IBIS-AMI Back-Channel System Optimization in Practice

Steven Parker, Marvell
Matthew Kelly, Marvell
Jared James, Cadence Design Systems
Ambrish Varma, Cadence Design Systems
Kumar Keshavan, Cadence Design Systems
Ken Willis, Cadence Design Systems

DesignCon IBIS Summit
Santa Clara, California
January 31, 2020

Authors

Steven Parker (Speaker)

Senior Staff Engineer, Marvell
sparker@marvell.com

Matthew Kelly

Senior Engineer, Marvell
mkelly@marvell.com

Authors (cont.)

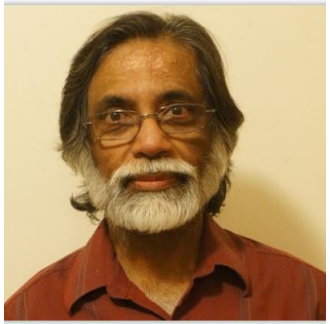
Jared James (Speaker)

Principal Product Engineer, Cadence Design Systems
jjames@cadence.com

Ambrish Varma

Senior Principal Software Engineer, Cadence Design Systems
ambrishv@cadence.com

Authors (cont.)



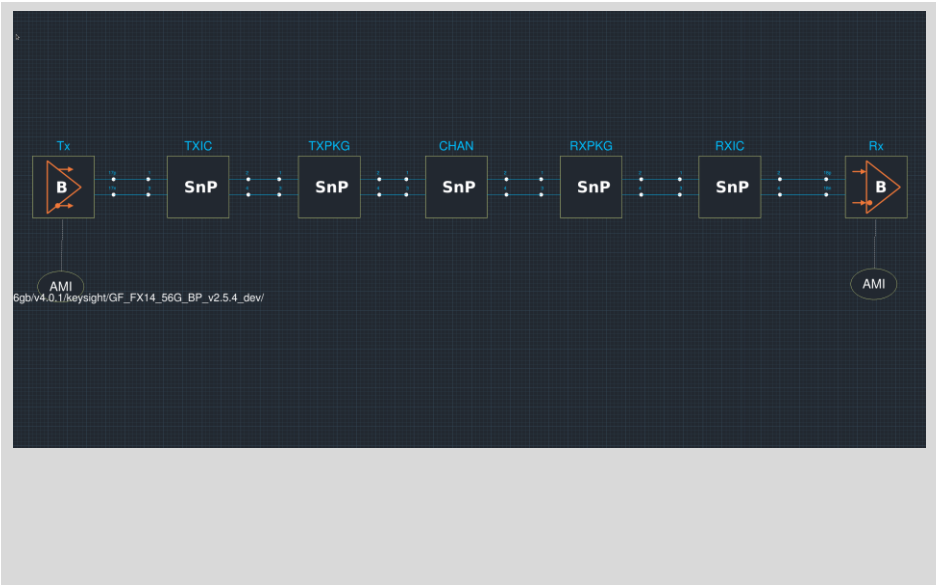
Kumar Keshavan

Senior Software Architect, Cadence Design Systems
ckumar@cadence.com

Ken Willis

