



Leveraging IBIS Capabilities for Multi-Gigabit Interfaces

Ken Willis - Cadence Design Systems
IBIS Summit – EDI CON 2017
Boston, Massachusetts
September 13, 2017

Overview

- In writing EDI CON paper “Signal Integrity Methodology for Double-Digit Multi-Gigabit Interfaces”, different IBIS modeling techniques were applied
- Wanted to share some of the IBIS modeling methods we’ve been using

Agenda

- [External Model]
- AMI equalization adaptation
- Backchannel training
- Applying IBIS-AMI to DDR applications

Agenda

- **[External Model]**
- AMI equalization adaptation
- Backchannel training
- Applying IBIS-AMI to DDR applications

Preliminary IBIS Modeling

- Can get started with:
 - Voltage swing
 - Pad capacitance
 - Output impedance
 - Rise time
- Even easier to model using [External Model] syntax

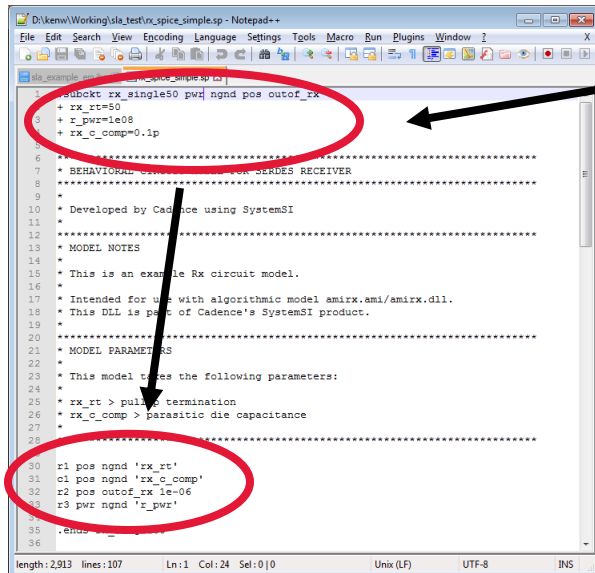
```

[Model]      pcie4_out
Model_type   Output
Polarity     Non-Inverting
Vmeas =      0.5
|
|
| variable          typ          min          max
C_comp        0.50pF          0.50pF          0.50pF
[Temperature Range]  20          80          -50
[Voltage range]     1.0          1.0          1.0
|
|*****
[Pulldown]
|
|          Voltage          I (typ)          I (min)          I (max)
|
|          | | | | |
|          | | | | | -1.0000e+00 -2.0000e-02 -2.0000e-02 -2.0000e-02
|          | | | | |  0.0000e-00  0.0000e-00  0.0000e-00  0.0000e-00
|          | | | | |  1.0000e+00  2.0000e-02  2.0000e-02  2.0000e-02
|
|*****
[Pullup]
|
|          Voltage          I (typ)          I (min)          I (max)
|
|          | | | | |
|          | | | | | -1.0000e+00  2.0000e-02  2.0000e-02  2.0000e-02
|          | | | | |  0.0000e-00  0.0000e-00  0.0000e-00  0.0000e-00
|          | | | | |  1.0000e+00 -2.0000e-02 -2.0000e-02 -2.0000e-02
|
|*****
[Ramp]
|
| variable          typ          min          max
dV/dt_r          0.60/0.020n  0.60/0.020n  0.60/0.020n
dV/dt_f          0.60/0.020n  0.60/0.020n  0.60/0.020n
|
R_load = 50
|

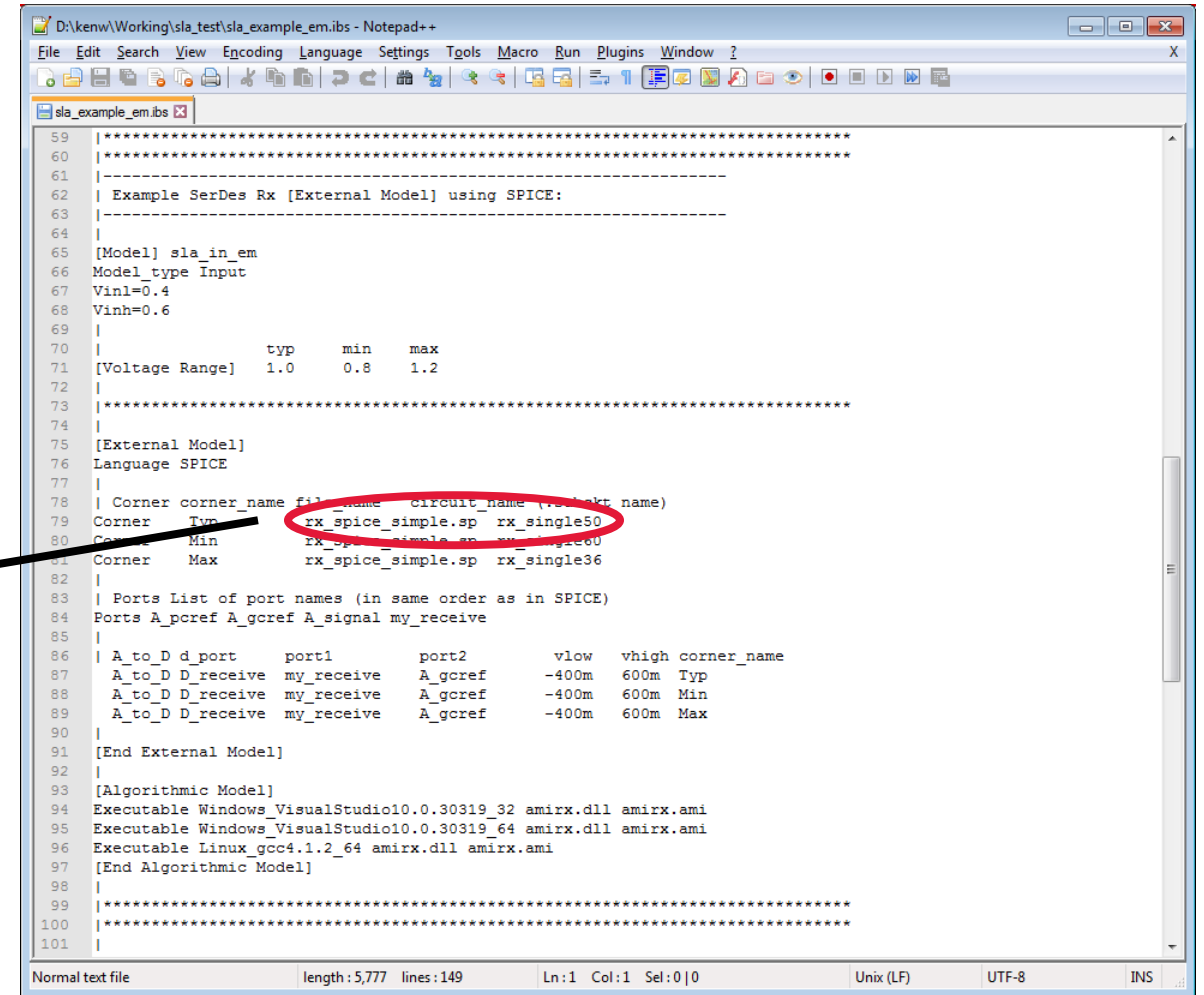
```

