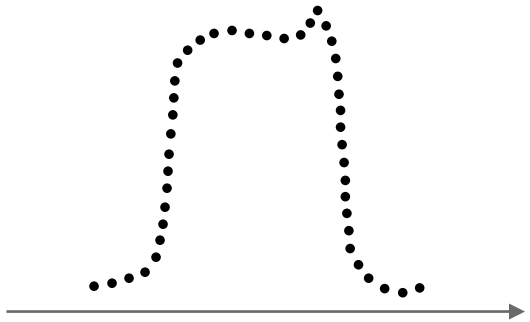


- Circuit simulation:
  1. **Kirchhoff's current law (KCL)**

     At any node (junction) in an electrical circuit, the sum of current flowing into that node is equal to the sum of currents flowing out of that node

  2. **Kirchhoff's voltage law (KVL)**

     The directed sum of the electrical potential differences (voltage) around any closed network is zero

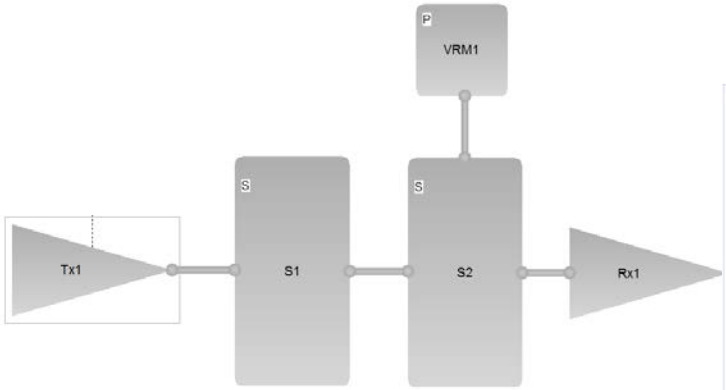


cadence

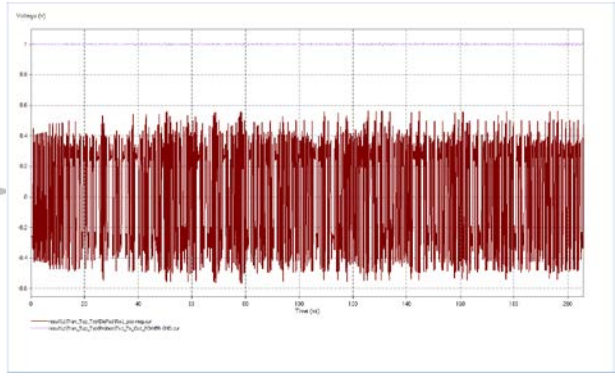
Traditional signoff flow –
Using transistor **SPICE netlist** model (con't)

Advantages:

- Accurate PI prediction under limited bits transmission
- Accurate jitter prediction under limited bits transmission

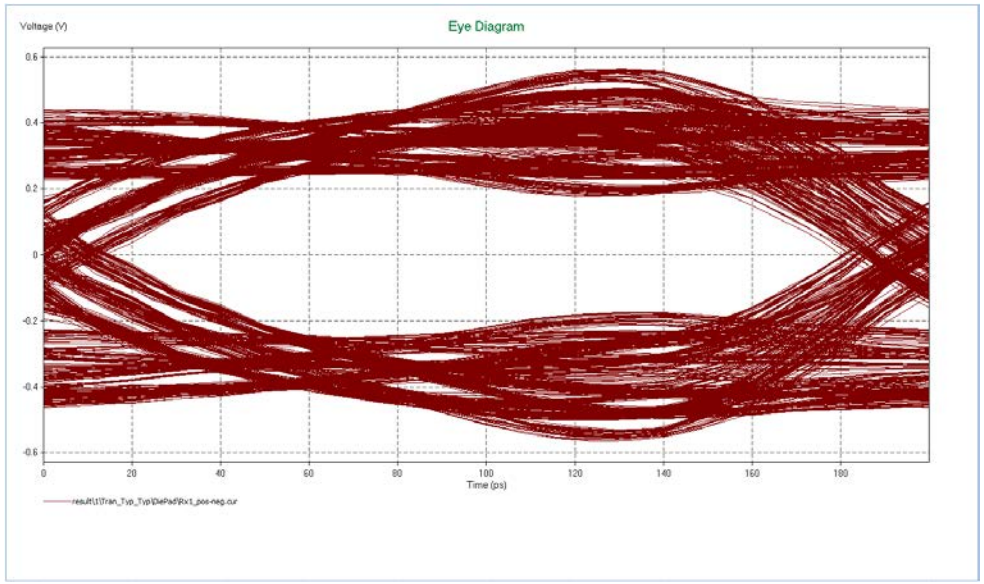


8hr34min for **1024** bit patter simulation



Disadvantages:

- Very slow for SPICE netlist model - Takes weeks/months to get bit error-rate (BER) prediction
- Can't model the adaptive mechanism in RX

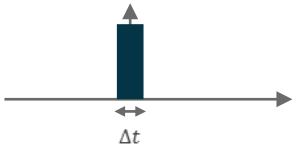


Agenda

- Traditional signoff flow – circuit simulation
- Channel simulation
  - LTI system
  - Channel simulation
- IBIS+AMI model
  - What is IBIS+AMI model
- IBIS+AMI model signoff flow
- Why Cadence?
- A real case
- Conclusion
- More than model generation/validation/simulation…

LTI – Linear time invariant (con't.)

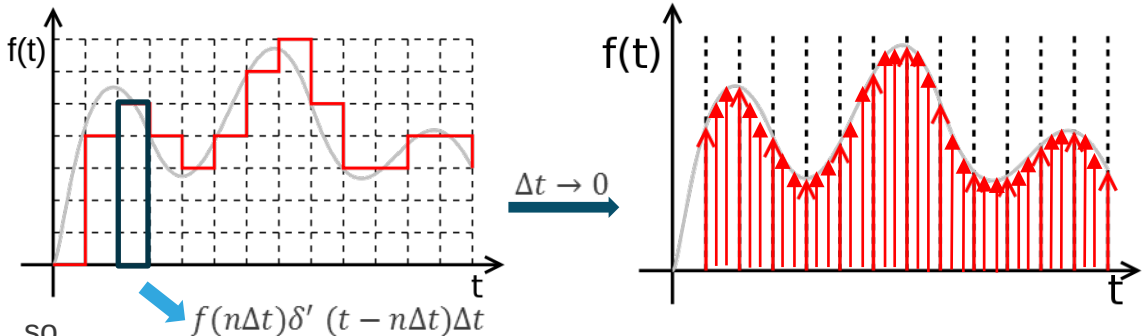
- Signal expressed in an impulse-train format:
  - Quasi-Impulse:



$$\delta'(t) = \begin{cases} 0, |t| > \dfrac{\Delta t}{2} \\ \dfrac{1}{\Delta t}, |t| \le \dfrac{\Delta t}{2} \end{cases}$$

so, $\int_{-\infty}^{\infty} \delta'(t)dt = 1$

  - Any Signal:

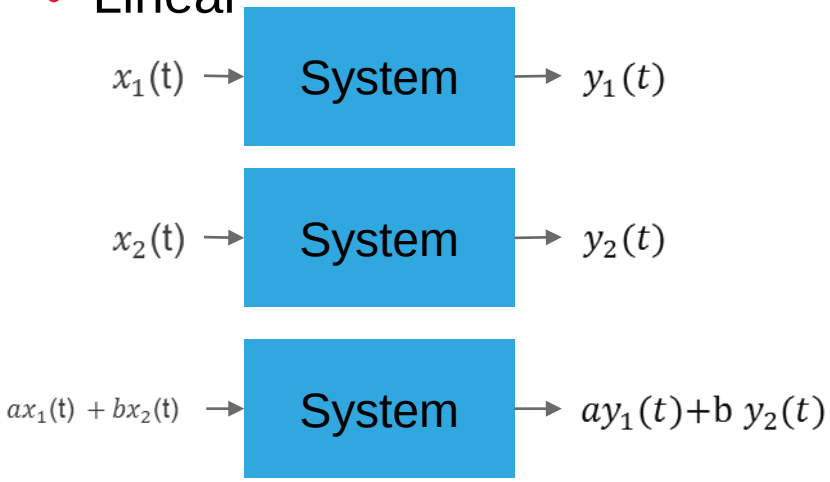


so,

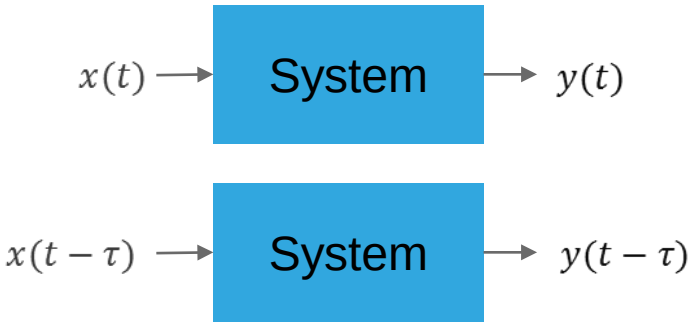
$$f(t) = \sum_{n=-\infty}^{\infty} f(n\Delta t)\delta'(t - n\Delta t)\Delta t \quad \rightarrow \quad f(t) = \int_{-\infty}^{\infty} f(\tau)\delta(t-\tau)d\tau$$

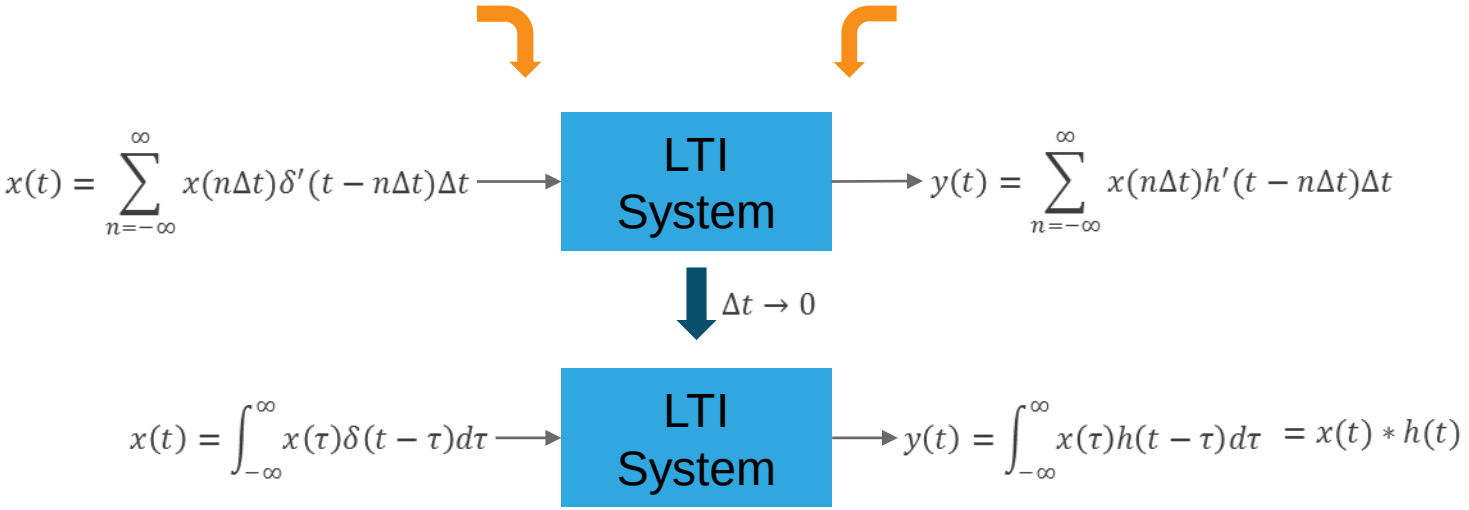
LTI – Linear time invariant (Con't.)

