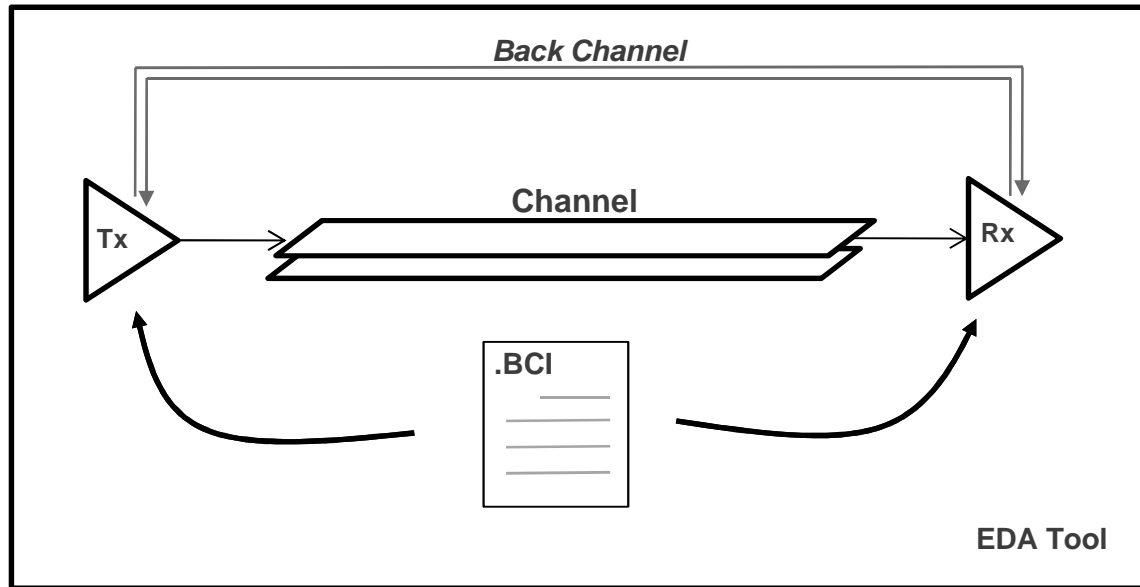# The BCI File

Ambrish Varma
05/06/2014

cādence®

Agenda

- BCI Basics

- Link to .BCI from .AMI

- What's in a BCI File?

- BCI File Examples
  - Published Protocols
  - Private Protocols

- BCI Communications

- Advantages of BCI

cādence®

BCI Basics

- Both Tx and Rx models will refer to the same BCI Protocol file.
  - This will remove any redundancy with Reserved_Parameters in Tx and BCI file in Rx.

- BCI Files will be discussed and approved in the IBIS/ATM every time there is a new protocol.

- All approved BCI Files will be publicly accessible at the IBIS/ATM website

cādence®

Link to BCI from AMI File

- ## 2 Top Level Reserved Parameters for .AMI file :

(**Training** (Usage In)(Type Integer) (List 0 1 2)
        (Default 0) (Description " This model supports both Init based and Getwave based backchannel training.
        Training Mode – 0 is off, 1 is Getwave based and 2 is Init based."))

AND

Pointers to BCI file

(**Backchannel_Protocol** (Usage In) (Type String) (List "None" "standard1.bci" "standard2.bci" "standard3.bci" "standard4.bci")
        (Default "standard1.bci") (Description "This Device can support back-channel training for multiple standards.")))

- ## Clean, keeps Backchannel related parameters in BCI file

cādence®

What is contained in a BCI File?

- **Training Pattern/Sequence.**
  - Most protocols define a data sequence with a set preamble and a postamble

- **Training Length/Max number of training bits.**
  - This is much lower than suggested in real devices where training can last for milliseconds. Software will be faster than actual device
    - safe to say 100-150K bits will be the norm. Considering the relatively slow adaptation of Tx (every 10k bits Vs 1k-2k bits for CTLE/DFE) it should give 10-15 alterations for Tx which is more than sufficient.