External Model

- Originally introduced in IBIS 4.1 !
- Enabled VHDL, Verilog-A, and Spice syntax to be used for buffer models instead of VI/VT table syntax
- Spice syntax has proven very useful to us



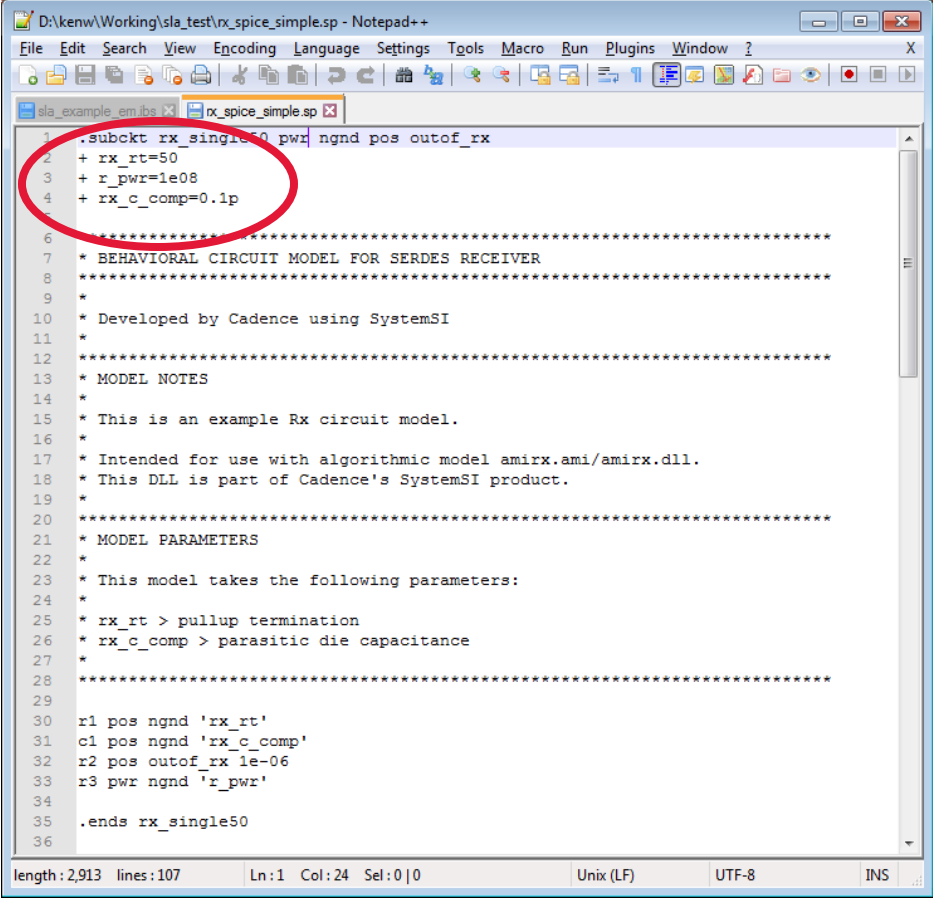
```
subckt rx_spice_simple pwr ngnd pos outof_rx
+ rx_rt=50
+ r_pwr=1e08
+ rx_c_comp=0.1p
.....
* BEHAVIORAL MODEL FOR SERDES RECEIVER
.....
* Developed by Cadence using SystemSI
.....
* MODEL NOTES
.....
* This is an example Rx circuit model.
.....
* Intended for use with algorithmic model amirx.amr/amirx.dll.
* This DLL is part of Cadence's SystemSI product.
.....
* MODEL PARAMETERS
.....
* This model uses the following parameters:
.....
* rx_rt > pullup termination
* rx_c_comp > parasitic die capacitance
.....
r1 pos ngnd 'rx_rt'
c1 pos ngnd 'rx_c_comp'
r2 pos outof_rx 1e-06
r3 pwr ngnd 'r_pwr'
```



```
.....
61 |-----
62 | Example SerDes Rx [External Model] using SPICE:
63 |-----
64 |
65 [Model] sla_in_em
66 Model_type Input
67 Vinl=0.4
68 Vinh=0.6
69 |
70 |          typ      min      max
71 [Voltage Range] 1.0      0.8      1.2
72 |
73 |-----
74 |
75 [External Model]
76 Language SPICE
77 |
78 | Corner corner_name file_name circuit_name (v.t.ckt name)
79 Corner Typ      rx_spice_simple.sp rx_single50
80 Corner Min      rx_spice_simple.sp rx_single60
81 Corner Max      rx_spice_simple.sp rx_single36
82 |
83 | Ports List of port names (in same order as in SPICE)
84 Ports A_poref A_gcref A_signal my_receive
85 |
86 | A_to_D_d_port  port1      port2      vlow  vhigh corner_name
87 A_to_D_D_receive my_receive A_gcref  -400m 600m Typ
88 A_to_D_D_receive my_receive A_gcref  -400m 600m Min
89 A_to_D_D_receive my_receive A_gcref  -400m 600m Max
90 |
91 [End External Model]
92 |
93 [Algorithmic Model]
94 Executable Windows_VisualStudio10.0.30319_32 amirx.dll amirx.amr
95 Executable Windows_VisualStudio10.0.30319_64 amirx.dll amirx.amr
96 Executable Linux_gcc4.1.2_64 amirx.dll amirx.amr
97 [End Algorithmic Model]
98 |
99 |-----
100 |-----
101 |
```

