# IBIS-AMI Modeling and Simulation of Link Systems using Duobinary Signaling


***Speakers***

***Timothy De Keulenaer (BiFAST)***
***Fangyi Rao (Keysight Technologies)***

***Authors***

***Hongtao Zhang (Xilinx Inc.), Fangyi Rao (Keysight Technologies),***
***Timothy De Keulenaer (BiFast), Ken Ly (Cisco Systems),***
***Ramses Pierco (BiFast), Geoff Zhang (Xilinx Inc.)***

# IBIS-AMI Modeling and Simulation of Link Systems using Duobinary Signaling


***Speakers***

***Timothy De Keulenaer (BiFAST)***
***Fangyi Rao (Keysight Technologies)***

***Authors***

***Hongtao Zhang (Xilinx Inc.)***
***Fangyi Rao (Keysight Technologies)***
***Timothy De Keulenaer (BiFast)***
***Ken Ly (Cisco Systems)***
***Ramses Pierco (BiFast)***
***Geoff Zhang (Xilinx Inc.)***

# SPEAKERS

**Timothy De Keulenaer, BiFAST, timothy@bifast.io**

is the CEO of BiFAST, an imec spin-off providing an electrical 112Gbps single-lane duobinary transceiver. He received the master degree in Applied Electrical Engineering from Ghent University, Belgium in 2010 at which point he started working at the INTEC Design laboratory which is part of the Department of Information Technology at Ghent University. He received the PhD degree in Applied Electrical Engineering in 2015 and immediately afterwards started the BiFAST spin-off. His main interests are on high speed integrated circuit design and signal integrity aspects for backplane communication. He was awarded the Best Paper Award at DesignCon 2015 in the High-Speed Signal Design category and his PhD dissertation was recognized for its technological contributions receiving the Nokia Bell Scientific Prize in 2016.

**Fangyi Rao, Keysight, Fangyi_rao@keysight.com**

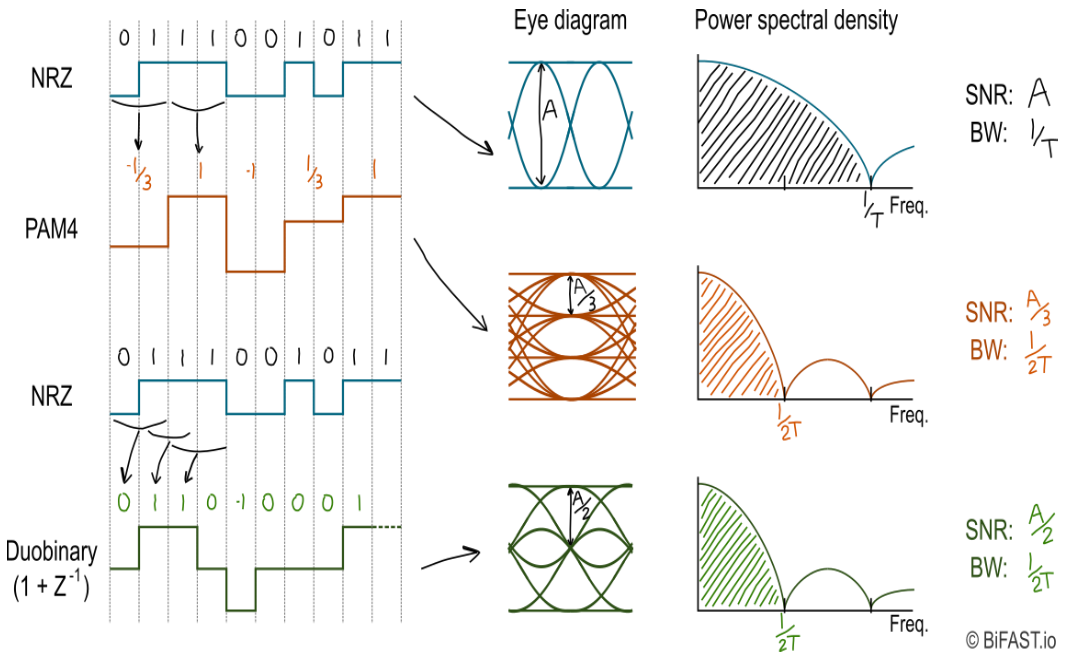
is a master R&D engineer at Keysight Technologies. He received his Ph.D. degree in theoretical physics from Northwestern University. He joined Agilent/Keysight EEsoft in 2006 and works on Analog/RF and SI simulation technologies in ADS. From 2003 to 2006 he was with Cadence Design Systems, where he developed SpectreRF Harmonic Balance technology and perturbation analysis of nonlinear circuits. Prior to 2003 he worked in the areas of EM simulation, nonlinear device modeling, and medical imaging.

