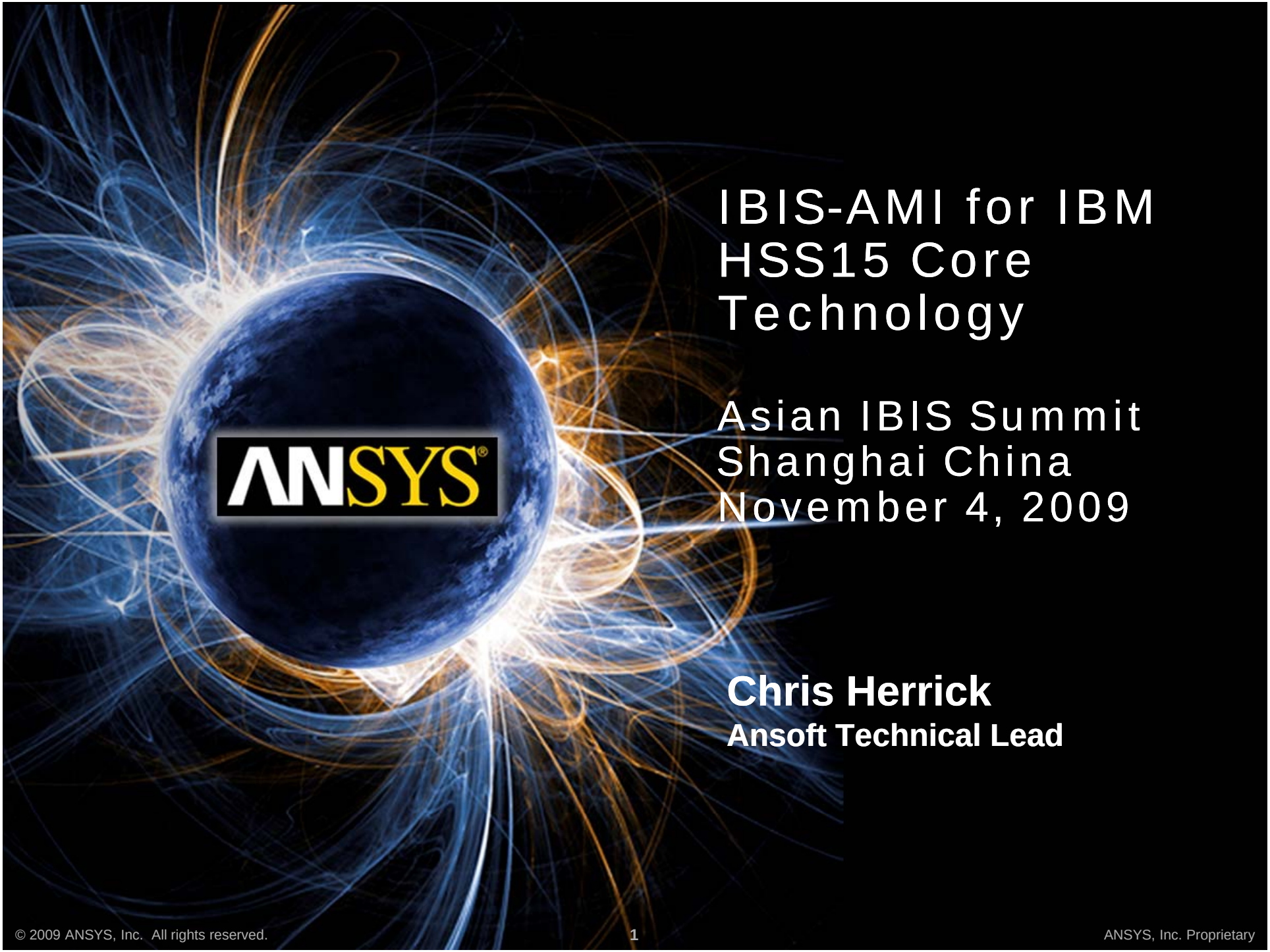


IBIS-AMI for IBM HSS15 Core Technology

Asian IBIS Summit
Shanghai China
November 4, 2009

Chris Herrick
Ansoft Technical Lead

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Agenda

- **AMI Standards Overview**
- **AMI Usage**
- **Design Kit Overview**
- **HSSCDR Comparison**
- **Case Studies**
- **Conclusions**

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

SI Today

- **High Speed Serial channels are pushing the current limits of simulation. Models/Simulator need to handle current challenges**
  - Need to accurately handle very high data rates
  - Simulate large number of bits to achieve low BER
  - Non-linear ,Time Variant Systems
  - TX/RX equalization and vendor specific device settings
  - Specific Data patterns and coding schemes
  - All types of jitter: RJ, DJ, UJ, PJ, etc.
  - XTLK
  - Clock Data Recovery circuits
  - TX and RX may come from different vendors

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Comparison of Simulations

Analysis Method	Advantages	Disadvantages
Traditional IBIS	Fast	Not accurate
Transistor Level	Potentially Accurate Handles Non-Linear	Very Slow No Rx Eq IP Liability Not interoperable
Fast Convolution	Very Fast Handles EQ Includes Bit Patterns	Not Silicon Specific LTI Assumption
Statistical	Very Fast Handles EQ	Not Silicon Specific No Bit Patterns LTI Assumption
IBIS-AMI	Fast Handles Vendor EQ Includes Bit Patterns Not LTI limited	Implementations vary

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

IBIS AMI

- **AMI stands for Algorithmic Modeling Interface**
- **It allows users to specify their own transmitter and receiver models as C-interface compiled libraries**
  - faster signal processing algorithms
  - intellectual property protection
- **Mainly used in convolution transient engines for channel simulation**
  - Designed to be used with fixed time step data
- **Introduced in IBIS 5.0 specs**
  - http://eda.org/pub/ibis/ver5.0/ver5_0.txt
  - IBIS stand for “I/O Buffer Information Specification”; high-level buffer specification for circuit modeling
  - In these specs the library is specified inside the IBIS wrapper and the interface is called IBIS-AMI
  - In fact, AMI concept is independent of IBIS

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

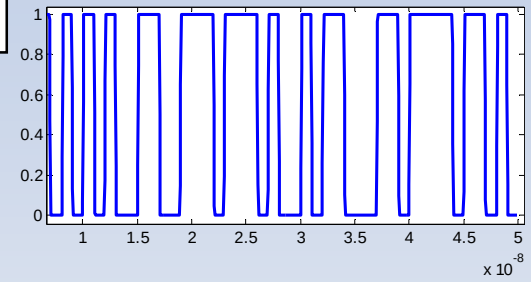
Data flow

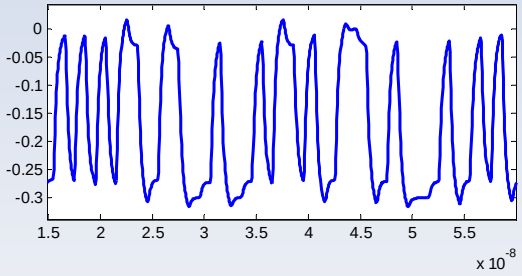
AMI Transmitter (Tx)

Linear Channel

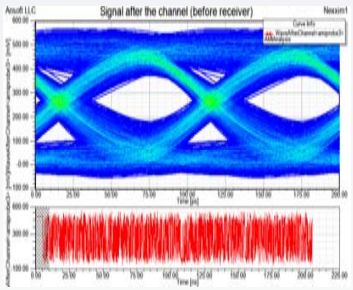
AMI Receiver (Rx)

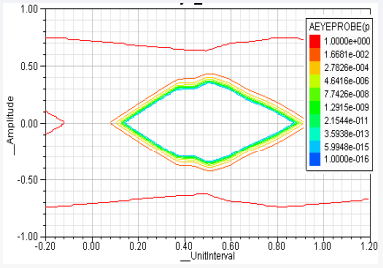
Engine post-processing

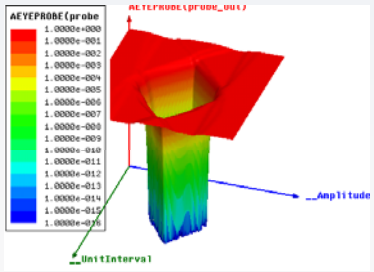




- **Very similar to Fast Convolution**
  - No backward dependency
- **Transmitter and receiver are based on user supplied libraries.**
- **Channel is characterized with impulse response function(s)**
- **All signals are sampled with the constant time step and handled in blocks.**







© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

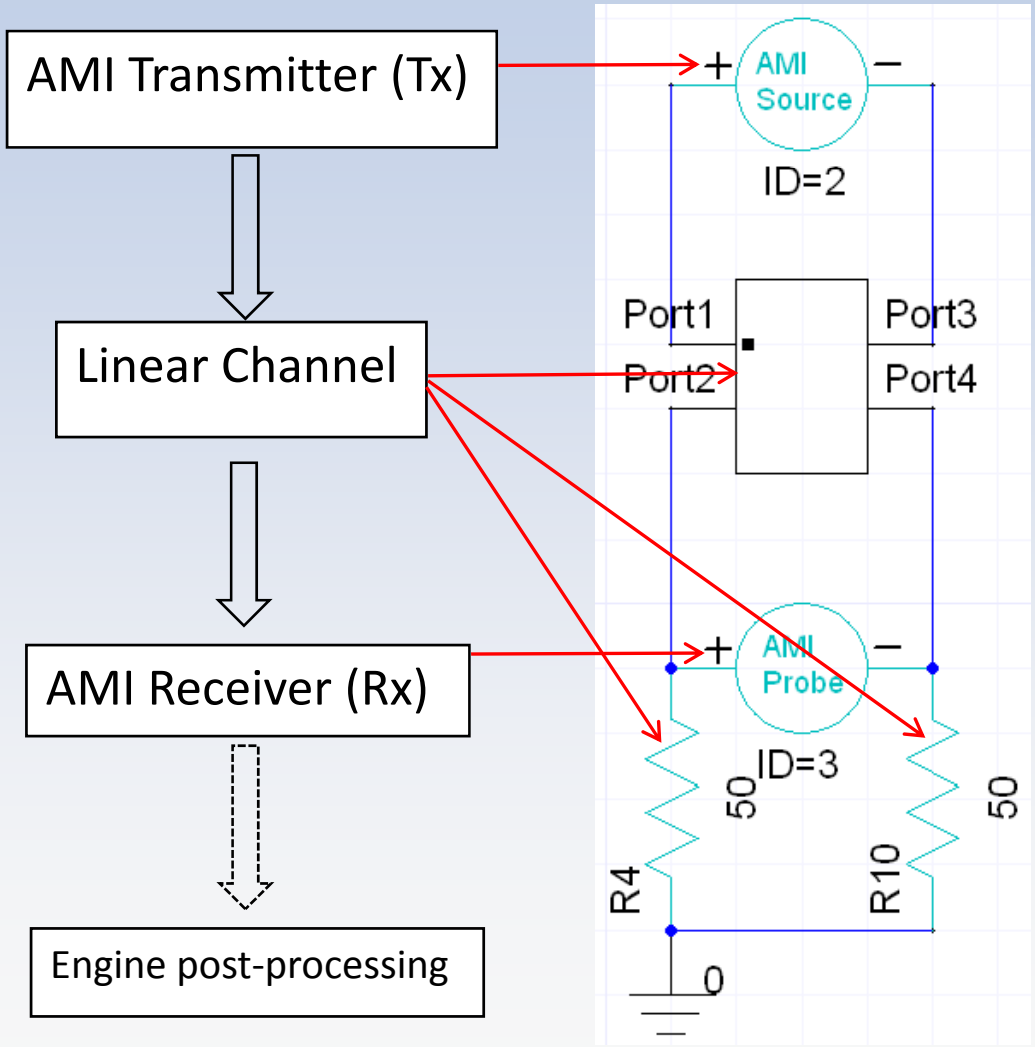
AMI Analysis flow

- **Compute impulse response of the channel (as well of aggressor impulse responses)**
  - Regular transient analysis
- **Initialize AMI libraries with the channel response**
  - Libraries can modify impulse responses
- **In a cycle**
  - Generate a block of transmitted bits
  - Convert a list of bits into a piecewise-linear rise-fall signal
  - Push the signal through the transmitter
  - Convolve the signal with the channel
  - Push the signal through the receiver
  - Post-process the results (eye and bit-error-rate plots)