Spice [External Model]s Convenient for Early Feasibility

- Sometimes the IBIS model you want is not available for preliminary / pre-design analysis
- Easy to write (or use) simple parameterized Spice subcircuits for IO buffers when IBIS availability does not align with your project schedule
- Can sweep parameters and cover a big portion of the design space quickly and easily



```
D:\kenw\Working\sla_test\rx_spice_simple.sp - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
rx_spice_simple.sp
1 .subckt rx_single50 pwr ngnd pos outof_rx
2 + rx_rt=50
3 + r_pwr=1e08
4 + rx_c_comp=0.1p
5
6 *****
7 * BEHAVIORAL CIRCUIT MODEL FOR SERDES RECEIVER
8 *****
9 *
10 * Developed by Cadence using SystemSI
11 *
12 *****
13 * MODEL NOTES
14 *
15 * This is an example Rx circuit model.
16 *
17 * Intended for use with algorithmic model amirx.aml/amirx.dll.
18 * This DLL is part of Cadence's SystemSI product.
19 *
20 *****
21 * MODEL PARAMETERS
22 *
23 * This model takes the following parameters:
24 *
25 * rx_rt > pullup termination
26 * rx_c_comp > parasitic die capacitance
27 *
28 *****
29
30 r1 pos ngnd 'rx_rt'
31 c1 pos ngnd 'rx_c_comp'
32 r2 pos outof_rx 1e-06
33 r3 pwr ngnd 'r_pwr'
34
35 .ends rx_single50
36
length: 2,913 lines: 107 Ln:1 Col:24 Sel:0|0 Unix (LF) UTF-8 INS
```

Analog Buffer Model Extensions

- Incorporated into IBIS 6.0

```
[External Model]
Language IBIS-ISS
|
| Corner corner_name file name circuit_name (.subckt name)
Corner    Typ    cdnstxrx.cir    tx_subckt
|
| List of parameters
Parameters TSFile1      = cdns_tx.param(CDNS_Tx(Model_Specific(Tstonefile)))
Parameters Tx_Rseries   = cdns_tx.param(CDNS_Tx(Model_Specific(Tx_R)))
|
| List of converter parameters
Converter_Parameters Vtx_h = cdns_tx.param(CDNS_Tx(Model_Specific(Tx_V)))
|
| Ports List of port names (in same order as in SPICE)
Ports A_signal_pos A_signal_neg my_driveP A_pdref A_puref
|
| D_to_A d_port port1 port2 vlow which trise tfall corner_name polarity
D_to_A D_drive my_driveP A_pdref 0 Vtx_h 0.05n 0.03n Typ
|
|
[End External Model]
```

Tx/Rx Model
described in
external file

- Replaces the VI/VT curves
- Can now be parameterized when Language 'IBIS-ISS' is used

Parameter
Definition is in a
separate file

- Similar to AMI Parameters
- Can be in .ami or any other file
- User Selects Parameter Values from GUI

Parameter
Value is passed
to Simulator

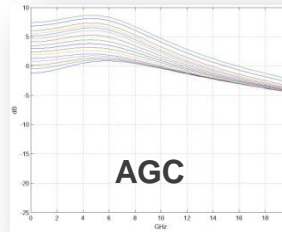
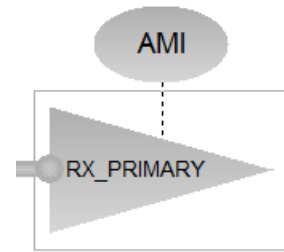
- Using .param, the parameter value is passed to the simulator
- Converter_Parameters are used appropriately in the stimulus/D_to_A

Agenda

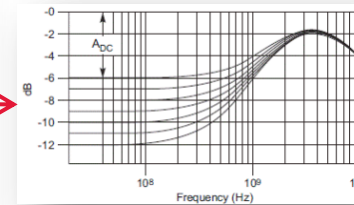
- [External Model]
- **AMI equalization adaptation**
- Backchannel training
- Applying IBIS-AMI to DDR applications

Anatomy of a Receiver AMI Model

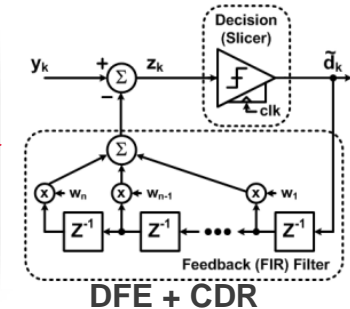
- These different modules typically adapt at different rates
- Initial modules usually adapt more slowly than later ones



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



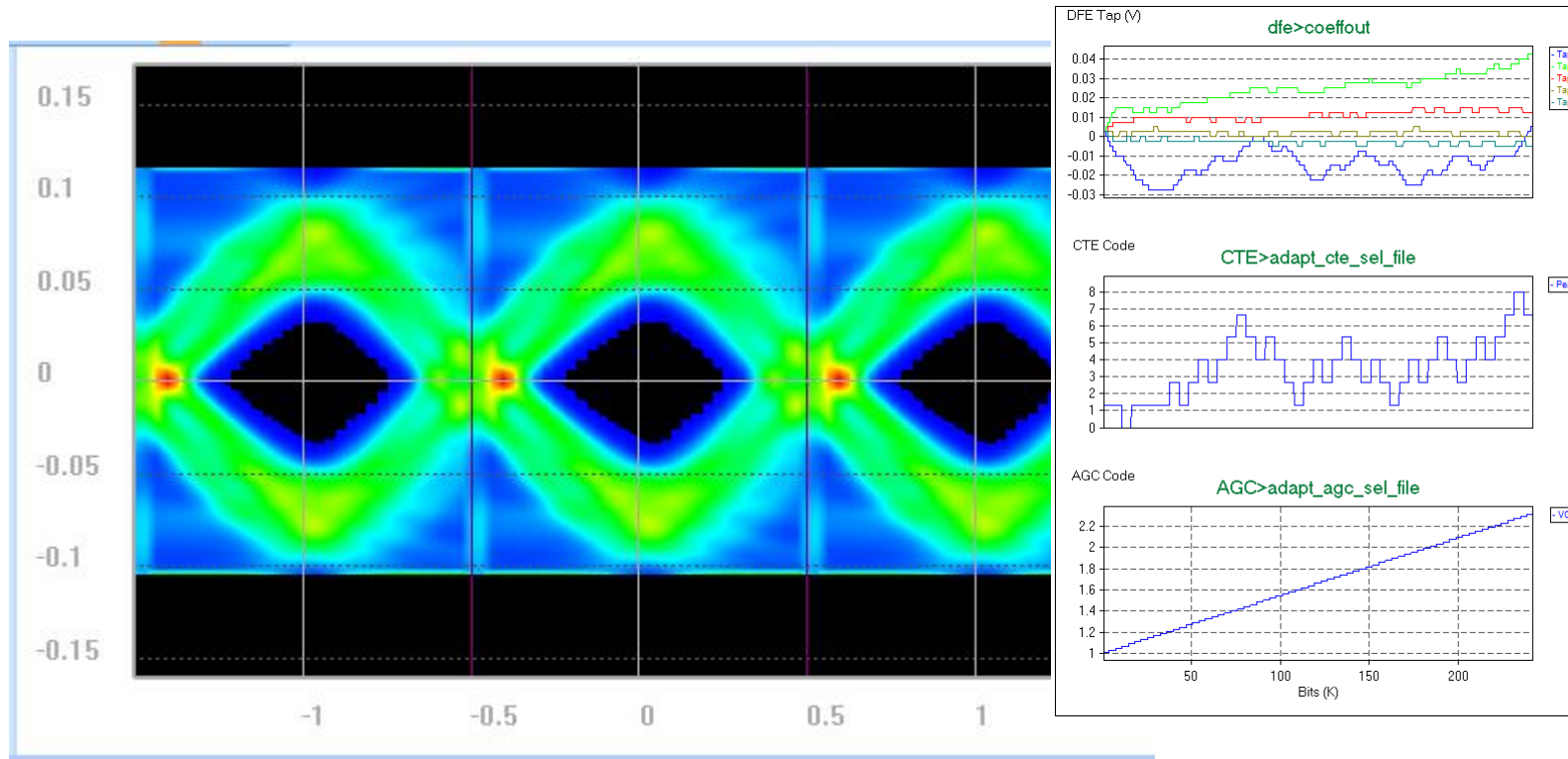
- Selective boosting at frequency of interest (high freq channel loss cancellation)
- Better for Area/Pwr considerations
- Noise and xtalk also gets boosted.
- Usually first order filter only