- Linear



- Time Invariant

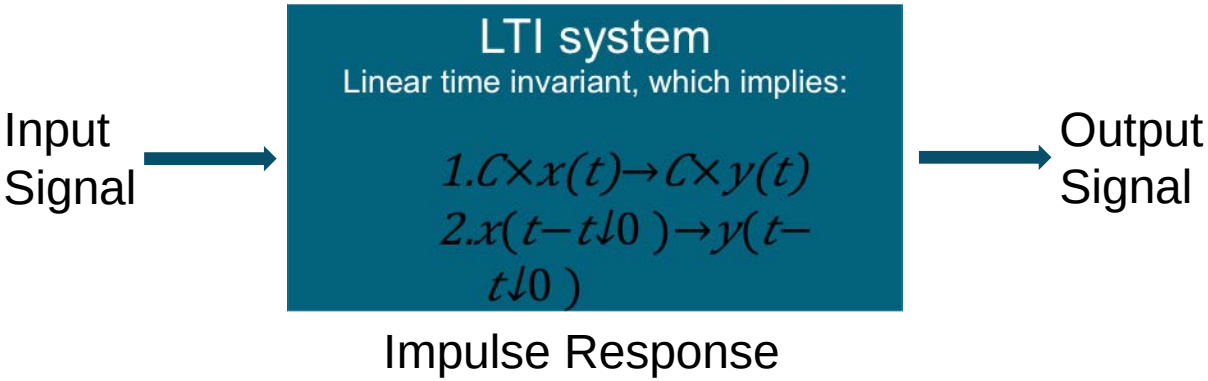




cadence

Channel-Sim

- Channel simulation :



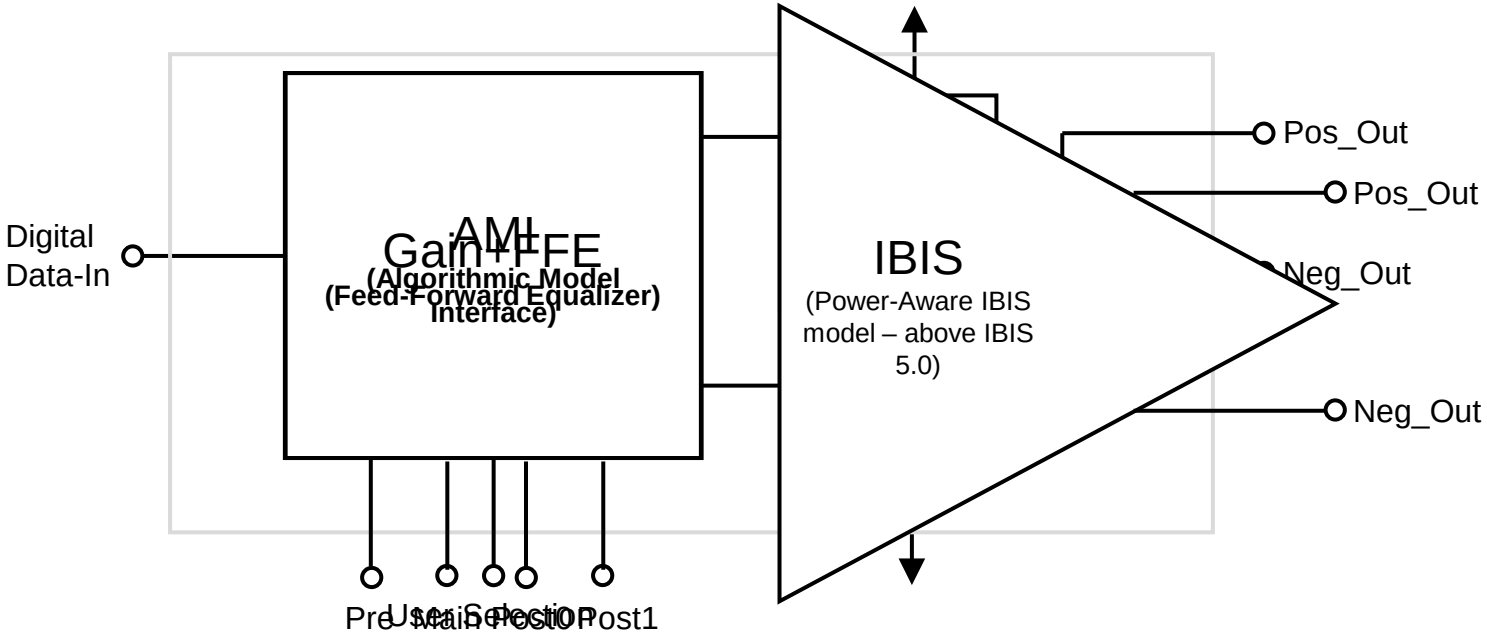
Multi-times faster than circuit simulation!!

What if the system is not an LTI one?

Agenda

- Traditional signoff flow – circuit simulation
- Channel simulation
  - LTI system
  - Channel Simulation
- IBIS+AMI model
  - What is IBIS+AMI model
  - And your concerns?
- IBIS+AMI model signoff flow
- Why Cadence?
- A real case
- Conclusion
- More than model generation/validation/simulation…

What is IBIS+AMI model (Example: TX)



Accompanied with channel simulator:

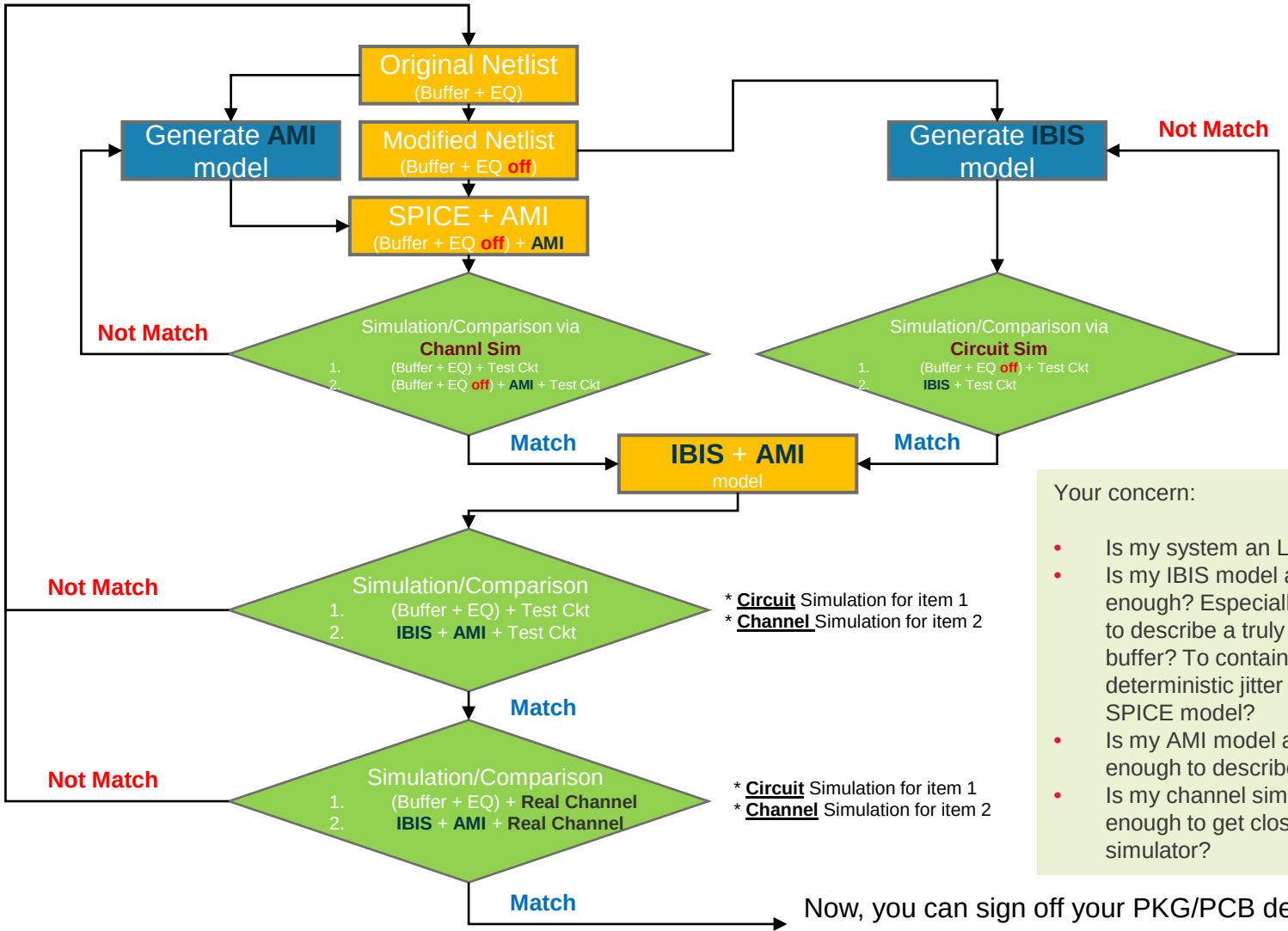
- PI prediction with good accuracy without limits on transmission bits
- Jitter prediction with good accuracy without limits on transmission bits
- Very fast for IBIS+AMI netlist model - Takes minutes to get BER prediction
- Can model the adaptive mechanism in RX

But you might be concerned:

Agenda

- Traditional signoff flow – circuit simulation
- Channel simulation
  - LTI system
  - Channel Simulation
- IBIS+AMI model
  - What is IBIS+AMI model
  - Algorithm overview
- IBIS+AMI model signoff flow
- Why Cadence?
- A real case
- Conclusion
- More than model generation/validation/simulation…

IBIS+AMI model signoff flow



Your concern:

- Is my system an LTI channel?
- Is my IBIS model accurate enough? Especially being used to describe a truly differential buffer? To contain the deterministic jitter existed in my SPICE model? V
- Is my AMI model accurate enough to describe my EQ? V
- Is my channel simulator accurate enough to get close to circuit simulator? V

Now, you can sign off your PKG/PCB design with the IBIS+AMI model so generated

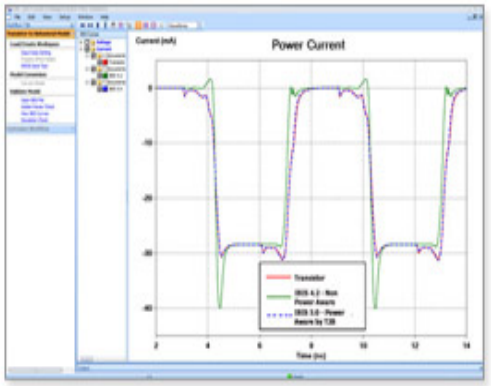
Agenda

- Traditional signoff flow – circuit simulation
- Channel simulation
  - LTI system
  - Channel Simulation
- IBIS+AMI model
  - What is IBIS+AMI model
  - Algorithmic Modeling?
- IBIS+AMI model signoff flow
- Why Cadence?
- A real case
- Conclusion
- More than model generation/validation/simulation…

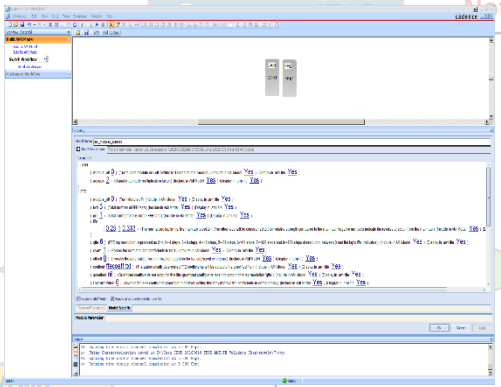
cadence®

Why Cadence?