# Outline

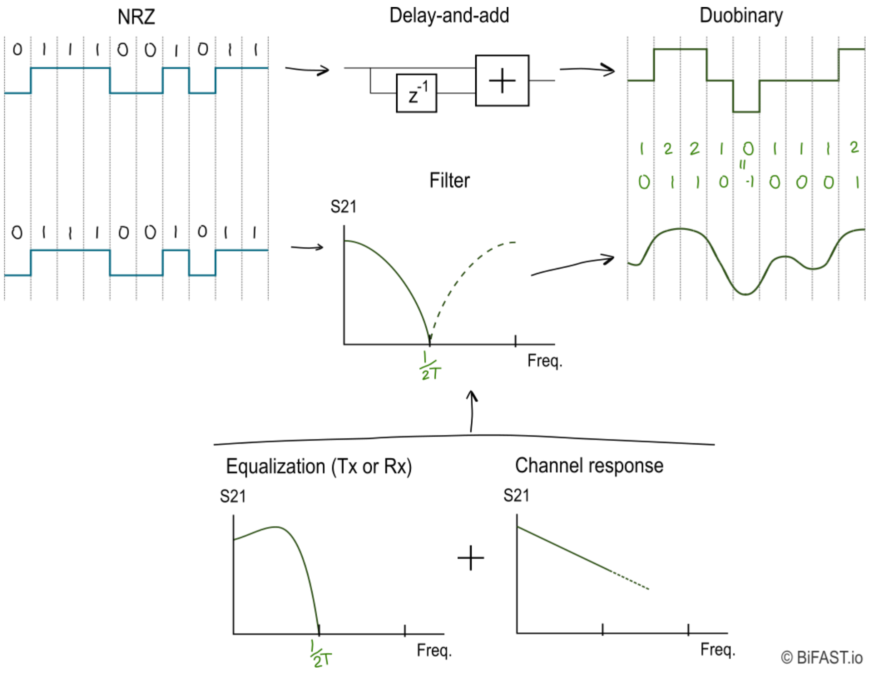
- **Introduction to Duobinary Signaling**
  - Duobinary basics
  - Duobinary signal detection
- **IBIS-AMI Modeling Overview**
- **IBIS-AMI Modeling for Duobinary**
  - TX input stimulus levels
  - RX slicer levels and timing skew
- **Duobinary Eye Measurements**
  - Eye Diagram
  - Bathtub Curves
- **IBIS-AMI Model Simulations for Duobinary Links**
- **Conclusions and Future Work**

# Duobinary Signal Overview

- Duobinary signal: combination of current NRZ bit with previous NRZ bit → multi-level signal
  - Duobinary signal has three signal levels.
  - Same power spectral density (PSD) up to frequency of 1/(2T) as PAM4 for the same data rate.
- Better SNR to bandwidth trade-off in case of duobinary.



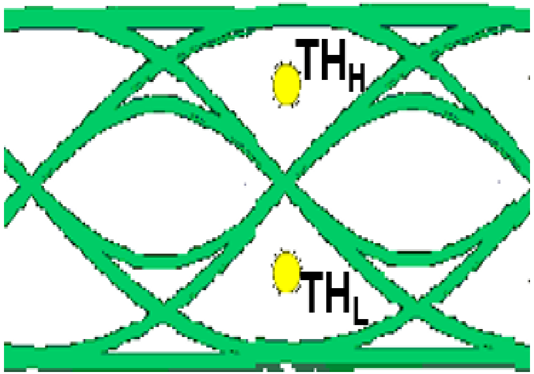
# Duobinary Signal Formulation




- Consecutive bits NRZ stream added together ($1+z^{-1}$)
  - Symbol rate equal to the NRZ bit rate.
  - No high speed signal transitions:
    - Impossible to go directly from a +1 to a -1 symbol and vice versa → factor two reduction in required bandwidth.
- Achieving duobinary in two different ways
  1. At the transmitter: putting an NRZ stream through a delay-and-add block.
  2. In the channel: $1+z^{-1}$ function realized by the combination of equalization and the channel insertion loss → less equalization needed

# Duobinary Signaling Detection – 1

- Duobinary signal can be detected based on the decision logic below



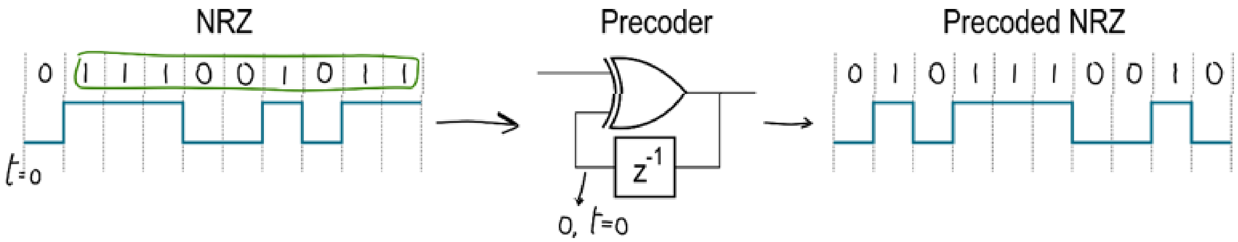

```
if s(k)>TH_H
      d(k) = 1;
elseif s(k)<TH_L
      d(k) = 0;
else
      if d(k-1) = 1
            d(k) = 0;
      else
            d(k) = 1;
      endif
endif
```

- Decision of current bit, d(k), depends on the previous bit, d(k-1).
- This causes error propagation if previous bit incorrectly detected.
- To avoid this, precoding on the TX side and de-precoding (demodulation) on the RX.

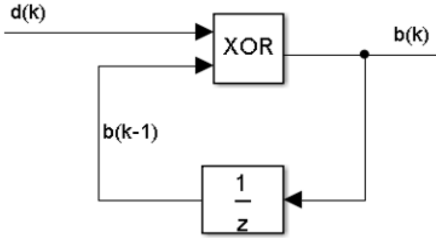
  → Straightforward in case of Duobinary

# Transmitter Precoding of the Signal

- Precoding is implemented by a simple XOR gate and a single-bit delay at the full rate.
  - Precoding can also be done at the half or quarter rate before the serialization.
- De-precoding/demodulation uses two slicers followed by a XOR gate.