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

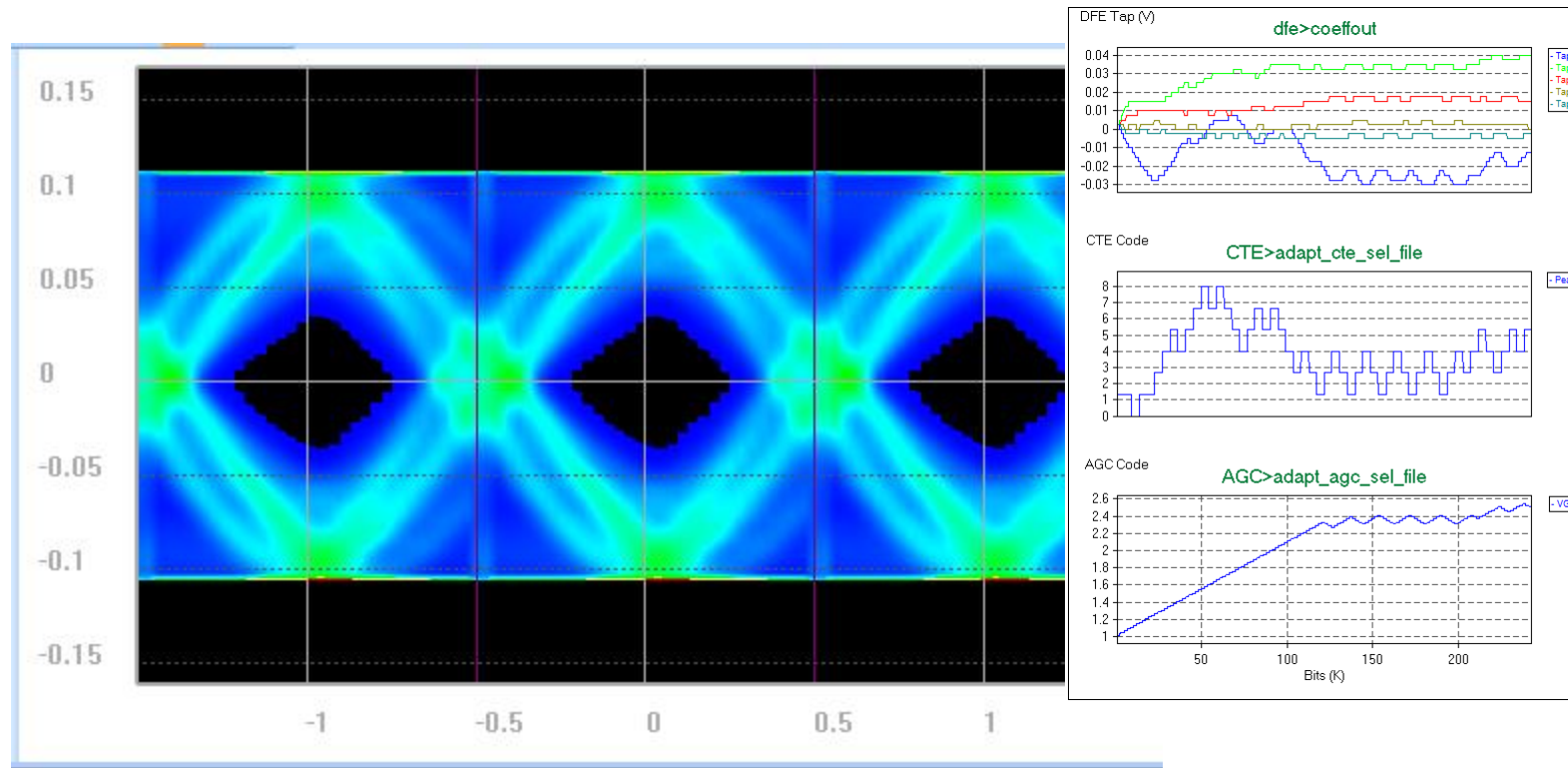
Rx With Default Adaptation

- Note that adaptation coefficients don't converge



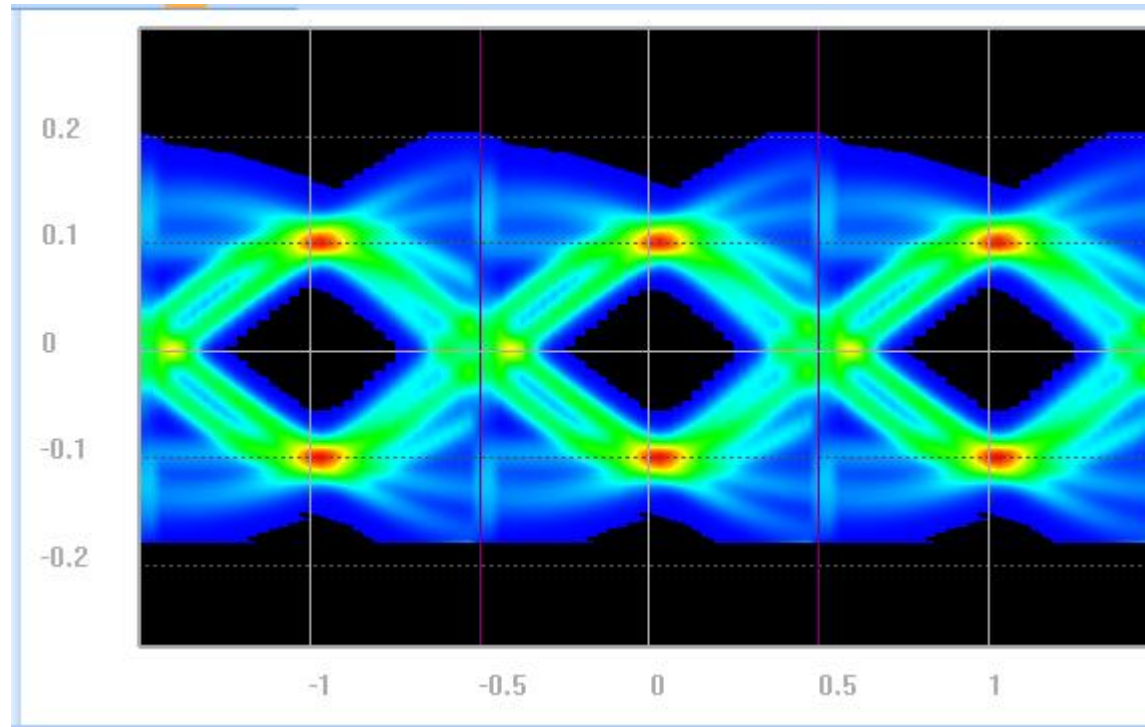
With Faster AGC Adaptation

- Coefficients converge, but after 150k bits of traffic are passed



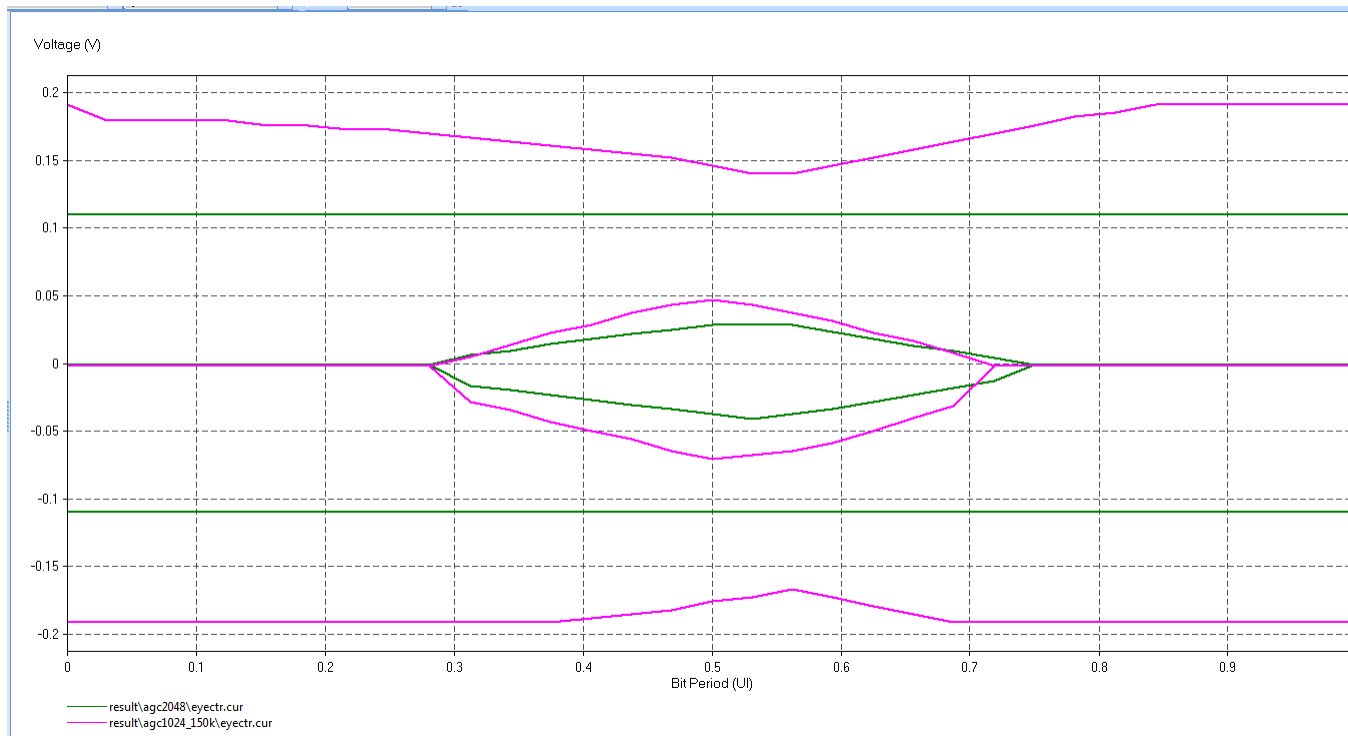
Ignoring the First 150k Bits

- Default was to ignore the first 40k bits
- Eliminates the noise from before coefficients converged
- Very important to be able to visualize how the adaptation is converging



Original Eye Contour vs. Final

- Adjusting AGC adaptation time and Ignore_Bits made a significant difference in eye height