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Simple Circuit Schematic



- **Random number generator and jitter added externally to AMI Tx model**
- **In this example, linear channel is represented as 4-port s-parameter block**
- **Additional linear parts of transmitter and receiver are treated as a part of the channel**

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

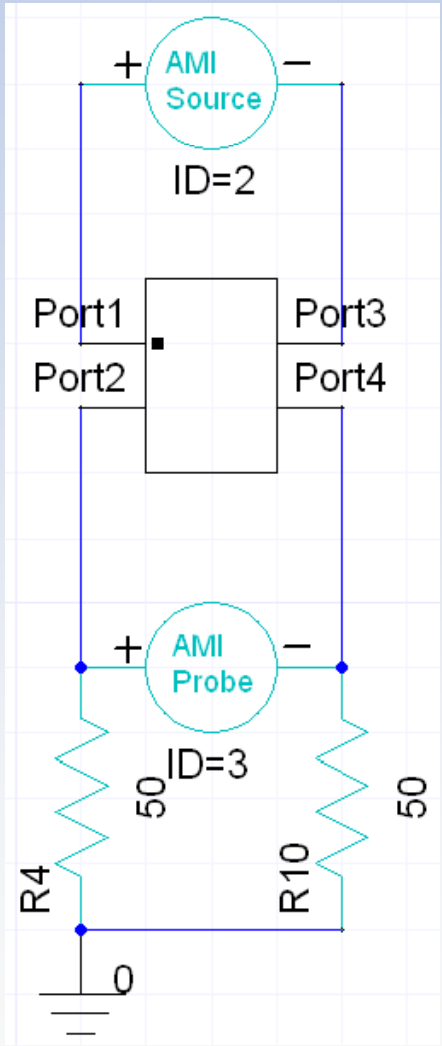
Inside AMI devices

- **AMI libraries define three (optional) functions with C interface**
- **AMI_Init()**
  - Impulse response for channel and aggressors
    - Can modify channel impulse response
  - Bit period (unit interval)
  - Time step
  - Number of aggressors
  - Whatever information user puts in .ami parameters text file
- **AMI_GetWave()**
  - Called for each block of incoming data
  - Input signal (wave) for the channel and aggressors
  - Can generate additional clock tics if the user needs to
- **AMI_Close()**

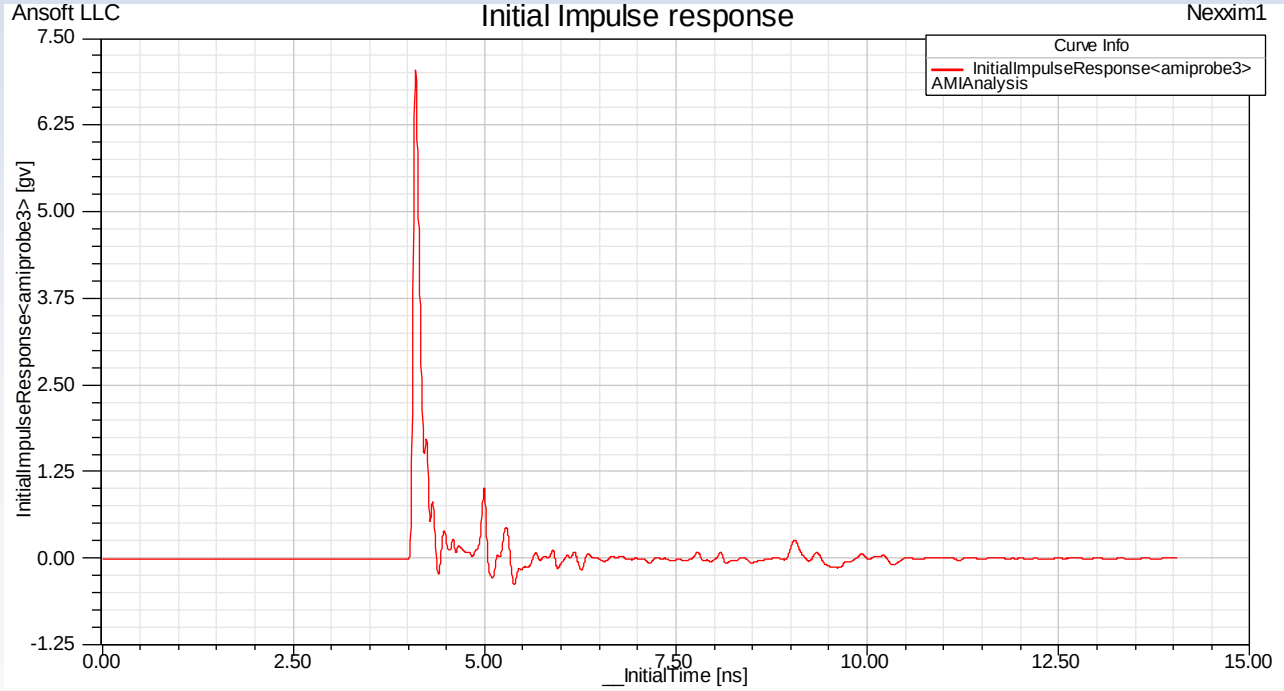
© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Generating Impulse Response



- **AMI source becomes an “impulse source”**
- **AMI probe becomes an empty device**
- **Full-scale transient analysis**
  - All transient options are valid

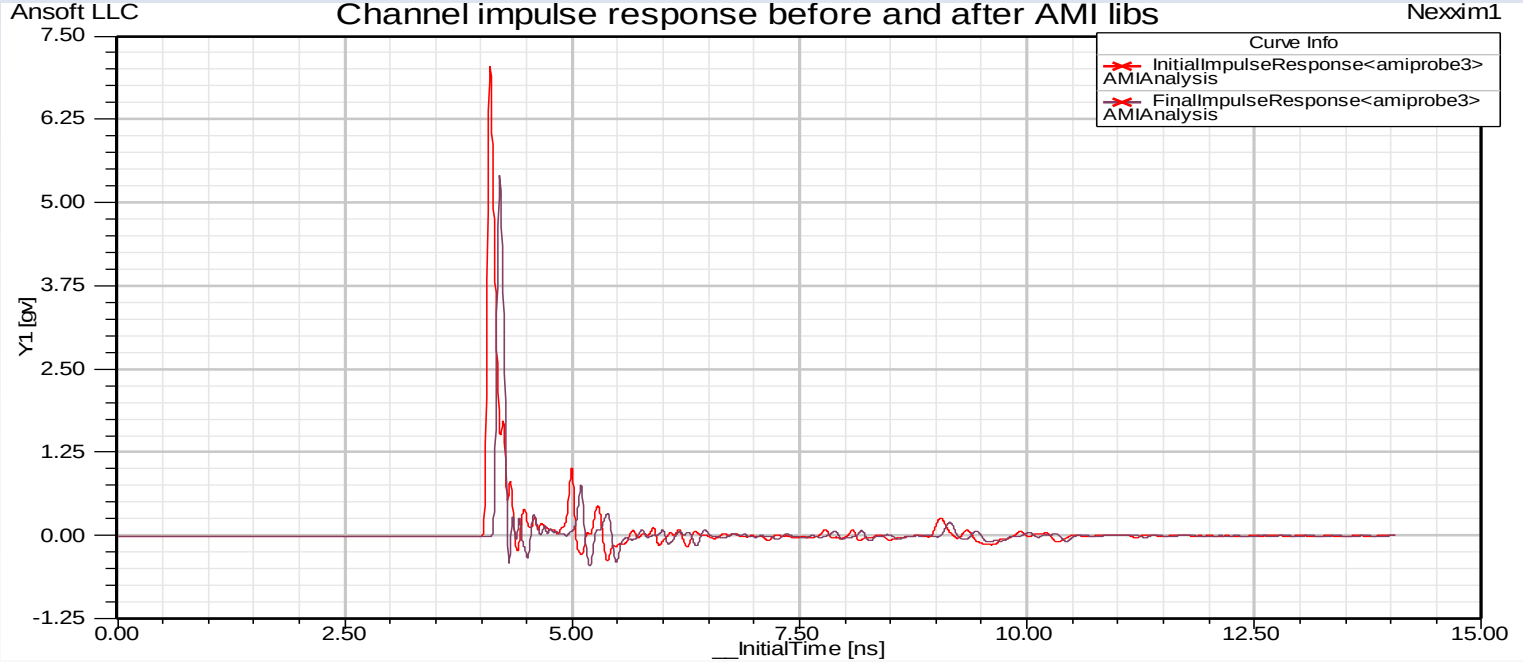


© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Initializing AMI devices with channel impulse response

- **AMI model receives impulse response and some other parameters**
  - Bit period, time step
  - Whatever text info user keeps in .ami file
  - If there are aggressors, AMI devices have access to their impulse responses too
- **AMI devices can modify the impulse response of the channel**
  - It could be modified on the initialization stage only

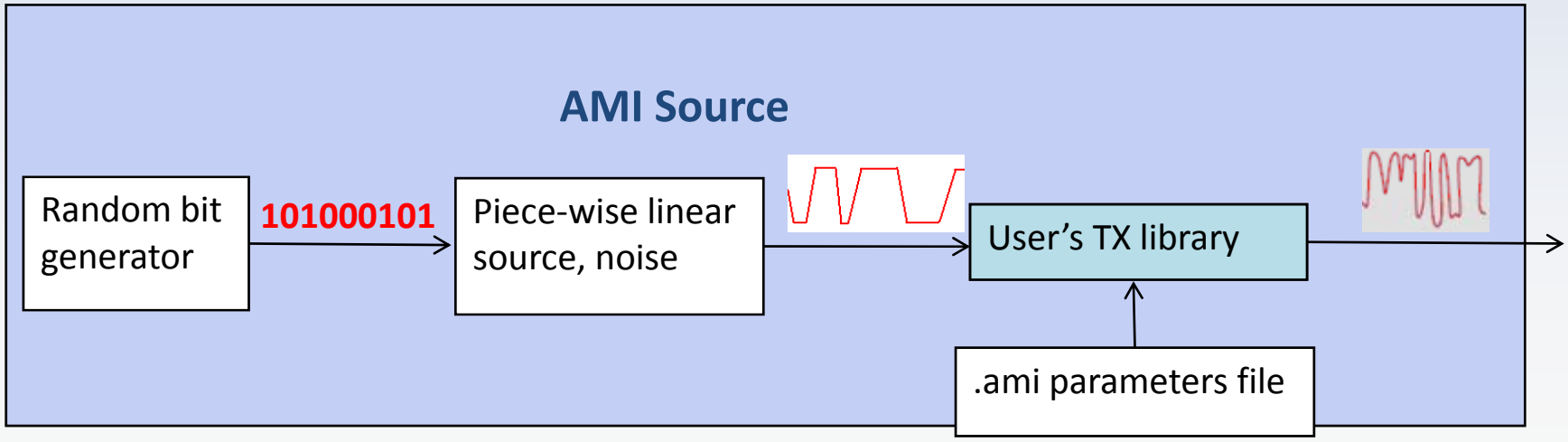


© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Example AMI Source Structure