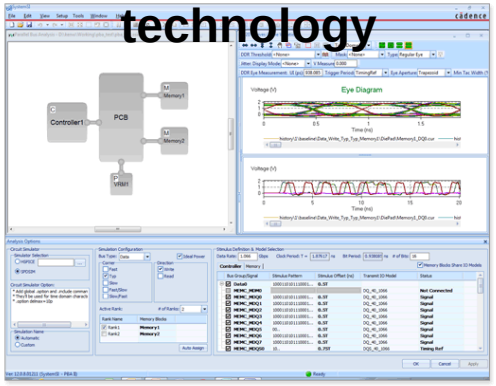
Sigrity™ T2B™



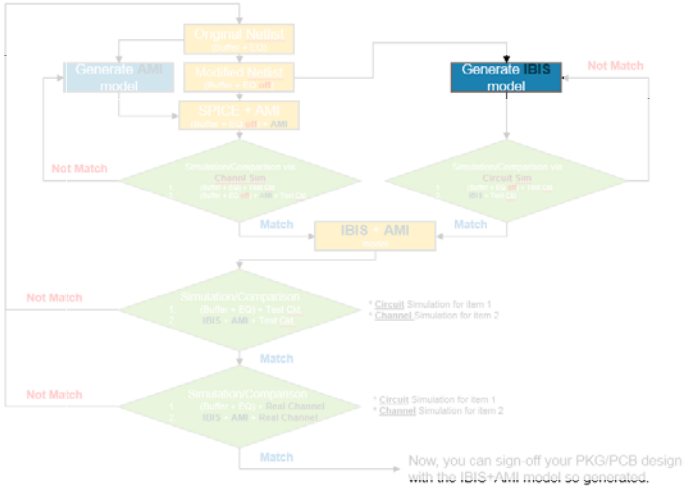
AMI Builder



Sigrity SystemSI™ technology

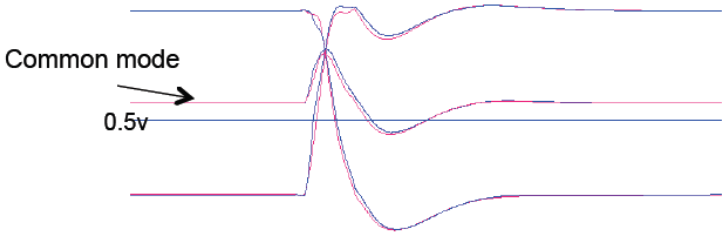


Why Cadence - IBIS generation

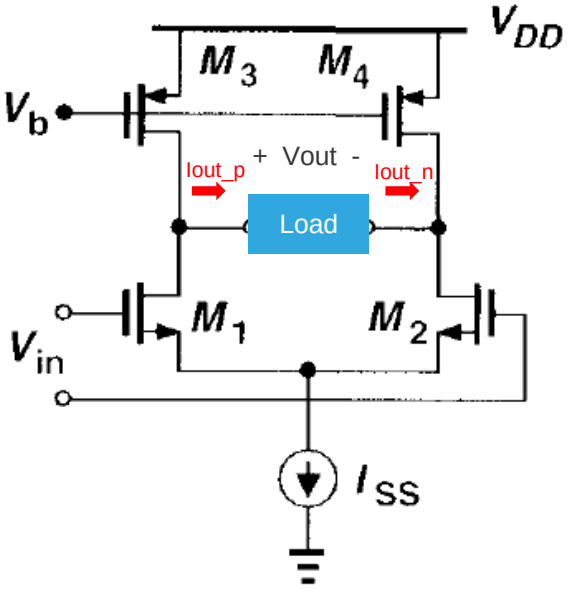


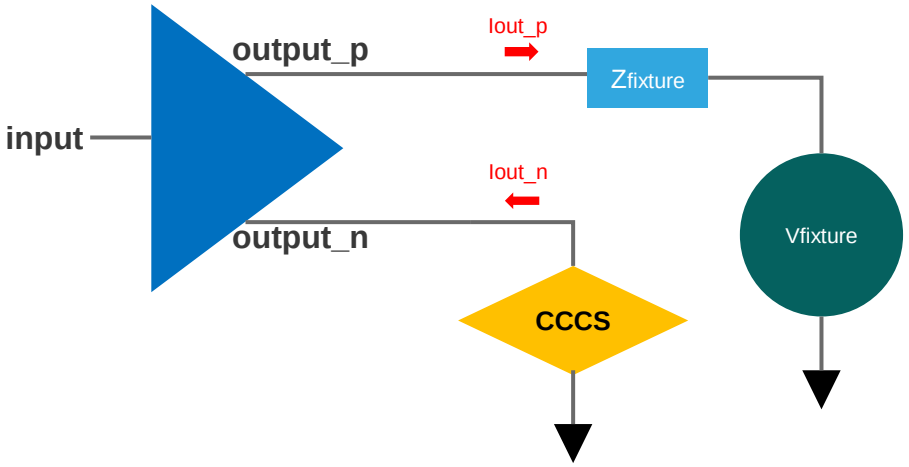
Better differential modeling for truly differential buffer

- For some differential buffer designs, the common voltage is not a fixed value



- To better model this kind of buffer, in Sigrity™ T2B™, no matter in V/T or V/I table extraction, the tool uses topologies as in the following:



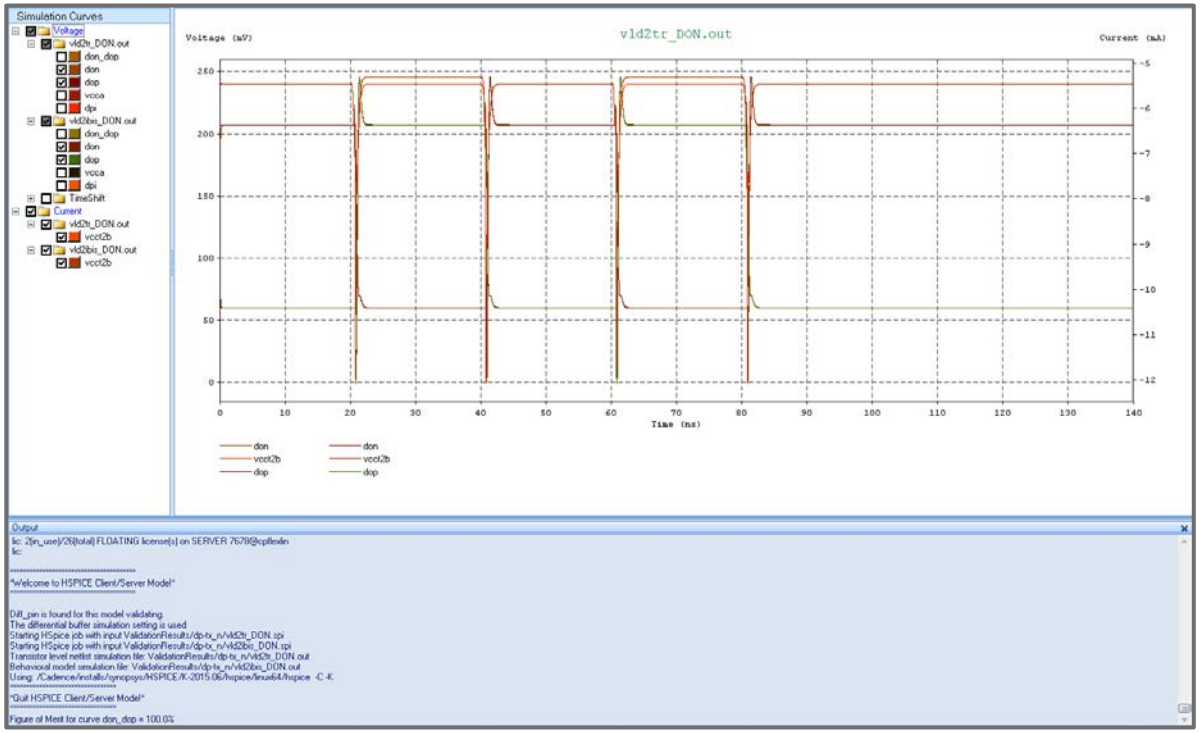


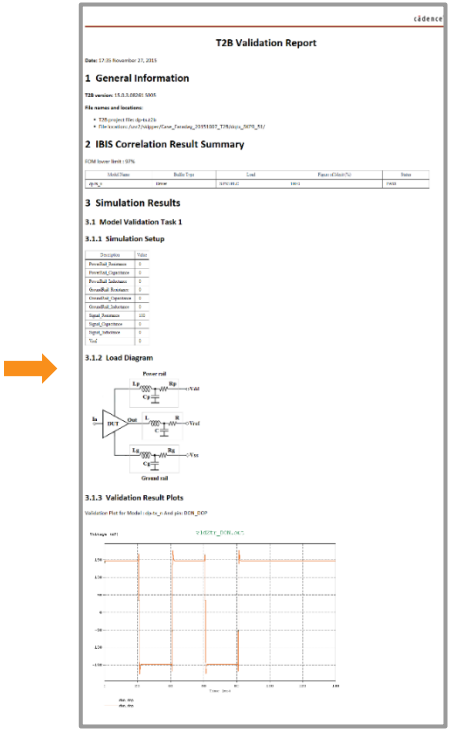
During modeling one of the positive and negative, Sigrity T2B will add a CCCS (current controlled current source) at the other side, with the value exactly the same as the current flowing out of the Vfixture but in the opposite direction, to make sure the current flowing out of the positive pin coincides with the current flowing into the negative pin

Why Cadence – IBIS generation: Sigrity T2B (con't.)

Validation and report

- Sigrity™ T2B™ provides an easy user interface for model validation and the report generation.





- A quantized mark for pass/fail criteria is introduced: FOM (Figure of Merit)

$$FOM = 100 \bullet \left[1 - \frac{\sum_{i=1}^{N} |Y_i(LAB) - Y_i(IBIS)|}{\Delta Y \bullet N} \right]$$

ΔY: (Max-Min) of HSPICE waveform

Why Cadence - IBIS generation: Sigrity T2B (con't.)

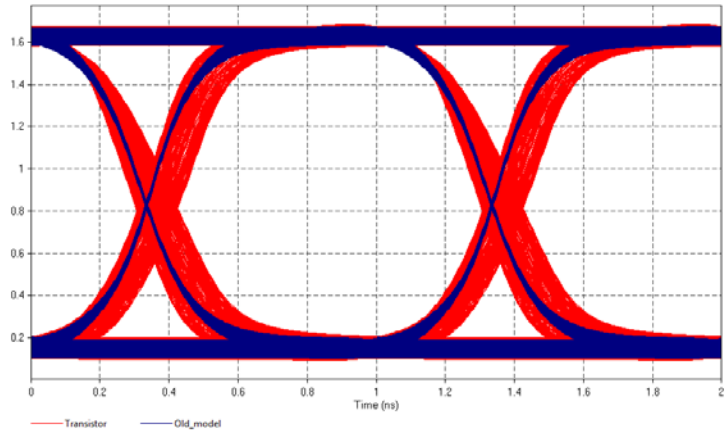
IBIS+ model

- Jitter aware and truly differential

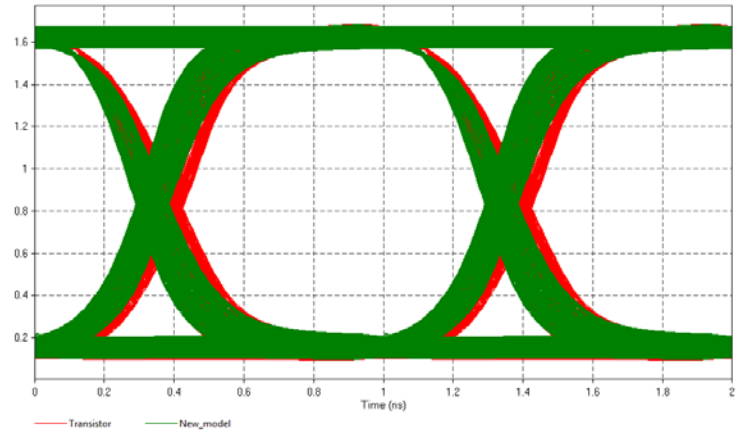
Patent Pending

Several models for different power voltages, such as, typ/min/max, will be built into the new model which can be dynamically composed according to the real power voltage during the circuit simulation

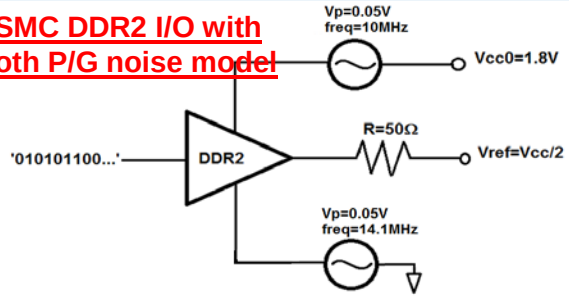
Without Power-Induced Jitter Model



With Power-Induced Jitter Model

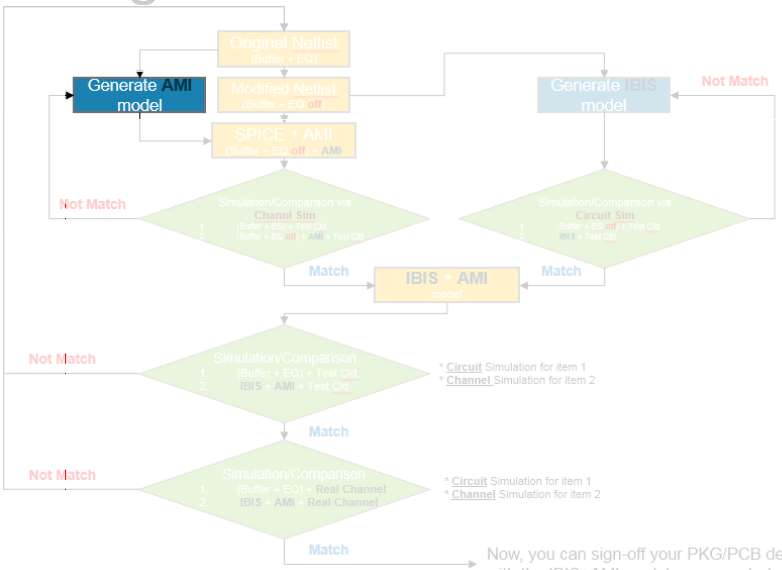


TSMC DDR2 I/O with both P/G noise model



- Transistor-level model (golden reference)
- No-jitter model
- Power-induced jitter model

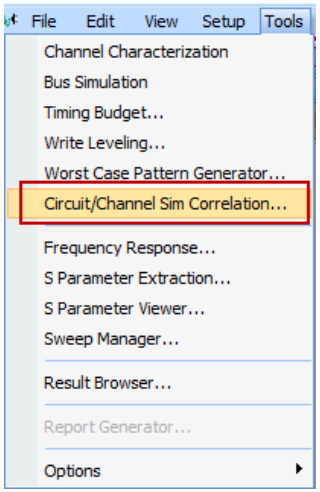
Why Cadence - AMI generation: advanced AMI Builder