- **Declaration for minimum amount of Tx taps which Rx needs to accommodate**
  - It does not prevent Tx declaring the existence of more taps and Rx giving alteration suggestion to them. This happens seamlessly using the backchannel string between the Tx and Rx.

cādence®

802.3KR BCI File

Only for BCI

```
(802.3KR
  (Reserved_Parameters
      (BCI_Version (Usage Info) (Type String) (Value "6.1"))
      (Training_Pattern
              (Preamble (Usage Info) (Type Bits) (Bit_Pattern
                              11111111111111110000000000000000 1)  (Description "Leading preamble pattern."))
              (Data (Usage Info) (Type Bits) (LFSR 1,9,11  11010101011  4096) (Description "Training  pattern with a seed of
                              11010101011."))
              (Postamble (Usage Info) (Type Bits) (Bit_Pattern 00 1) (Description "Trailing postamble  pattern."))
       )
      (Max_Train_Bits (Usage Info) (Type Integer) (Value 500000)
           (Description "Number of total training bits allowed"))
      (Training_Done (Usage InOut) (Type Boolean) (Default False)
           (Description "If True then training is done"))
  )
```

Analogous to Model_Specific

```
(Protocol_Specific
     (BCI
         (taps
               (-1 (Usage InOut) (Type Tap) (Range 0 -1  1) (Default 0)
                                              (Description "Parameter name is standard-specific, and can be any legal Type"))
               (0  (Usage InOut) (Type Tap) (Range 0 -1 1) (Default 0)
               (Description "Parameter name is standard-specific, and can be any legal Type"))
               (1  (Usage InOut) (Type Tap) (Range 0 -1 1) (Default 0)
               (Description "Parameter name is standard-specific, and can be any legal Type"))
   ))))
```

cādence®

PCIe3 BCI File

```
(PCIe3
 (Reserved_Parameters
     (BCI_Version (Usage Info) (Type String) (Value "6.1"))
     (Training_Pattern
             (Data (Usage Info) (Type Bits) (Bit_Pattern r 4096) (Description "Random Training  pattern ."))
 )
     (Max_Train_Bits (Usage Info) (Type Integer) (Value 500000)
         (Description "Number of total training bits allowed"))
     (Training_Done (Usage InOut) (Type Boolean) (Default False) (Description "If True then training is done"))
)


(Protocol_Specific
     (BCI
         (taps
             (-1 (Usage InOut) (Type Tap) (Range 0 -1  1) (Default 0)
             (Description "Pre cursor"))
             (0  (Usage InOut) (Type Tap) (Range 0 -1 1) (Default 0)
             (Description " Main cursor "))
             (1  (Usage InOut) (Type Tap) (Range 0 -1 1) (Default 0)
             (Description " Post cursor "))
  ))))
```

cadence®

12GBPS SAS-3

- Mine the Phy layer and Link layer sections for information for the BCI file:

Table 73 — Transmitter training pattern

Transmitter training pattern	Description
Train_Tx pattern	Sequence of: 1) pattern marker (see 5.10.4.2.3.4.3); 2) TTIU (see 5.10.4.2.3.5); and 3) 58 data dwords set to 00000000h that are transmitted scrambled and 8b10b encoded.

Table 63 — COEFFICIENT 1 REQUEST field, COEFFICIENT 2 REQUEST field, and COEFFICIENT 3 REQUEST field

Code	Name	Description	Type
00b	hold	The local phy shall make no adjustment to the specified coefficient.	M
01b	increment	The local phy shall adjust the specified coefficient by one increment. [a]	M
10b	decrement	The local phy shall adjust the specified coefficient by one decrement. [a]	M