- An example is shown below

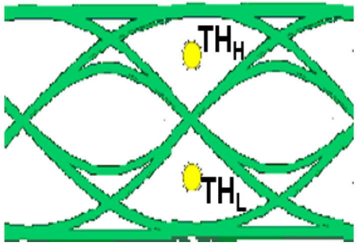


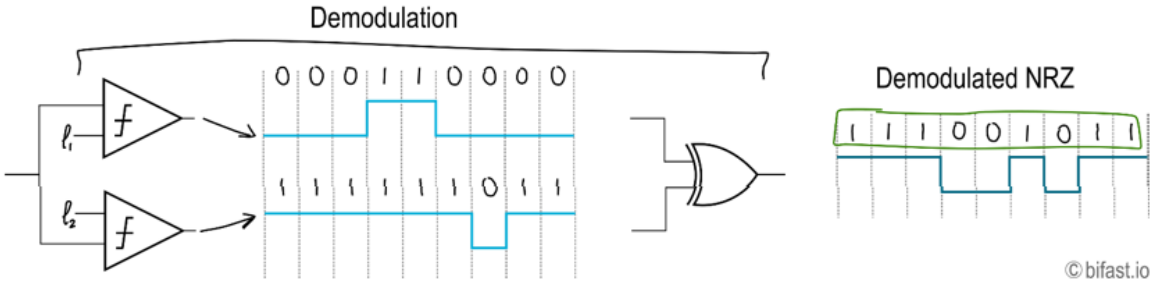

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **d(k)** | x | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| **b(k)** | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |

# Duobinary Signaling Detection – 2

- For duobinary with precoding, detection requires de-precoding/demodulation
  - This process uses two slicers followed by a XOR
  - Error propagation is avoided

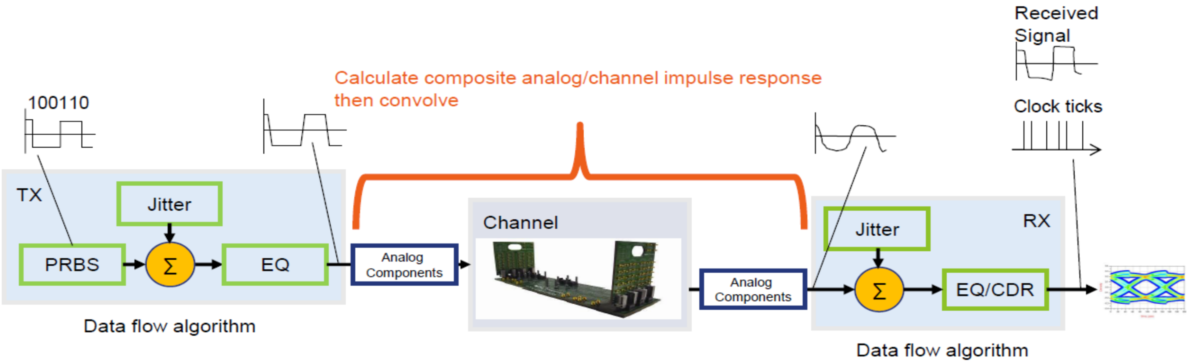
*if* $s(k)>TH_H$ *or* $s(k)<TH_L$
    $d(k) = 0;$
*else*
    $d(k) = 1;$
*endif*



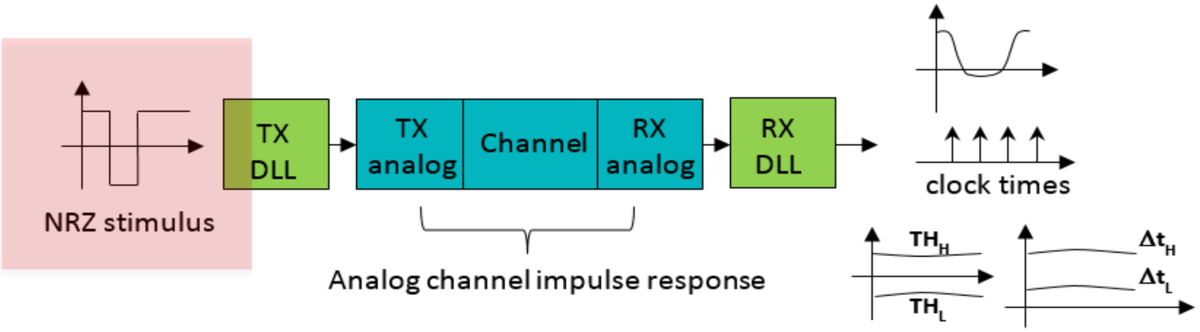
# IBIS-AMI Modeling Overview

- TX DLL input is switching between 0.5V and -0.5V
- TX output is convolved with channel impulse response
- The simulator sends binary sequences to TX IBIS-AMI model
- The resultant waveform is input to the RX DLL
- RX data segments are processed sequentially with each AMI_GetWave() call
- RX sends equalized signal and clock ticks to the simulator



