Accurate Modelling of PCIe® 3.0 Analog Buffers

Shivani Sharma

Design Engineer II

Cadence Design Systems

Disclaimer

- **Presentation Disclaimer**: All opinions, judgments, recommendations, etc. that are presented herein are the opinions of the presenter of the material and do not necessarily reflect the opinions of the PCI-SIG®.

Agenda

- Problem Statement
- Overview of IBIS-AMI Modeling
- Accurate Modeling of PCIe® SerDes IO for 16Gbps or Higher Speeds
  - ✓Using Parameterized AMI Blocks
  - ✓Using Virtual Reference Design (VRD) Flow for Quick Sign-off
- Conclusion

Agenda

- Problem Statement
- Overview of IBIS-AMI Modeling
- Accurate Modeling of PCIe SerDes IO for 16Gbps or Higher Speeds
  - ✓Using Parameterized AMI Blocks
  - ✓Using Virtual Reference Design (VRD) Flow for Quick Sign-off
- Conclusion

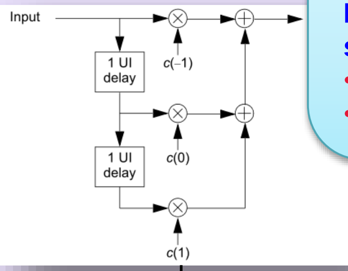
Problem Statement

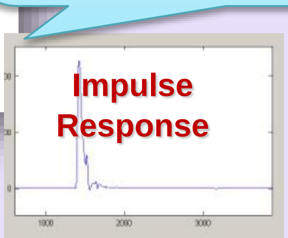
- PCIe 3.0 and other current and future high speed protocols require complex equalization schemes to open the eye at the receiver sampler
- These equalizers (AGC, CTE, DFE, etc.) interact with each other during adaptation
- Managing this sequence of events can be challenging
- Correlation is even more challenging

Channel Simulation; EQ Basics

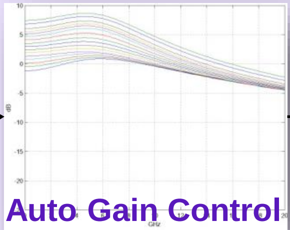
- Preemphasis or deemphasis to compensate for channel loss
- Can do precursors and post cursors
- Can optimize for the channel as impulse Response is provided in software simulation, else supports backchannel
- Static, no real adaptation
- Avoid if pwr is main concern

FFE

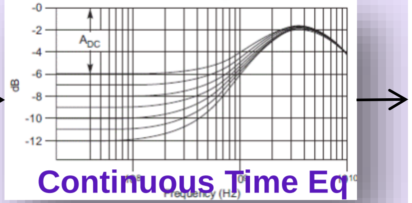




Channel

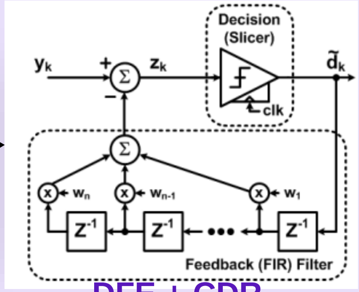


Auto Gain Control (AGC)



Continuous Time Eq (CTLE)

- Selective boosting at frequency of interest (high freq channel loss cancellation)
- Better for area/power considerations
- Noise and xtalk also gets boosted
- Usually first order filter only



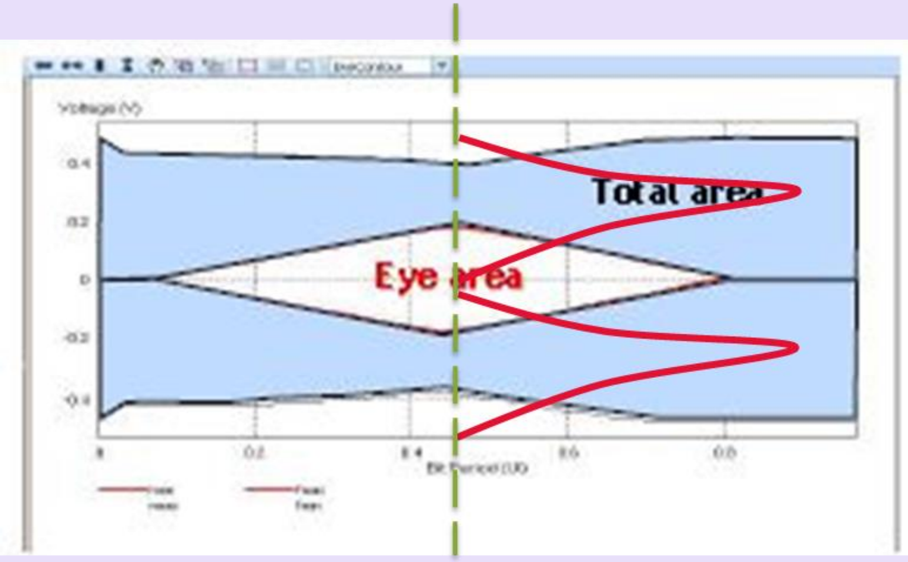
DFE + CDR

- Curve based function to boost the incoming signal so that it could be detected at the output
- Noise also gets boosted

- Feeds back previous bit decisions to cancel post cursor ISI caused by them
- Can model non-linearity
- Adaptively tunes tap coefficient
- Level sensitive
- Cannot cancel pre-cursor
- Needs 'something' to work with → AGC/CTLE

Objectives of Equalization

- Maximize SNR
  - ✓ Done by reducing the area/spread of the PDF
- Auto DFE figures out the peak and tries to maximize the peak
  - ✓ Focus on the center, works better in xtalk minimization
  - ✓ Digitally aware
- CTE follows the DFE
  - ✓ Analog (indirectly sharpens IR)
  - ✓ Cheap/less power/area
  - ✓ Can amplify noise
- VGA amplifies the target to achieve the target
  - ✓ Which also amplifies the noise
  - ✓ Lower dv_target – the better for noise – IF the Rx can detect (Tradeoff)



Agenda

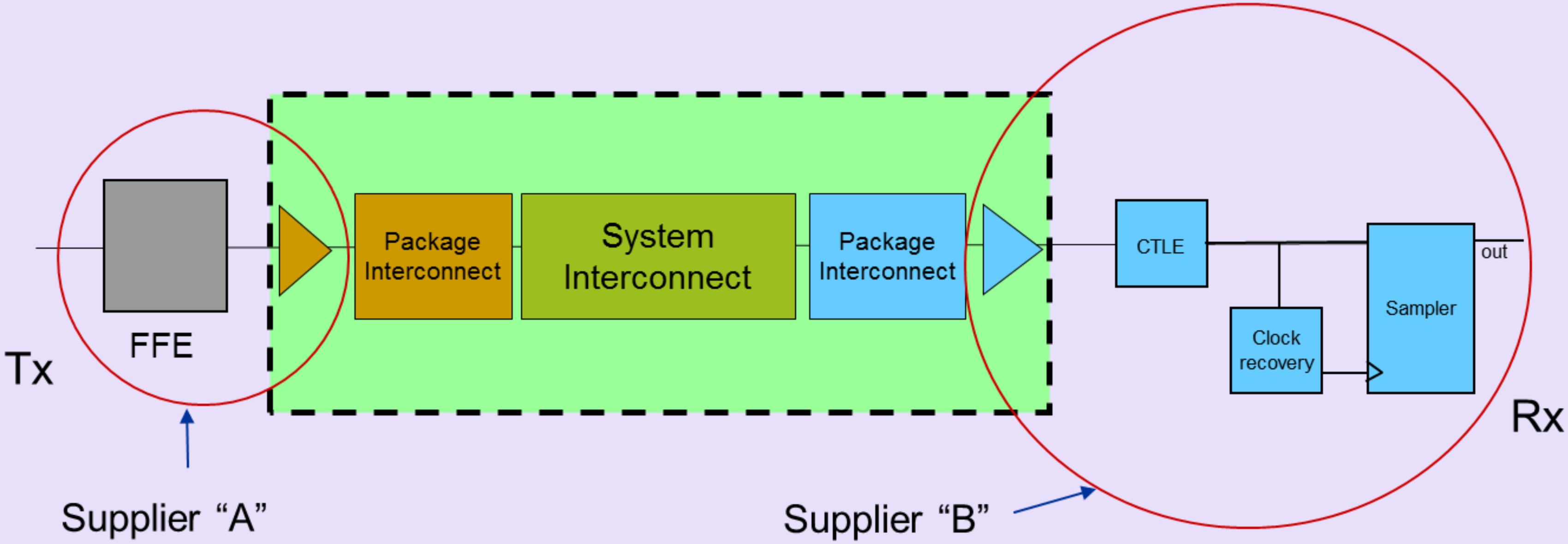
- Problem Statement
- Overview of IBIS-AMI Modeling
- Accurate Modeling of PCIe SERDES IO for 16Gbps or Higher Speeds
  - ✓Using Parameterized AMI Blocks
  - ✓Using Virtual Reference Design (VRD) Flow for Quick Sign-off
- Conclusion

AMI > Algorithmic Modeling Interface

- Extension made to IBIS in 2007
  - ✓ **Cadence** at the forefront of driving AMI through the standardization process
- Enables executable, software-based, algorithmic models to work together with traditional IBIS circuit models
  - ✓ Allows deeper access to on-chip technology/secret sauce
- Enables SerDes adaptive equalization algorithms to be modeled and used during channel simulation
  - ✓ Fast, accurate and flexible