cādence®

SAS-3 BCI File

```
(SAS3
 (Reserved_Parameters
     (BCI_Version (Usage Info) (Type String) (Value "6.1"))
     (Training_Pattern
             (Preamble (Usage Info) (Type Bits) (Bit_Pattern
                             11111111111111111100000000000000000000 1)  (Description "Pattern Marker."))
             (Data (Usage Info) (Type Bits) (LFSR 1,4,13,15,16  1111111111111111 4096) (Description "Training  pattern with a seed of
                             1111111111111111."))
      )
     (Max_Train_Bits (Usage Info) (Type Integer) (Value 500000)
          (Description "Number of total training bits allowed"))
     (Training_Done (Usage InOut) (Type Boolean) (Default False)
          (Description "If True then training is done"))
 )


(Protocol_Specific
     (BCI
         (taps
             (-1 (Usage InOut) (Type Tap) (Range 0 -1  1) (Default 0)
                                             (Description "Parameter name is standard-specific, and can be any legal Type"))
             (0  (Usage InOut) (Type Tap) (Range 0 -1 1) (Default 0)
             (Description "Parameter name is standard-specific, and can be any legal Type"))
             (1  (Usage InOut) (Type Tap) (Range 0 -1 1) (Default 0)
             (Description "Parameter name is standard-specific, and can be any legal Type"))
  ))))
```

cādence®

Other Protocols…

DesignCon 2012

TX back channel adaptation
algorithm and protocol emulation
with application to PCIe, SAS, FC,
and 10GBASE-KR

Mohammad S. Mobin, LSI Corporation
md.mobin@lsi.com, 610.712.5829

Amaresh Malipatil, LSI Corporation
[amaresh.malipatil@lsi.com, 408.433.7195]

Pervez Aziz, LSI Corporation
[pervez.aziz@lsi.com, 972.244.5025]

Adam Healey, LSI Corporation
[adam.healey@lsi.com, 781.672.3508]

cādence®

'Private' BCI File

From 04/29/14 Meeting Minutes

> **- Michael M.: Are we really saying a proprietary interface would not be supported?**
>   **- Companies might agree to make something work a certain way, independent of a formal protocol.**
>
> **- Walter: You could agree to a private BCI file if the EDA tool knows nothing about it.**
>   **- My proposal would not work for proprietary protocols, BIRD 147 would.**

● Company A can make a 'private' .BCI file and send it to Company B to disclose what/how the Backchannel interaction is performed for their Tx/Rx.

  – Convenient, wholly contained

  – Separate from .AMI

  – Works in the same fashion as published protocol Backchannel training.

  – EDA tool does not need to do anything different for private .BCI.

cādence®

A 'Private' BCI File

```
(MyPrivateBCI
 (Reserved_Parameters
     (BCI_Version (Usage Info) (Type String) (Value "6.1"))
     (Training_Pattern
             (Preamble (Usage Info) (Type Bits) (Bit_Pattern
                               1111111111111110000000000000000 1)  (Description "Leading preamble pattern."))
             (Data (Usage Info) (Type Bits) (LFSR 1,9,11  11010101011  4096) (Description "Training  pattern with a seed of
                         11010101011."))
             (Postamble (Usage Info) (Type Bits) (Bit_Pattern 00 1) (Description "Trailing postamble  pattern."))
      )
     (Max_Train_Bits (Usage Info) (Type Integer) (Value 5000)
         (Description "Number of total training bits allowed"))
     (Training_Done (Usage InOut) (Type Boolean) (Default False)
         (Description "If True then training is done"))
 )


(Protocol_Specific
    (BCI
       (taps
```

This method may only need 2 pre taps

```
             (-2  (Usage InOut) (Type Tap) (Range 0 -1 1) (Default 0)
             (Description "2nd Pre Cursor"))
             (-1 (Usage InOut) (Type Tap) (Range 0 -1  1) (Default 0)
             (Description "1st Pre Cursor"))
             (0  (Usage InOut) (Type Tap) (Range 0 -1 1) (Default 0)
             (Description "Main Cursor"))
))))
```

cādence®

How/When Will a BCI file Originate

- Be based on prior approved BCI files

- Incorporate Minimal Change, if any.