# AMI Modeling for Duobinary – Transmitter

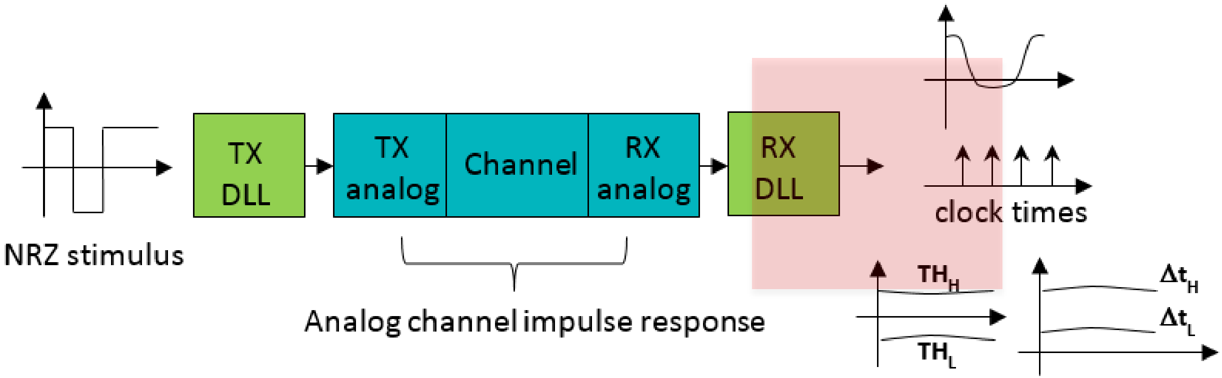
- It is proposed that the TX DLL input stimulus remain the same as that for NRZ.
- The EDA tool is responsible for precoding the binary data stream.
  - Thus, for the TX DLL there is no difference whether the binary bits are precoded or not, but for the RX decision, with or without it makes a big difference.
- Note that the TX DLL can either perform the delay-and-add operation or leverage the channel to apply a low-pass filtering on the input NRZ signal to get the duobinary signal.
  - The delay-and-add block is not implemented in this paper. This implies the equalization, jointly delivered by the TX and the RX, has less burden in terms of the amount ISI that needs to be removed.



# AMI Modeling for Duobinary – Receiver

- $TH_H$ and $TH_L$ is proposed to be returned through the *AMI_parameters_out* string.
  - The model returns a pointer to the string as the value of this argument.
  - The content of the string is formatted as a tree structure of parameters with Usage Out and InOut.
  - The tree structure is scalable and extendible, making it easy to add new parameters to the string.

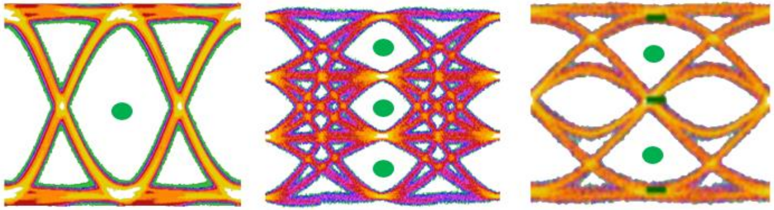
```
long AMI_GetWave ( double *wave,
                   long wave_size,
                   double *clock_times,
                   char **AMI_parameters_out,
                   void *AMI_memory );
```



# AMI Modeling for Duobinary – Receiver (Con't)

- For optimal sampling result,
  - The two slicers can possibly sample the signal at different times. The sampling time skew can be adjusted adaptively and time dependent. It is proposed that two new AMI reserved parameters, $\Delta t_H$ and $\Delta t_L$, for upper and lower slicer sampling time offsets, relative to clock tick times.
  - Assuming that the skew varies slowly after the adaptation converges, it is sufficient to update the offsets once in each AMI_GetWave call.
  - Similar to slicer levels, the offset values are returned by the model through the AMI_parameters_out string argument of the AMI_GetWave function.
- In each AMI_GetWave, the RX model will write name-value pairs of $TH_H$, $TH_L$, $\Delta t_H$ and $\Delta t_L$ into the AMI_parameters_out string and pass it back to the simulator
  - The simulator will parse the string to extract slicer levels and sampling time offsets relative to clock tick times. The slicer levels are used to decide duobinary logic on bits processed in this AMI_GetWave call for BER calculations.
  - The decision logic is implemented in the EDA. Eye diagrams, BER contours, and voltage and timing bathtub curves can all be derived from the four sets of data.