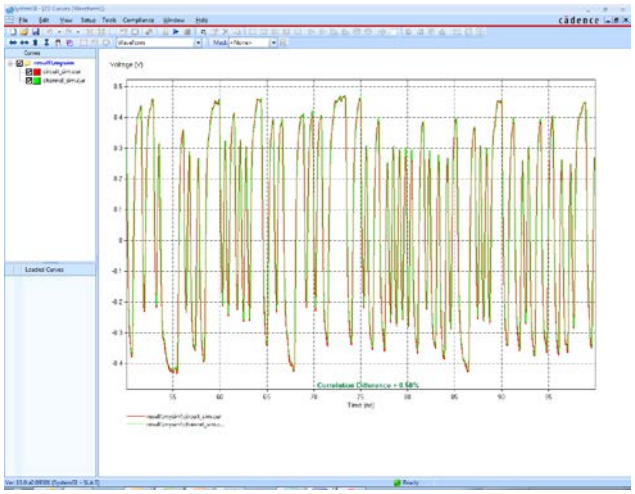


- An abundant and rich library for you to describe your EQ design
- A friendly user interface to guide you through the model generation process
- Fast and accurate

Why Cadence – simulation: Sigrity SystemSI

- One click to know if your system/channel to be analyzed can be treated as LTI or not:





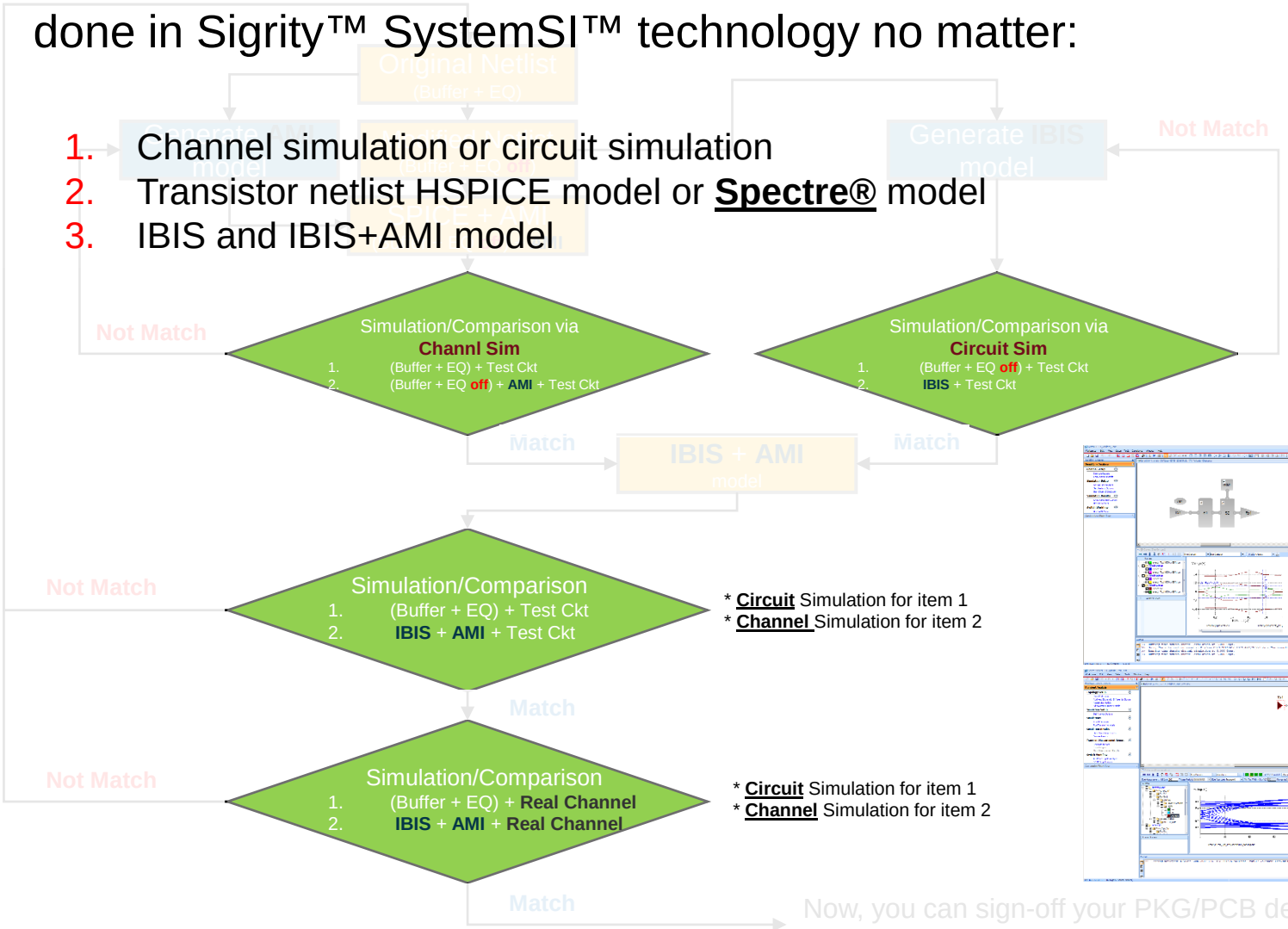
Your concern:

- Is my system an LTI channel? V
- Is my IBIS model accurate enough? Especially being used to describe a truly differential buffer? To contain the deterministic jitter existed in my SPICE model? V
- Is my AMI model accurate enough to describe my EQ? V
- Is my channel simulator accurate enough to get close to circuit simulator? V

Why Cadence – simulation: Sigrity SystemSI (con't.)

- All these four blocks of simulation for model's validation can be done in Sigrity™ SystemSI™ technology no matter:

1. Channel simulation or circuit simulation
2. Transistor netlist HSPICE model or **Spectre®** model
3. IBIS and IBIS+AMI model



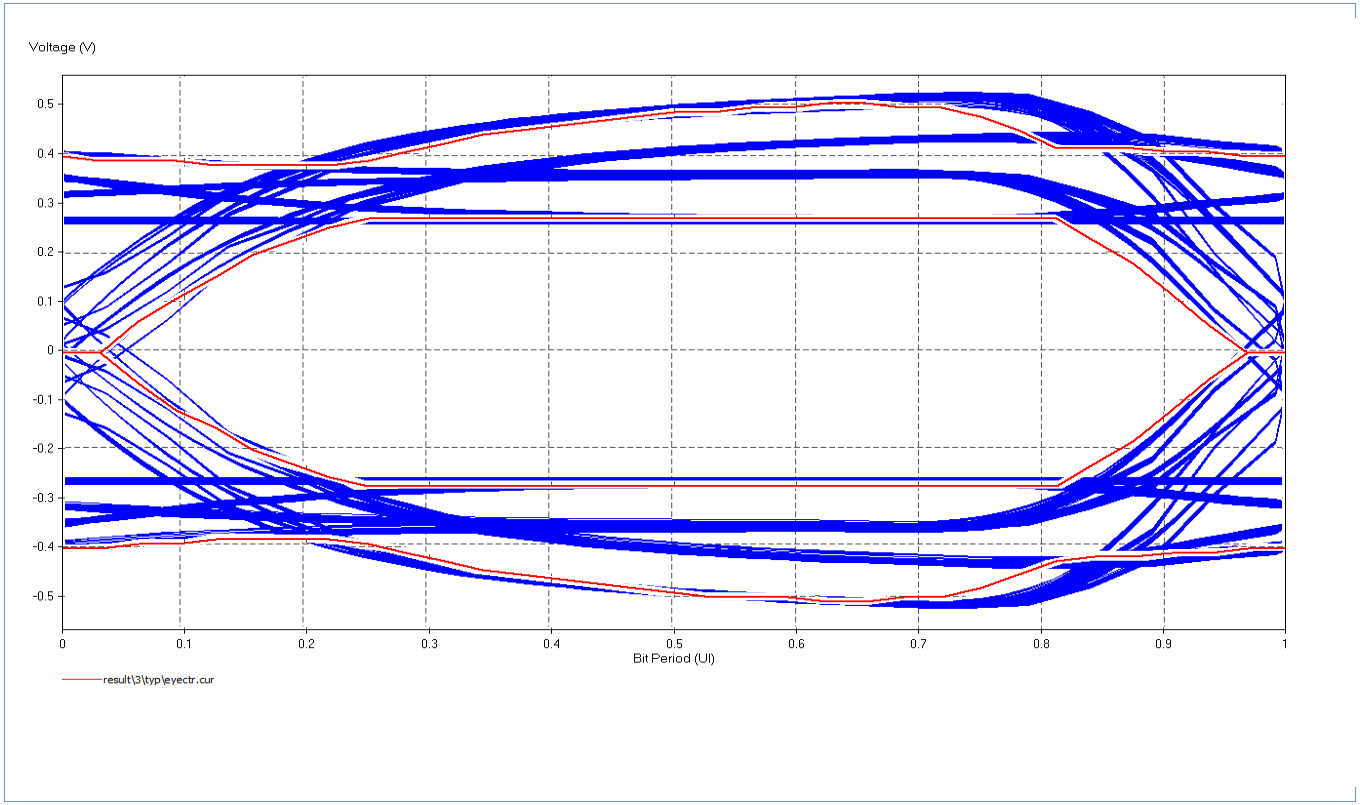
* **Circuit** Simulation for item 1
* **Channel** Simulation for item 2

* **Circuit** Simulation for item 1
* **Channel** Simulation for item 2

Why Cadence – simulation: Sigrity SystemSI (con't.)

- All these four blocks of simulation for model's validation can be done in Sigrity™ SystemSI™ technology no matter:

1. Channel simulation or circuit simulation
2. Transistor netlist **HSPICE** model or **Spectre®** model
3. IBIS and IBIS+AMI model



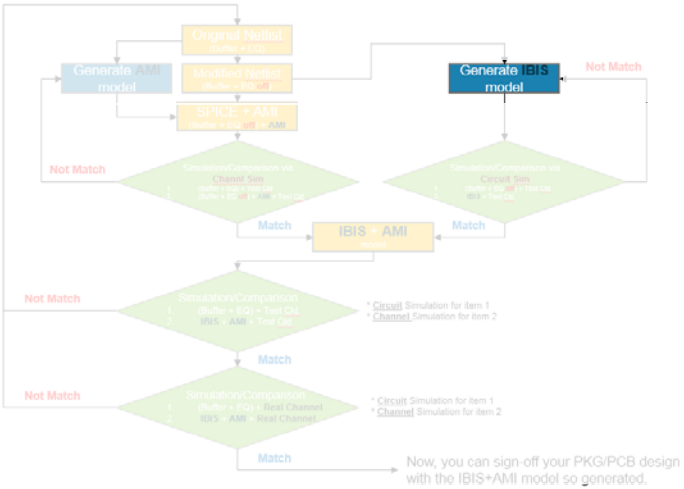
cadence®

Why Cadence ?

- Now, please help yourself by checking:

  <u>For IBIS model:</u>

1. Does your IBIS generation tool describe a truly differential buffer well enough?
2. Are you forced to give up the V/T information in your IBIS during channel simulation due to channel simulation engine or extraction methodology?
3. Or, can you just use a “Dummy” IBIS model (generated by AMI model generation tool) which behaves in just a different way from your own design?

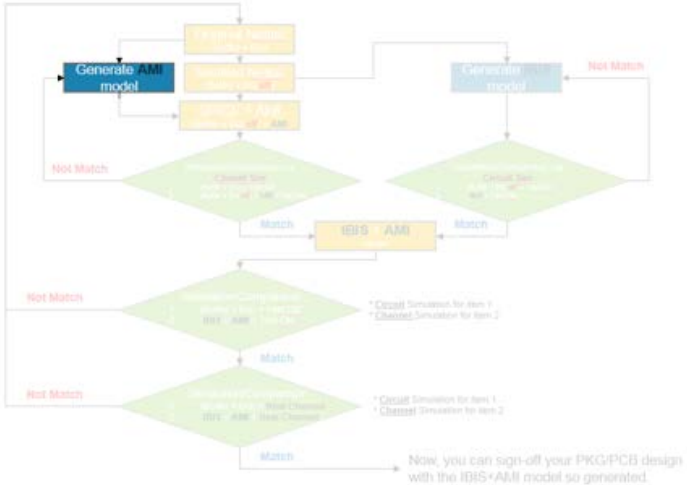


Why Cadence? (con't.)

- Now, please help yourself to check:

  For AMI model:

  1. Does your AMI generation tool provide you a convenient way to validate your AMI model by:
     - Channel simulation over your netlist of Buffer+EQ_**on**
     - Channel simulation over your netlist of Buffer+EQ_**off** combined with **AMI model** so generated?

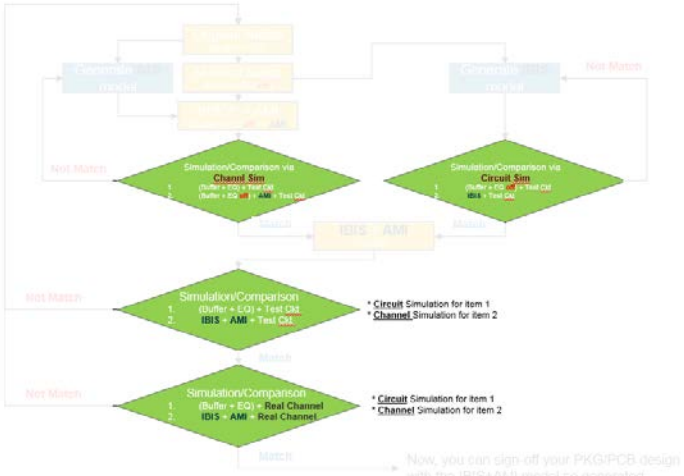


Why Cadence ? (con't.)