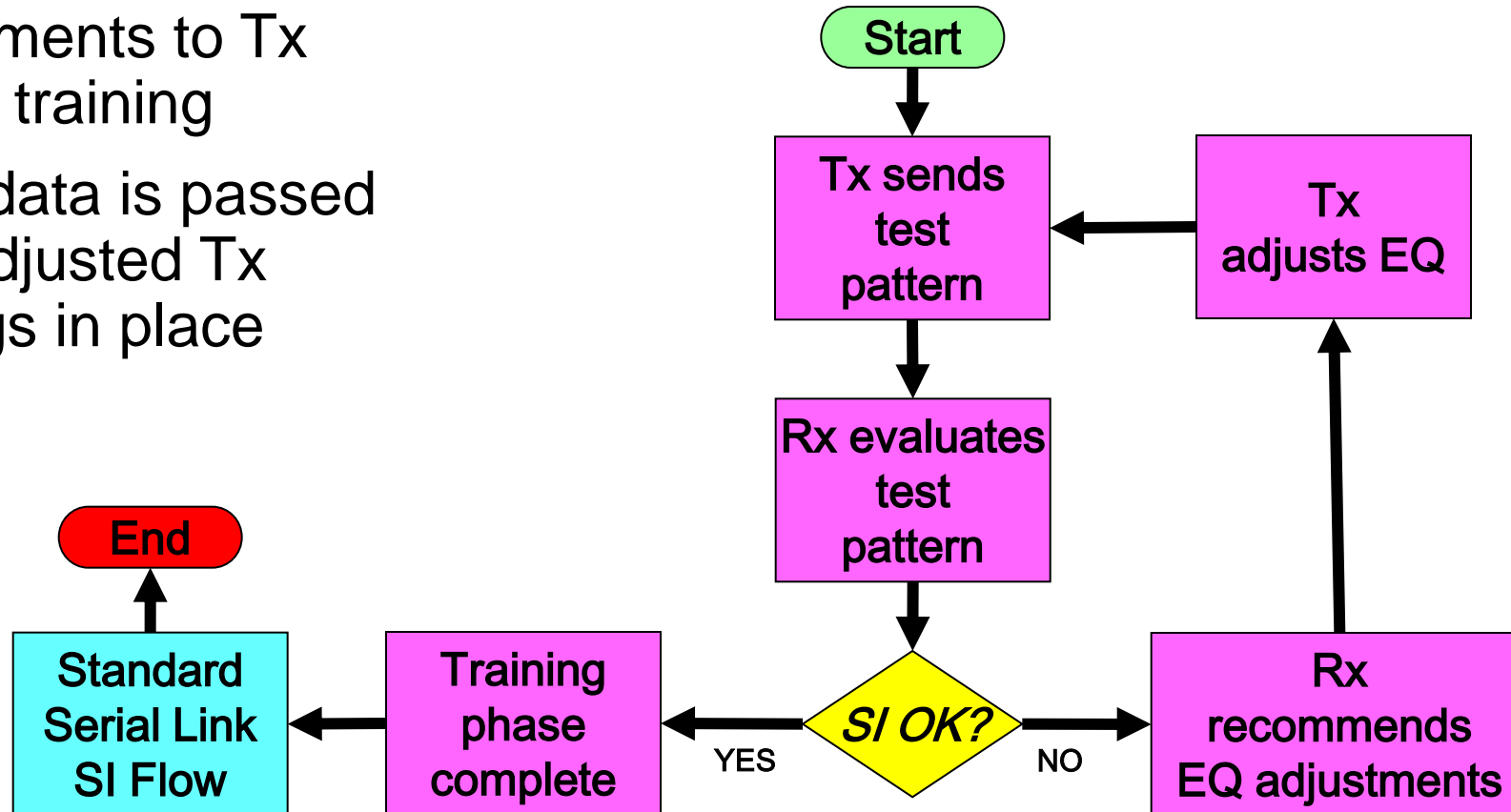


Agenda

- [External Model]
- AMI equalization adaptation
- **Backchannel training**
- Applying IBIS-AMI to DDR applications

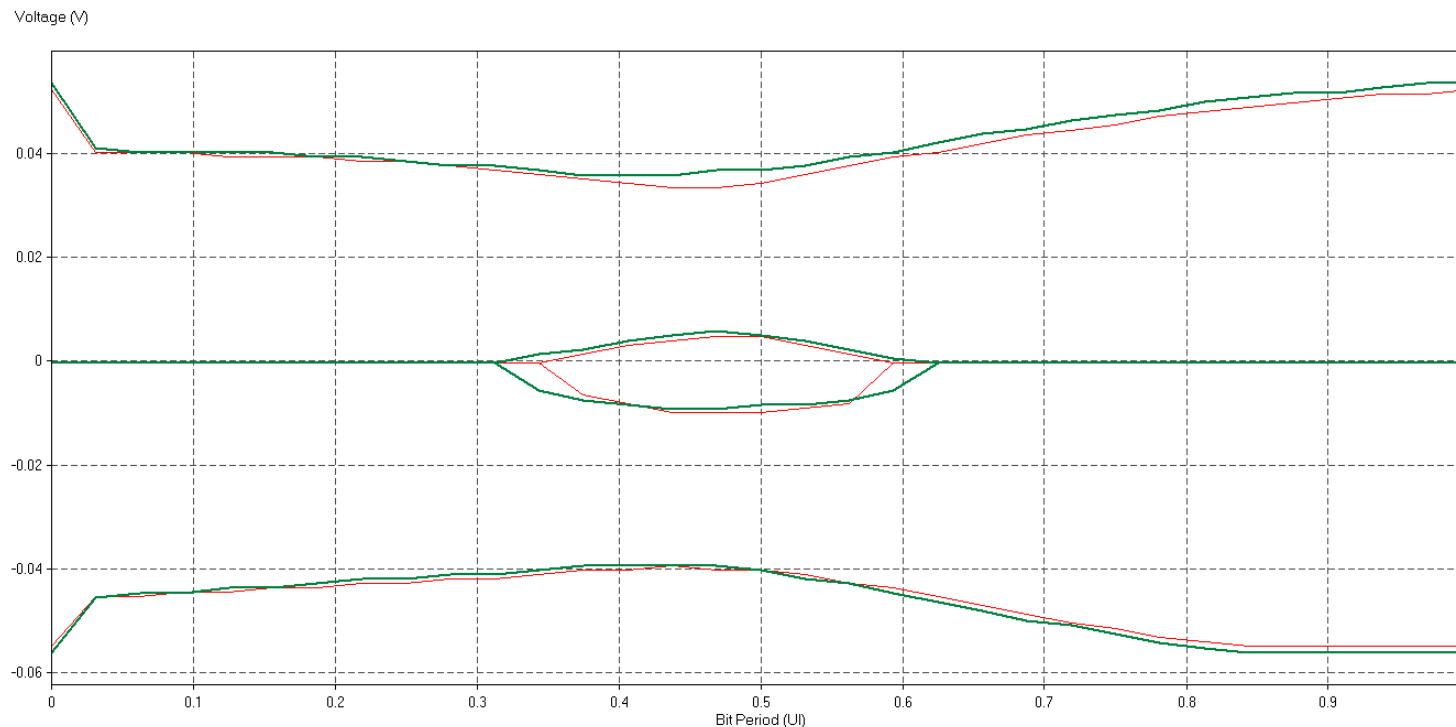
Backchannel Training (IBIS BIRD 147, in IBIS 7.0)

- Rx feeds back EQ adjustments to Tx during training
- Then data is passed with adjusted Tx settings in place



With and Without Backchannel

- Backchannel turned down Tx FFE settings somewhat
- Leaves more “heavy lifting” to Rx and its advanced adaptation
- Improves overall signal quality significantly