- When there is a new Standard, need to decide on
  - Training Pattern/Sequence
  - Training Length/Max number of training bits.
  - Declaration for minimum amount of Tx taps which Rx needs to accommodate
  - Any other protocol specific parameters that may be needed to be passed back and forth
    - Tap Resolution (for ex?)

cādence®

How is information extracted from the BCI file

- EDA Tool

- AMI Model

cādence®

EDA Tool and the BCI file

```
(802.3KR
 (Reserved_Parameters
      (BCI_Version (Usage Info) (Type String) (Value "6.1"))
      (Training_Pattern
                (Preamble (Usage Info) (Type Bits) (Bit_Pattern
                               1111111111111110000000000000000 1)  (Description "Leading preamble pattern."))
                (Data (Usage Info) (Type Bits) (LFSR 1,9,11  11010101011  4096) (Description "Training  pattern with a seed of
                               11010101011."))
                 (Postamble (Usage Info) (Type Bits) (Bit_Pattern 00 1) (Description "Trailing postamble  pattern."))
       )
      (Max_Train_Bits (Usage Info) (Type Integer) (Value 500000)
            (Description "Number of total training bits allowed"))
      (Training_Done (Usage InOut) (Type Boolean) (Default False)
            (Description "If True then training is done"))
  )
)
```

‘Reads’ the Reserved Parameters section

  – Implements the Training Pattern
  – Notes the Max_Train_Bits and shuts off communication after max_train_bits
  – Based on the ‘Training’ parameter, operates in Getwave or Init modes.

cādence®

How is information extracted from the BCI file

- EDA Tool

- AMI Model

cādence®

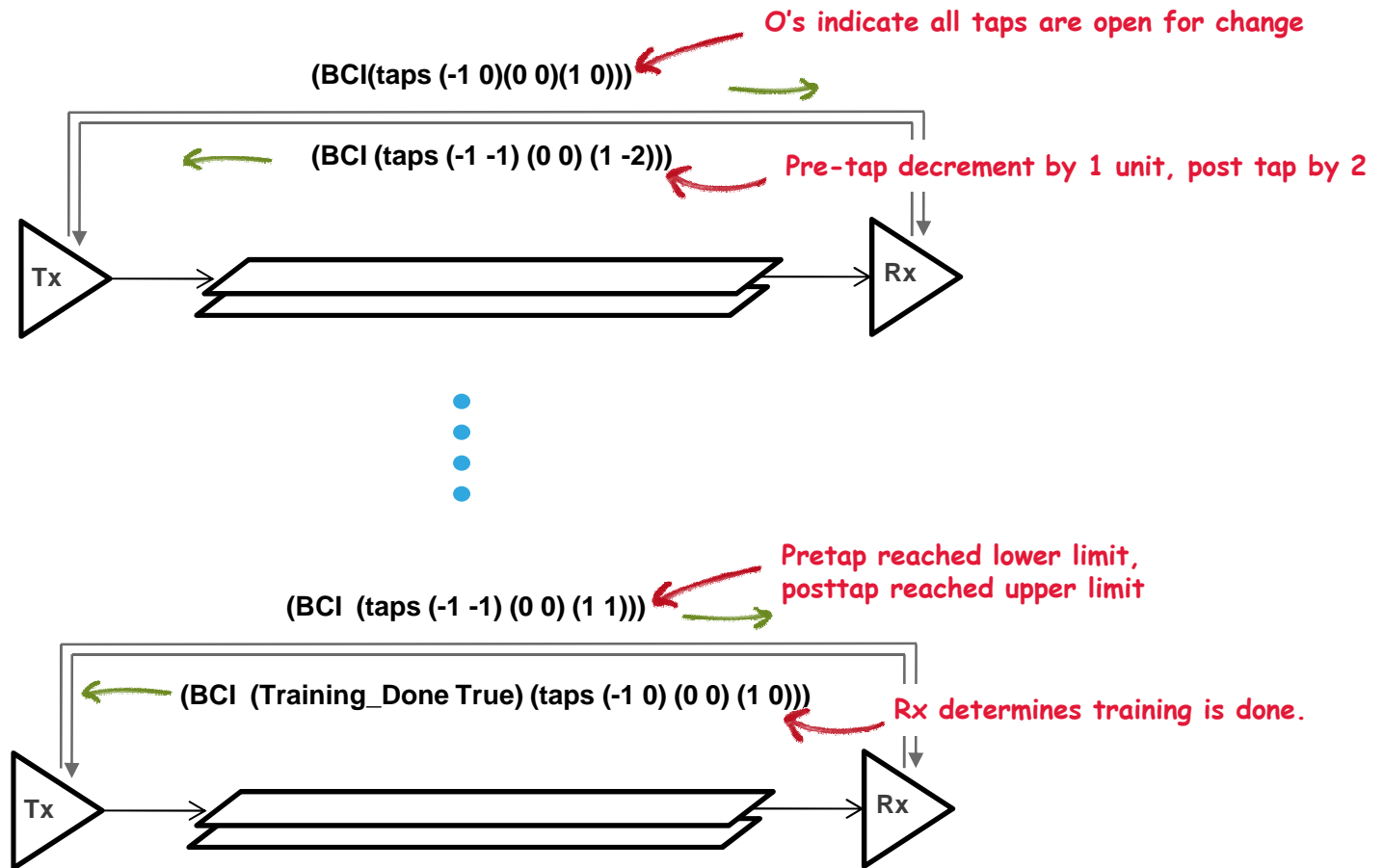# AMI Model and the BCI file

```
(802.3KR
(Protocol_Specific
    (BCI
        (taps
                (-1 (Usage InOut) (Type Tap) (Range 0 -1  1) (Default 0)
                                    (Description "Parameter name is standard-specific, and can be any legal Type"))
                (0  (Usage InOut) (Type Tap) (Range 0 -1 1) (Default 0)
                (Description "Parameter name is standard-specific, and can be any legal Type"))
                (1  (Usage InOut) (Type Tap) (Range 0 -1 1) (Default 0)
                (Description "Parameter name is standard-specific, and can be any legal Type"))
    ))))
```