- **AMI source serves as a wrapper for the AMI library**
- **Generate random bit sequence**
  - Unlimited user defined encoding schemes are available
- **Construct piece-wise linear signal**
  - Separate Rise/fall, high/low
  - Several types of noise – DCD, RJ, PJ, UJ
- **Push the signal through the user's transmitter (TX) library**

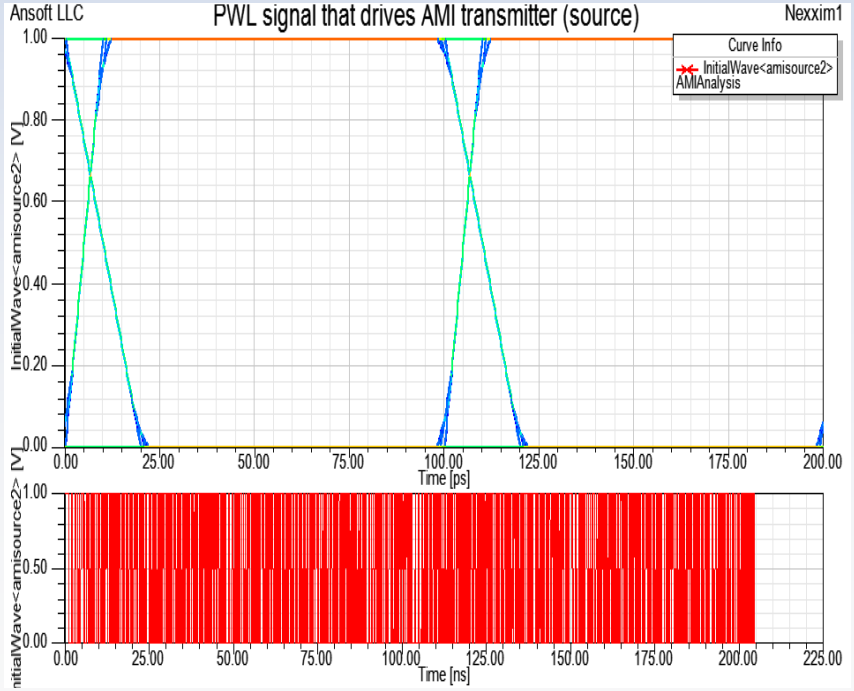


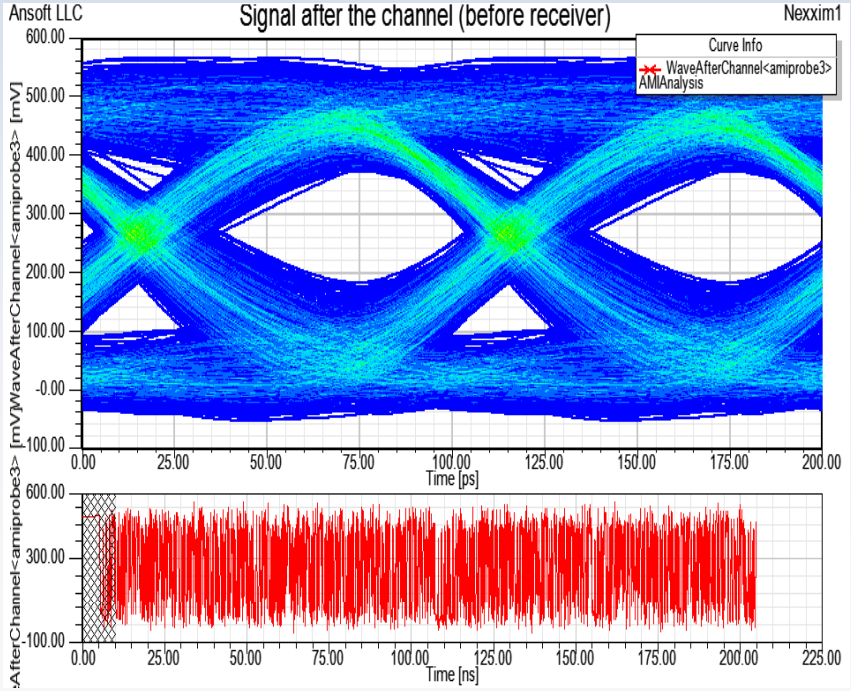
© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

AMI channel

- **Straightforward convolution with the modified impulse response**
  - Engine has to keep history , because the data is handled in blocks
- **Contributions from the aggressors are added linearly**



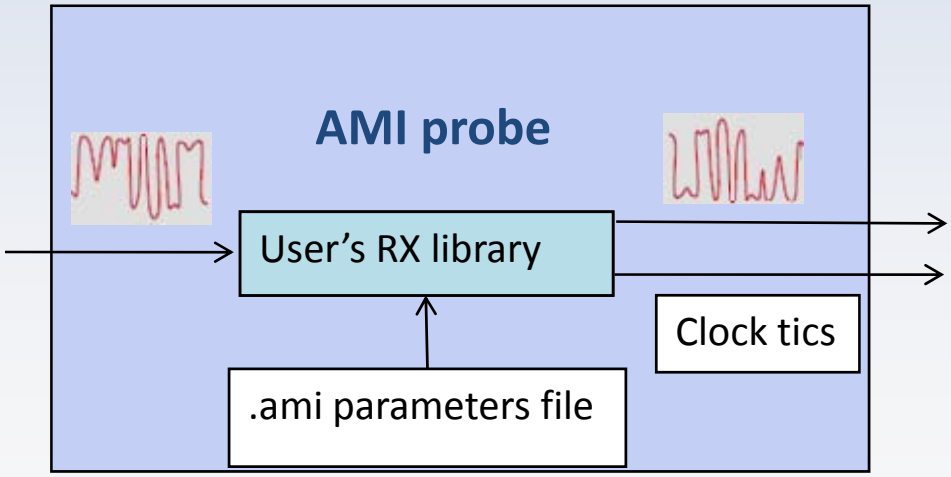


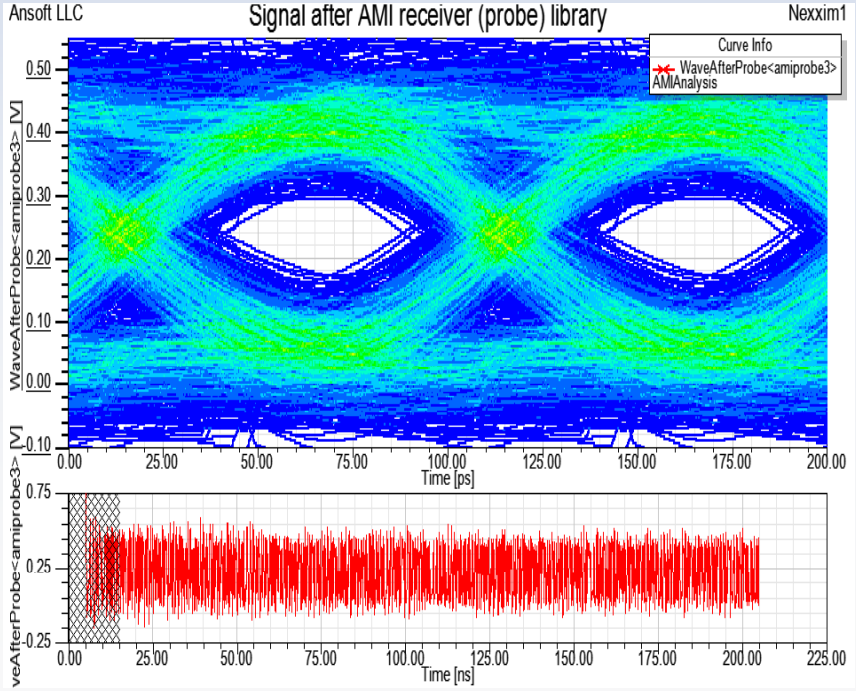
© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

AMI probe structure

- **User's libraries have exactly the same interface as AMI transmitters**
- **User can compute the clock ticks internally and provide it to the simulator for post-processing and plotting**





© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

AMI Challenges

- **AMI models are compiled DLLs and text files**
  - No graphical representation
- **Package model standard not finalized**
  - User needs to manually add IC/package parasitics to channel model
- **Each IC vendor has different parameter set**
  - No standards set
  - Each vendor must document their models
- **No standard way to sweep parameters**
  - Need to create multiple .AMI files
  - EDA tools need to parse arbitrary .AMI parameters

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

AMI Design Kit Goals

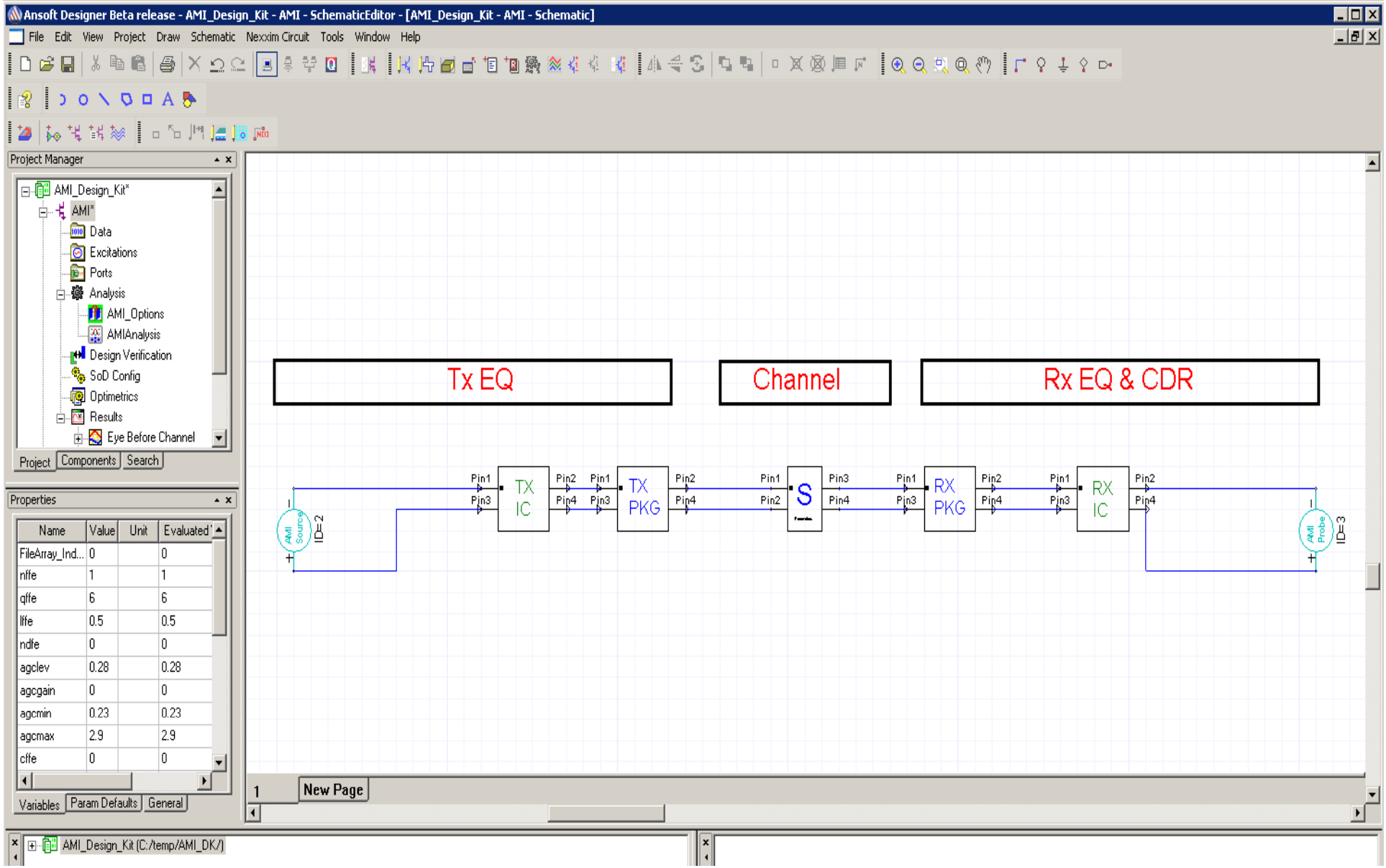
Design Kits provide another level of abstraction outside the AMI standard. They can be used by specific models to compensate for limitations of standardization.