# Duobinary Eye Measurement

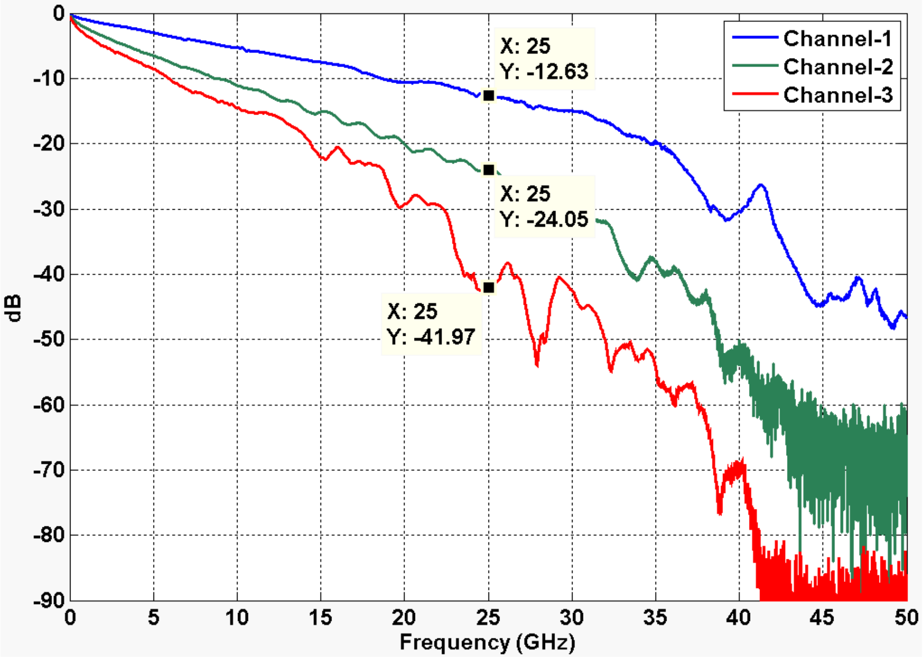


- Eye diagrams
  - EDA tool can construct the duobinary eye diagrams similar to  the NRZ and PAM eye diagrams
- Bathtub curves
  - EDA tool separates the waveforms into the Upper and the Lower eyes depending on the resolved bit values
  - The instantaneous slicer levels are used for the BER calculation
  - The waveform segments are aligned according to the instantaneous clock ticks and timing offset

| Eye | Traces | Horizontal eye center |
|---|---|---|
| Upper | $v_1(t)-TH_H(t)$ and $v_2(t)-TH_H(t)$ | $t_{clk}(n)+\Delta t_H(n)+UI/2$ |
| Lower | $v_0(t)-TH_L(t)$ and $v_1(t)-TH_L(t)$ | $t_{clk}(n)+\Delta t_L(n)+UI/2$ |

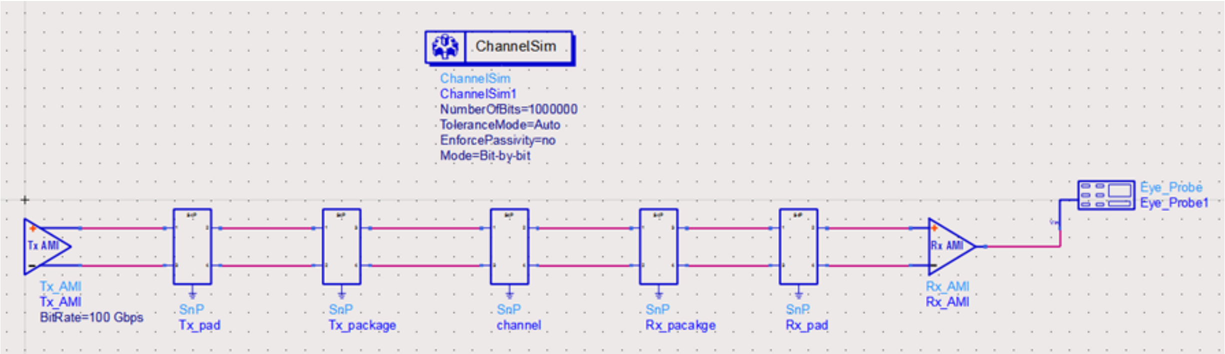
# Simulation Examples – Channel Description

- The simulation is to verify the proposed approach for modeling duobinary.
- No crosstalk is included without affecting the purpose of the proposed methodology.
- 100Gbps is the data rate for the work.
- The three channel losses, compared at 25GHz, are shown below, targeting different reaches
  - 12.6dB
  - 24.1dB
  - 42.0dB



# Duobinary Link Simulation in ADS

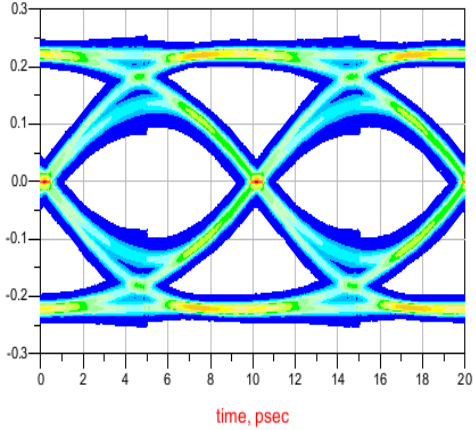
- Simulation conditions include
  - The data rate is 100Gbps. The data pattern is PRBS23 (before precoding).
  - The channel is formed by cascading TX on-die, TX package, channel, RX package and RX on-die S-parameters.
  - 1.5 million bits are simulated for each channel, with 0.5 million bits ignored for RX adaptation.
  - The TX de-emphasis consists of 3-tap FIR (pre + main + post). The tap weights are manually programmed.
  - On the RX side, there is a 2-stage CTLE, an AGC, plus 20-tap DFE.
  - All the RX side parameters are adaptive, including baseline wander cancellation.
  - Baud rate CDR is implemented for timing recovery.



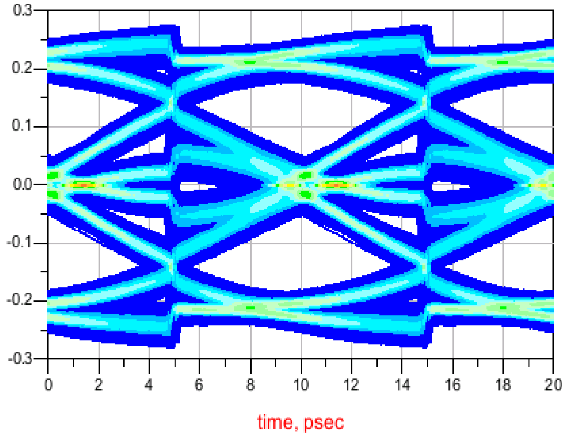
# Example of Using AMI Model for Duobinary