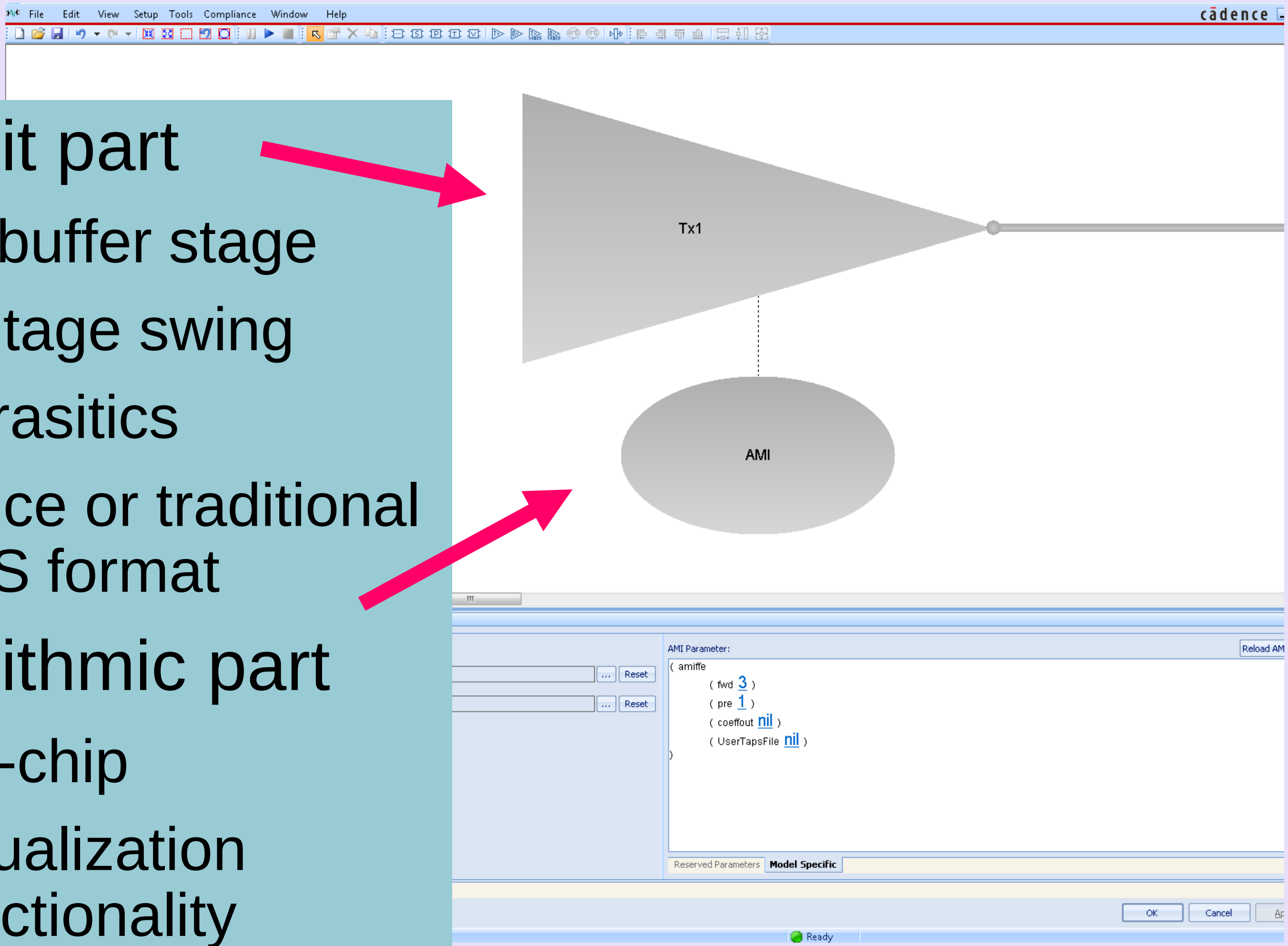
Motivation for AMI

- Interoperability: IBIS-AMI allows plug-and-play simulation compatibility between SerDes models from different suppliers, in a standard commercial EDA format.
- IP Protection: Shared Objects (dlls) are compiled. EDA tool communicates with the dll using the standardized API.
- Flexibility: The Model Maker can use any high level programming language (ex C) to describe the eq at both ends.



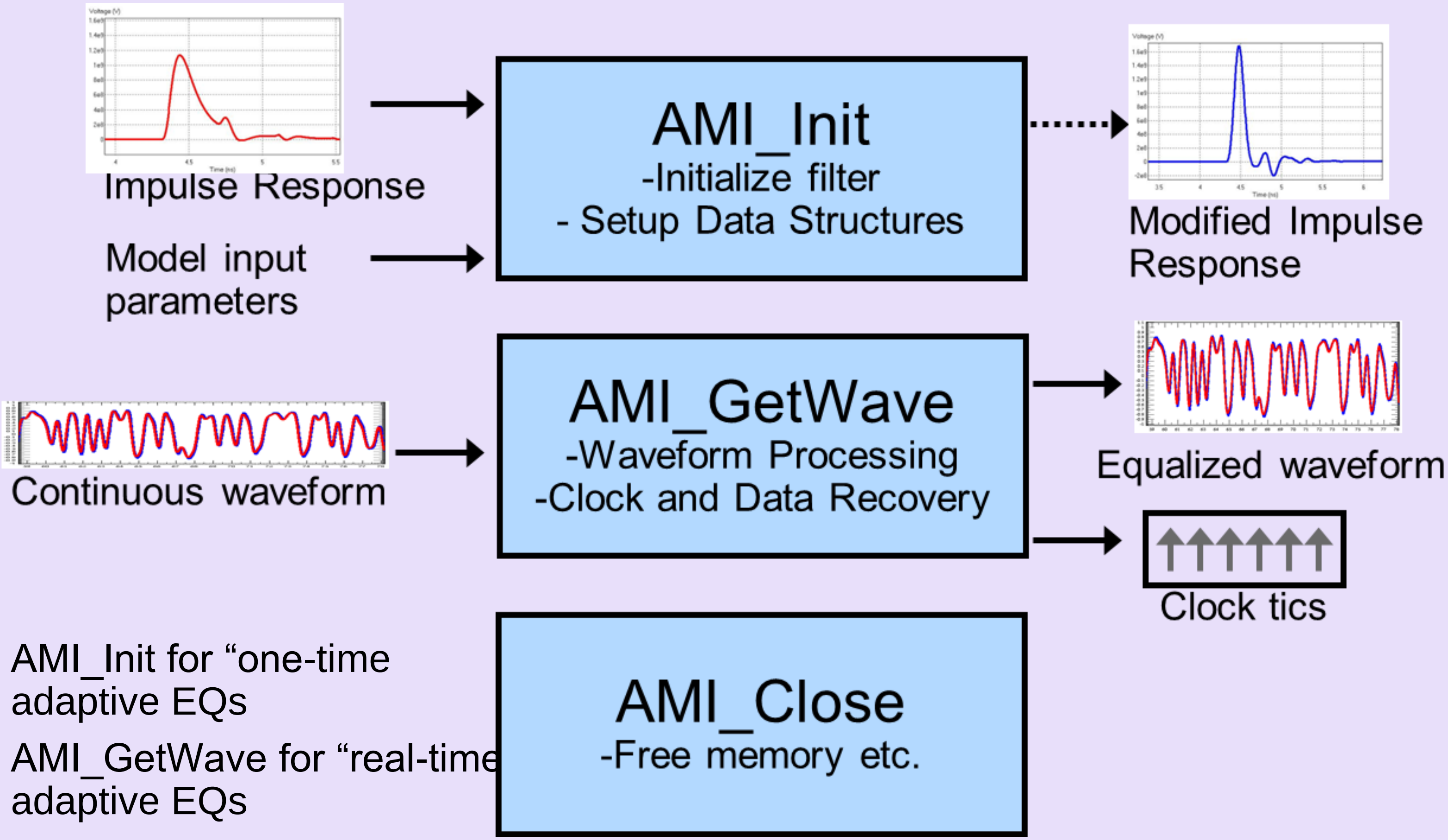
IBIS-AMI Model Sub-Components

- Circuit part
  - ✓ IO buffer stage
  - ✓ Voltage swing
  - ✓ Parasitics
  - ✓ Spice or traditional IBIS format
- Algorithmic part
  - ✓ On-chip
  - ✓ Equalization functionality
  - ✓ DLL + AMI file



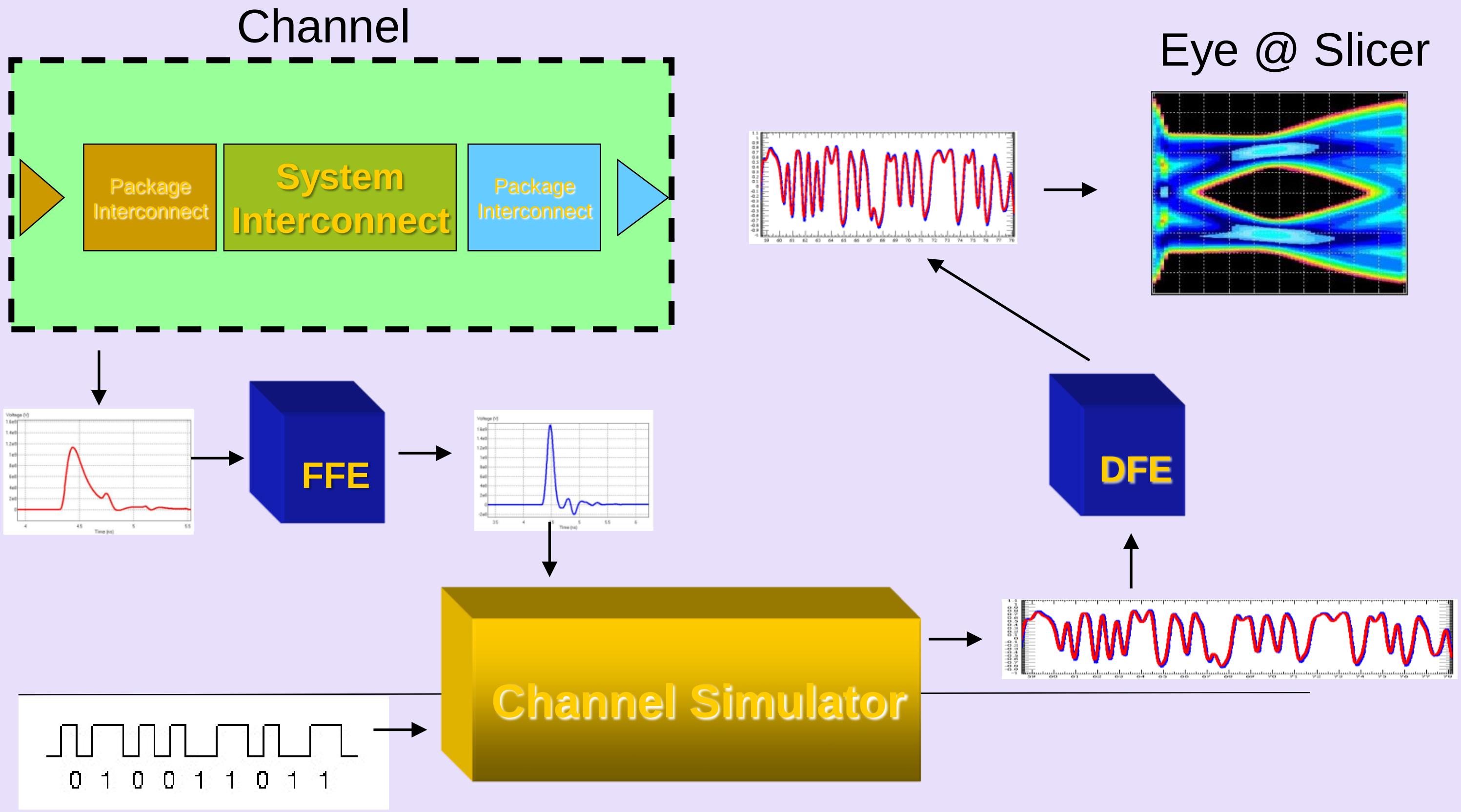
```
( amiffe
        ( fwd 3 )
        ( pre 1 )
        ( coeffout nil )
        ( UserTapsFile nil )
)
```