Product Engineering Architect, Cadence Design Systems
kenw@cadence.com

Objectives and Topics Covered

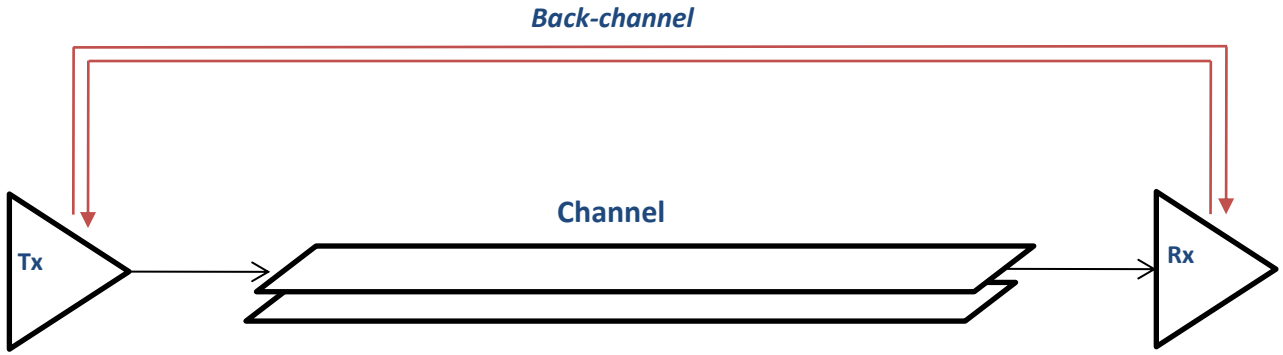


Cadence / Marvell IBIS AMI Back-channel Interoperability Demo

- Overview of the IBIS-AMI Back-channel Feature
- Interoperability considerations from the Cadence / Marvell IBIS AMI Back-channel Demo
- Results with and without BCI
- Potential enhancements to improve interoperability

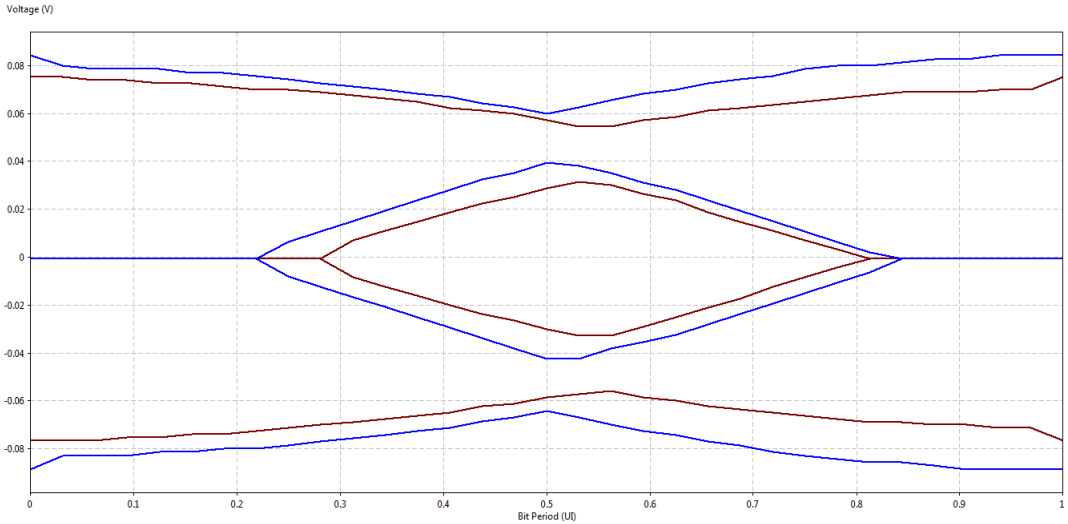
What is Back-channel Training?

Ability for a SerDes receiver in a serial link to automatically tune the equalization settings of its SerDes transmitter

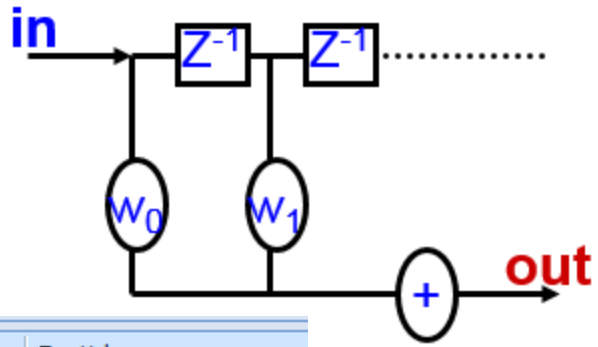


Why is Back-channel Training Important?

Optimizing the Tx and Rx settings ***together in combination*** often produces better results than optimizing either one by themselves



Feed Forward Equalization (FFE, at the Tx)