- **Provide a quick start channel simulation**
  - Simply replace channel s-parameters with desired channel
- **Fully parameterized models**
  - Easily sweep variables
  - Bypass different components
- **Link IC and Package parasitics to core AMI models**
  - Automatically reference correct corner cases
- **Fully documented features**
  - All parameters are explained and linked with simulator
- **Common Plots already created**
  - Results automatically plotted

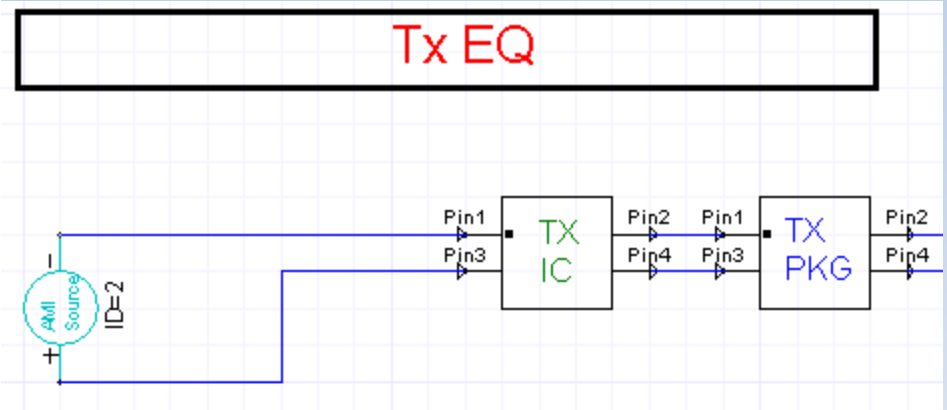
© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Example Design Kit for IBM Models



Transmitter



TX .AMI File

```
(
(path G:\Projects\AMI_Design_Kit\dll_lib) Full path to libraries
(ibis 1 )                                 Specifies IBIS-compliant AP
(nffe _subst_nffe )                       Number of FFE Taps: 1 for n
(cffe _subst_cffe )                       Sets FFE design algorithm c
(qffe _subst_qffe )                       Number of unsigned bits for
(rffe 1 )                                 Sets resolution of FFE taps
(lffe 0.25 1.0 0.5 _subst_lffe _subst_lffe _subst_lffe _subst_lffe _s
(ffeout FFE_taps.txt)                     Writes out a file for savin
(cdrstepos 2)                             Select simulation steps per
(vtt 1.5)                                 Set VTT for use in VTT depe
(txpow 115)                               Set Tx power register value
(txlev 600)                               Maximum peak level from Tx
)
```

Vendor-specific syntax, official IBIS-AMI syntax has additional required content

AMI source controls data rate, risetime, bit pattern and AMI parameters.

Source may point to different .AMI files and DLL libraries

Transmitter generally controls output levels and FFE

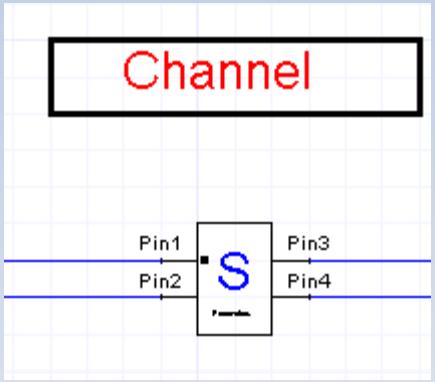
Design Kit includes a .AMI file that has been parameterized _subst_ placeholders in the text file

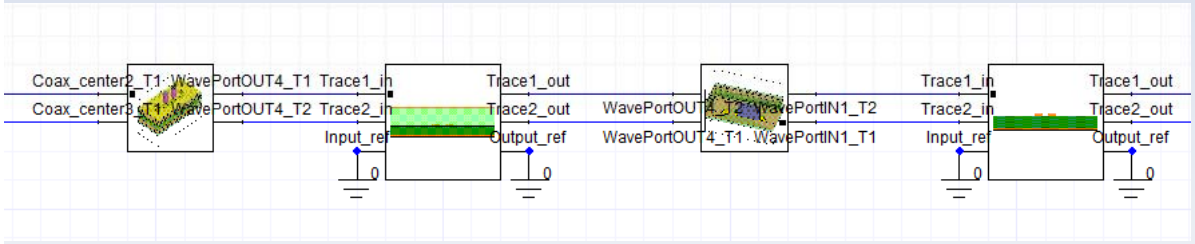
IC and PKG files are selected based on the Driver type and Corner

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Channel





This block is modified by the user to represent his/her channel

Channel may be replaced by any user defined s-parameters

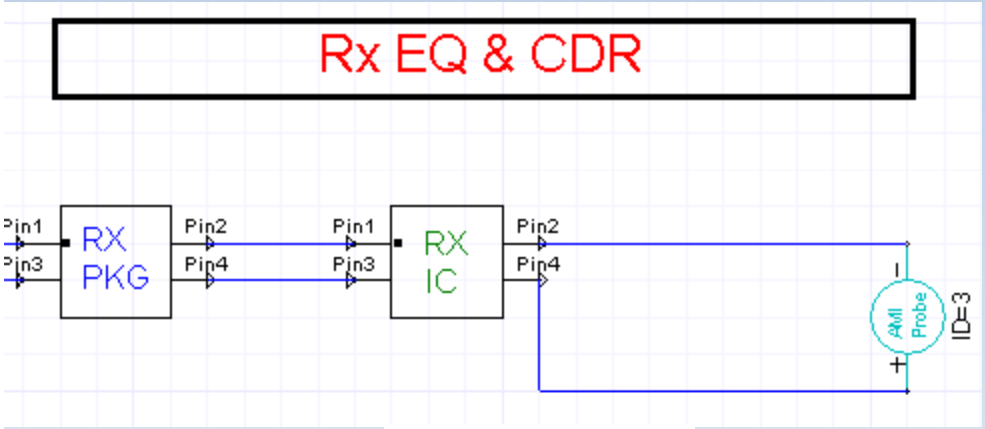
Multiple touchstone files may be swept.

Channel may also contain other spice circuitry or other include files for the transient analysis

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Receiver



RX .AMI File

```
(
(path G:\Projects\AMI_Design_Kit\dll_lib) Path to DLL Folder
(ibis 1 )                       Specifies IBIS-Compliant API
(agcmax _subst_agcmax )         Max gain of Rx Variable Gain Amplifier fro
(agcmin _subst_agcmin )         VGA min gain of Rx VGA from 0.1 to 1.0 (de
(agclev _subst_agclev )         Differential target amplitude level (V)
(agcgain _subst_agcgain )       linear gain of Rx VGA:The gain range is fr
(am .015)                       minimum latch overdrive (V)
(dfealg 3 )                     DFE Algorithim:0 = adapt averaged dfe taps
(ndfe _subst_ndfe )             Number of DFE taps
(dfelimit _subst_dfelimit )     DFE coefficient limits: 0= Don't constrair
(cdrsteps 32 )                  CDR steps per UI:27 and 32 are the valid s
(cdr 0 )                        CDR algorithm: 0 = cdrver[1] cdrclock[8](c
(cdrclock 8 )                   Select CDR clock rate in UI/CDR clock:UI p
(cdrver 4 )                     Select CDR version: 0-255 are valid settir
(cdrstepos 2)                   Select simulation steps per phase rotator
(lockwait _subst_lockwait )     Number of bits to wait before applying DFE
(rotlin G:\Projects\AMI_Design_Kit\include\pr_wc_6g4.dat ) rotator linea
(dfeadaptoff 0)                 Turns off adaptive DFE
(dfeoff 0)                      Turns off DFE
)
```

AMI probe controls AMI parameters.

Source may point to different .AMI files and DLL libraries

Very similar to AMI Source but generally contains more parameters

Receiver controls DFE and CDR settings

Vendor-specific syntax, official IBIS-AMI syntax has additional required content

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

IBM HSSCDR

- **The HSSCDR (“High Speed SerDes/Clock Data Recovery”) program is a proprietary MATLAB based system-level signal integrity simulator that supports IBM’s HSS cores.**
- **Current standard for evaluating user channels with IBM ICs.**
- **This is a standalone program that does not integrate with other simulators, requires a single s-parameter block and does not allow IC models from outside vendors**
- **IBM is looking to translate the models into the more flexible IBIS-AMI format.**

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Channel Analysis Settings

- **Used a long channel consisting of two backplanes and a cable**
- **15.6 GB/s**
- **Used HSS15 CU045 technology**
- **Simulated 1 Million Bits – PRBS31**
  - Only plotted 8192 bits for HSSCDR (Maximum Limit)
- **Worst Case PKG and IC parameters**
- **Simulated on x86 Intel Xeon Quad Core 2.4 GHz**