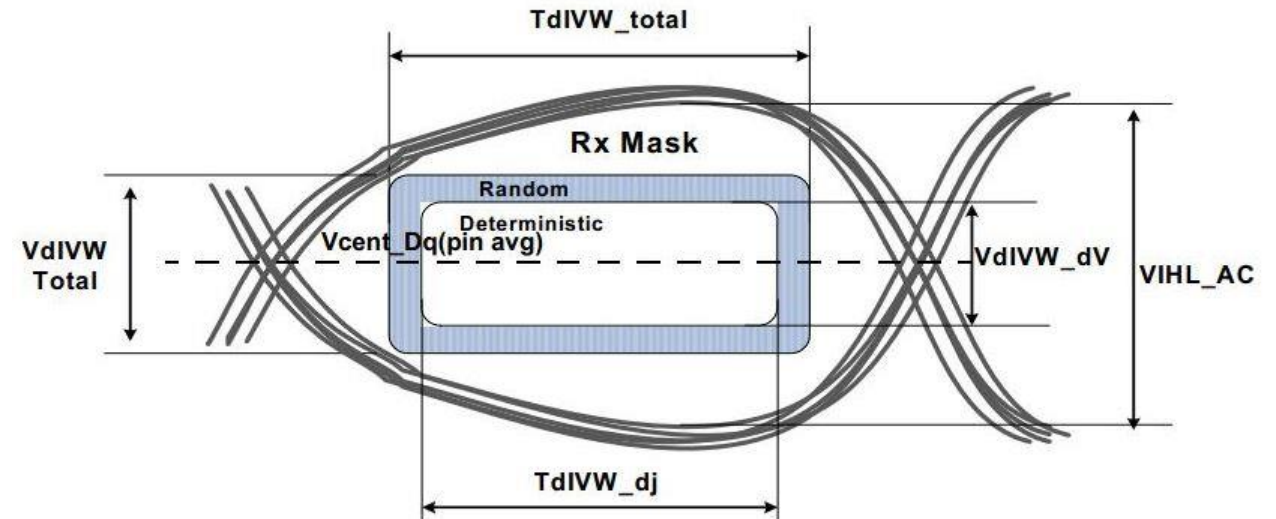


Agenda

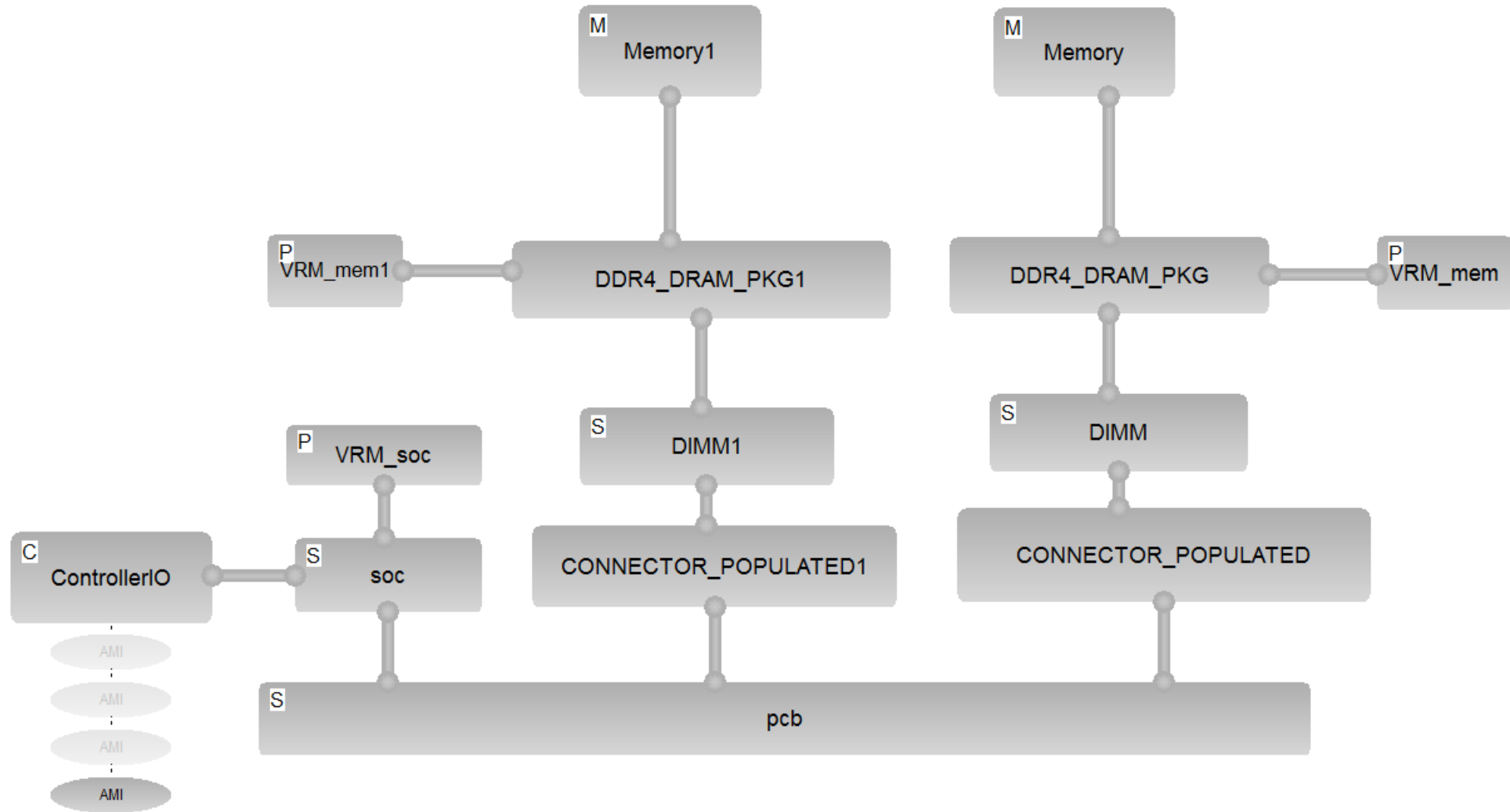
- [External Model]
- AMI equalization adaptation
- Backchannel training
- **Applying IBIS-AMI to DDR applications**

DDR4 Brought Some New Requirements

- Specified DQ mask compliance checking at a particular BER
- BER analysis requires extrapolation (bathtubs)
- Extrapolation requires a lot of traffic to be passed (need a lot of samples)
 - Channel simulation can be applied
- Started to see equalization used at Controller side
 - AMI modeling can be applied
- Worked with IP division to develop AMI model for DDR4 IP