APIs in IBIS-AMI Modeling



- AMI_Init for “one-time adaptive EQs
- AMI_GetWave for “real-time adaptive EQs

IBIS-AMI and Channel Simulation



Agenda

- Problem Statement
- Overview of IBIS-AMI Modeling
- Accurate Modeling of PCIe SerDes IO for 16Gbps or Higher Speeds
  - Using Parameterized AMI Blocks
  - Using Virtual Reference Design (VRD) Flow for Quick Sign-off
- Conclusion

Accurate Modeling for 16 Gbps or Higher Speeds

- SerDes transmitter and receiver modules
  - ✓ Transmitter
    - FFE
    - Edge-boost
    - TX matching network
  - ✓ Receiver
    - Rx matching network
    - Continuous-time linear equalizer (CTLE)
    - Variable gain amplifier (VGA)
    - Decision feedback equalizer (DFE)
    - Clock and data recovery (CDR)

Accurate Modeling for 16 Gbps or Higher Speeds

- Tx AMI
  - ✓ FFE is modelled as C code
  - ✓ Boost circuit characterized by step response
  - ✓ Tx matching network characterized by step response
- RX AMI
  - ✓ Tx matching network characterized by step response
  - ✓ CTLE, VGA are characterized by step responses
  - ✓ DFE, CDR are modelled as C code
  - ✓ Data path and a parallel clock path

Accurate Modeling for 16 Gbps or Higher Speeds

- **Use s-parameters to model matching network in analog portion**
  - ✓Boost circuit is implemented at Tx side to improve rise/fall time of the signal
  - ✓Boost circuit is a current-mode driver present in parallel with the voltage mode driver module

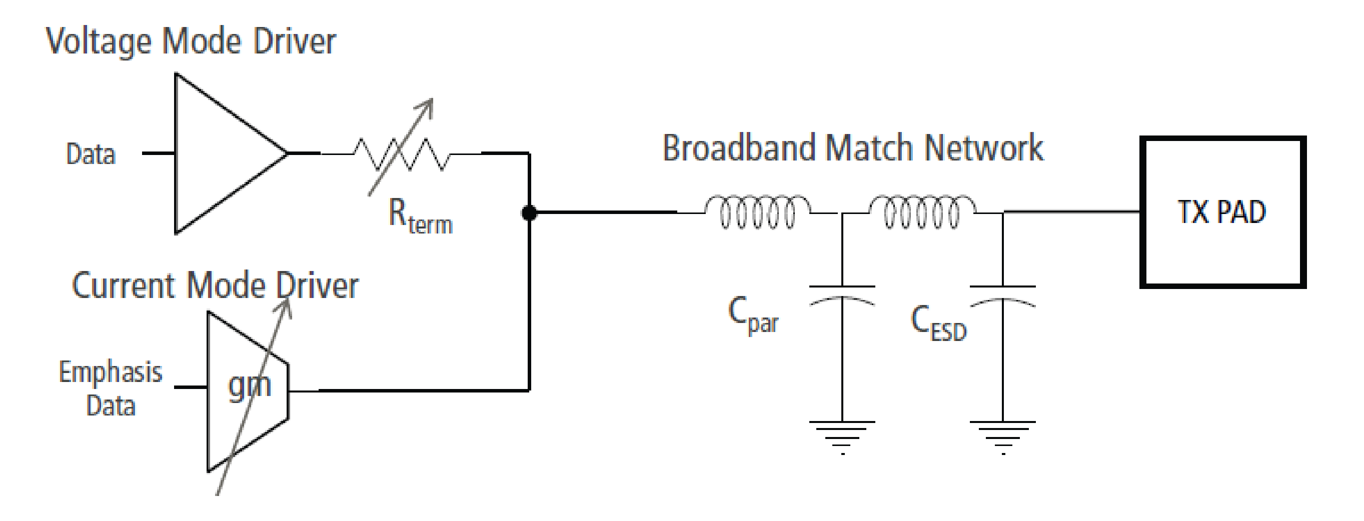
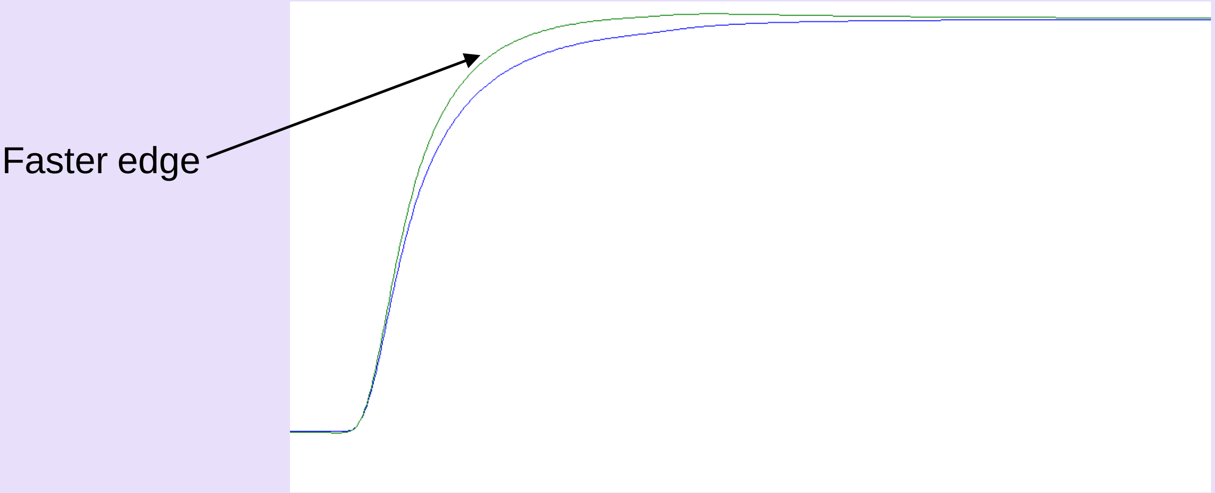


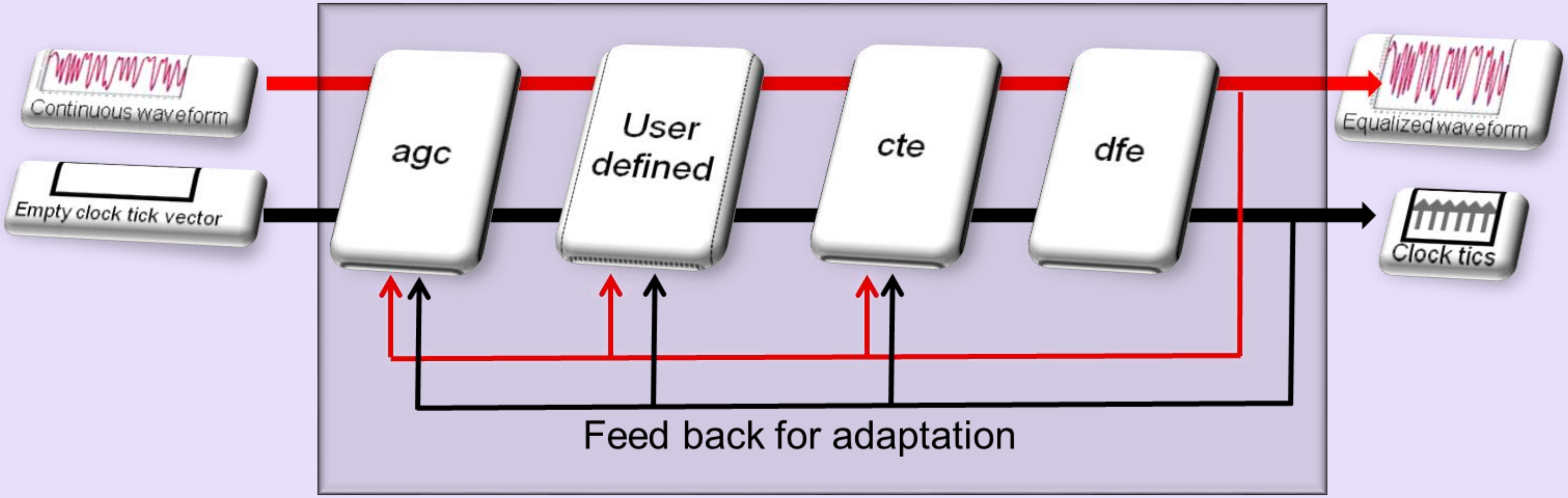
Fig5: Tx Edge-boost circuit in parallel path

Accurate Modeling for 16 Gbps or Higher Speeds

- **Modeling Boost effects as IR or s-parameters**
  - ✓ Input signal is differentiated and fed to a voltage-controlled current source. Rise/fall time is improved by pumping extra current at the Tx output node.
  - ✓ Assuming boost circuit is linear, enable boost and characterize Tx while terminating in reference impedance. (Repeat steps 1 and 2)

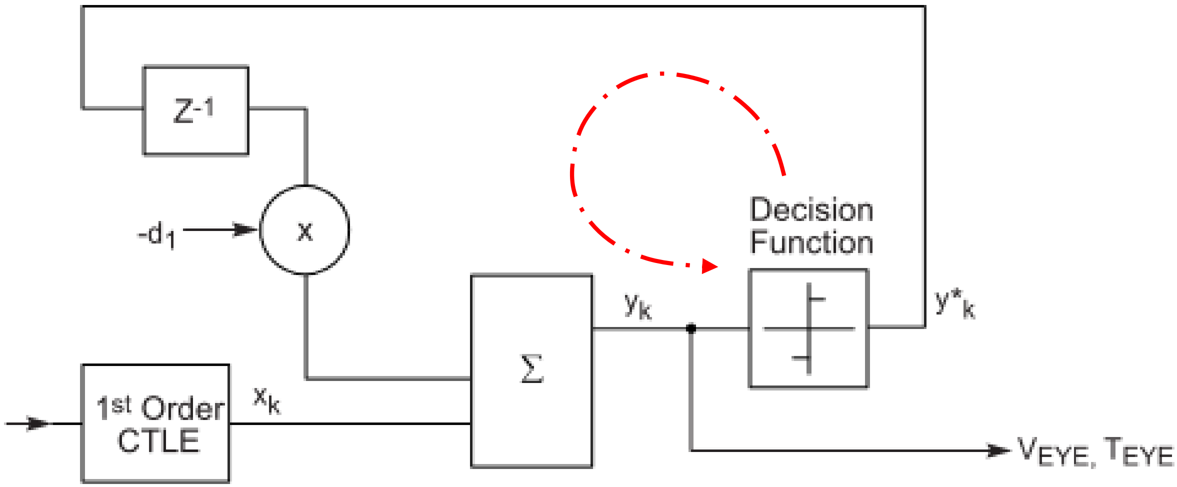


Accurate Modeling of PCIe SERDES IO Using AMI Blocks



- High level of architectural abstraction
- Extremely powerful and flexible
- Capability to model Tx/Rx end to end
- Parameterized blocks

CTE and DFE for PCIe 3.0



$y_k = x_k - d_1 \mathrm{sgn}(y_{k-1})$

y_k = DFE summer differential output voltage.