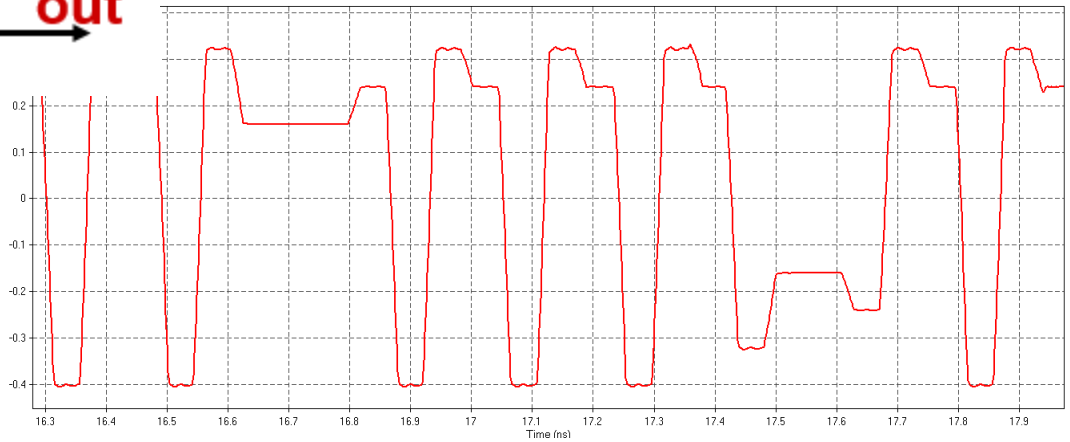


FFE configured in “taps” or boost drivers

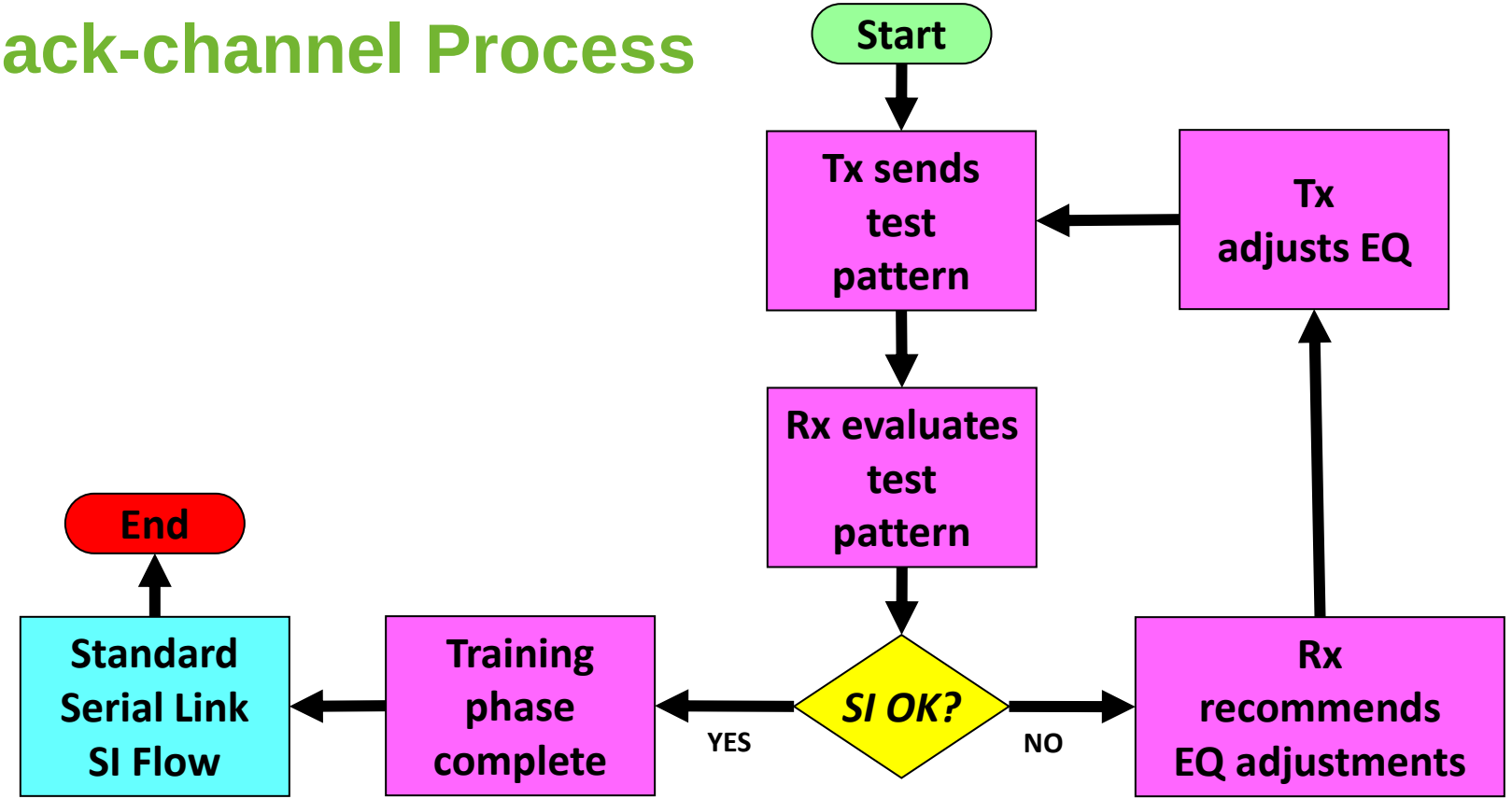
FFE taps have drive strength coefficients and limits