Simulation Testbench

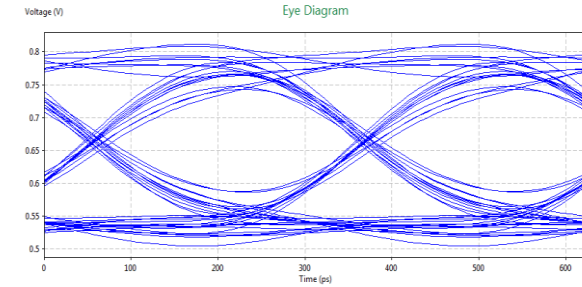
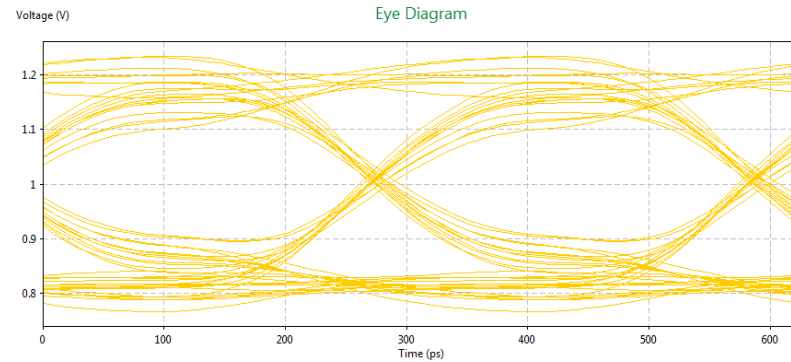


CTLE Correlation: 3200Mbps

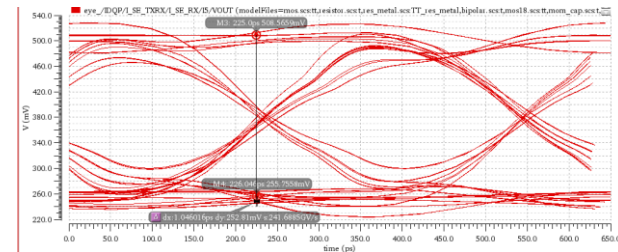
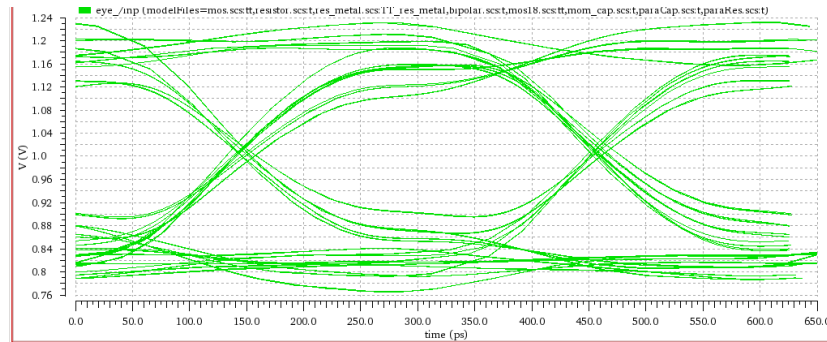
Input of receiver @pad

Output of CTLE

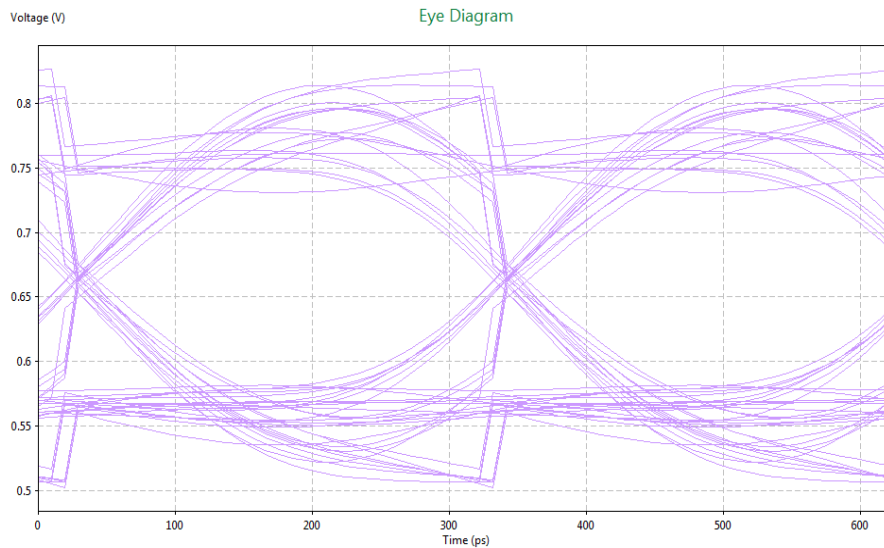
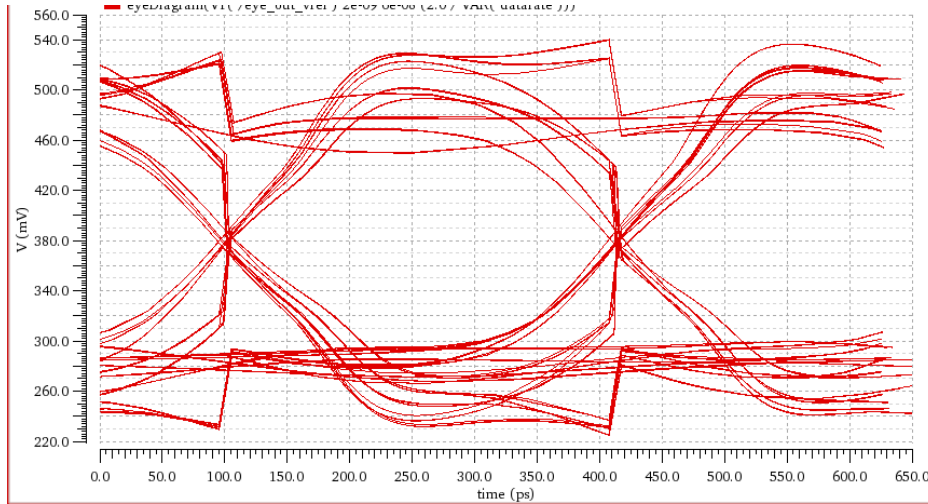
IBIS-AMI
channel sim



Transistor-
Level
circuit sim



CTLE + DFE Correlation: 3200Mbps



Summary

- External Model syntax can be very useful for pre-design modeling, when detailed IBIS models are not available, and has had some recent additions in capability
- Building IBIS-AMI models is not the obstacle it used to be
- Adaptive equalization often has interplay between multiple sub-modules in real devices, and therefore also in AMI models
- If adaptive, understand if your EQ coefficients converge during simulation
- Backchannel training enables interplay between the Tx and Rx in simulation, and can produce more realistic results for devices that use backchannel
- Channel simulation and AMI modeling has been successfully applied to DDR4 IP (and more of this is expected with DDR5)

cā dence®

© 2017 Cadence Design Systems, Inc. All rights reserved worldwide. Cadence, the Cadence logo, and the other Cadence marks found at www.cadence.com/go/trademarks are trademarks or registered trademarks of Cadence Design Systems, Inc. All other trademarks are the property of their respective owners.