y^*_k = decision function output voltage. $|y^*_k| = 1$

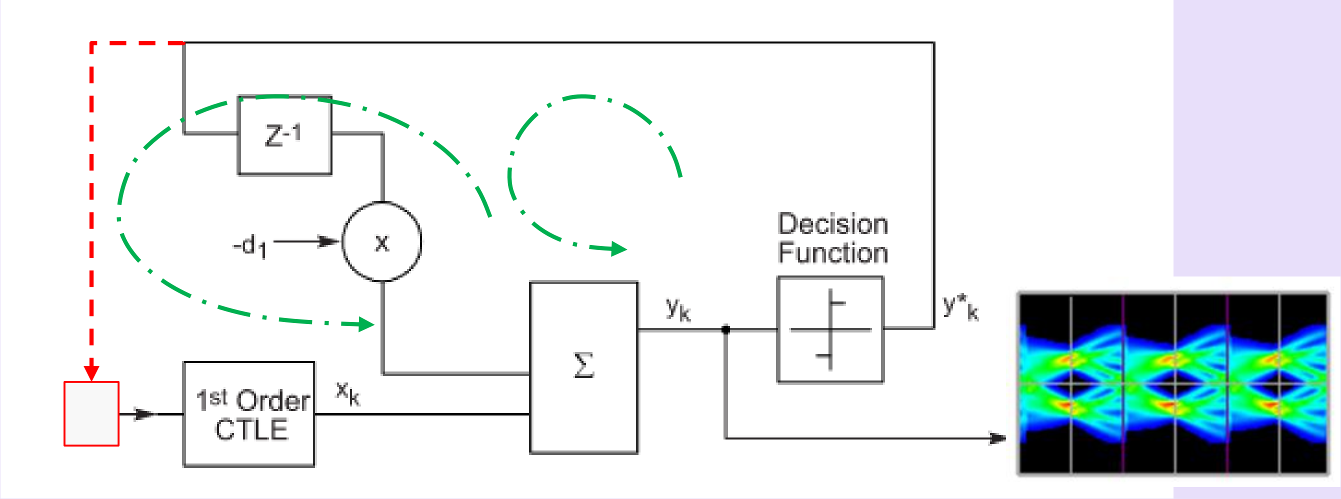
x_k = DFE differential input voltage

d_1 = feedback coefficient

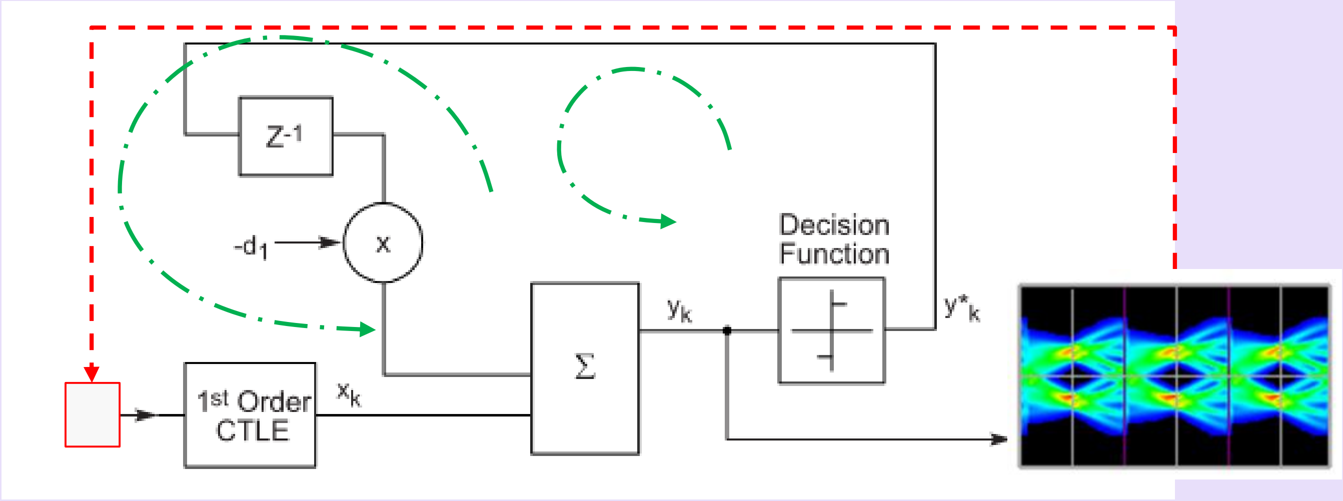
k = sample index in UI

A-0830

CTE Adapting with DFE

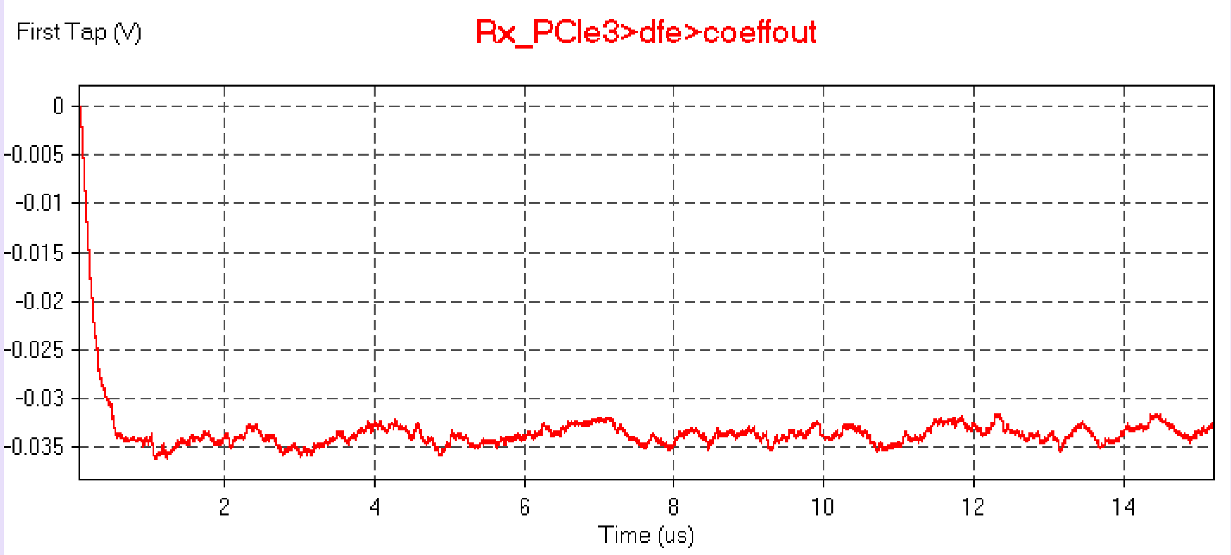


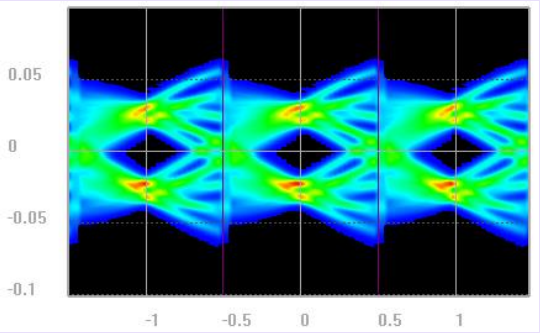
CTE Adapting with Other Metrics (Ex SNR)



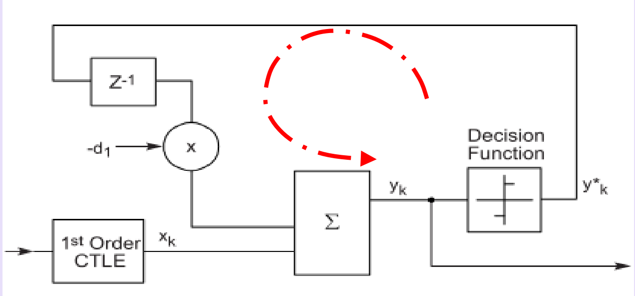
Adaptation can get complex with multiple blocks adapting affecting each other