These can be adjusted by the Rx thru back-channel

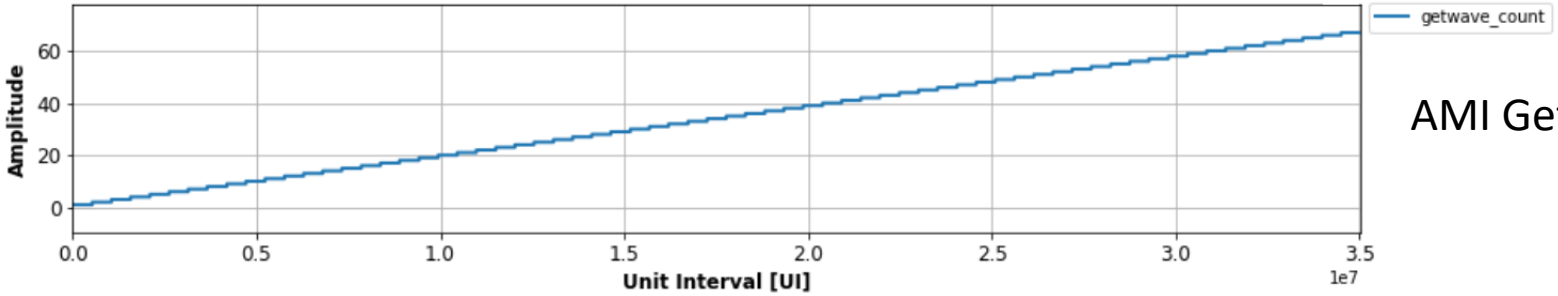
#	Tap Values
P0	0,0.75,-0.25
P1	0,0.833,-0.167
P2	0,0.8,-0.2
P3	0,0.875,-0.125
P4	0,1,0
P5	-0.1,0.9,0
P6	-0.125,0.875,0
P7	-0.1,0.7,-0.2
P8	-0.125,0.75,-0.125
P9	-0.166,0.834,0



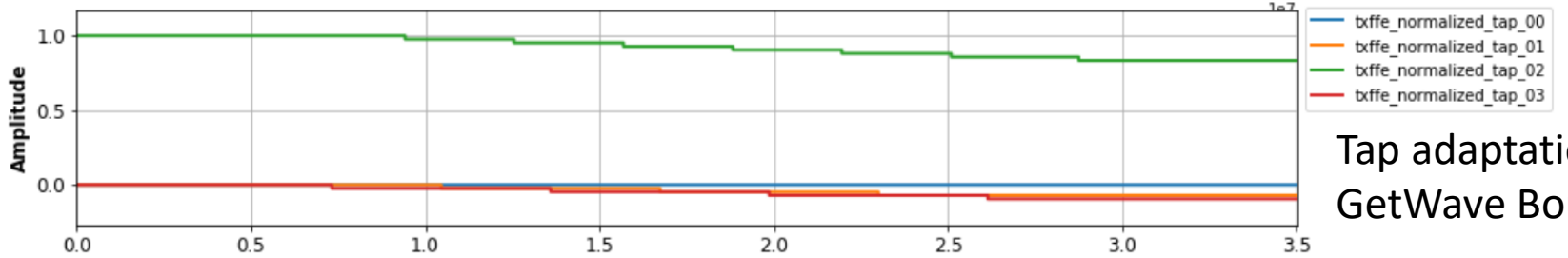
Back-channel Process



BCI Training Sequence Example



AMI GetWave Count



Tap adaptation on GetWave Boundaries

AMI Reserved_Parameters - Tx

BCI_Protocol

- Name of protocol for both Tx and Rx to use

BCI_ID

- Prefix for data file AMI models will pass back and forth

BCI_State

- Training On or Off

```
(BCI_Protocol (Usage Info ) (Type String ) (Value "cdns_bci")(Description "Name of private protocol." ))
(BCI_ID (Usage Info ) (Type String )  (Value "cdns_bci")(Description "BCI file related param." ))
(BCI_State (Usage Info )(Type String )  (Value "Training")(Description "Use 'off' to turn off BCI." ))
```