- Eye diagrams for channel 1, 2, 3
  - Eye margin gets reduced with higher loss channels
  - 50 GHz frequency is noticeably reduced for channel 2 and 3
  - No significant DFE is required even for high loss channels

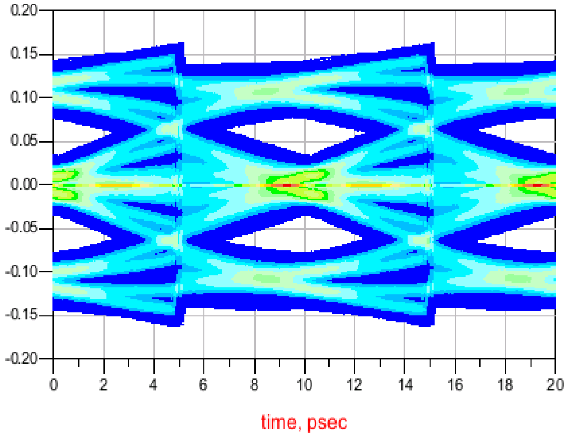
**Channel 1**




**Channel 2**

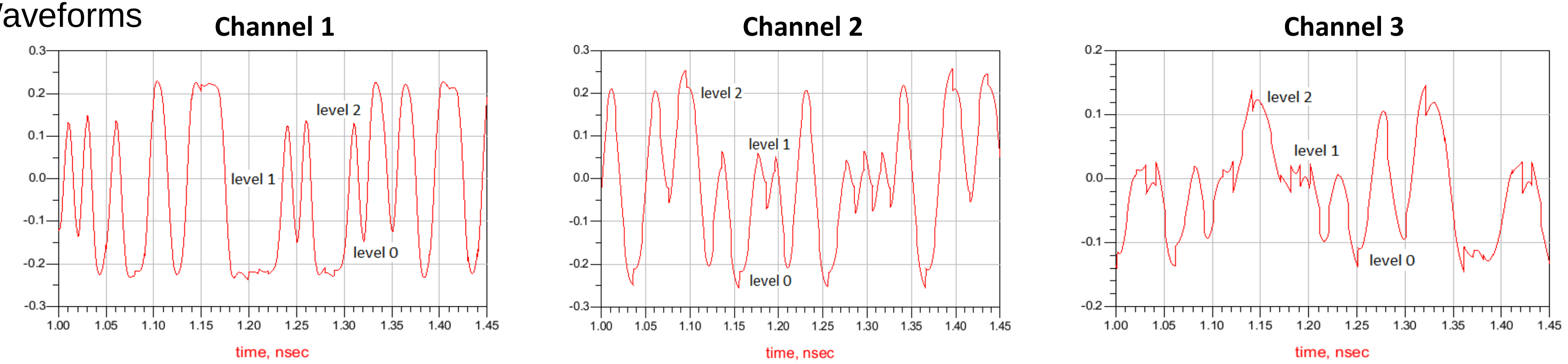



**Channel 3**

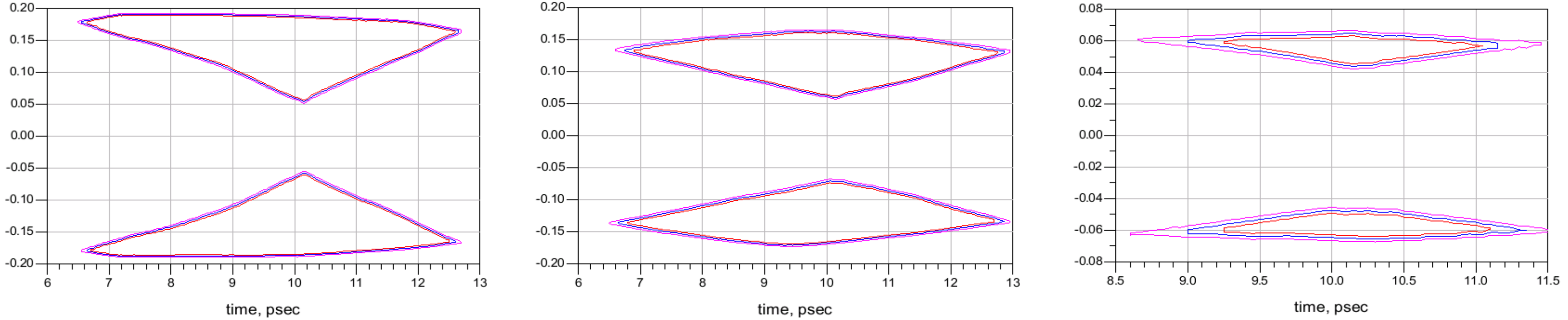


# Example of Using AMI Model for Duobinary

- Waveforms

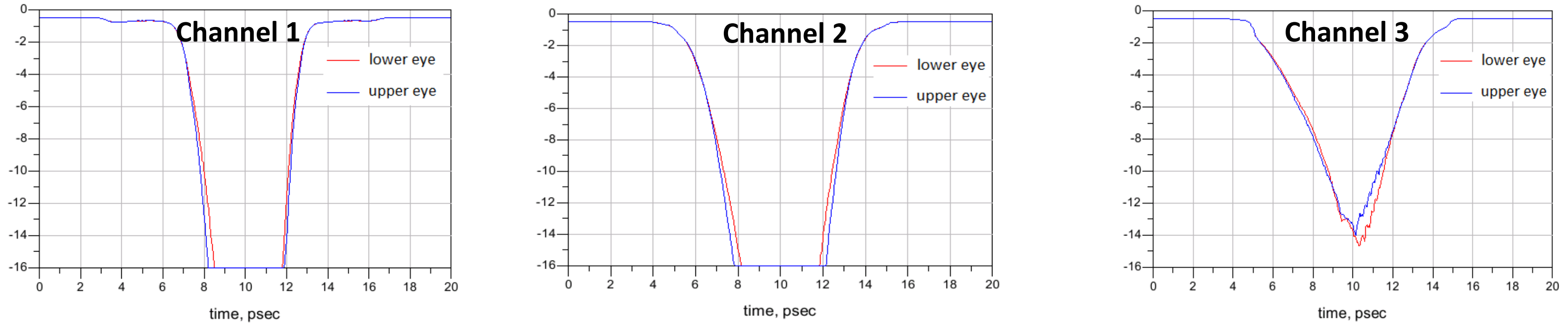



- Eye Contour at 1e-10, 1e-11, and 1e-12

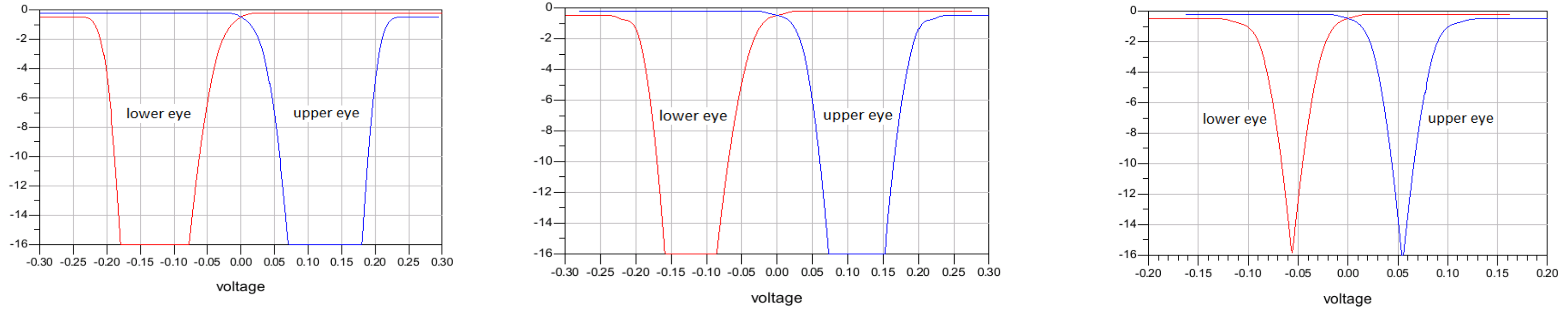


# Example of Using AMI Model for Duobinary

- Timing Bathtub Curves