DFE Only Adaptation at 16Gbps

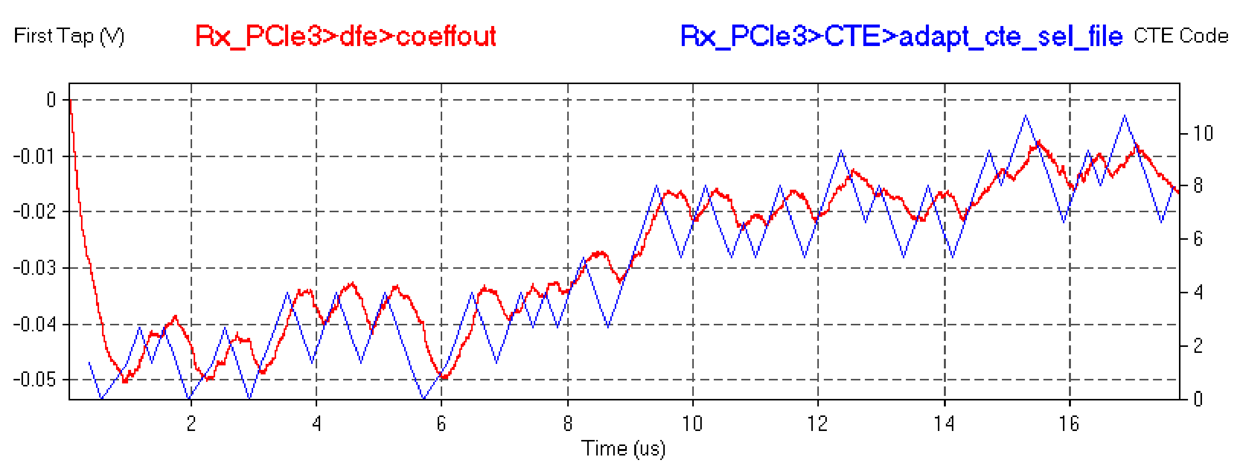


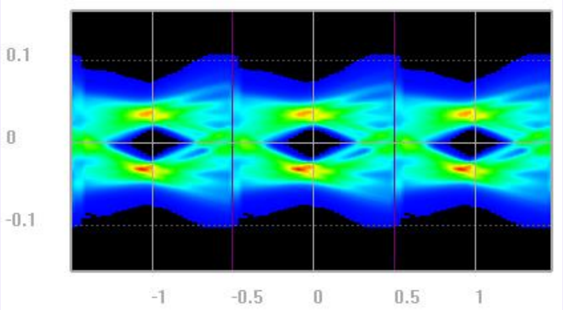


- CTE Adaptation based on SNR

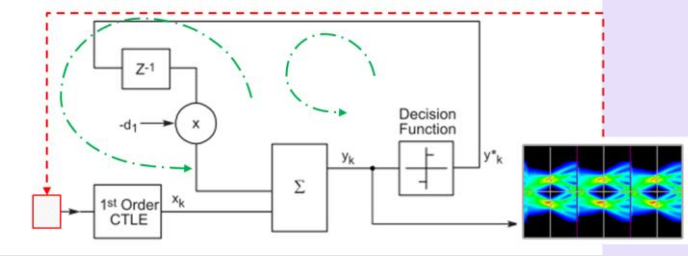


Adapting CTE May Throw the DFE Off

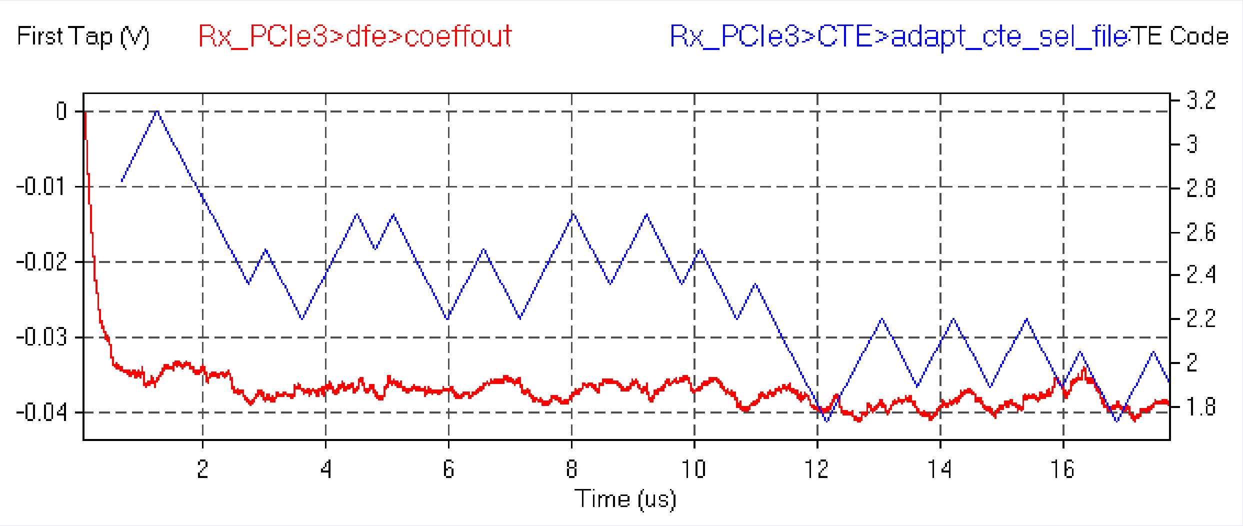




- CTE Adaptation based on SNR

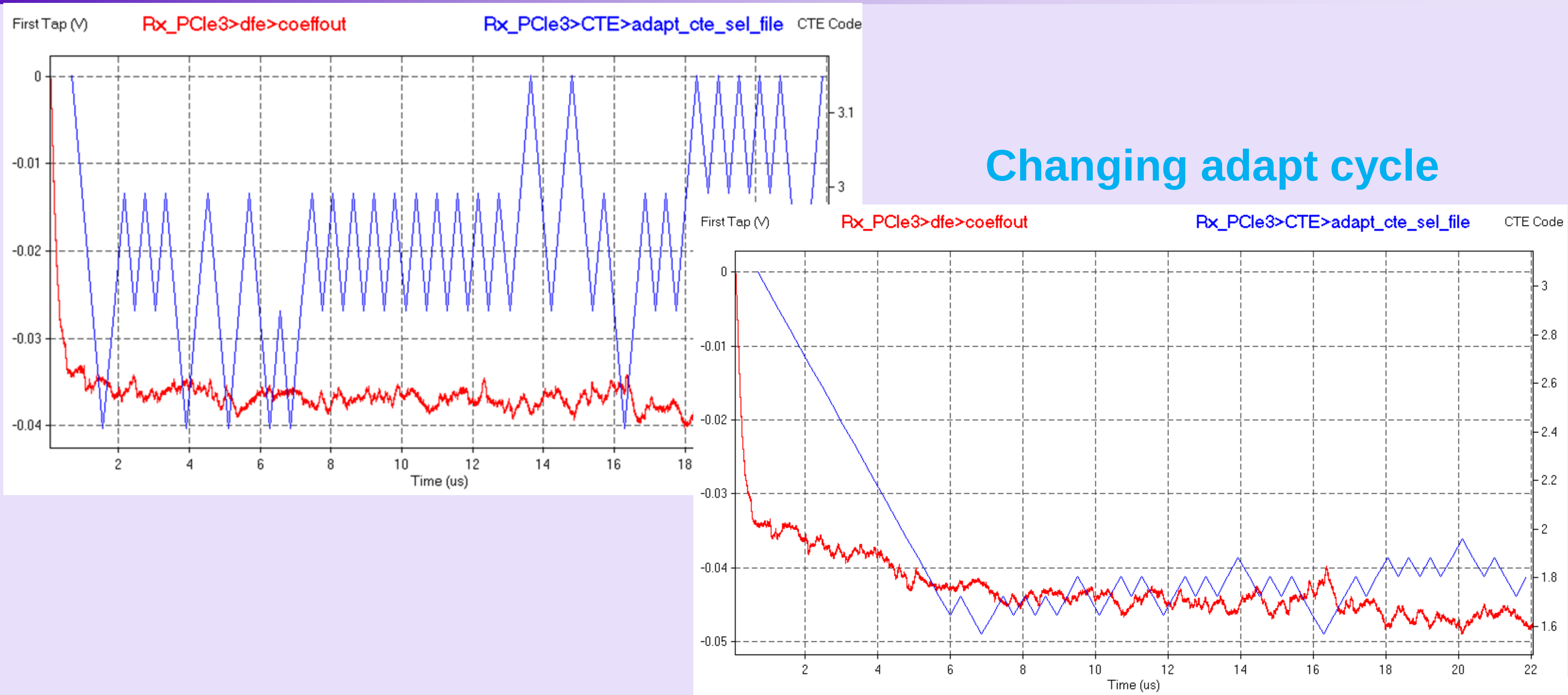


Changing CTE Adaptation Cycle



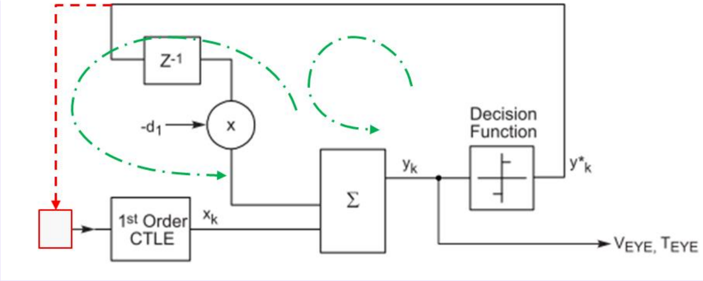
- Making the adapt cycle slower may be beneficial

Effects of Changing Adaptation Algorithm for CTE

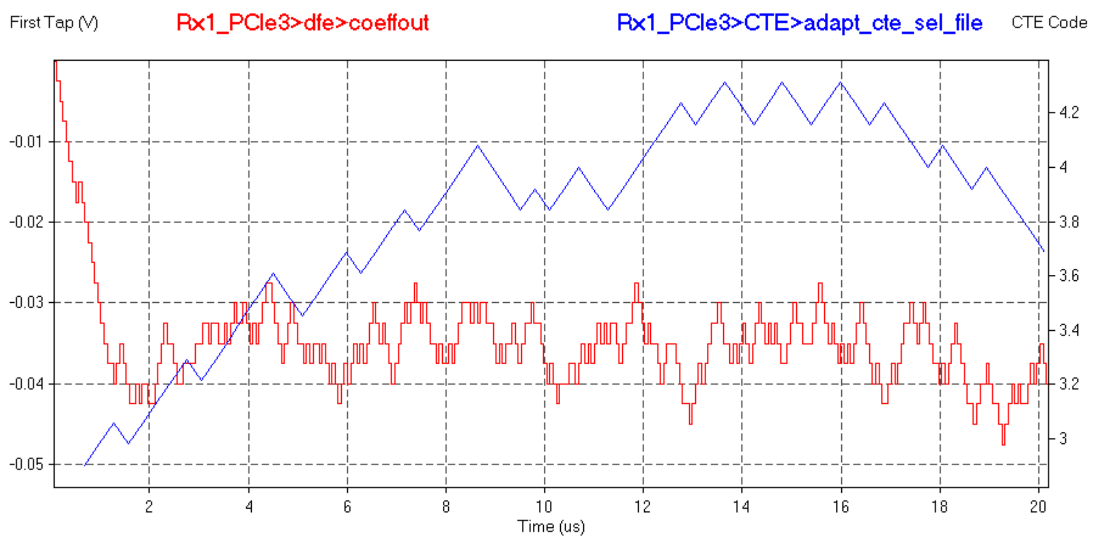


Changing adapt cycle

- CTE adaptation based on 1st DFE Tap



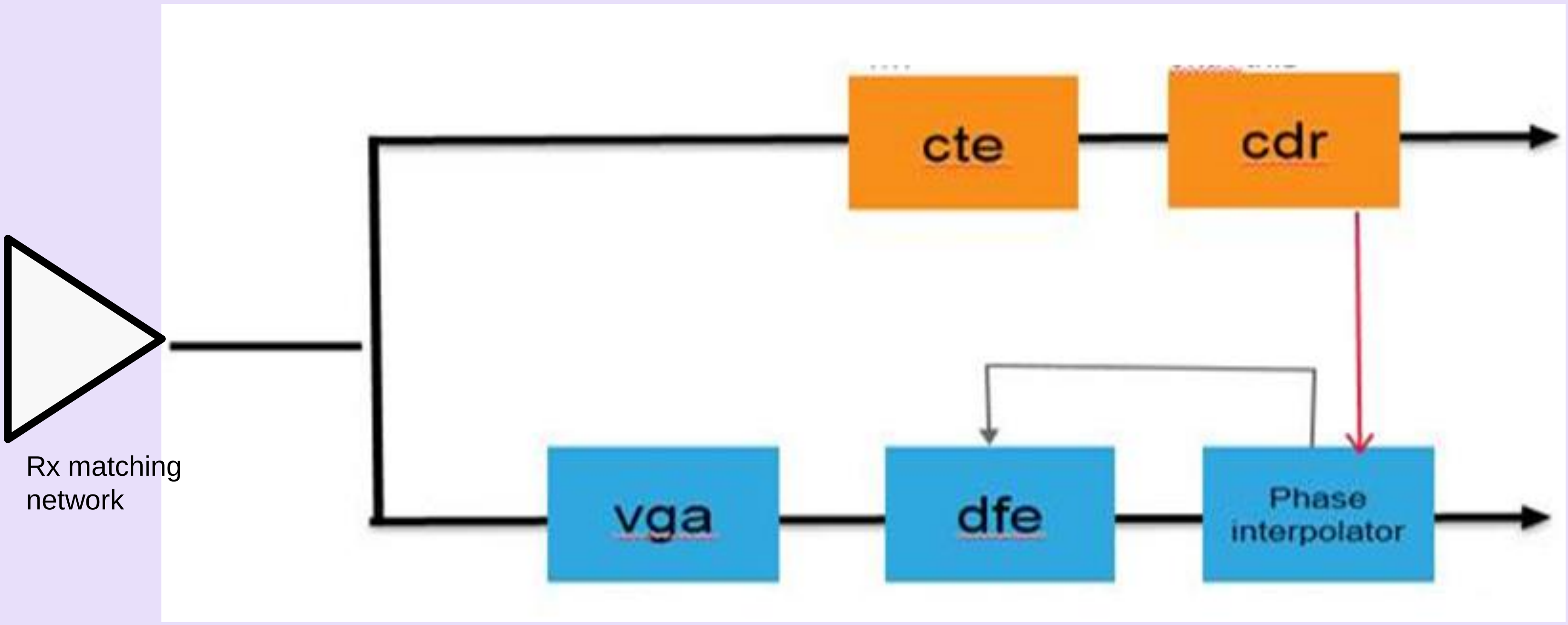
Digital DFE and Adding Adapting AGC Effects DFE