- Tx model 'Reads' the Protocol_Specific section and constructs backchannel communication string with root name 'BCI'.
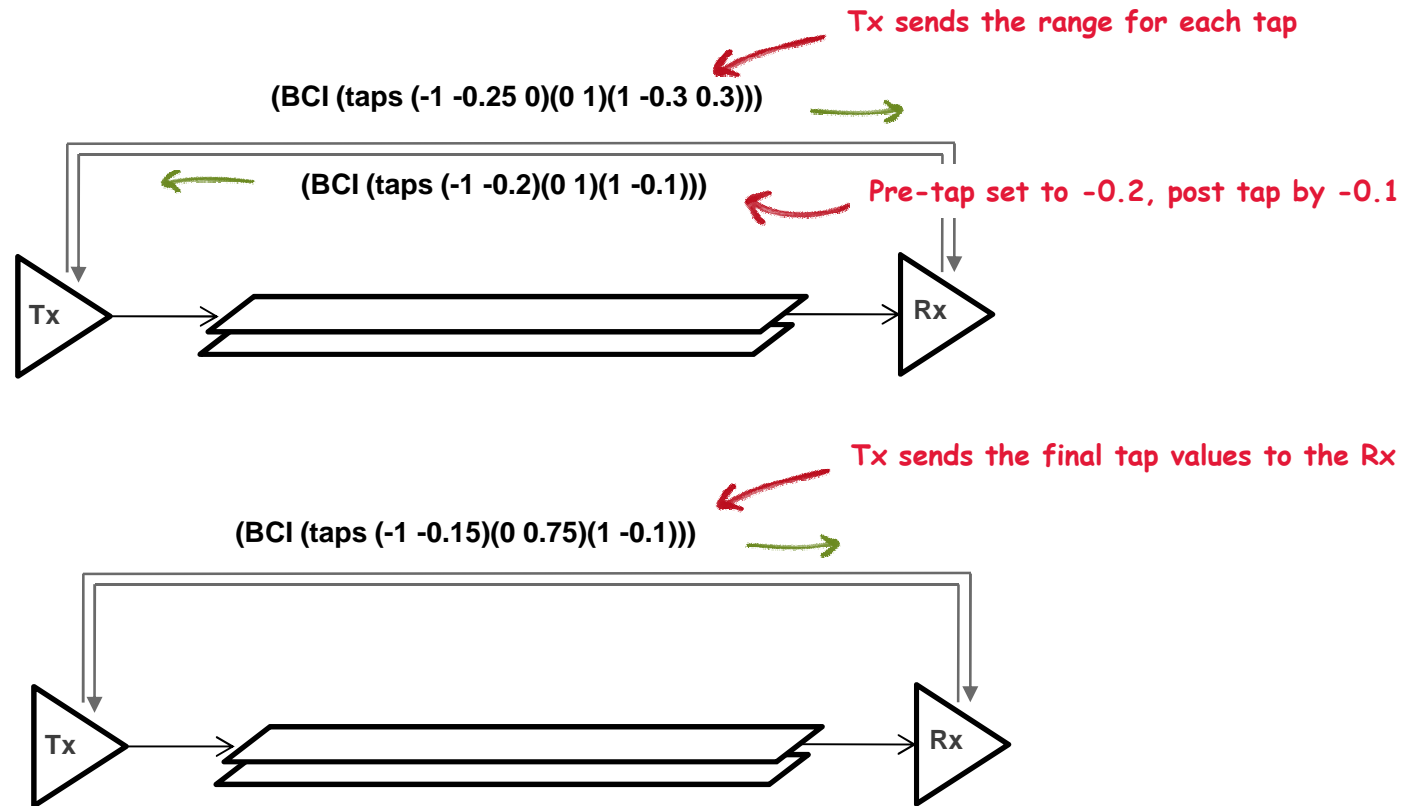
(BCI(taps (-1 0)(0 0)(1 0)))