- Now, please help yourself to check:

  For simulation:

1. Does your channel simulator tell you how LTI your system is with only one-button click?
2. Does your channel/circuit simulator work with **Spectre®** netlist?
3. Does your channel simulator give results close enough to the result generated by circuit simulator? Is it easy to validate this item in your simulator?
4. Does your channel simulator provide power-aware result?

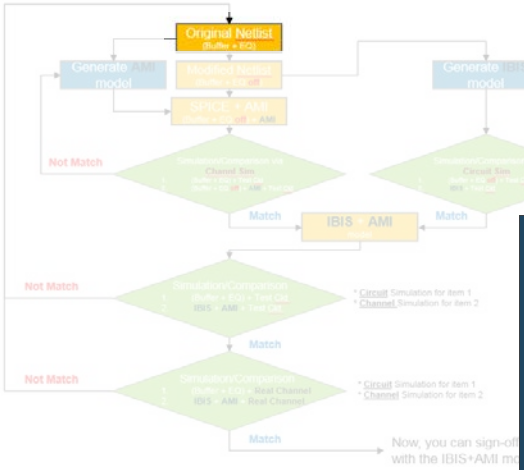


Agenda

- Traditional signoff flow – circuit simulation
- Channel simulation
  - LTI system
  - Channel Simulation
- IBIS+AMI model
  - What is IBIS+AMI model
  - Algorithmic modeling?
- IBIS+AMI model signoff flow
- Why Cadence?
- A real case
- Conclusion
- More than model generation/validation/simulation…

A real case

- The original SPICE netlist:

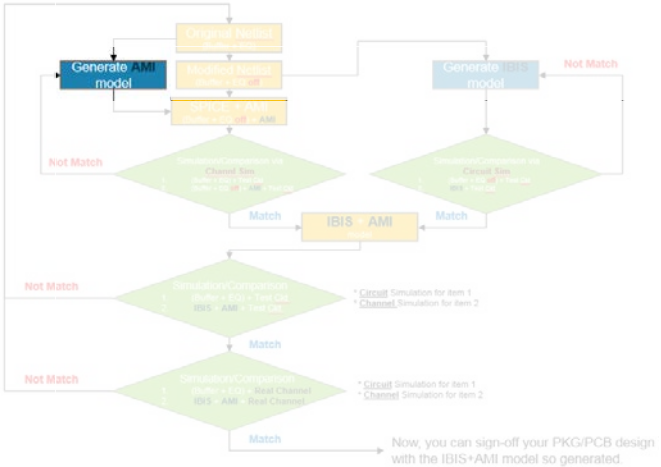



```
* Simulation testbench for Hypercore65 Transceiver model
* Generated on Jan. 14, 2007

.options post brief captab acout=0 nopage nopiv nomod method=gear probe

.lib "./log65lp.l" TT
.lib "./log65lp.l" TT_25
.lib "./log65lp.l" MID_RES
.lib "./log65lp.l" MID_DIO

.param per=200p
.param eWIDTH=200p
.param ePHASE=0

.include 'hypercore65_rx_hspice_model.enc'
.include 'hypercore65_tx_hspice_model.enc'
*.inc 'ttm_table.sp'
.include 'prbs_100ns.inc'

XTX_hspice_model data idriver<3> idriver<2> idriver<1> idriver<0>
+ preemphasis_post<3> preemphasis_post<2> preemphasis_post<1> preemphasis_post<0>
+ preemphasis_pre<2> preemphasis_pre<1> preemphasis_pre<0> sel_halfrate
+ tpoutp tpoutn txvdd txgnd hypercore65_tx_hspice_model
XRX_hspice_model rpinp rpinn rx_eoutp rx_eoutn rxvdd rxgnd eqsel<2> eqsel<1> eqsel<0> hypercore65_rx_hspice_model

vtxvdd txvdd txgnd 1.2v
vtxgnd txgnd 0 0
vrxvdd rxvdd rxgnd 1.2v
vrxgnd rxgnd 0 0
vgnd gnd 0 0

vidriver<3> idriver<3> 0 1.2
vidriver<2> idriver<2> 0 0
vidriver<1> idriver<1> 0 0
vidriver<0> idriver<0> 0 0

vpre_post<3> preemphasis_post<3> 0 0
vpre_post<2> preemphasis_post<2> 0 0
vpre_post<1> preemphasis_post<1> 0 0
vpre_post<0> preemphasis_post<0> 0 0
vpre_pre<2> preemphasis_pre<2> 0 0
vpre_pre<1> preemphasis_pre<1> 0 0
vpre_pre<0> preemphasis_pre<0> 0 0
vsel        sel_halfrate 0  0

veq2 eqsel<2> 0 1.2
```

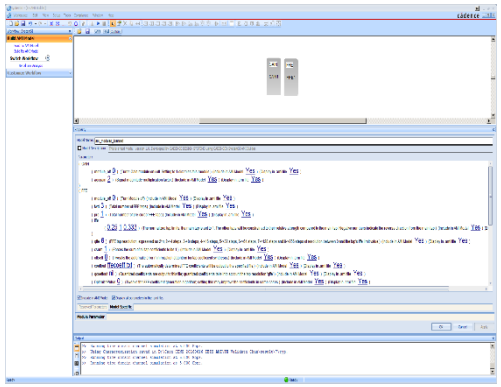

cadence

A real case

- AMI model generation:



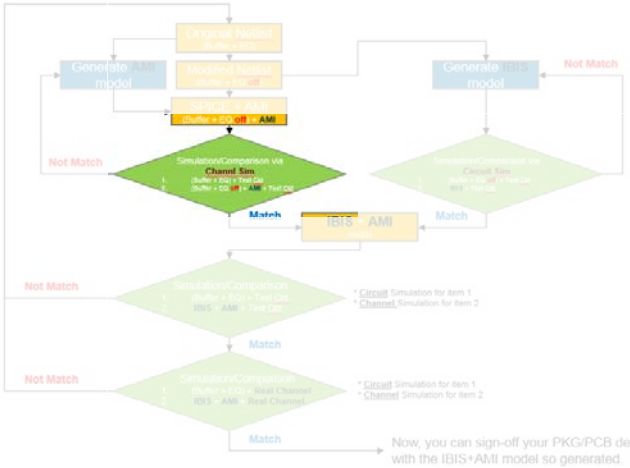
AMI Builder

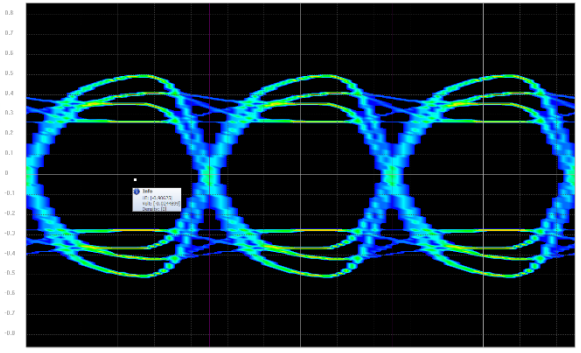


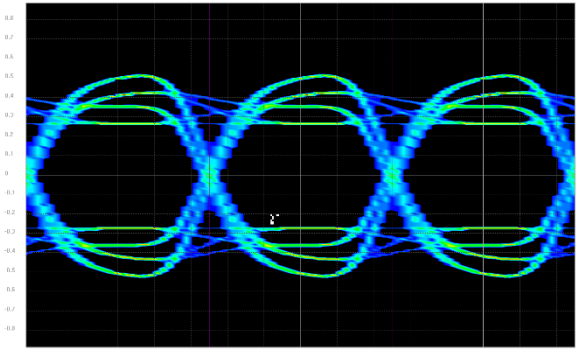
```
  (Usage Info )
  (Type Boolean )
  (Default True )
  (Description "Getwave present in this model." )
 )

)
(Model_Specific(FFE

 (module_off
  (Usage In )
  (Type Integer )
  (List 0 1 )
  (Default 0 )
  (Description "Turn Module off" )
 )
 (Number_of_FFE_Taps
  (Usage In )
  (Type Integer )
  (Range 3 1 10 )
  (Default 3 )
  (Description "The total number of FFE taps" )
 )
 (Number_of_FFE_Pre_Taps
  (Usage In )
  (Type Integer )
  (Range 1 1 10 )
  (Default 1 )
  (Description "The total number of pre-cursor FFE taps" )
 )
 (coeffout
  (Usage In )
  (Type String )
  (Value "ffecoeffout.txt" )
  (Description "The automatically determined FFE coefficients will be output to the specified file." )
 )
(preset_taps (def
(P0 (Usage In) (Type String) (Value "0, 1, 0"))
(P1 (Usage In) (Type String) (Value "-0.06348943, 0.541411754, -0.111743042"))
)
(Default_Preset_Value (Usage In) (Type String) (List "P0" "P1" "nil") (Default "P0") (Description "nil"))
 ) )

)
)
```

A real case

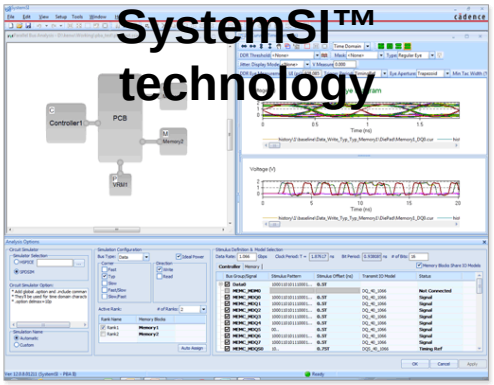
- AMI model validation:

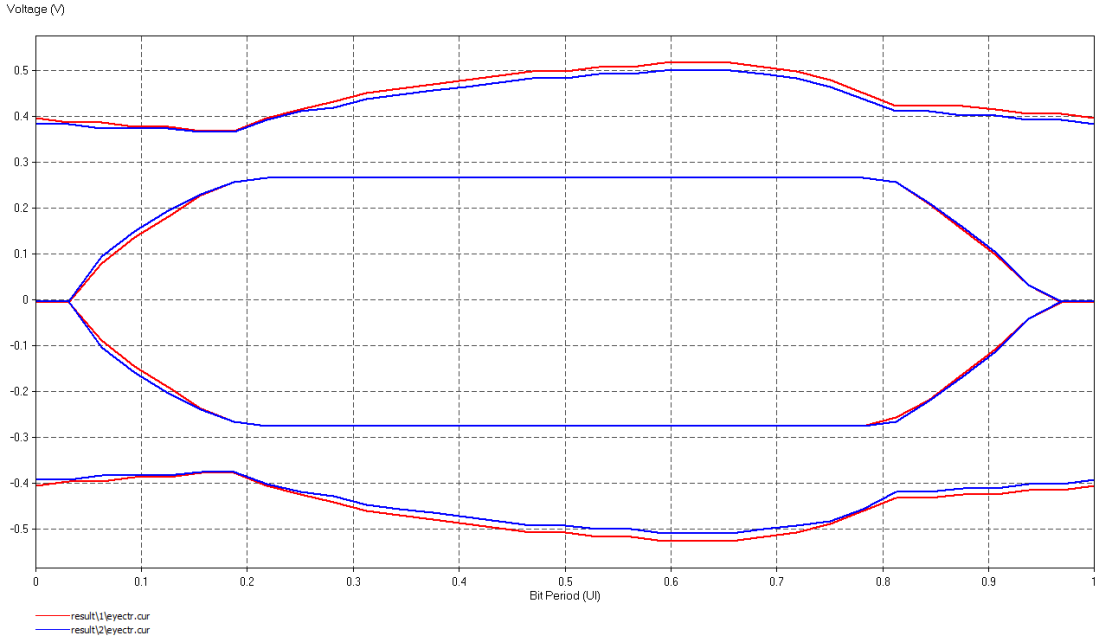






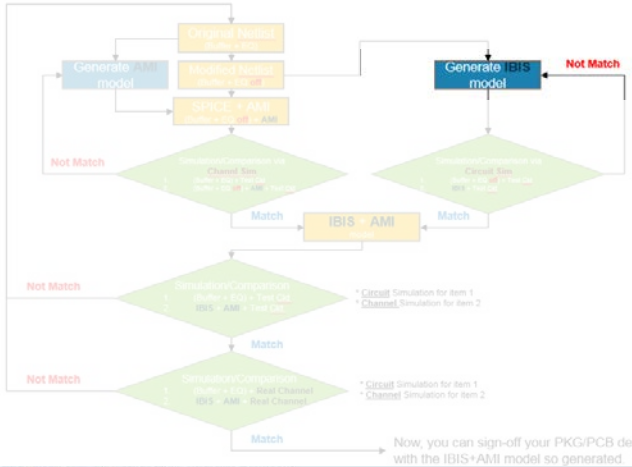
Sigrity™ SystemSI™ technology





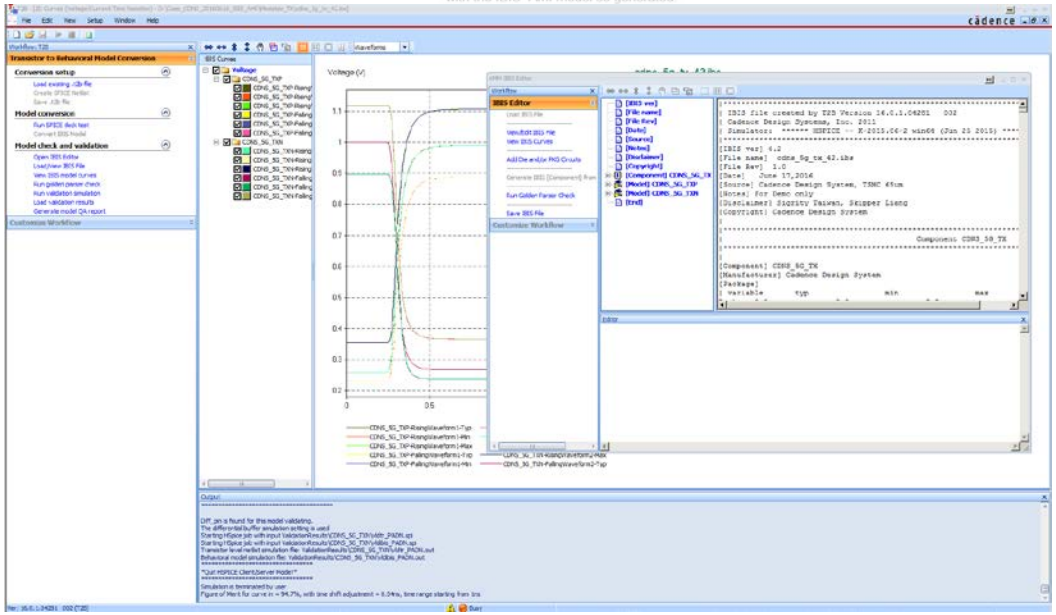
A real case

- IBIS model generation:

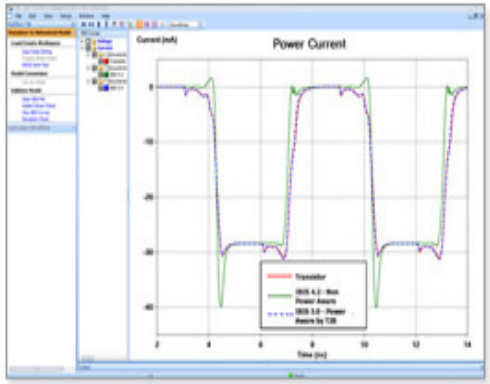



```
* Simulation testbench for Hypercore65 Transceiver model
* Generated on Jan. 14, 2007

.options post brief captab acout=0 nopage nopiv nomod method=gear probe

.lib "./log651p.l" TT
.lib "./log651p.l" TT_25
.lib "./log651p.l" MID_RES
.lib "./log651p.l" MID_DIO

.param per=200p
.param eWIDTH=200p
.param ePHASE=0

.include 'hypercore65_rx_hspice_model.enc'
.include 'hypercore65_tx_hspice_model.enc'
*.inc 'ttm_table.sp'
.include 'prbs_100ms.inc'

XTX_hspice_model data idriver<3> idriver<2> idriver<1> idriver<0>
+ preemphasis_post<3> preemphasis_post<2> preemphasis_post<1> preemphasis_post<0>
+ preemphasis_pre<2> preemphasis_pre<1> preemphasis_pre<0> sel_halfrate
+ tpoutp tpoutn txvdd txgnd hypercore65_tx_hspice_model
XRX_hspice_model rpinp rpinn rx_eoutp rx_eoutn rxvdd rxgnd eqsel<2> eqsel<1> eqsel<0> hypercore65_rx_hspice_model

vtxvdd txvdd txgnd 1.2v
vtxgnd txgnd 0 0
vrxvdd rxvdd rxgnd 1.2v
vrxgnd rxgnd 0 0
vgnd gnd 0 0

vidriver<3> idriver<3> 0 1.2
vidriver<2> idriver<2> 0 1.2
vidriver<1> idriver<1> 0 1.2
vidriver<0> idriver<0> 0 1.2

vpre_post<3> preemphasis_post<3> 0 0
vpre_post<2> preemphasis_post<2> 0 0
vpre_post<1> preemphasis_post<1> 0 0
vpre_post<0> preemphasis_post<0> 0 0
vpre_pre<2> preemphasis_pre<2> 0 0
vpre_pre<1> preemphasis_pre<1> 0 0
vpre_pre<0> preemphasis_pre<0> 0 0
vsel        sel_halfrate 0  0

veq2 eqsel<2> 0 1.2
```

Turn off EQ



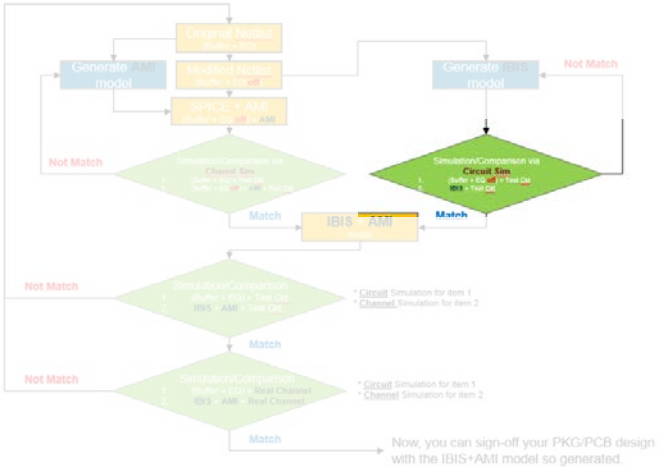
Sigrity™ T2B™



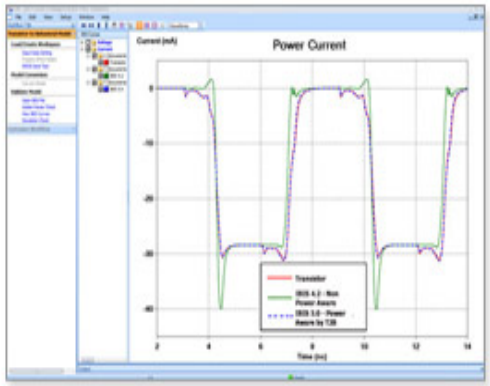
cadence

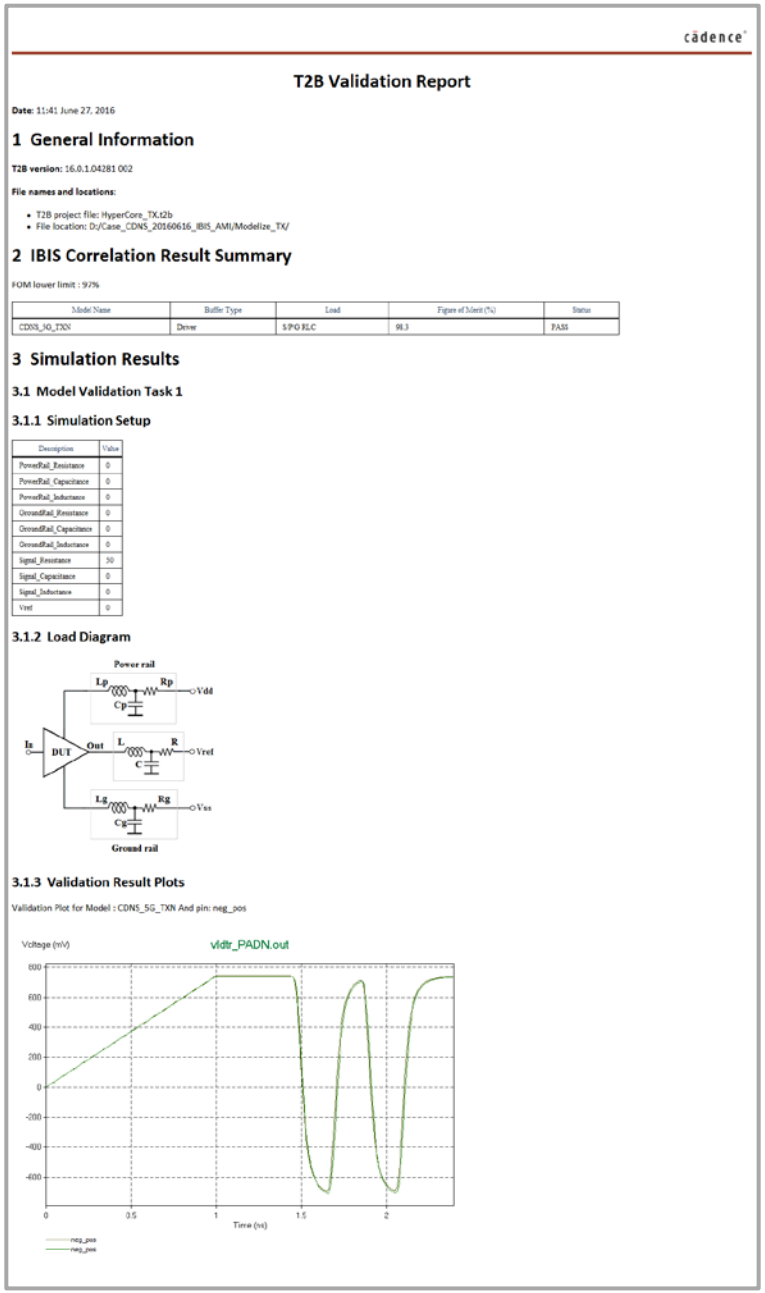
A real case

- IBIS model validation



Sigrity™ T2B™





cadence

T2B Validation Report

Date: 11:41 June 27, 2016

1 General Information

T2B version: 16.0.1.04281 002

File names and locations:

- T2B project file: HyperCore_TX.t2b
- File location: D:/Case_CDNS_20160616_IBIS_AMI/Modelize_TX/

2 IBIS Correlation Result Summary

FOM lower limit : 97%

Model Name	Buffer Type	Load	Figure of Merit (%)	Status
CDNS_5G_TXN	Driver	S/P/G RLC	98.3	PASS

3 Simulation Results

3.1 Model Validation Task 1

3.1.1 Simulation Setup

Description	Value
PowerRail_Resistance	0
PowerRail_Capacitance	0
PowerRail_Inductance	0
GroundRail_Resistance	0
GroundRail_Capacitance	0
GroundRail_Inductance	0
Signal_Resistance	50
Signal_Capacitance	0
Signal_Inductance	0
Vref	0

3.1.2 Load Diagram

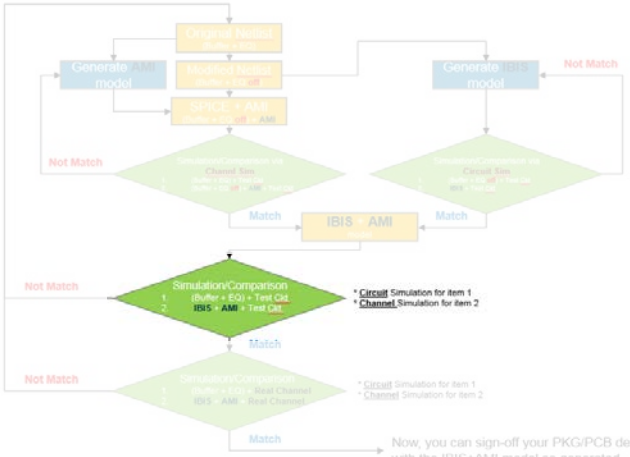
3.1.3 Validation Result Plots

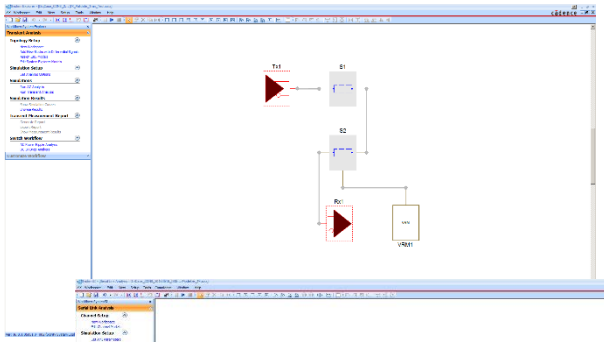
Validation Plot for Model : CDNS_5G_TXN And pin: neg_pos