AMI Reserved_Parameters - Rx

All the same parameters as the Tx, plus two more

BCI_Message_Interval_UI

- Suggested waveform block size

BCI_Training_UI

- Max number of unit intervals to evaluate for training purposes

```
(BCI_Protocol (Usage Info ) (Type String ) (Value "cdns_bci")(Description "Name of private protocol." ))
(BCI_ID (Usage Info ) (Type String )  (Value "cdns_bci")(Description "BCI file related param." ))
(BCI_State (Usage Info )(Type String )  (Value "Training")(Description "Use 'off' to turn off BCI." ))
(BCI_Message_Interval_UI (Usage Info ) (Type Integer ) (Default 256) (Description "Suggested block size (not
used)." ))
(BCI_Training_UI (Usage Info ) (Type Integer ) (Default 150000) (Description "Max BCI training periond." ))
```

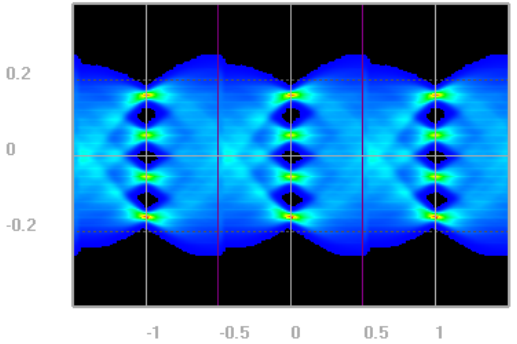
What the Back-channel BIRD specifies and what it doesn't….

- **"This specification describes an underlying mechanism for the AMI .ami file and the executable model to allow information to be transferred from the Tx to the Rx and from the Rx to the Tx without requiring the EDA tool to understand the content of this information, or even for the EDA tool to know that back-channel communications is occurring."**
- **"With the information provided in this specification, IC Vendors can develop models that support Back Channel Training in current IBIS AMI EDA tools. "**

This presentation focuses on how two IC vendors developed models to support interoperable Back-channel Training.

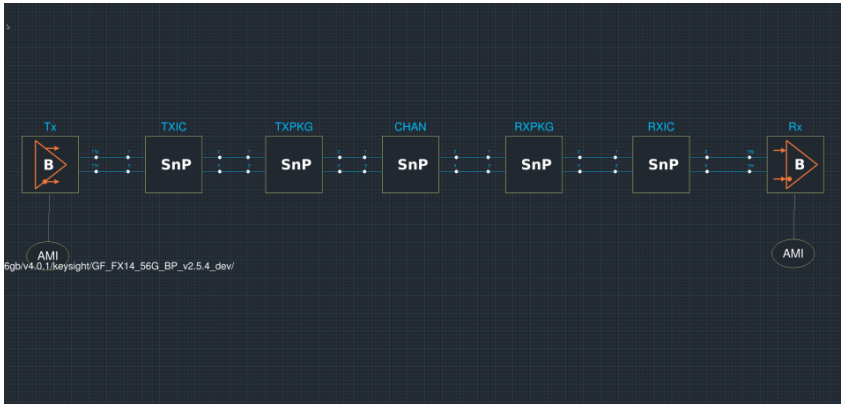
Back-channel Interoperability Demo

- **56G PAM4 model chosen**
- **Cadence EDA Software used for demonstration**
- **IEEE Contributed Channel “Bch2_a0_7_t.s4p” used for demo**