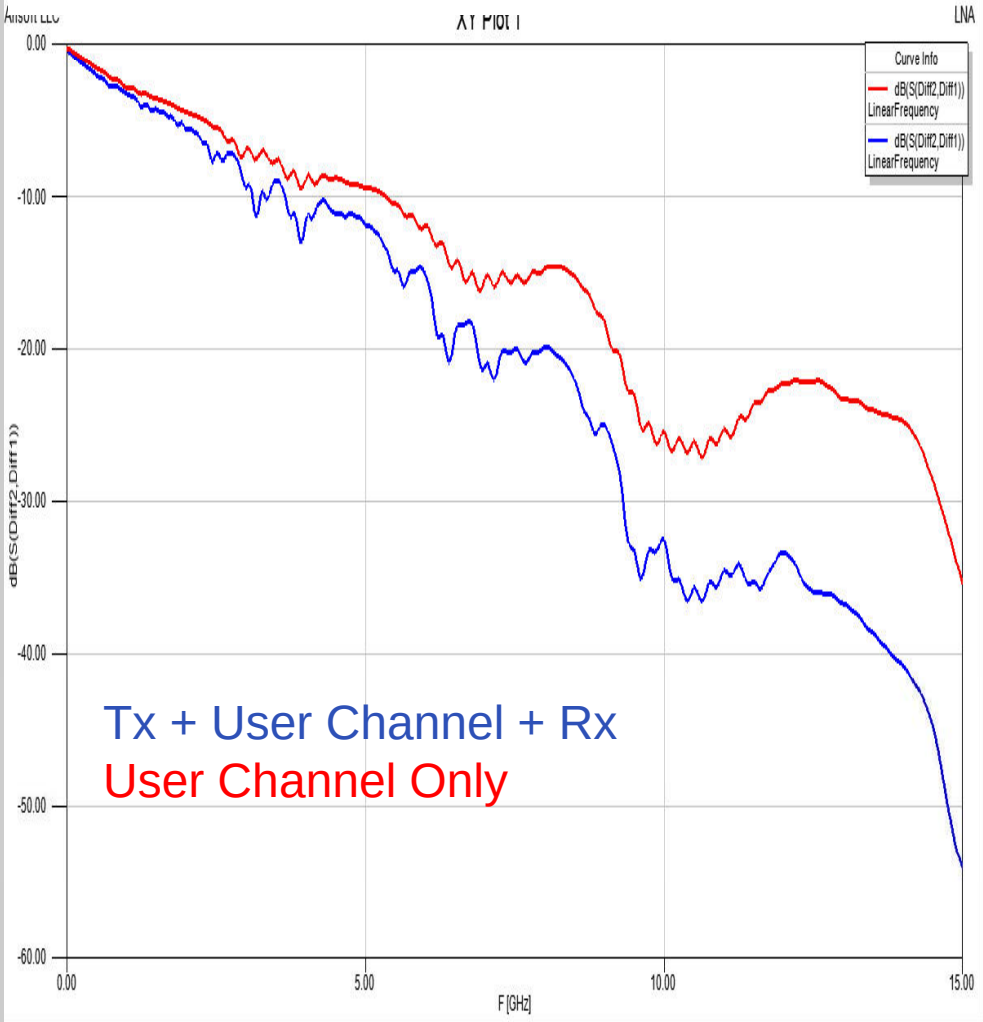
© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Channel Comparison