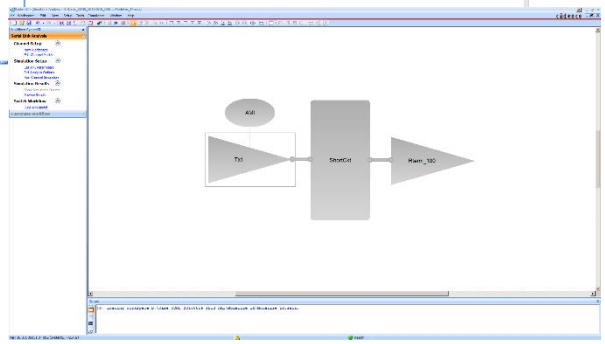
cadence

A real case

- IBIS+AMI model validation

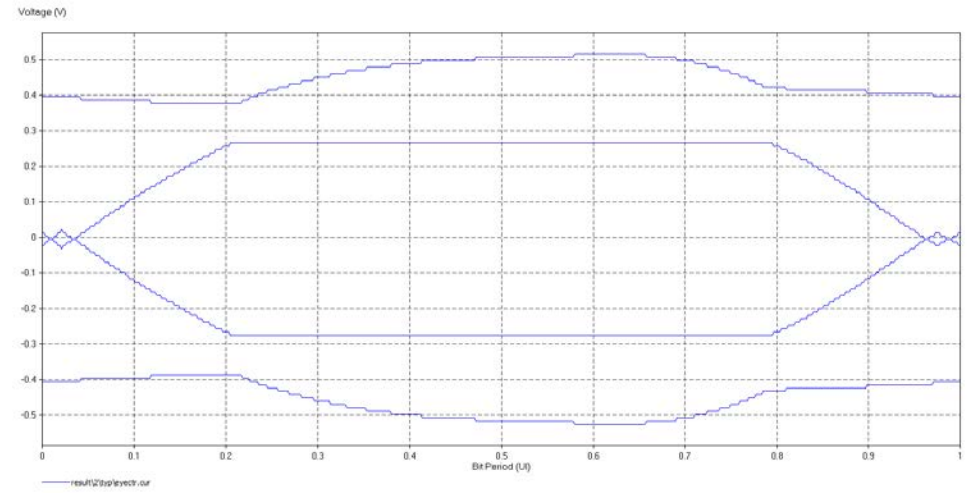






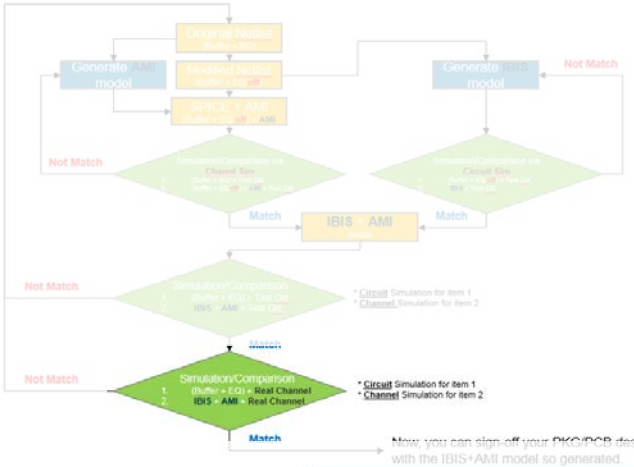
Circuit sim over transistor model in Sigirty™ SystemSI™ technology

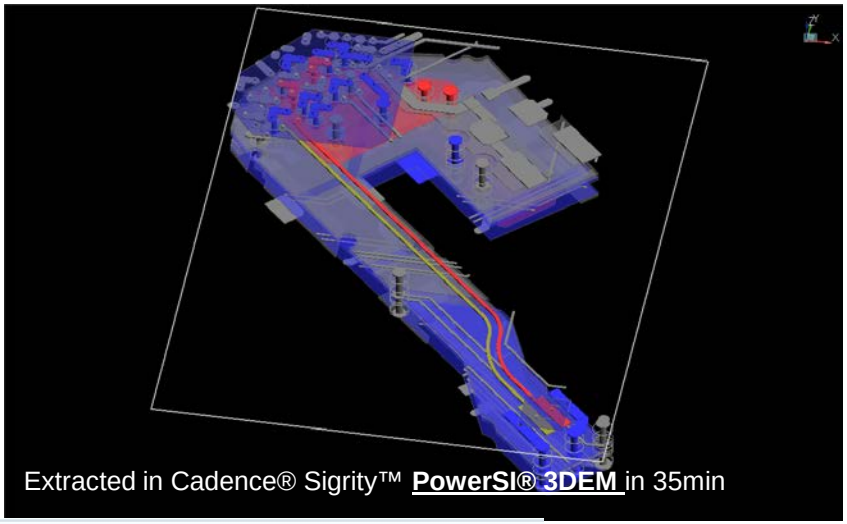
Channel sim over IBIS+AMI model in Sigrity SystemSI technology

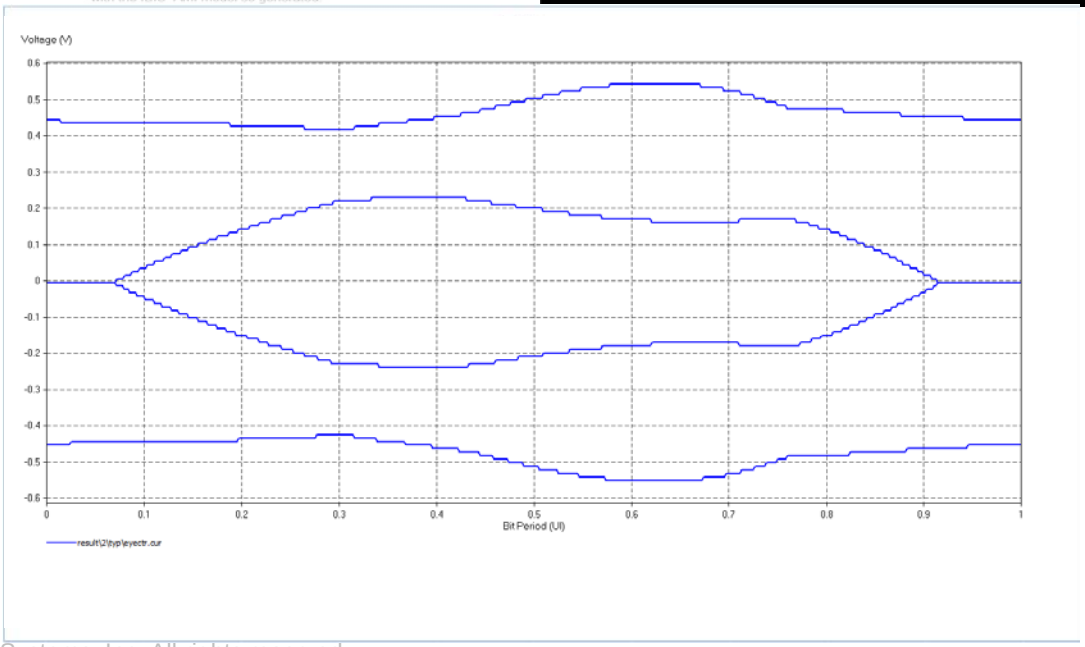


A real case

- IBIS+AMI model validation over a real channel







Agenda

- Traditional signoff flow – circuit simulation
- Channel simulation
  - LTI system
  - Channel Simulation
- IBIS+AMI model
  - What is IBIS+AMI model
  - Algorithmic modeling?
- IBIS+AMI model signoff flow
- Why Cadence?
- A real case
- Conclusion
- More than model generation/validation/simulation…

Conclusion

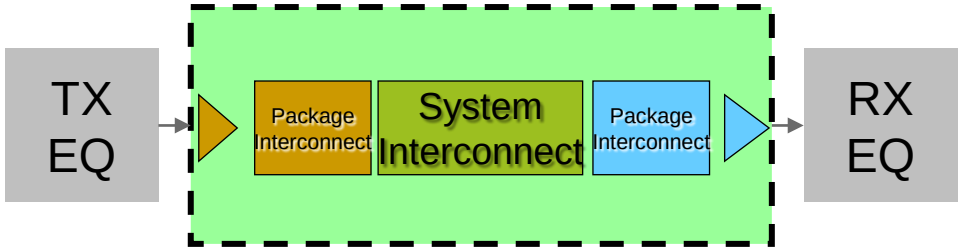
- An accurate IBIS+AMI model could be an alternative approach to validate your “system” design versus a transistor netlist model
- An accurate IBIS+AMI model consists of two parts – an **accurate IBIS model** and **an accurate AMI model** – **validation** is the key
- An accurate IBIS should be generated by a tool which can well describe a **truly differential pair** in all V/I, V/T and I/T curves, such as Sigrity™ T2B™ technology
- An accurate AMI model should be generated by a tool with a rich library such that the generation tool can use all available means in the library to describe all your possible designs, such as Cadence® AMI Builder
- A simulation environment which supports transistor netlist models – not only HSPICE, but also Spectre® simulation - is fundamental for IBIS+AMI model generation/validation, such as Sigrity™ SystemSI™ technology
- Sigrity SystemSI technology provides **good correlation between channel simulation and circuit simulation** Now, only with Sigrity SystemSI, you can trust channel simulation as you used to trust circuit simulation.
- More than model generation/validation and system simulation, Sigrity SystemSI + AMI Builder tools also provide a handy EQ design environment……

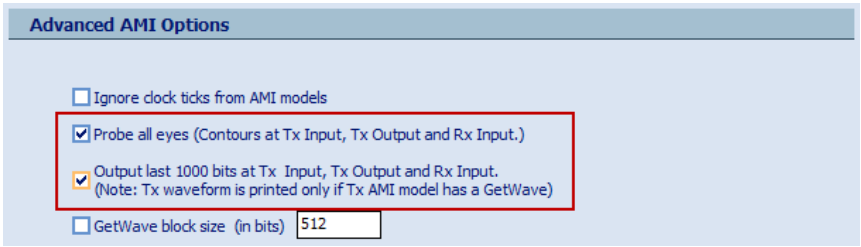
Agenda

- Traditional signoff flow – circuit simulation
- Channel simulation
  - LTI system
  - Channel Simulation
- IBIS+AMI model
  - What is IBIS+AMI model
  - Any improvement?
- IBIS+AMI model signoff flow
- Why Cadence?
- A real case
- Conclusion
- More than model generation/validation/simulation…

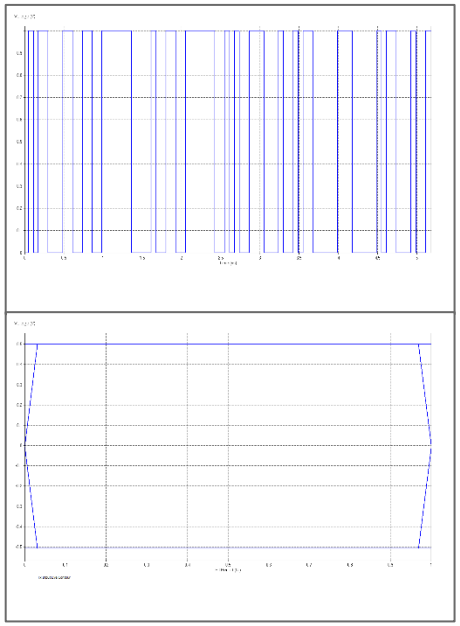
Sigrity SystemSI – probe all eyes at every stage

- Accompanied with Sigrity™ SystemSI™ SLA, advance EQ designer can use "**Probe all eyes**" and "**Output last 1000 bits at Tx_input, Tx_output, Rx_input**" to validate if the EQ works properly