PAM4 Eye After Equalization including Back-channel optimization

Cadence Transmitter



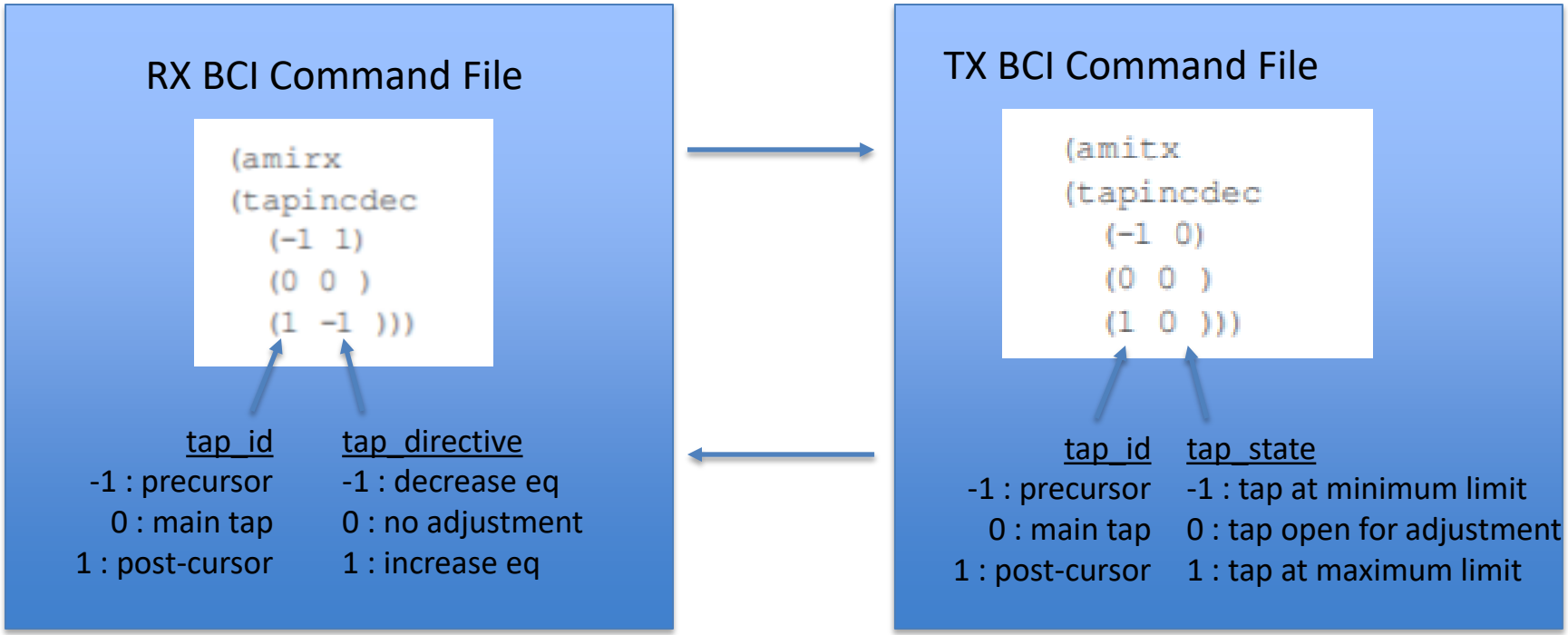
Marvell Receiver

Development Methodology

1. Marvell (development team from GlobalFoundries prior to acquisition) develop 56G PAM4 models and testbench, implementing “GF_HSS_BCI_V1” BCI_PROTOCOL in Marvell internal EDA software.
2. Cadence develops reference 56G PAM4 models and testbench, implementing “cdns_bci_default” BCI_PROTOCOL in Cadence EDA software.

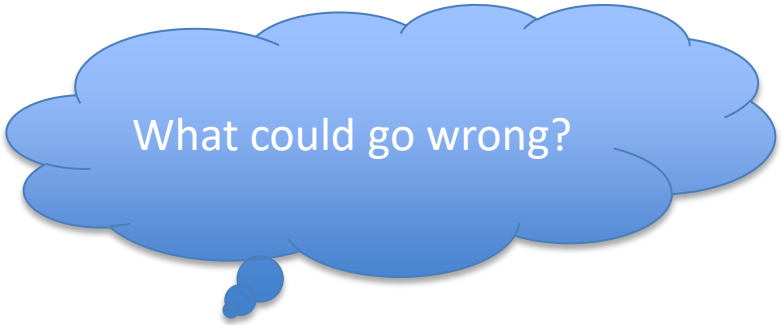
   Note: Although developed independently and not compatible, these protocols accomplish the same task.
3. Marvell 56G PAM4 models modified to implement “cdns_bci_default” BCI_PROTOCOL.
4. Cadence’s 56G PAM4 TX connected to Marvell’s 56G PAM4 RX implementing “cdns_bci_default” BCI_PROTOCOL in Cadence EDA software.

Implementing BCI_Protocol



What did we learn going through this?