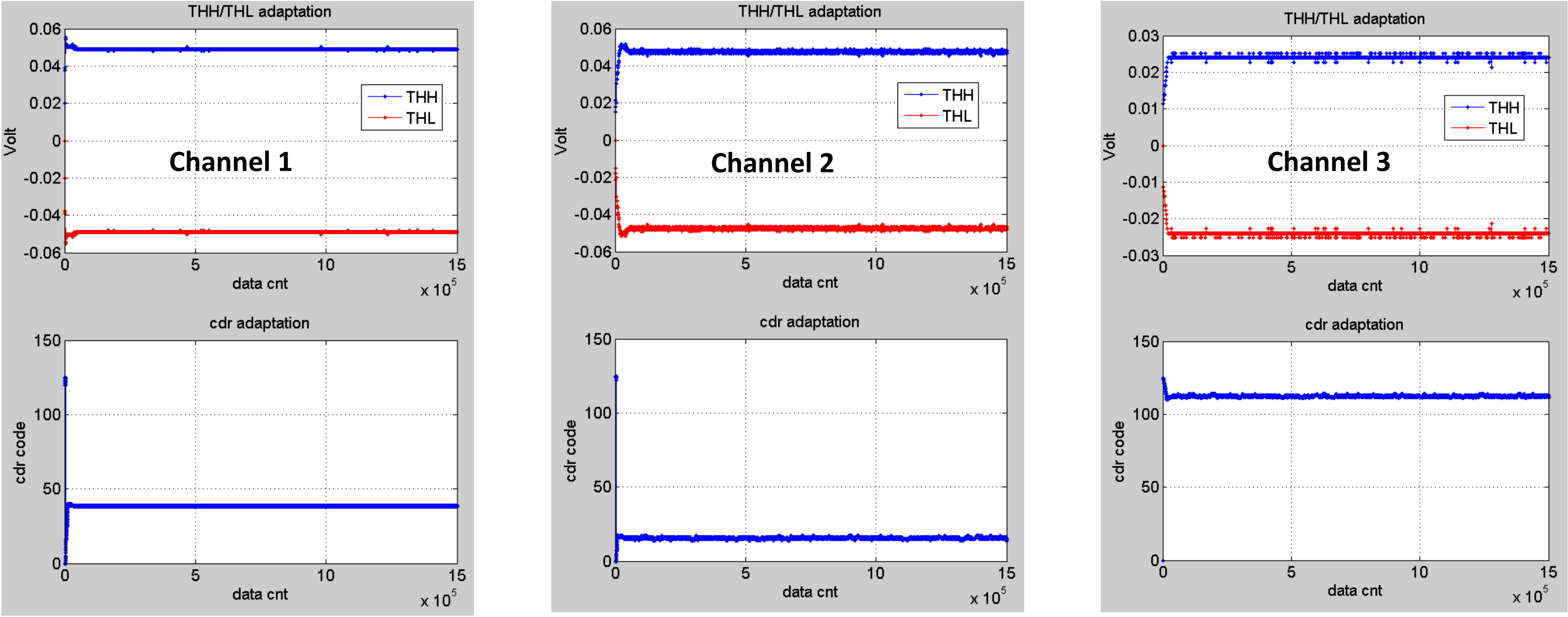



- Voltage Bathtub Curves

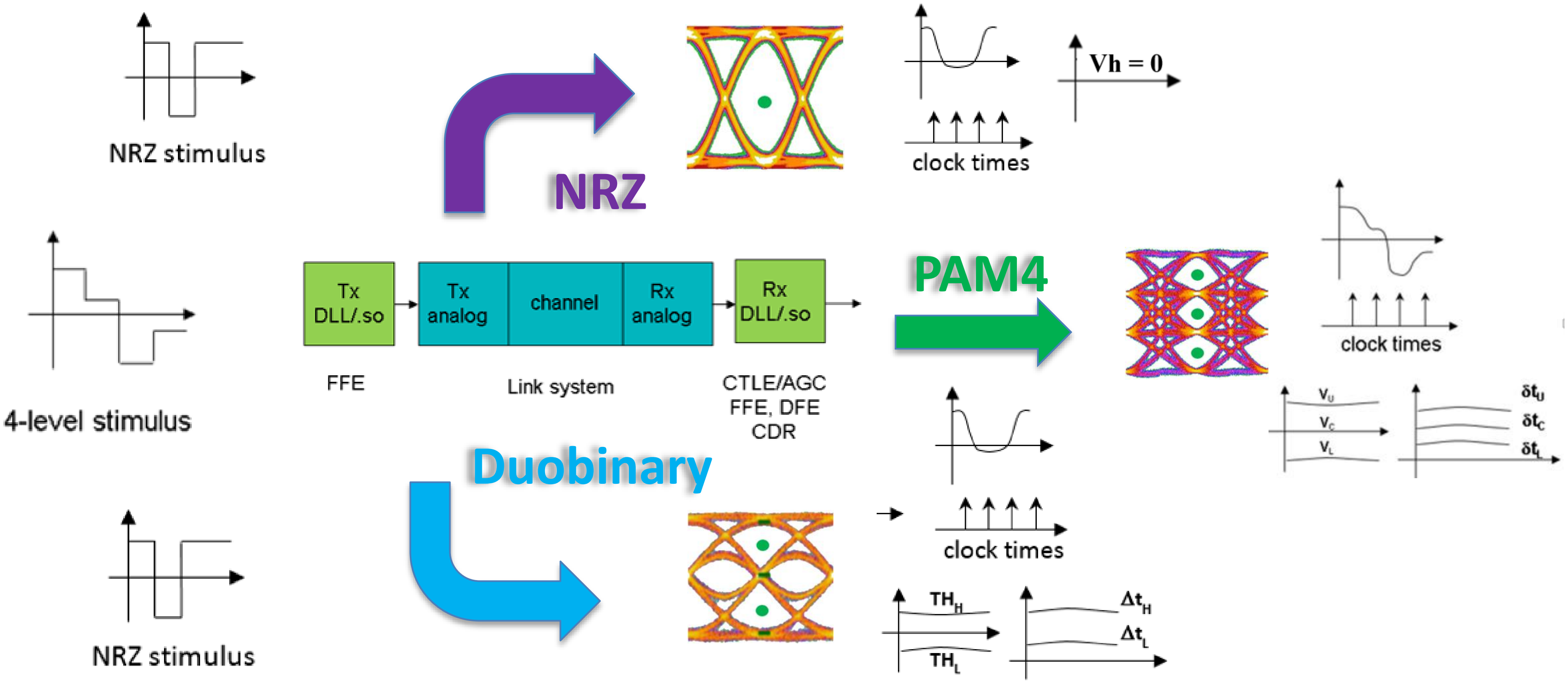


# Example of Using AMI Model for Duobinary

- Slicing level and CDR adaptation



# IBIS-AMI Modeling – A Unified Solution



# Conclusions and Future Work

- We proposed what is needed to enable IBIS-AMI simulation for duobinary
  - There is no change on the TX side in the AMI model
  - RX DLL needs to send two slicing levels, $TH_H$ and $TH_L$, to the EDA tool
  - EDA tool can opt for the precoding and de-precoding
- We verified this flow through simulations over three selected channels
- The EDA tool should be able to switch between NRZ, PAM4 and duobinary modulation schemes
  - A unified solution is discussed

# References

[1] A. Lender, “The duo-binary technique for high-speed data transmission”, IEEE Trans. Communications Electronics, vol. 82, no. 2, pp. 214-218, May 1963

[2] Timothy De Keulenaer, et al, “56+ Gb/s Serial Transmission using Duobinary Signaling”, DesignCon 2015

[3] Joris Van Kerrebrouck, et al, “100 Gb/s Serial Transmission over Copper using Duobinary Signaling”, DesignCon 2016

[4] Jeffrey Sinsky, et al, “Circuitry and method for multi-level signals”, US 9369317 B2

[5] Hongtao Zhang, et al, “IBIS-AMI Modeling and Simulation of 56G PAM4 Link Systems”, DesignCon 2015

# Thank you!

---

**QUESTIONS?**