cadence®

AMI_Getwave based flow

O's indicate all taps are open for change

(BCI(taps (-1 0)(0 0)(1 0)))

(BCI (taps (-1 -1) (0 0) (1 -2)))

Pre-tap decrement by 1 unit, post tap by 2

Tx   **Rx**

Pretap reached lower limit,
posttap reached upper limit

(BCI (taps (-1 -1) (0 0) (1 1)))

(BCI (Training_Done True) (taps (-1 0) (0 0) (1 0)))

Rx determines training is done.

Tx   **Rx**

- Increment/Decrement for training Tx taps

- Rx can change taps every cycle, or wait to receive enough bits from Tx before suggesting change

- Tx can change taps based on Rx suggestions. Tx may not change at all based on its constraints.

cādence®

AMI_Init based flow

Tx sends the range for each tap

(BCI (taps (-1 -0.25 0)(0 1)(1 -0.3 0.3)))

(BCI (taps (-1 -0.2)(0 1)(1 -0.1)))

Pre-tap set to -0.2, post tap by -0.1

Tx sends the final tap values to the Rx

(BCI (taps (-1 -0.15)(0 0.75)(1 -0.1)))

- 2 pass Tx-Rx communication
- Rx uses the Impulse Response that has been modified using the feedback from the Rx, then runs its own AMI_Init.

cādence®

Advantages of BCI

- Declare training pattern to EDA tool and Models

- Lets one AMI model support multiple protocols
  - Significant when we consider future generations for each protocols.

- Prevents adding Reserved_Parameters in AMI file in the future

- Allows 'Protocol_Specific' parameters that are specific to a particular protocol to be included without adding new Reserved_Parameters in .AMI.

- Quicker time to market
  - Bypasses yearly approval cycle for IBIS to approve new Reserved_Parameters

- Allows a standard, consistent way of performing back channel without allowing different vendors their own flavor of the high speed protocols.

cādence®

cadence®

References for Backchannel training

- http://www.hgst.com/tech/techlib.nsf/techdocs/9BF6D9A994F2E94586257B8600679A52/$file/12GbpsSAS_KeyConsid_WP.pdf.

- http://cp.literature.agilent.com/litweb/pdf/5991-4080EN.pdf.

```
From: si-list-bounce@xxxxxxxxxxxxx [mailto:si-list-bounce@xxxxxxxxxxxxx]
On Behalf Of Ravinder.Ajmani@xxxxxxxxxxxxxx
Sent: Tuesday, December 13, 2011 10:28 AM
To: si-list@xxxxxxxxxxxxx
Subject: [SI-LIST] Backchannel Training in IBIS-AMI Modeling

Hi Experts,
The IBIS-AMI models currently being provided by our ASIC vendor do not
have Backchannel Training function.  I believe the IBIS-AMI simulators
also do not support this feature.  There was some discussion about this in

DesignCon 2011.  Can someone comment on when this is likely to be
implemented.

Thanks.
```

cādence®