HSSCDR

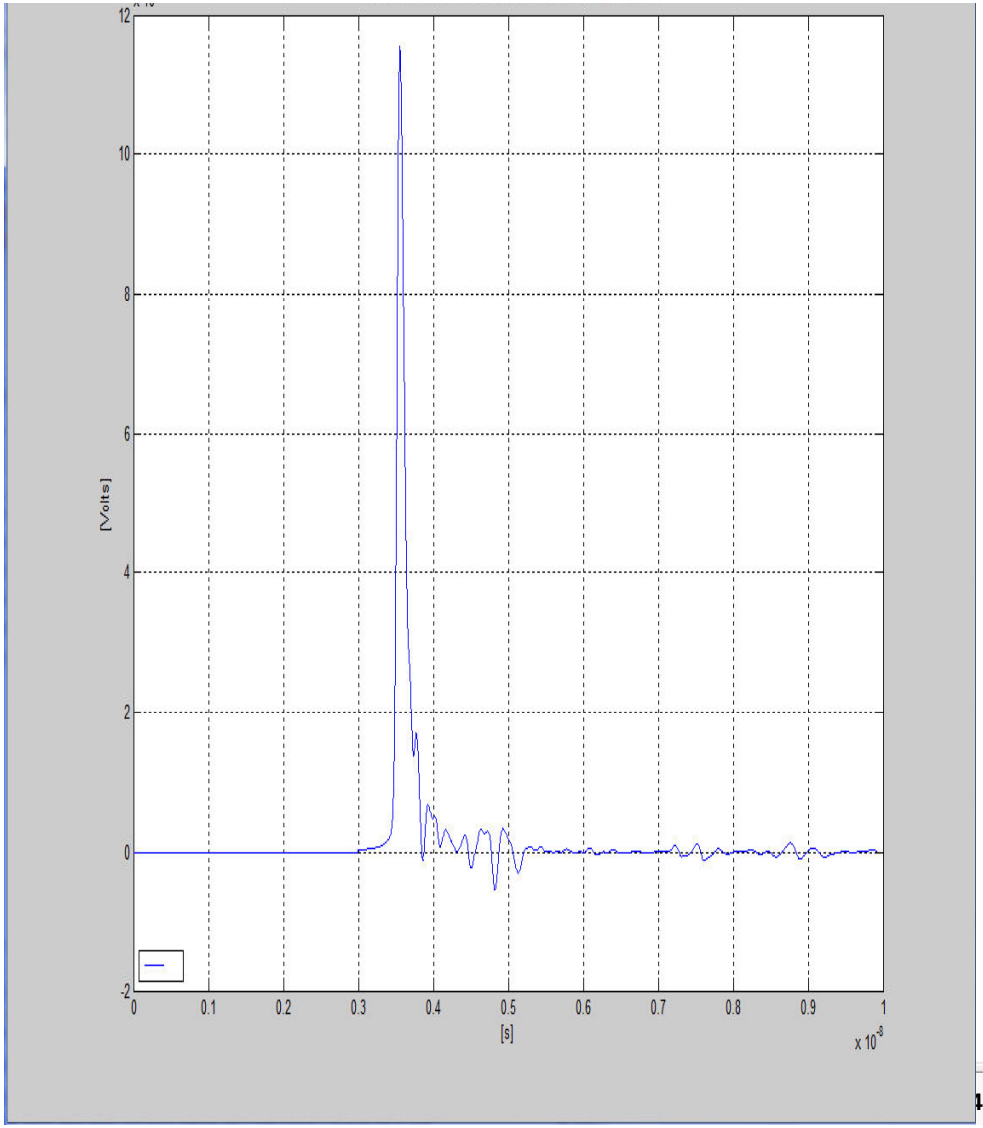
IBIS-AMI



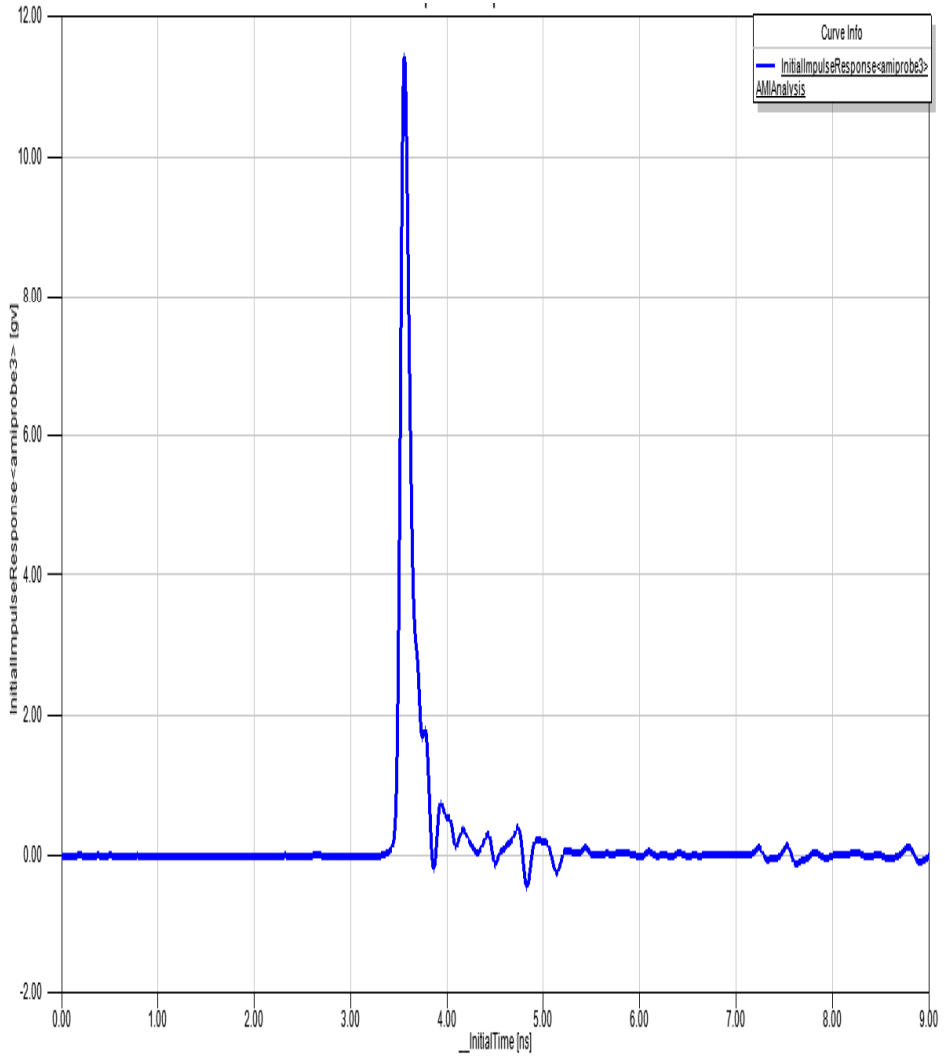
ANSYS, Inc. Proprietary

Impulse Response Comparison

HSSCDR



IBIS-AMI

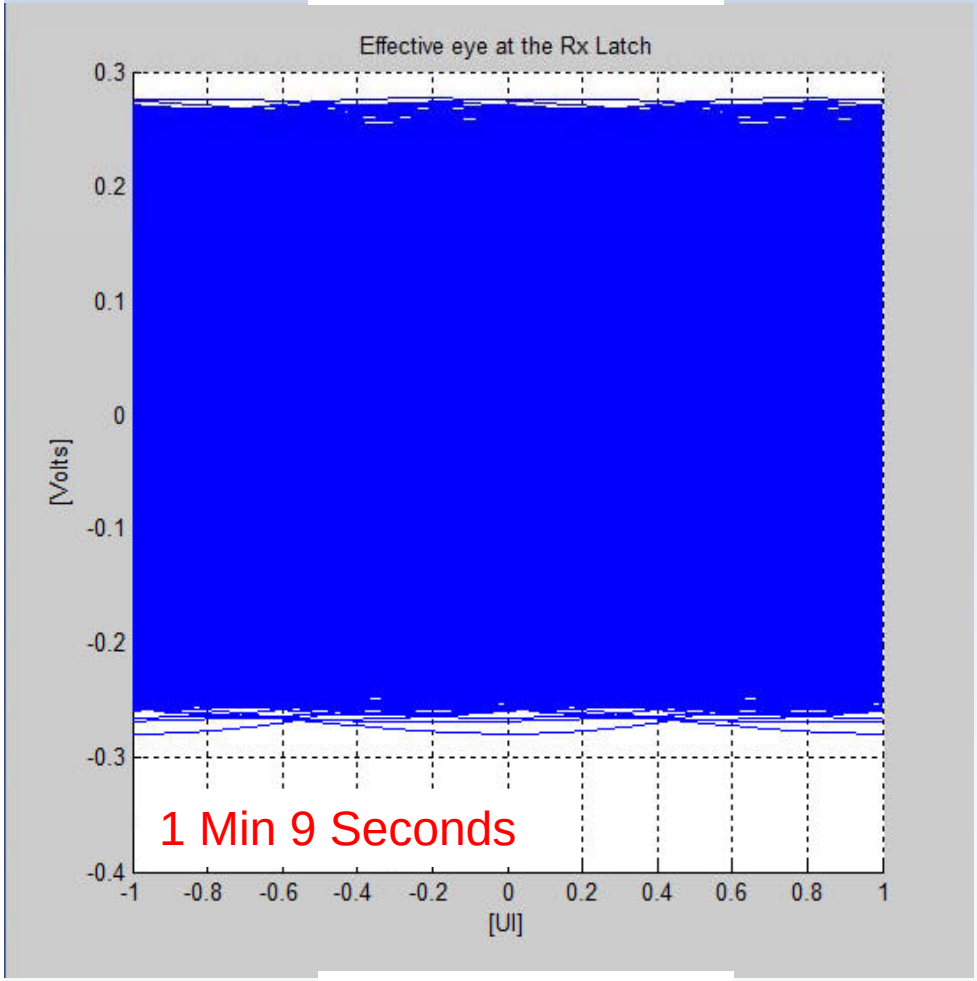


ANSYS, Inc. Proprietary

Full Channel, No FFE, No DFE

HSSCDR

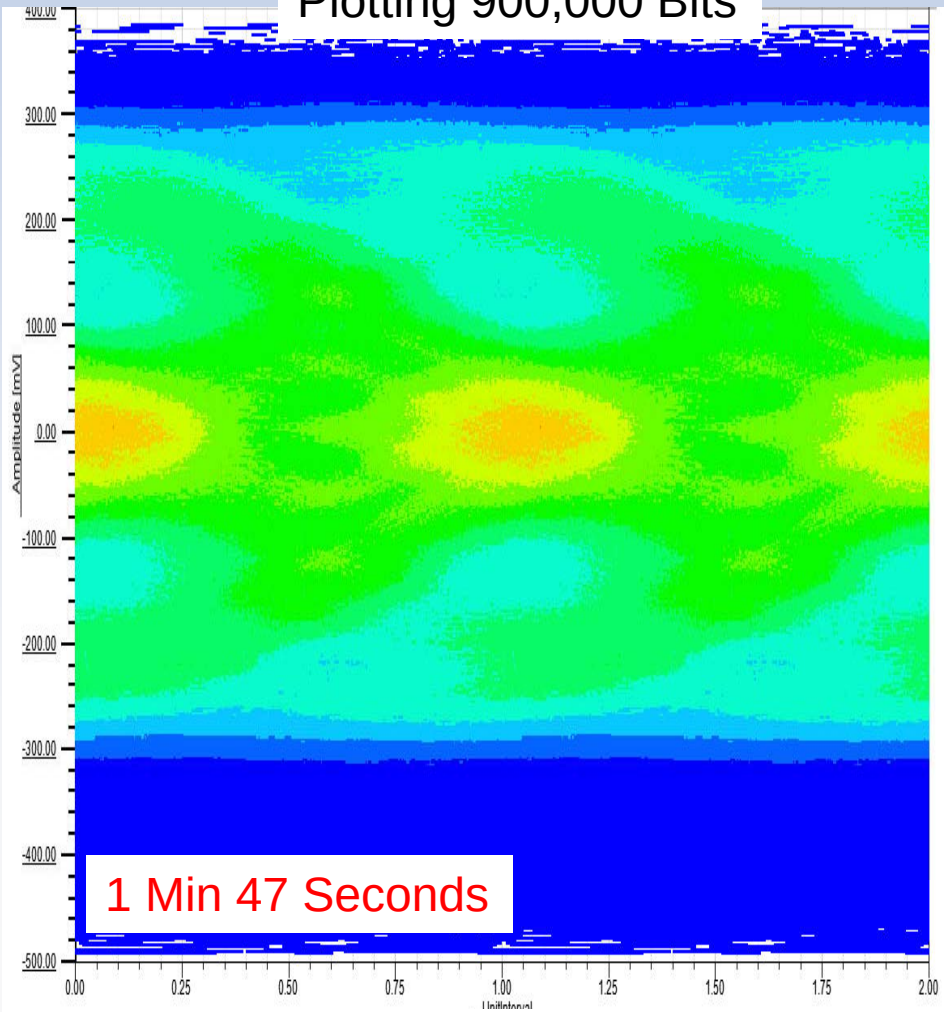
Plotting 8192 Bits



Plotting 8192 Bits

IBIS-AMI

Plotting 900,000 Bits



Plotting 900,000 Bits

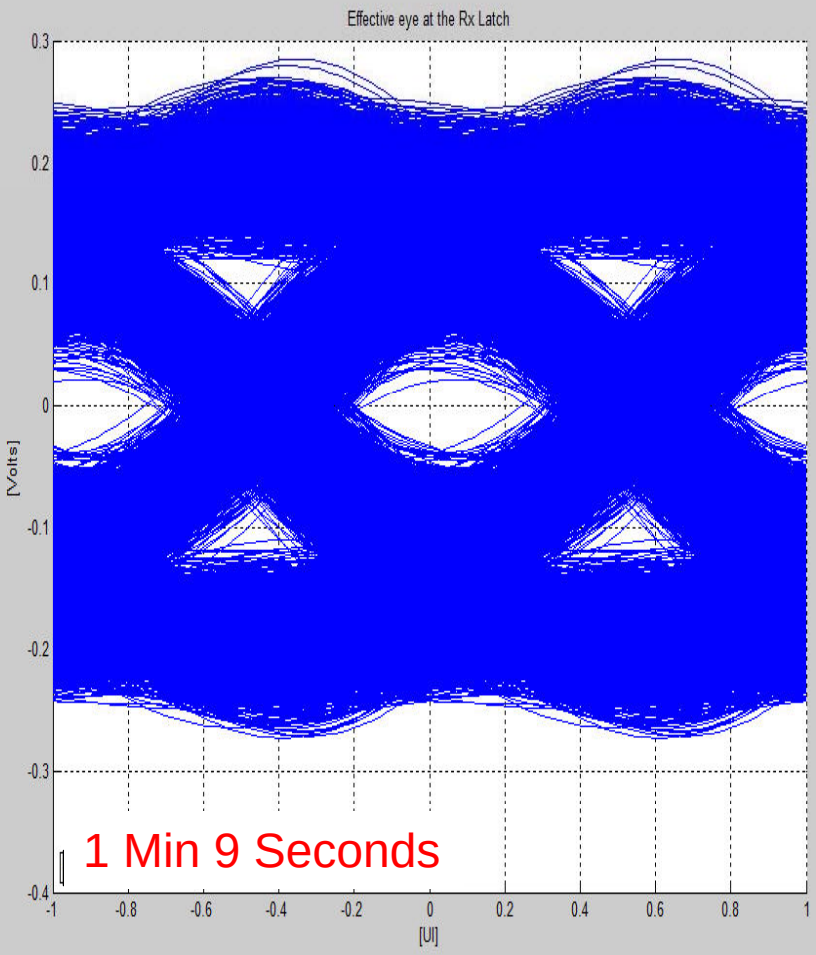
© 2009 ANSYS, Inc. All rights