- Digital DFE only updates after 1024 bits instead of every bit (analog).
- Generally takes longer to finish adapting

Parallel Clock Path and Data Path

Clock path



Data path

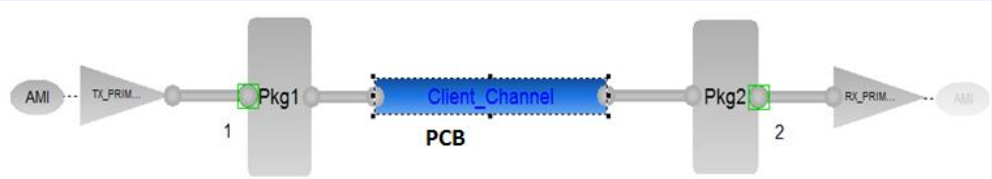
Agenda

- Problem Statement
- Overview of IBIS-AMI Modeling
- Accurate Modeling of PCIe SerDes IO for 16Gbps or Higher Speeds
  - ✓Using Parameterized AMI Blocks
  - ✓Using Virtual Reference Design (VRD) Flow for Quick Sign-off
- Conclusion

What’s in a VRD (Design-in Kit)?

- A ***virtual reference design***
  - ✓ All the stuff in here:

  - ✓ Is modeled here:



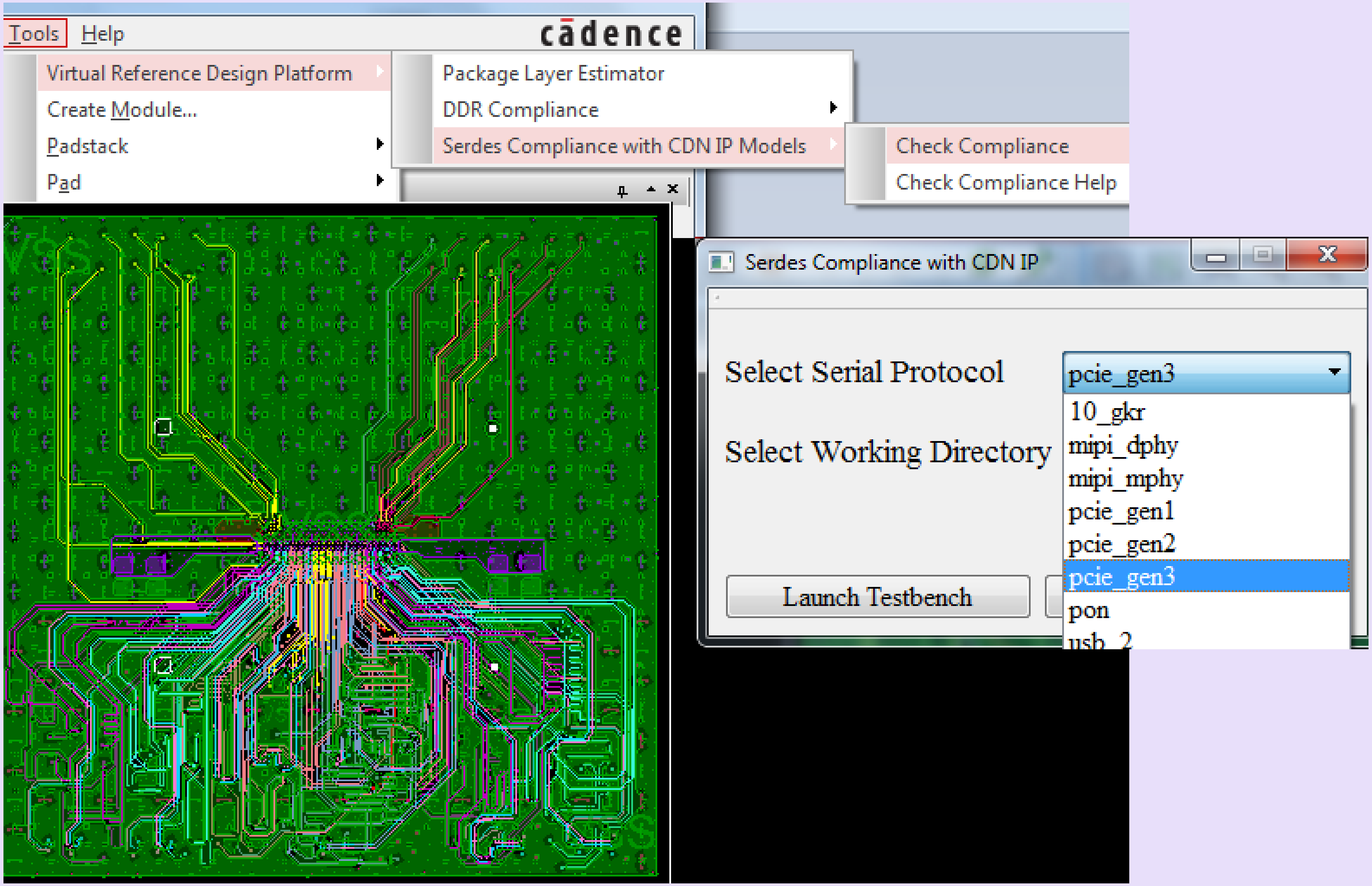
Measurement results are replaced with simulation results

  - ✓ And simulated/ measured here:

What’s in a VRD?

- Parameterized AMI blocks to quickly model various architectures
- Pre-created test-benches to quickly simulate as per compliance
  - ✓ Ability to switch channel models
- Ability to quickly transfer PCB/Package model into the test-bench
  - ✓ Can estimate Package for given nets and Package size
- Run simulations and get compliance reports

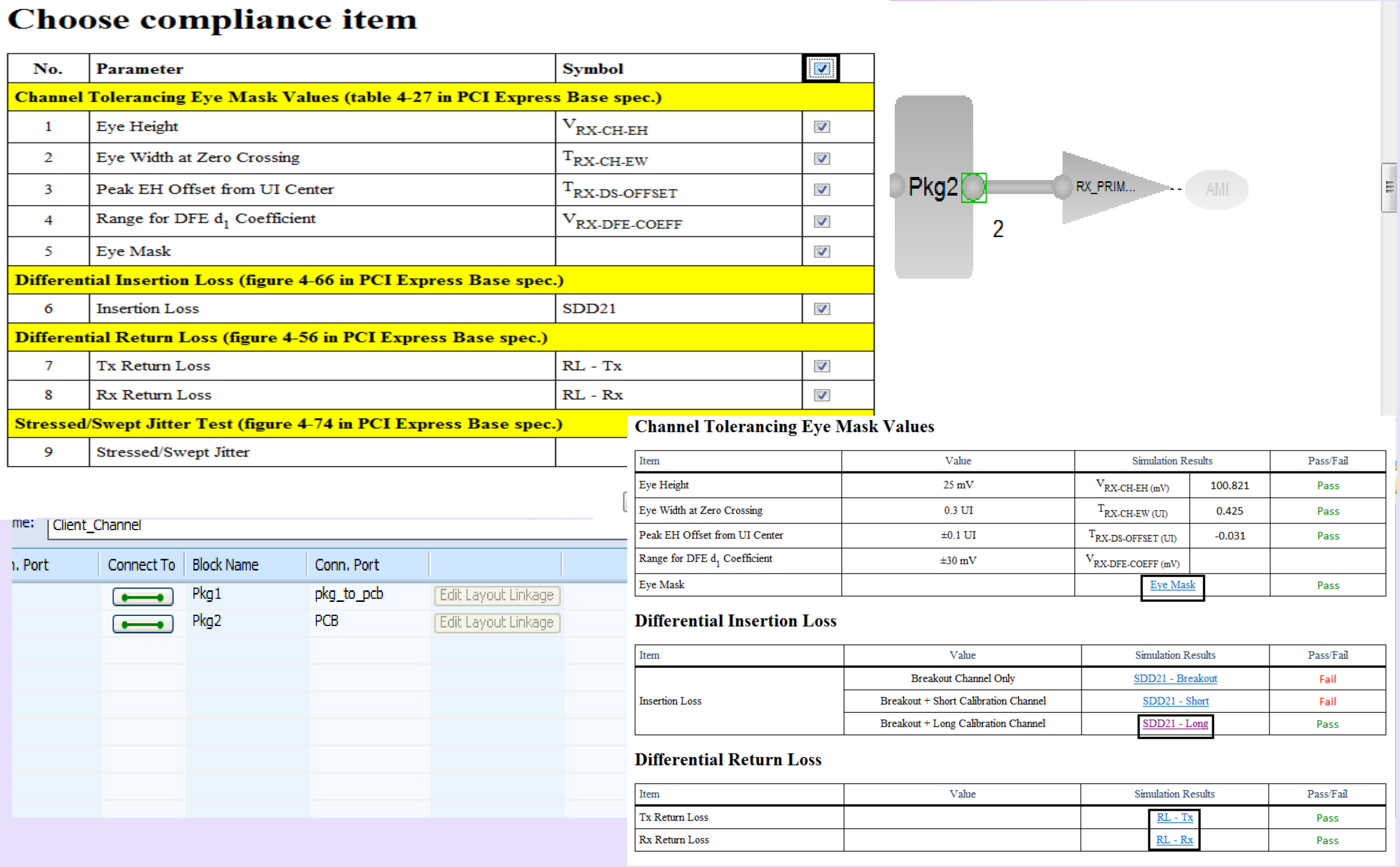
VRD Flow for PCIe 3.0



Select Serial Protocol: pcie_gen3

Select Working Directory

VRD Flow for PCIe 3.0



Choose compliance item

No.	Parameter	Symbol	☑
Channel Tolerancing Eye Mask Values (table 4-27 in PCI Express Base spec.)			
1	Eye Height	$V_{RX\text{-}CH\text{-}EH}$	☑
2	Eye Width at Zero Crossing	$T_{RX\text{-}CH\text{-}EW}$	☑
3	Peak EH Offset from UI Center	$T_{RX\text{-}DS\text{-}OFFSET}$	☑
4	Range for DFE d_1 Coefficient	$V_{RX\text{-}DFE\text{-}COEFF}$	☑
5	Eye Mask		☑
Differential Insertion Loss (figure 4-66 in PCI Express Base spec.)			
6	Insertion Loss	SDD21	☑
Differential Return Loss (figure 4-56 in PCI Express Base spec.)			
7	Tx Return Loss	RL - Tx	☑
8	Rx Return Loss	RL - Rx	☑
Stressed/Swept Jitter Test (figure 4-74 in PCI Express Base spec.)			
9	Stressed/Swept Jitter		

me: Client_Channel

n. Port	Connect To	Block Name	Conn. Port	
	⟷	Pkg1	pkg_to_pcb	Edit Layout Linkage
	⟷	Pkg2	PCB	Edit Layout Linkage