- **BCI_Protocol Design Issues**
  - No command acknowledgement
    - *The only feedback from the TX was that an FFE tap hit a limit, there was no positive acknowledgement a command had been issued*
  - No sequence checking
    - *No checking for out of order command or stale BCI file*
- **BCI_Message_Interval_UI**
  - Since the BCI is written on GetWave boundaries, it is very important to set a BCI_Message_Interval_UI which is best for the RX
    - *Too small, and the RX model will not have sufficiently reacted to the previous TX update*
    - *Too large and the simulation time will be excessive.*
    - *1000 found to be ideal for Marvell / Cadence Interoperability Study*

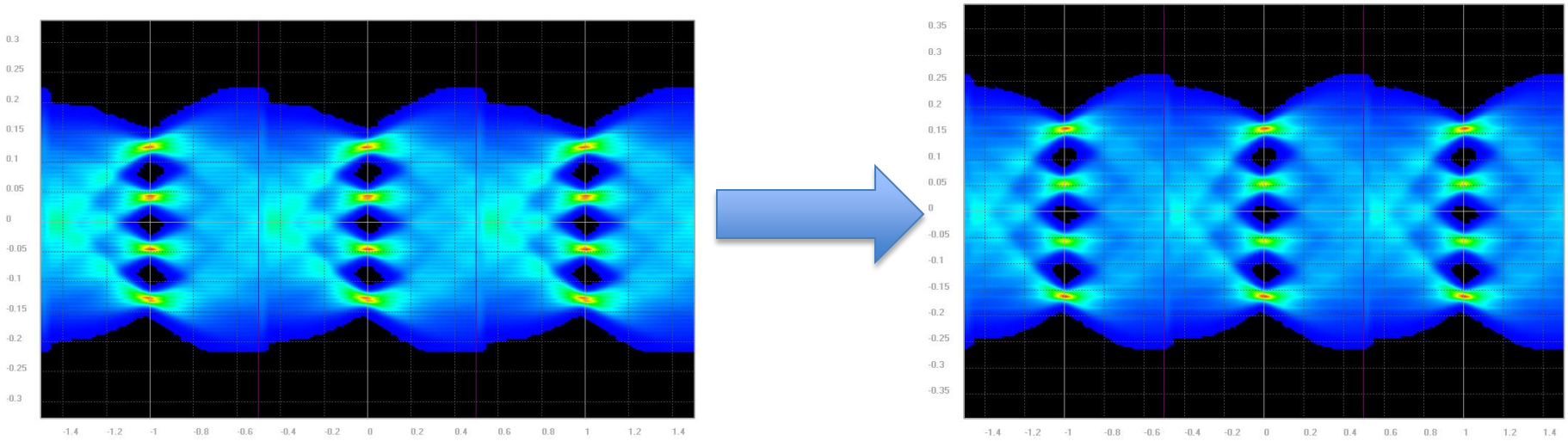


What did we learn going through this?

- **BCI_ID (base name for any data file written by model)**
  - Since both the TX and RX both use the same file and the BCI protocol looked very similar for both command and status, it was difficult to know which model wrote the file last
  - EDA tool renames this
  - BCI_ID in the GUI, not necessarily the filename used for the BCI file

- **Debugging Tips**
  - Have the TX model report a positive response on FFE commands which are executed
  - Be able to uniquely identify output from an TX vs RX

Results (with and without BCI)…



15-20% Eye Height Improvement seen in Interoperability Testbench

Potential Interoperability Pitfalls (and ideas to avoid them in the future)

- Implementing an incompatible BCI_Protocol
  - Since every model maker is left to implement their own BCI Protocol, no two protocols may be compatible. To improve compatibility:
    - *Propose a de facto standard BCI_Protocol to support major training schemes*
    - *Model makers could open source their BCI Protocol schemes to ensure common usage*
    - *IBIS could consider a follow-on BIRD to standardize a protocol standard*

- Debugging a BCI_Protocol
  - A BCI_Protocol should include the following:
    - *Positive acknowledgement of TX commands whether executed or not executed due to some limitation*
    - *Sequence tracing of the BCI packets*
  - IBIS AMI models supporting the BCI_Protocol should have verbose modes to report training status. Detailed log files for debug.

Summary

- The IBIS-AMI Back-channel support provides a useful means to support Back-channel in AMI simulations
- Inter-vendor Back-channel simulations require upfront planning to be efficiently implemented
- More work can be done to improve inter-vendor Back-channel interoperability

Thank you!

QUESTIONS?