NSYS, Inc. Proprietary

Full Channel, 3 tap FFE, No DFE

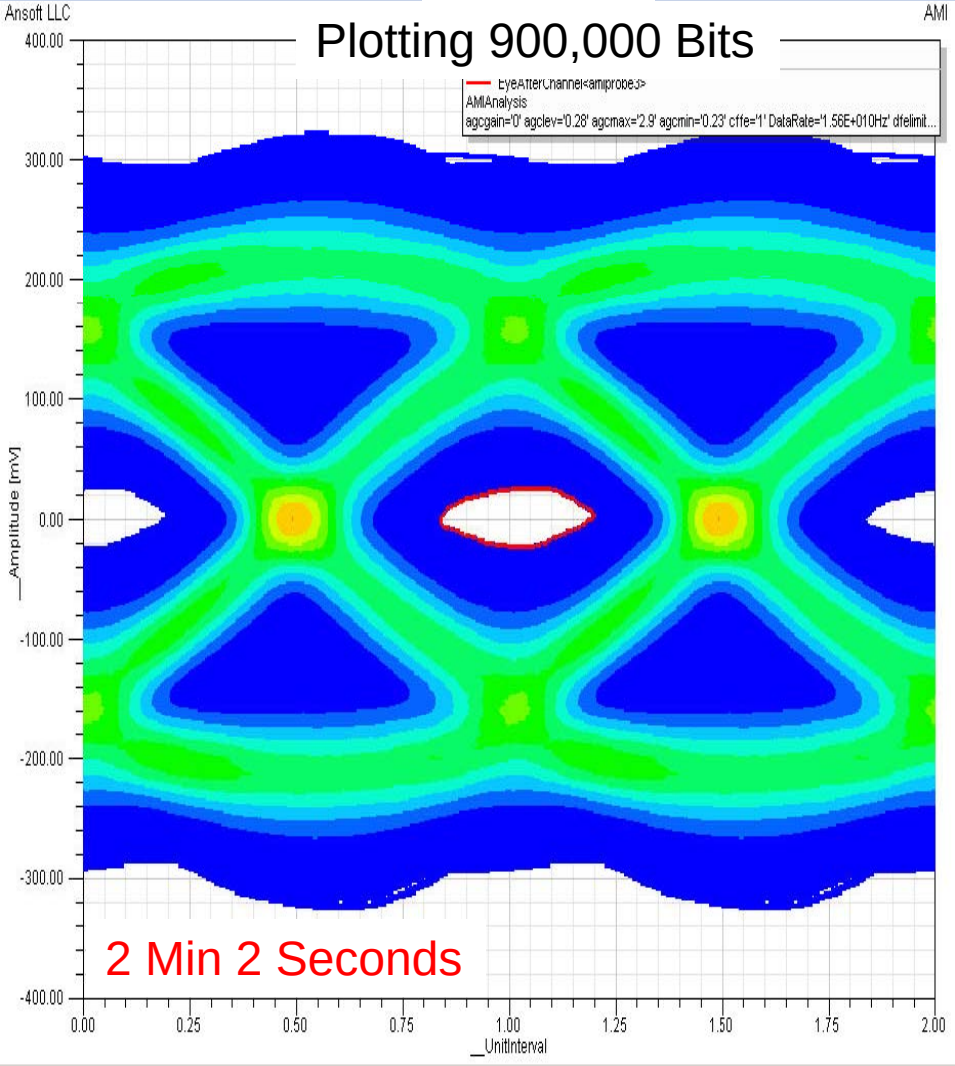
HSSCDR

Plotting 8192 Bits



IBIS-AMI

Plotting 900,000 Bits

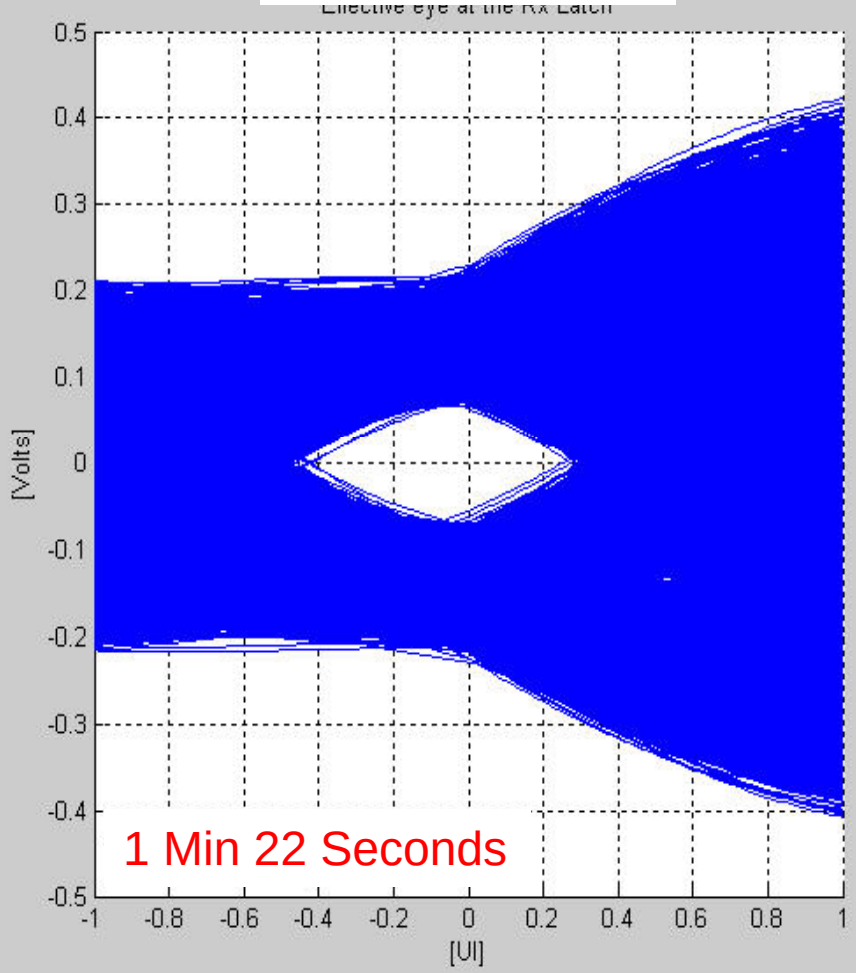


ANSYS, Inc. Proprietary

Full Channel, No FFE, 10 Tap DFE

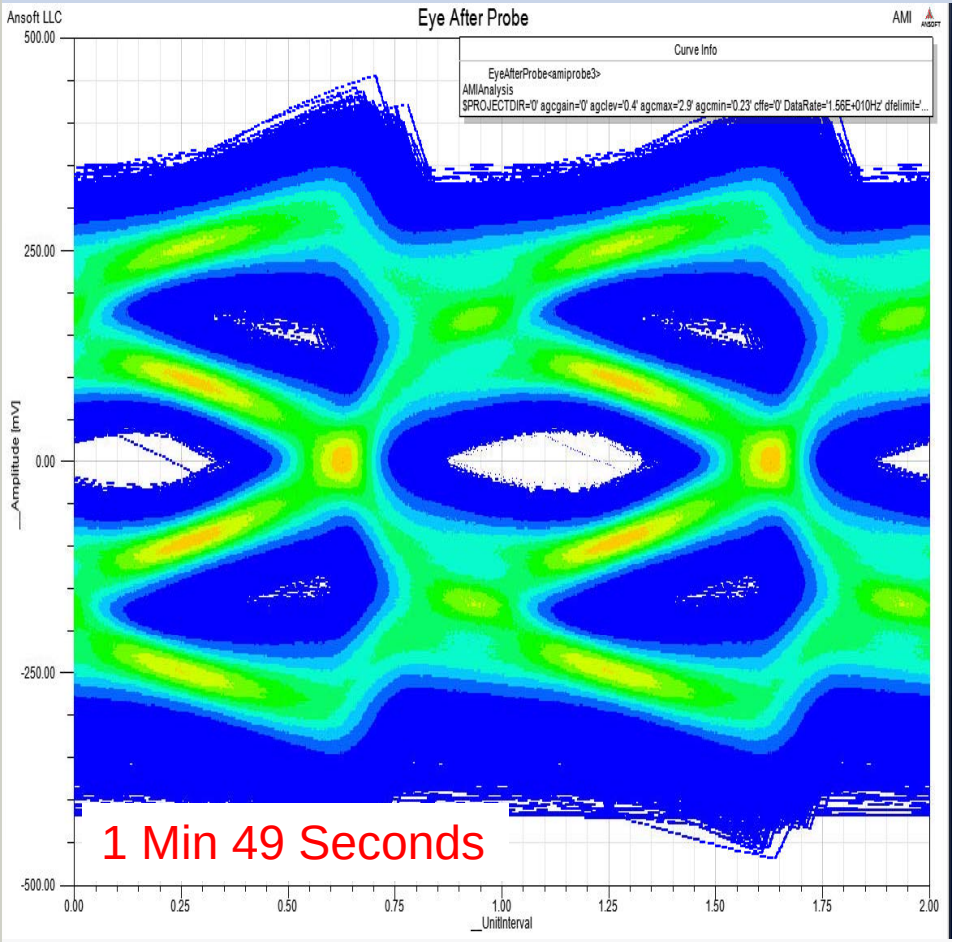
HSSCDR

Plotting 8192 Bits



IBIS-AMI

Plotting 900,000 Bits



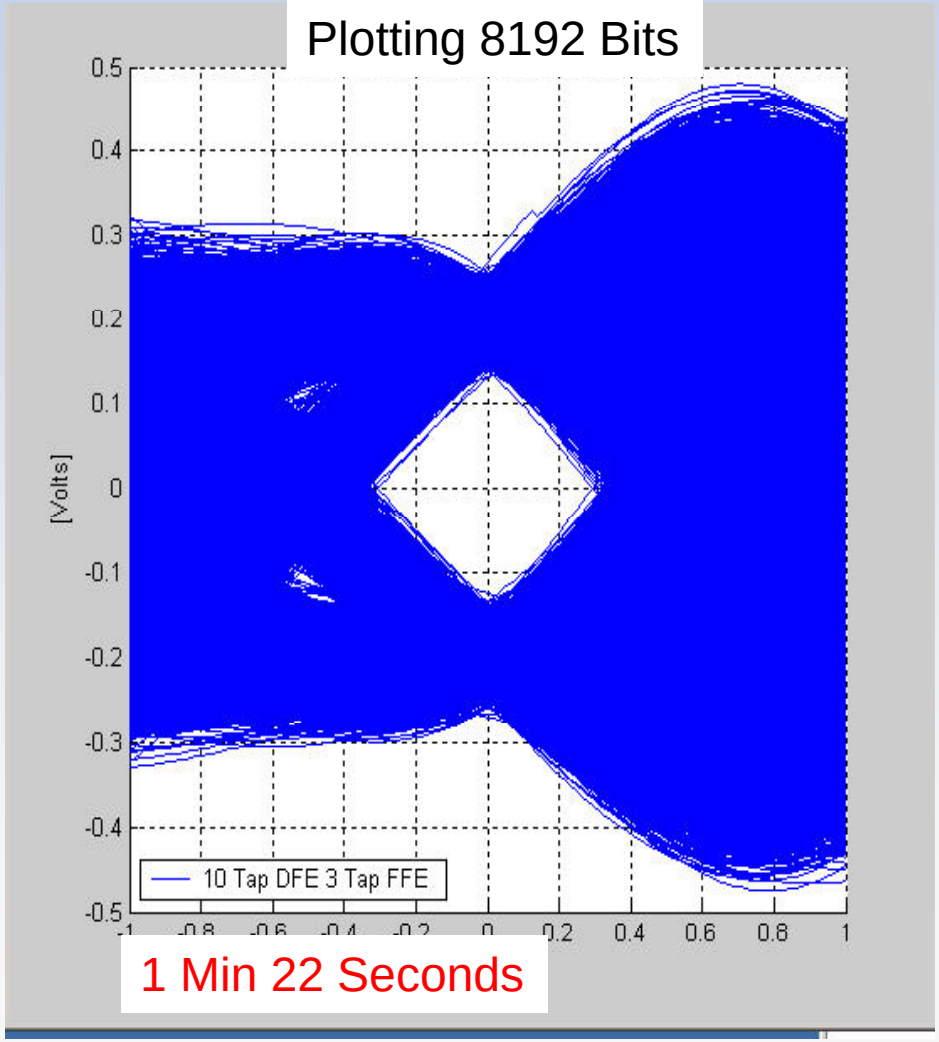
© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Full Channel, 3 tap FFE, 10 Tap DFE