Channel Tolerancing Eye Mask Values

Item	Value	Simulation Results		Pass/Fail
Eye Height	25 mV	$V_{RX\text{-}CH\text{-}EH\ (mV)}$	100.821	Pass
Eye Width at Zero Crossing	0.3 UI	$T_{RX\text{-}CH\text{-}EW\ (UI)}$	0.425	Pass
Peak EH Offset from UI Center	±0.1 UI	$T_{RX\text{-}DS\text{-}OFFSET\ (UI)}$	-0.031	Pass
Range for DFE d_1 Coefficient	±30 mV	$V_{RX\text{-}DFE\text{-}COEFF\ (mV)}$		
Eye Mask		Eye Mask		Pass

Differential Insertion Loss

Item	Value	Simulation Results	Pass/Fail
Insertion Loss	Breakout Channel Only	SDD21 - Breakout	Fail
	Breakout + Short Calibration Channel	SDD21 - Short	Fail
	Breakout + Long Calibration Channel	SDD21 - Long	Pass

Differential Return Loss

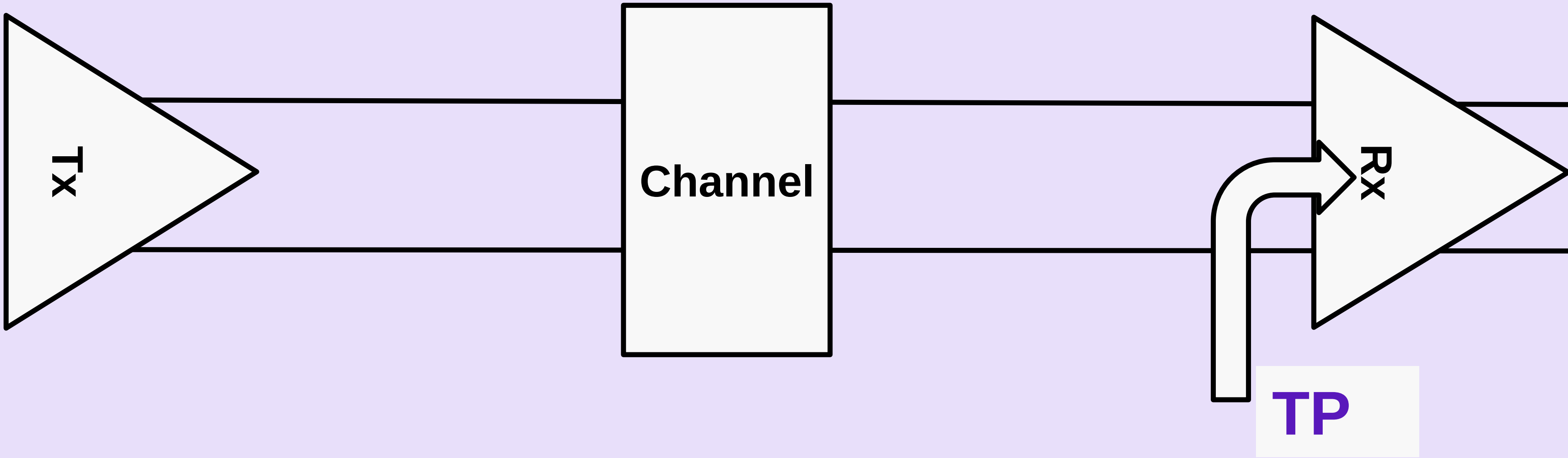
Item	Value	Simulation Results	Pass/Fail
Tx Return Loss		RL - Tx	Pass
Rx Return Loss		RL - Rx	Pass

Correlation Techniques

- Three level of correlations
  - ✓ Block by block correlations against transistor level simulations
  - ✓ Adaption tests against behavioral simulations
  - ✓ Tx to Rx link simulations and correlations against Hardware lab measurements

Correlation Results - 1

- Design
  - ✓ 16Gbps serial link-PCIe 4.0
  - ✓ 16 FinFET technology node

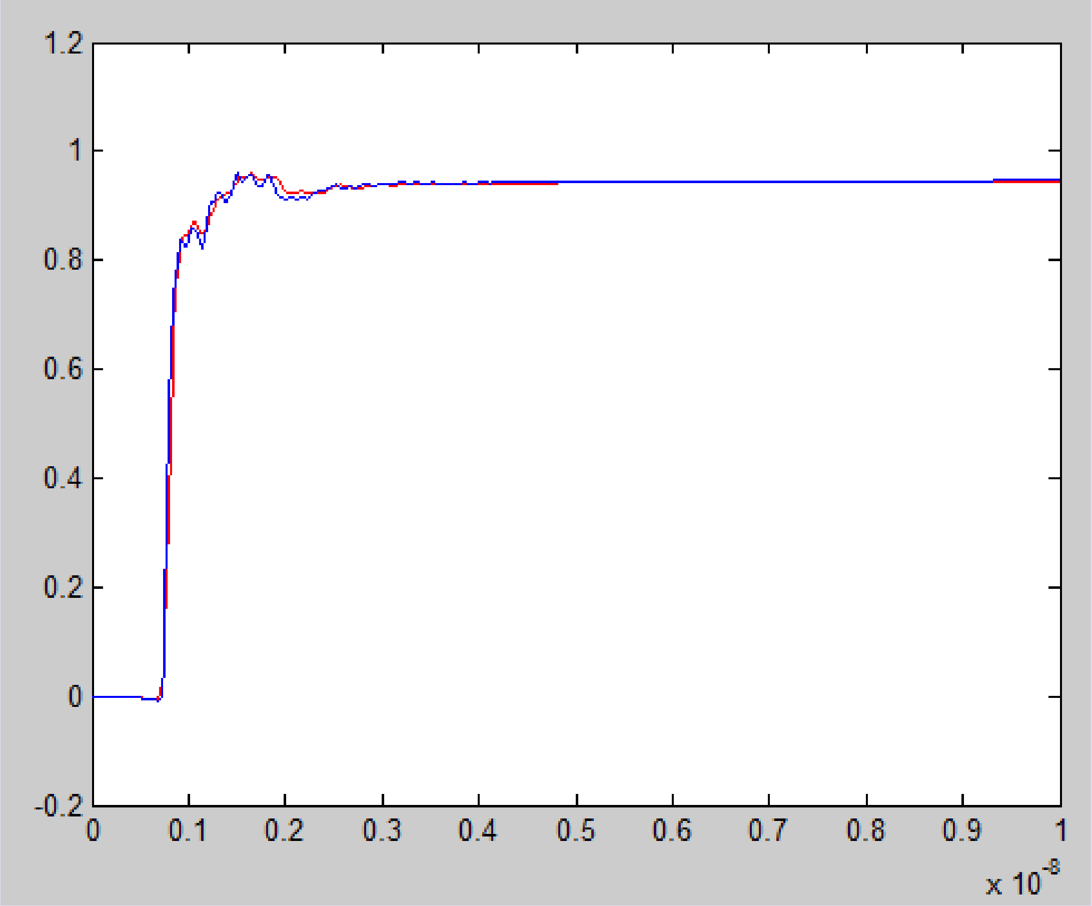


Correlation Results - 1

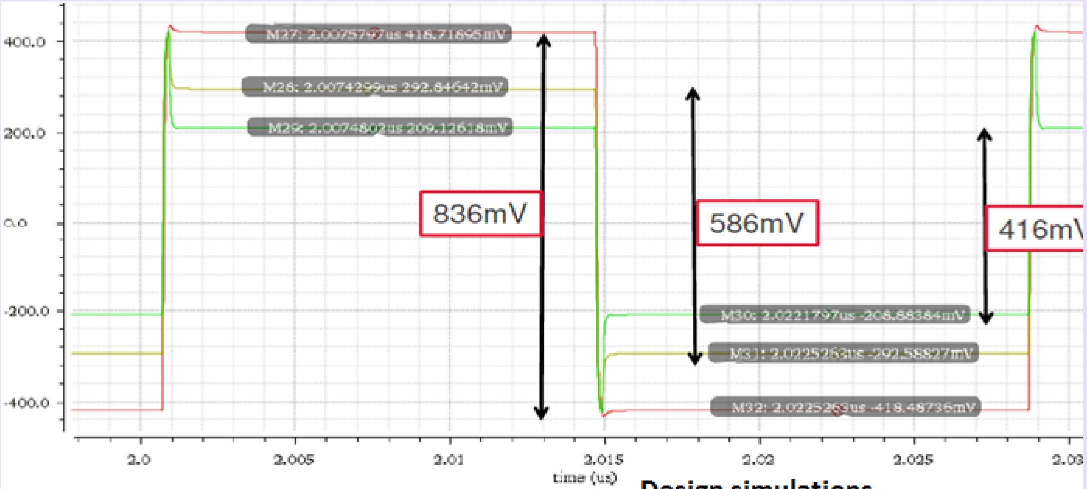
Testing Analog portion - Step response at TP