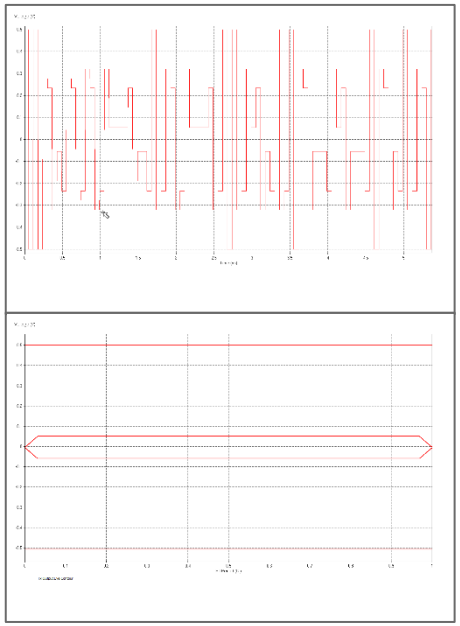




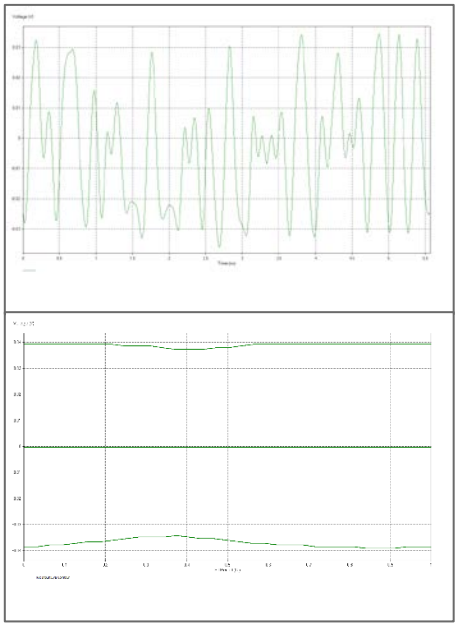
Eye **before TX** EQ



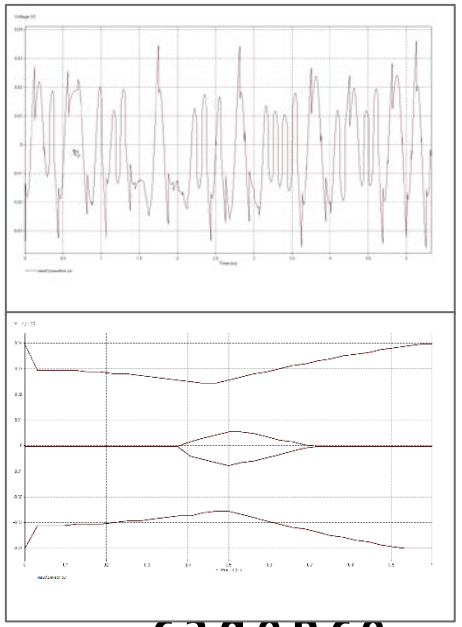
Eye **After TX** EQ



Eye **Before RX** EQ

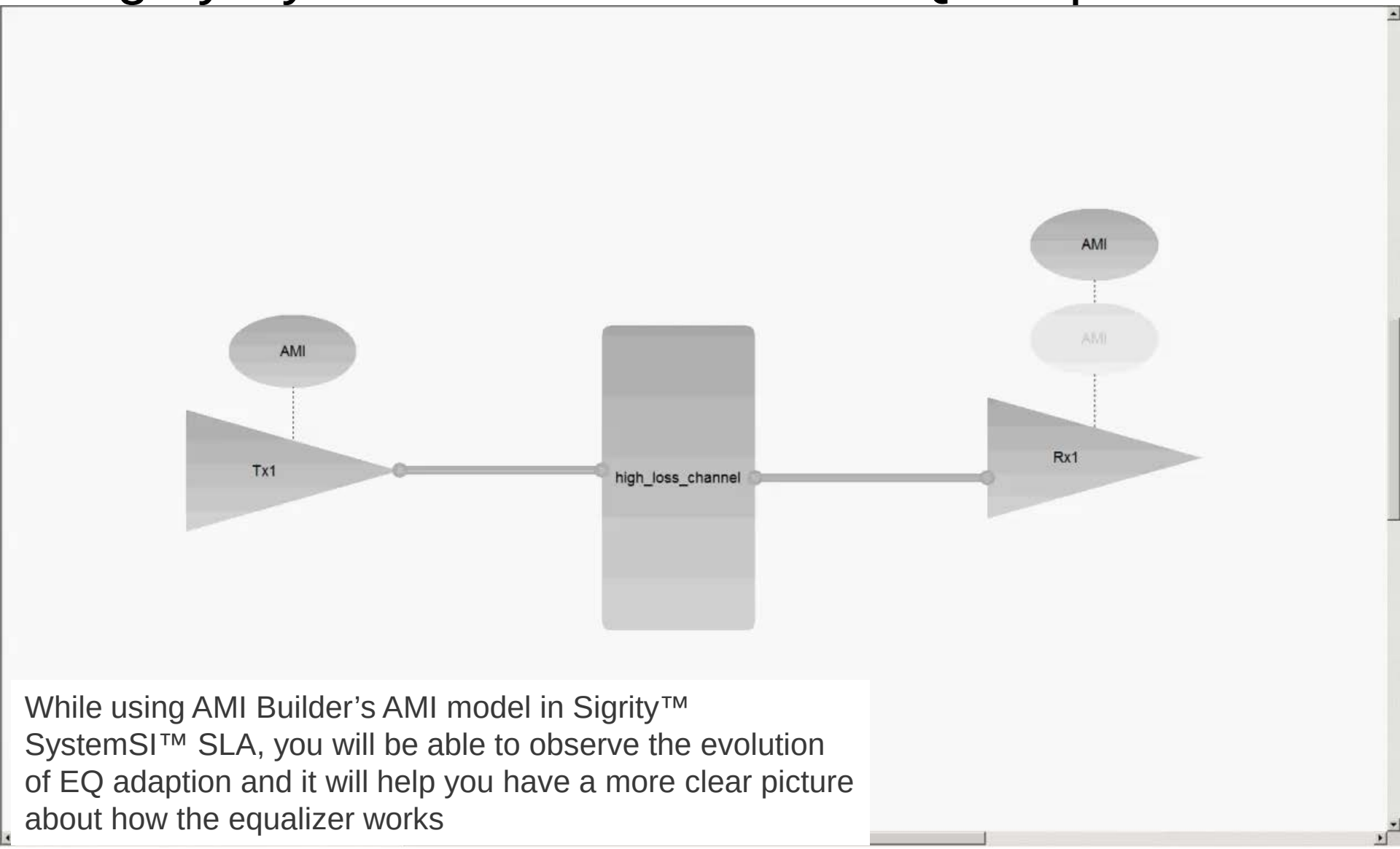


Eye **After RX** EQ



cadence

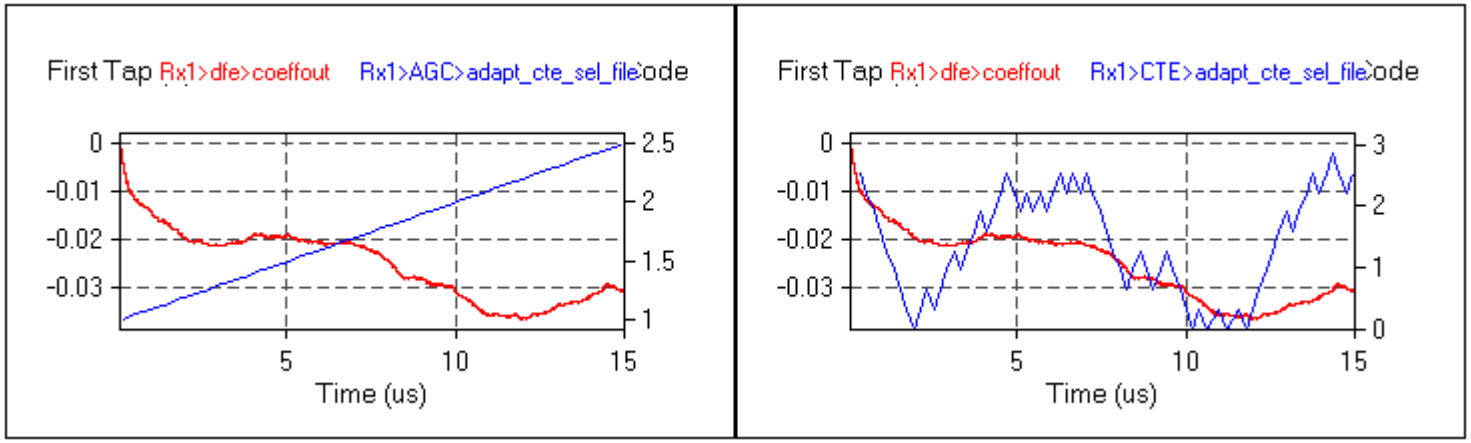
Sigrity SystemSI – evolution of EQ adaption



While using AMI Builder's AMI model in Sigrity™ SystemSI™ SLA, you will be able to observe the evolution of EQ adaption and it will help you have a more clear picture about how the equalizer works

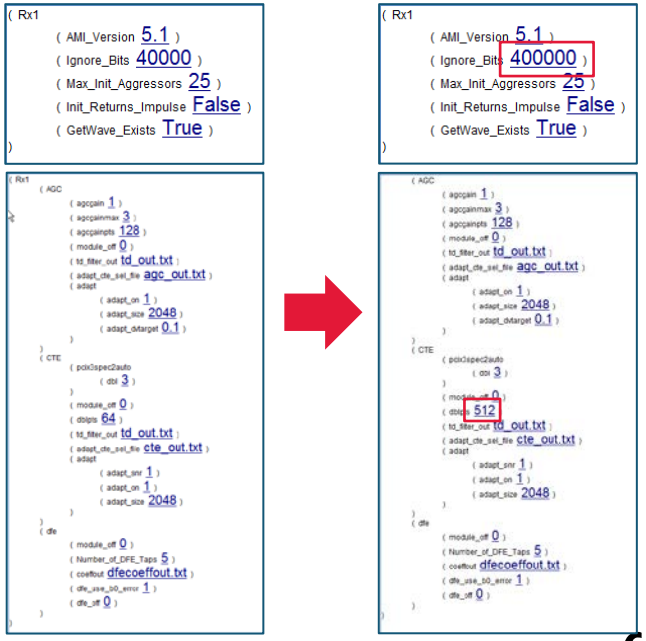
Sigrity SystemSI – evolution of EQ adaption

As you can see, in this simulation, the adaption is not stable yet. The parameters of AGC, CTLE, and DFE are still under tuning to find a better value.

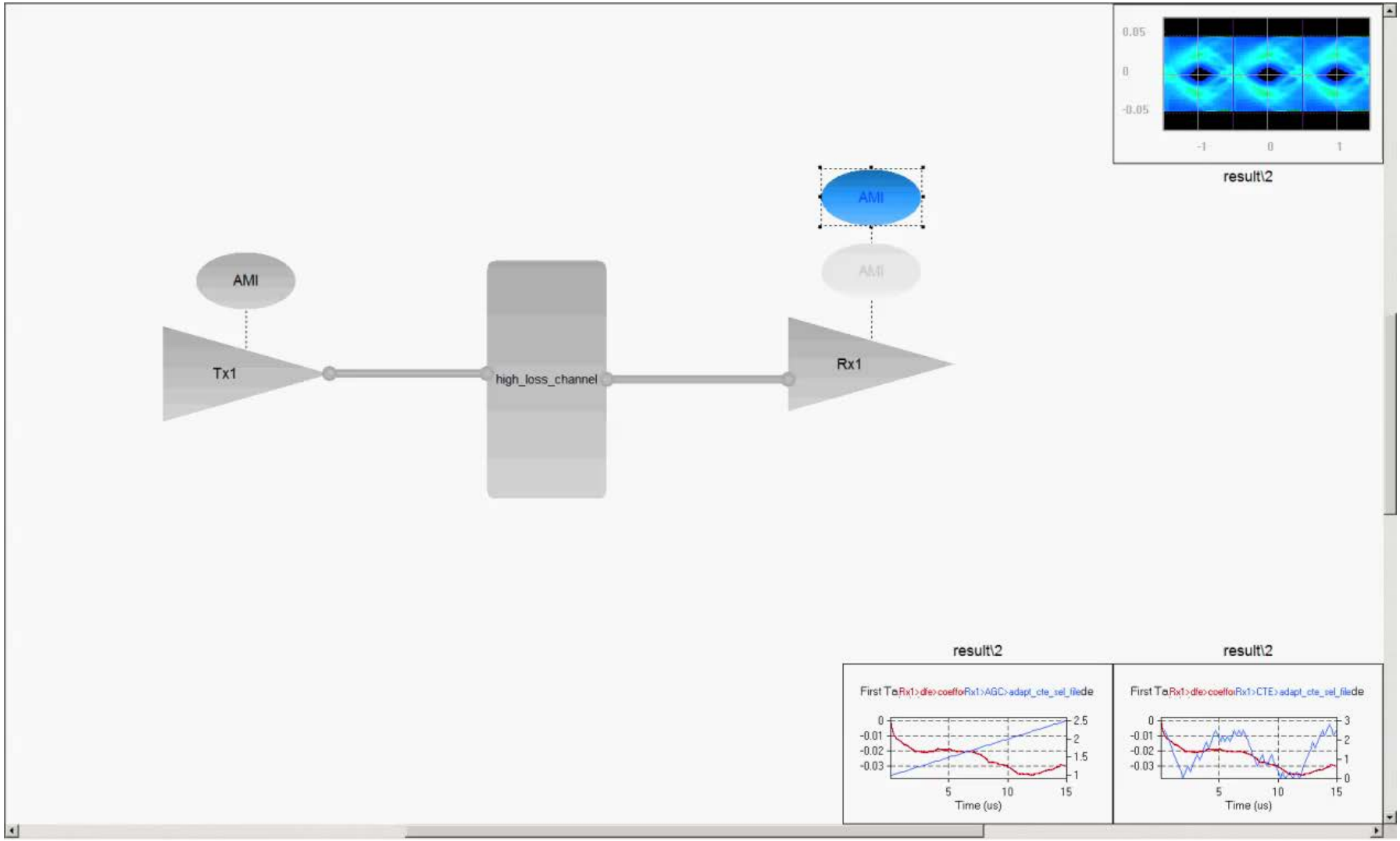


You will notice some significant facts:

1. AGC parameter's value keeps moving up from 0 to 15us
2. CTLE parameter's value jumps between the range from 0 to 3 in the period from 0 to 15us
3. DFE parameter's value seems to be stable after 10us



Sigrity SystemSI – evolution of EQ adaption



Sigrity SystemSI – evolution of EQ adaption