HSSCDR

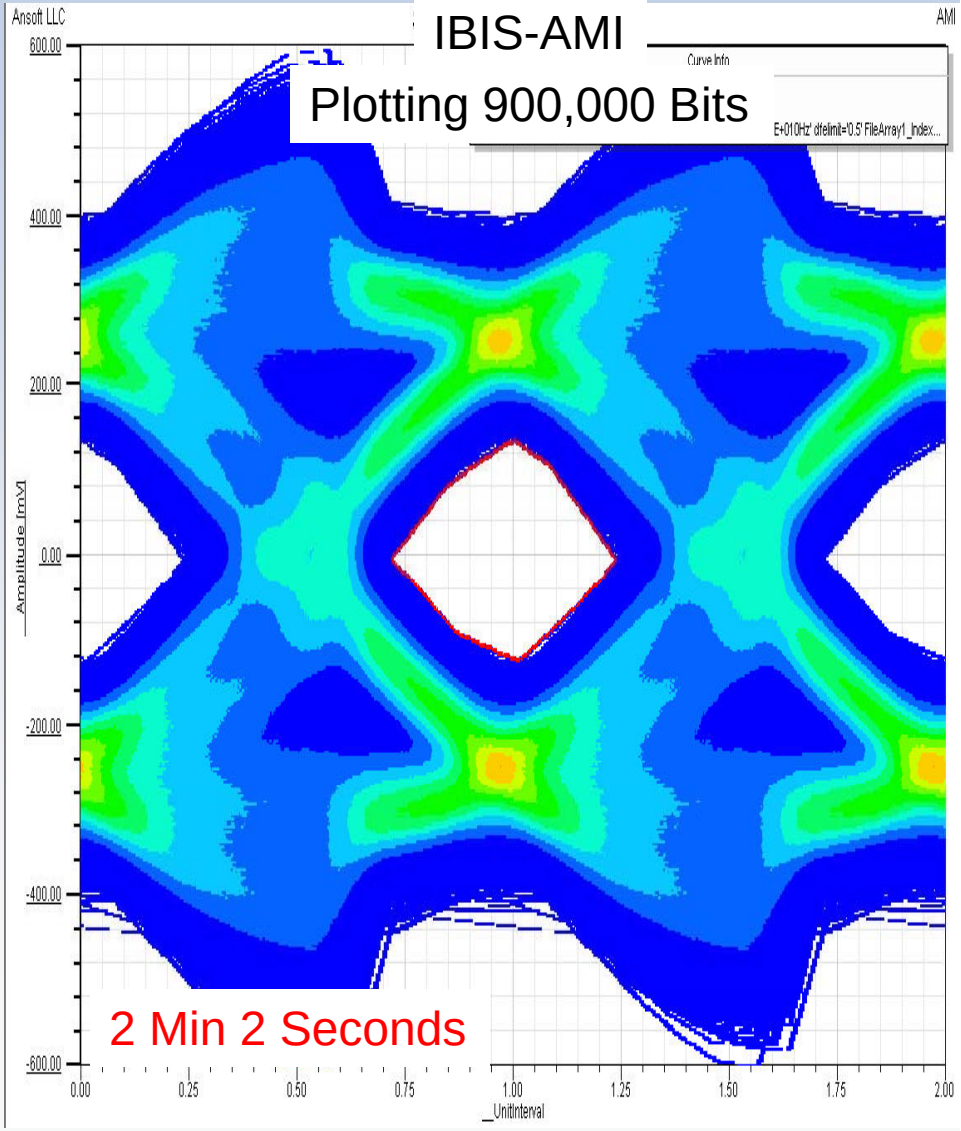
Plotting 8192 Bits



1 Min 22 Seconds

IBIS-AMI

Plotting 900,000 Bits

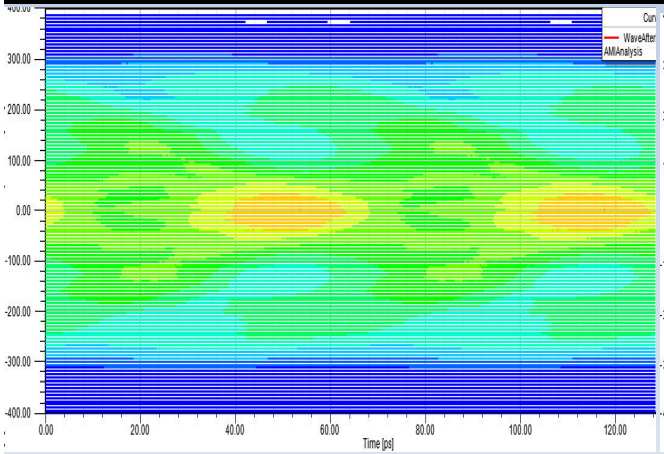


2 Min 2 Seconds

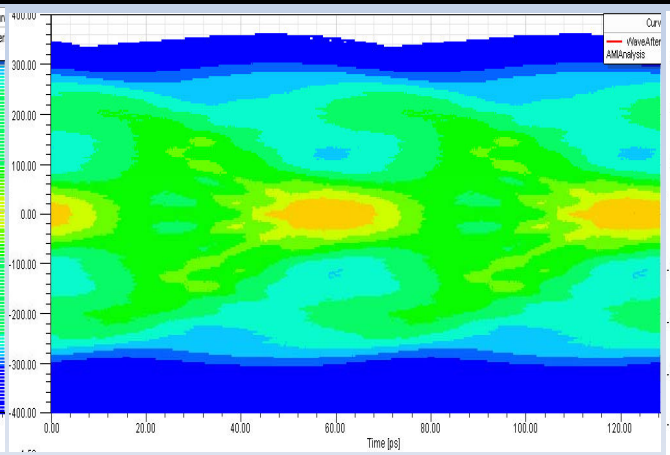
© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

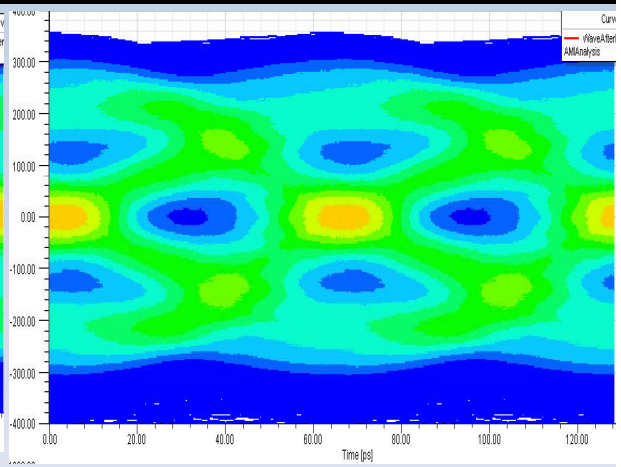
Sweeping FFE



1 Tap = No FFE

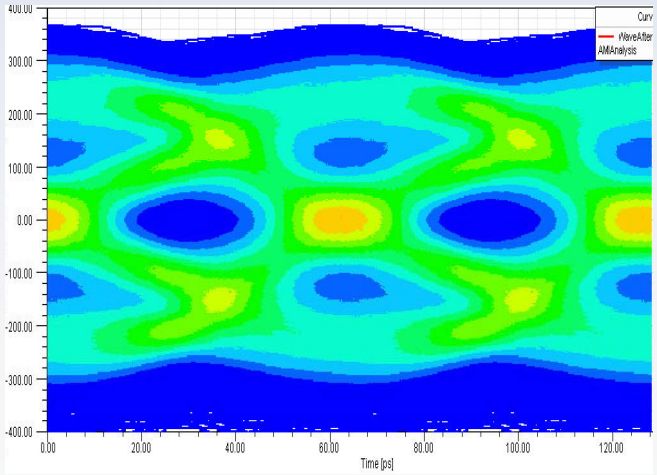


2 Taps

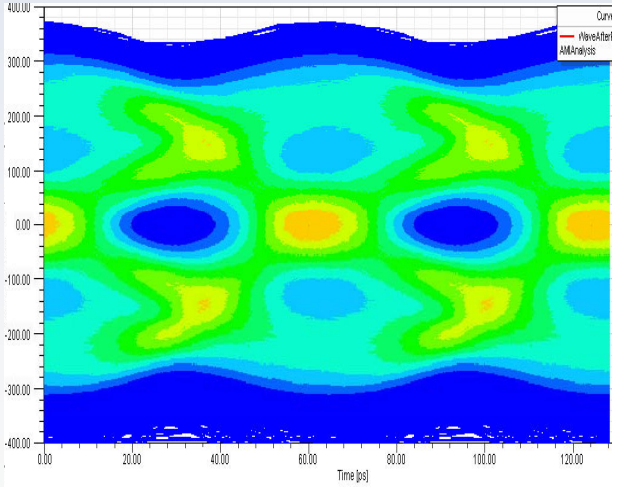


3 Taps

FFE swept from 1 to 5 taps
CFFE = NFFE-1
DFE turned OFF
Data Rate 15.6 GB/s
PRBS31 used
1 Millions bits simulated
Eye Plotted after Rx



4 Taps

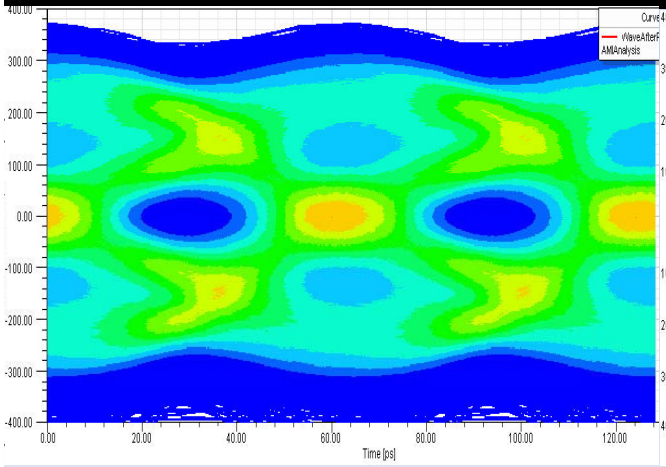


5 Taps

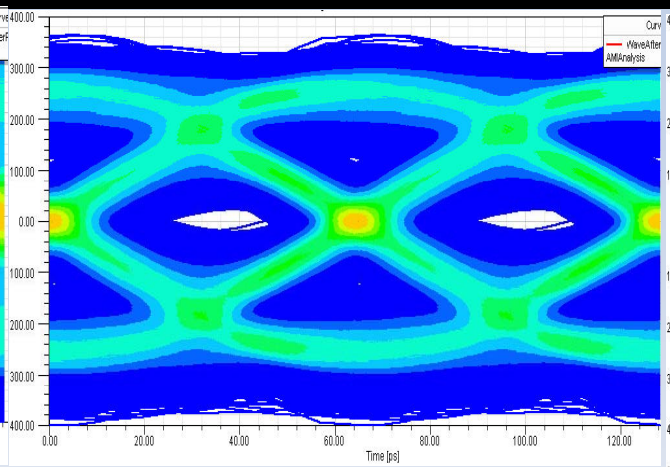
© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

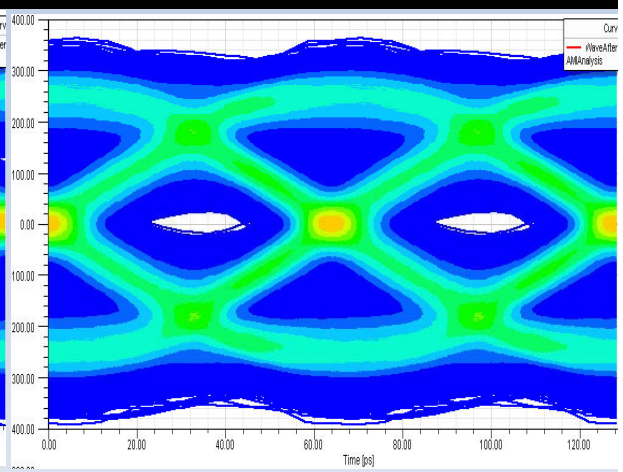
Sweeping CFFE (Cursor Lag)



CFFE = 4

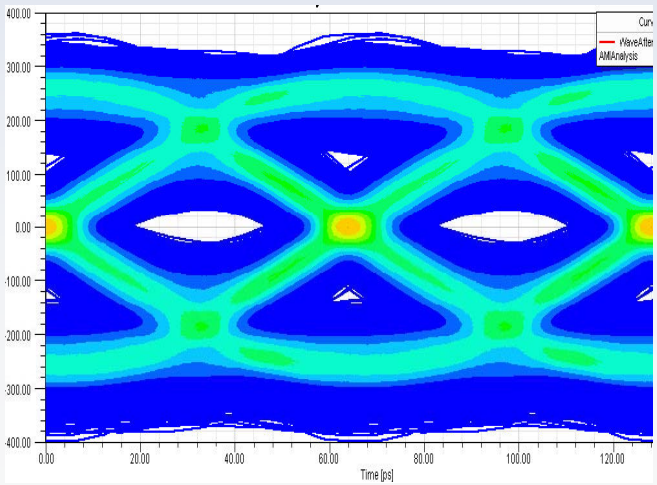


CFFE = 3

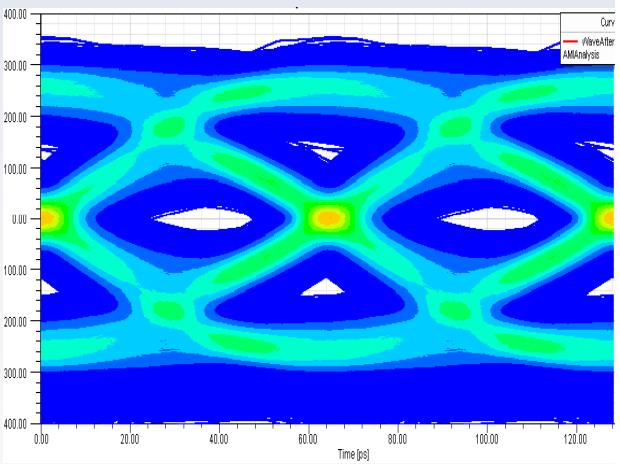


CFFE = 2

CFFE swept from 4 to 0
NFFE = 5 Taps
DFE turned OFF
Data Rate 15.6 GB/s
PRBS31 used
1 Millions bits simulated
Eye Plotted after Rx



CFFE = 1



CFFE = 0

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary

Conclusions

- **Traditional models used in simulation are no longer feasible for the latest high speed serial designs**
- **IBIS – AMI offers a promising new means of modeling Transmit and Receiver equalization without compromising intellectual property.**
- **AMI Design Kits offers a quick start for successful simulation of IBM's HSS core technology**
- **For questions or additional information, please email Cherrick@ansoft.com**

© 2009 ANSYS, Inc. All rights reserved.

ANSYS, Inc. Proprietary