Red-Using S-params

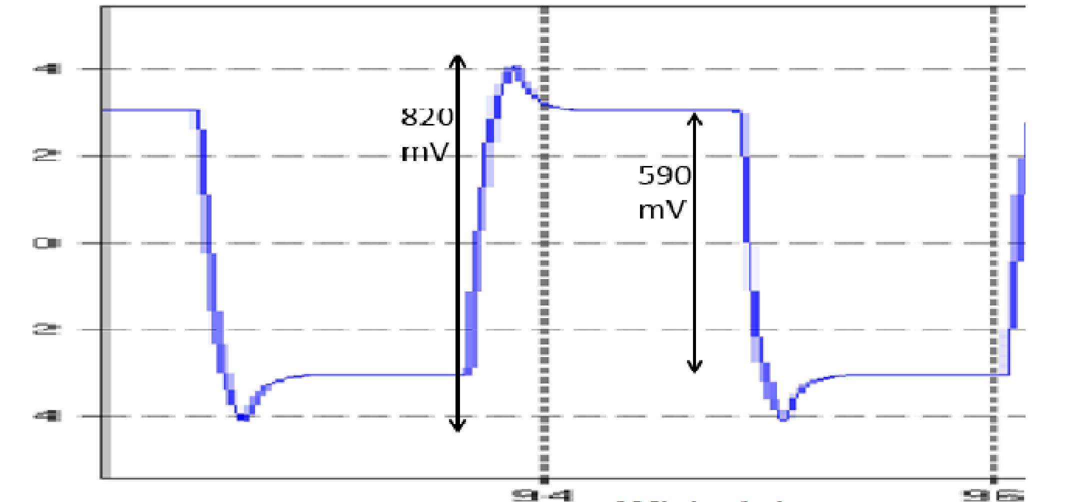
Blue-Design simulation



Correlation Results - 2



Design simulations

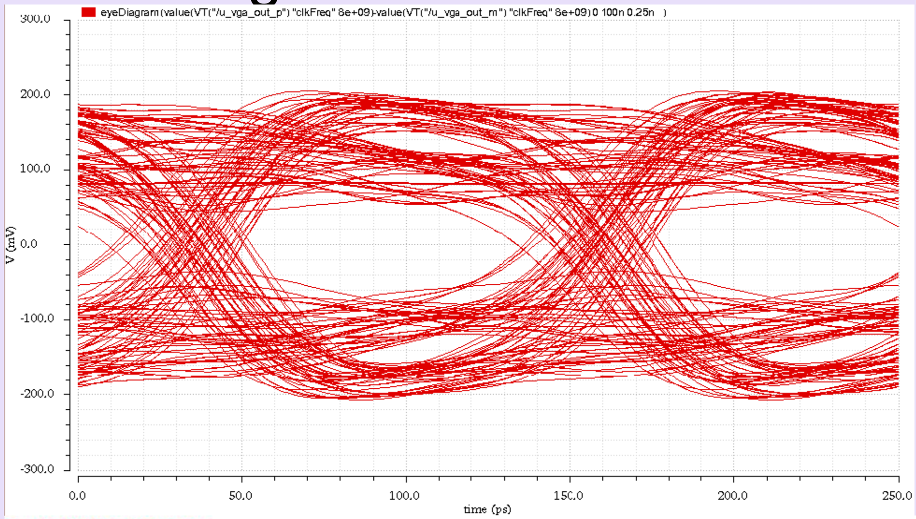


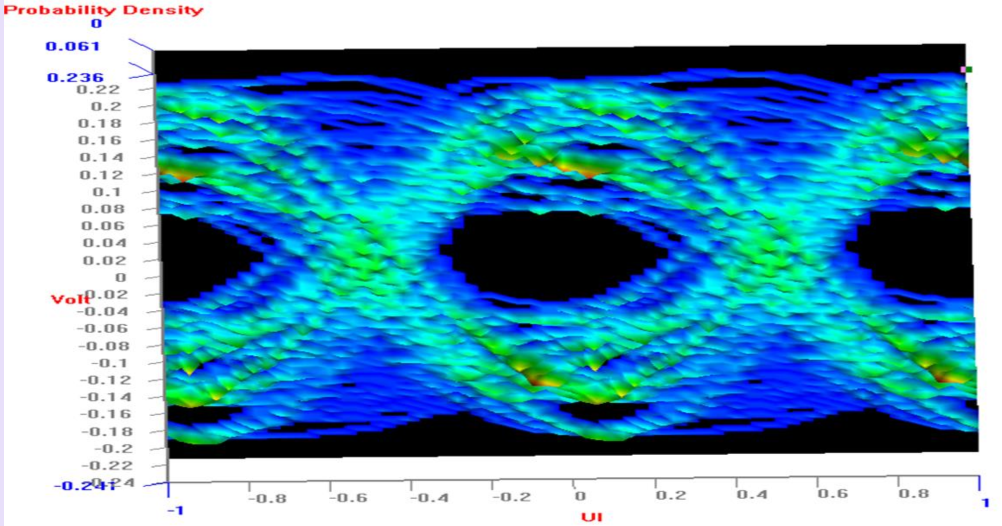
AMI simulations

- Testing pre-emphasis and boost

Correlation Results - 3

- Modelling non-linear VGA



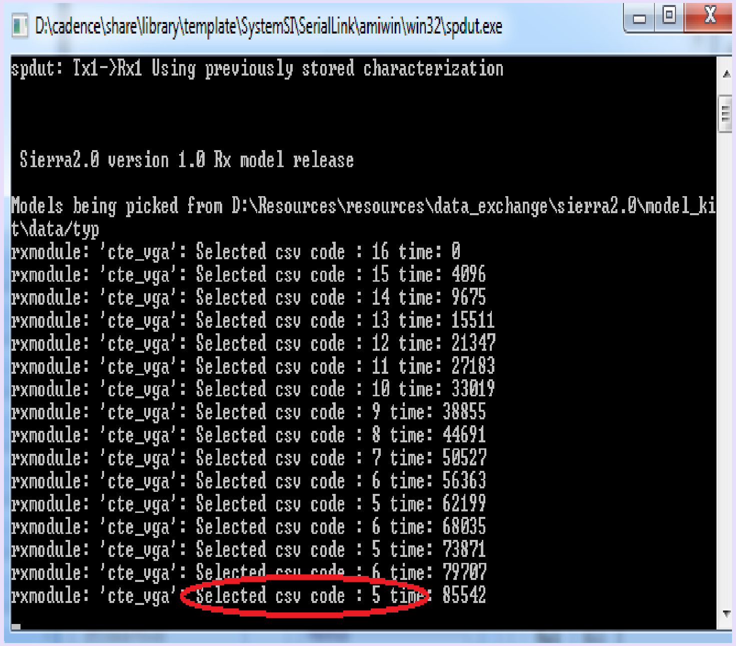


Design Simulation
Eye Amplitude:392mV
Eye Height:117mV
Jitterpp:0.35UI
800 bits

Model simulation
Eye Amplitude:409mV
Eye Height:115mV
Jitterpp:0.37UI
100,000 bits

Correlation Results - 4

- VGA adaptation
  - ✓ VGA codes 0 to 16 for different VGA settings



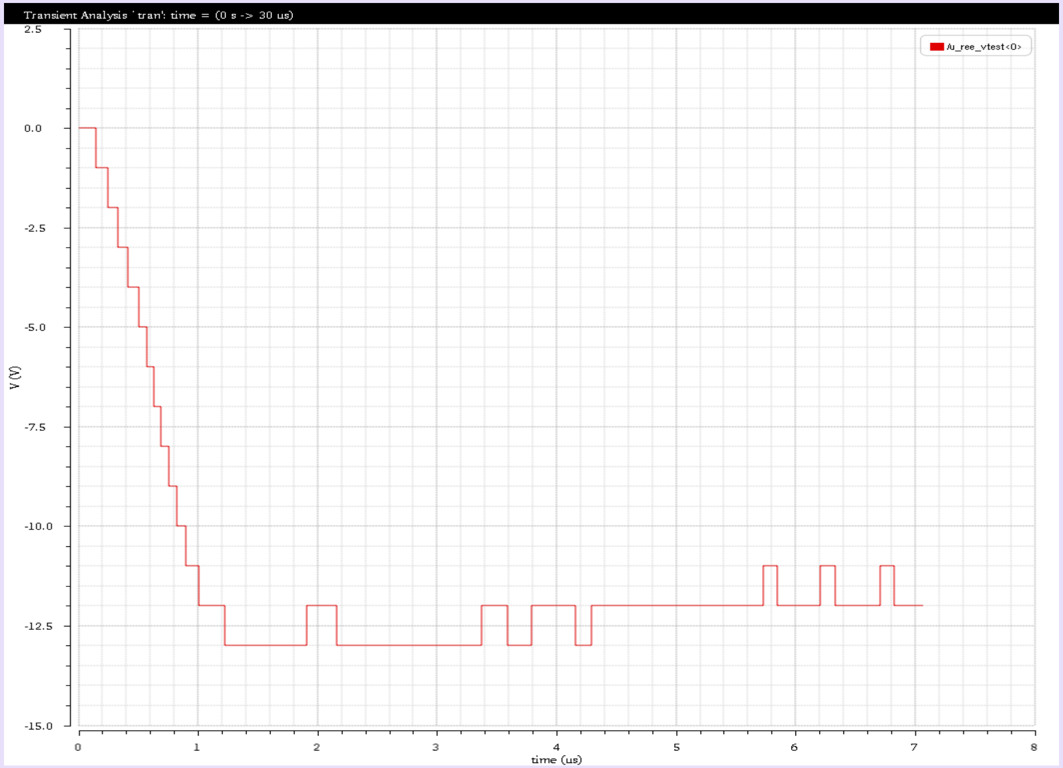
D:\cadence\share\library\template\SystemSI\SerialLink\amiwin\win32\spdut.exe

```
spdut: Tx1->Rx1 Using previously stored characterization



 Sierra2.0 version 1.0 Rx model release

Models being picked from D:\Resources\resources\data_exchange\sierra2.0\model_ki
t\data/typ
rxmodule: 'cte_vga': Selected csv code : 16 time: 0
rxmodule: 'cte_vga': Selected csv code : 15 time: 4096
rxmodule: 'cte_vga': Selected csv code : 14 time: 9675
rxmodule: 'cte_vga': Selected csv code : 13 time: 15511
rxmodule: 'cte_vga': Selected csv code : 12 time: 21347
rxmodule: 'cte_vga': Selected csv code : 11 time: 27183
rxmodule: 'cte_vga': Selected csv code : 10 time: 33019
rxmodule: 'cte_vga': Selected csv code : 9 time: 38855
rxmodule: 'cte_vga': Selected csv code : 8 time: 44691
rxmodule: 'cte_vga': Selected csv code : 7 time: 50527
rxmodule: 'cte_vga': Selected csv code : 6 time: 56363
rxmodule: 'cte_vga': Selected csv code : 5 time: 62199
rxmodule: 'cte_vga': Selected csv code : 6 time: 68035
rxmodule: 'cte_vga': Selected csv code : 5 time: 73871
rxmodule: 'cte_vga': Selected csv code : 6 time: 79707
rxmodule: 'cte_vga': Selected csv code : 5 time: 85542
```



Agenda

- Problem Statement
- Overview of IBIS-AMI Modeling
- Accurate Modeling of PCIe SerDes IO for 16Gbps or Higher Speeds
  - ✓ Our methodology
- Using Virtual Reference Design (VRD) Flow for Quick Sign-off
- Conclusion

Conclusions

- Need parameterized AMI blocks
  - ✓ To quickly build AMI models to mimic hardware
  - ✓ Allow electrical engineer designer to create AMI models who knows little programming

- Need a flow that allows rapid building of test-benches to cover various correlations at level of:
  - ✓ Transistor level simulations
  - ✓ Architecture level simulations
  - ✓ Hardware lab measurements

References

- http://www.eda.org/ibis/
- Eric Naviasky, “Defining a New High-Speed, Multi-Protocol SerDes Architecture for Advanced Nodes” <http://ip.cadence.com/knowledgecenter/resources/know-dip-wp>

Acknowledgements

- Taranjit Kukal
- Ritabrata Bhattacharya
- Steve Williams
- Eric Naviasky
- Ken Willis
- Kumar Keshavan
- Ambrish Varma
- Jasleen Ahuja

Thank you for attending the
PCI-SIG Developers Conference 2015.

For more information please go to
www.pcisig.com