BIRD201 - Back-channel Statistical Optimization

Walter Katz

Eric Brock

DesignCon IBIS Summit
Santa Clara, California
January 31, 2020

Overview

- Back-channel Time Domain Optimization (BIRD 147)
- Back-channel Statistical Optimization (BIRD 201)
- BCI_Protocol
- DDR5 DQ Write Protocol
- Results Presented on Two Examples
  - DDR5 DQ Write – comparing both BCI statistical and time domain optimization
  - 56G PAM4 - BCI statistical optimization
- Future Optimization Methods Unleashed by Back-channel Statistical Optimization

Back-channel Time Domain Optimization (BIRD 147)

- BIRD147 has been approved in IBIS 7.0
- Uses the existing AMI_GetWave calls
- Tx and Rx communicate using file I/O (filename determined by the AMI Reserved Parameters BCI_ID)

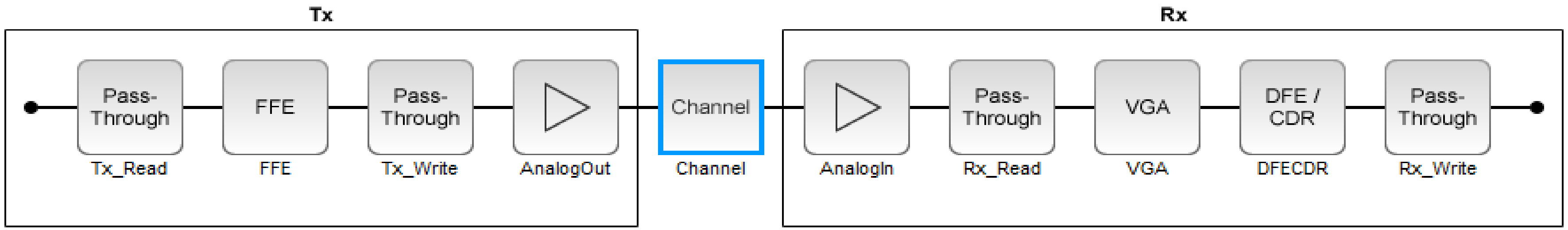
Back-channel Statistical Optimization (BIRD 201)

- Currently under consideration in the IBIS Open Forum
- New AMI Reserved Parameter BCI_Training_Mode that allows statistical optimization, time domain optimization or both.
- New AMI function AMI_Impulse. The EDA tool alternatively calls the Tx and Rx AMI_Impulse function until BCI_State is “Converged”.
- Reserved AMI Parameters **BCI_parameters_out and *BCI_parameters_in are used to communicate between the Tx and Rx AMI_Impulse functions

BCI_Protocol

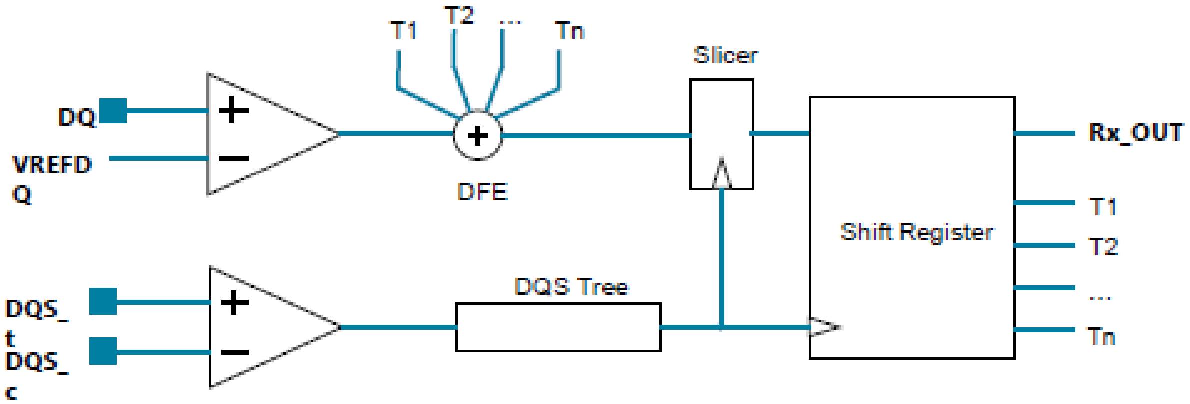
- BIRDs 147 and 201 only describe the method that the Tx and Rx communicate.
- The BCI_Protocol determines the content of what is communicated in the BCI_ID file and the **BCI_Parameters_out string.
- The BCI_Protocol can be a private protocol, a published protocol or a protocol approved by the IBIS Open Forum.
- The details of how BCI_Protocol's are approved by the IBIS Open Forum has yet to be determined.
- The model maker is responsible for developing optimization methods.
- Results presented in this presentation use simple sweeping of tap setting and either Eye Height or COM (S/N) metrics.

Statistical Iteration When Tx is Training (DDR5 Write)



1. Tx_Read reads Rx rx2tx protocol data, calculates metric, decides next settings
2. Tx runs FFE and outputs Tx Impulse Response
3. Tx_Write writes Tx tx2rx protocol data
4. Rx_Read reads Tx tx2rx protocol data, sets VGA Gain and DFE taps
5. Rx runs VGA and DFE/CDR
6. Rx_Write writes Rx rx2tx protocol data
7. If BCI_State is still “Training” then go to Step 1 above.

DDR5 DQ Write Hardware Overview



- DDR5 DQ is single ended. The DDR5 DRAM input buffer has a differential input/gain stage and a 4 tap DFE
- The reference voltage at the differential input is set by a VrefDQ register.
- The DRAM input has no CDR or DFE adaptation.
- The data is sampled at the latch using a DQS clock that is generated by the controller. The controller is responsible for setting the VrefDQ, Gain, DFE taps and DQS/DQ skew.
- The controller determines the channel eye height margin by sweeping the VrefDQ register and detecting errors in a PRBS pattern.

DDR5 DQ Write Protocol, Tx **BCI_parameters_out

- Tells Rx how to set its gain and taps

```
(BCI
    (BCI_State “Training|Converged”)
     (Gain    <gain in dB>)   | Integer, (Range 0 -3 3)
     (DFE1  <tap1>)           | Integer, (Range 0 -40 10)
     (DFE2  <tap2>)           | Integer, (Range 0 -15 15)
     (DFE3  <tap3>)           | Integer, (Range 0 -12 12)
     (DFE4  <tap4>)           | Integer, (Range 0 -9 9))
```

```
(BCI (BCI_State “Training”) (Gain 0) (DFE1 0 ) (DFE2 0 ) (DFE3 0 ) (DFE4 0))
```

DDR5 DQ Write Protocol, Rx **BCI_parameters_out

| Rx sets up Gain and DFE taps from Tx instructions in Tx (BCI …)

| Rx tells Tx what it did with the following (BCI …)

```
(BCI
        (BCI_State “Training”)
        (BER <used BER>)     | May be > 1e-16, if no 1e-16 contour
        (Eye_Height <eye height at used BER>)
        (Eye_Area <eye area at used BER>) )
```

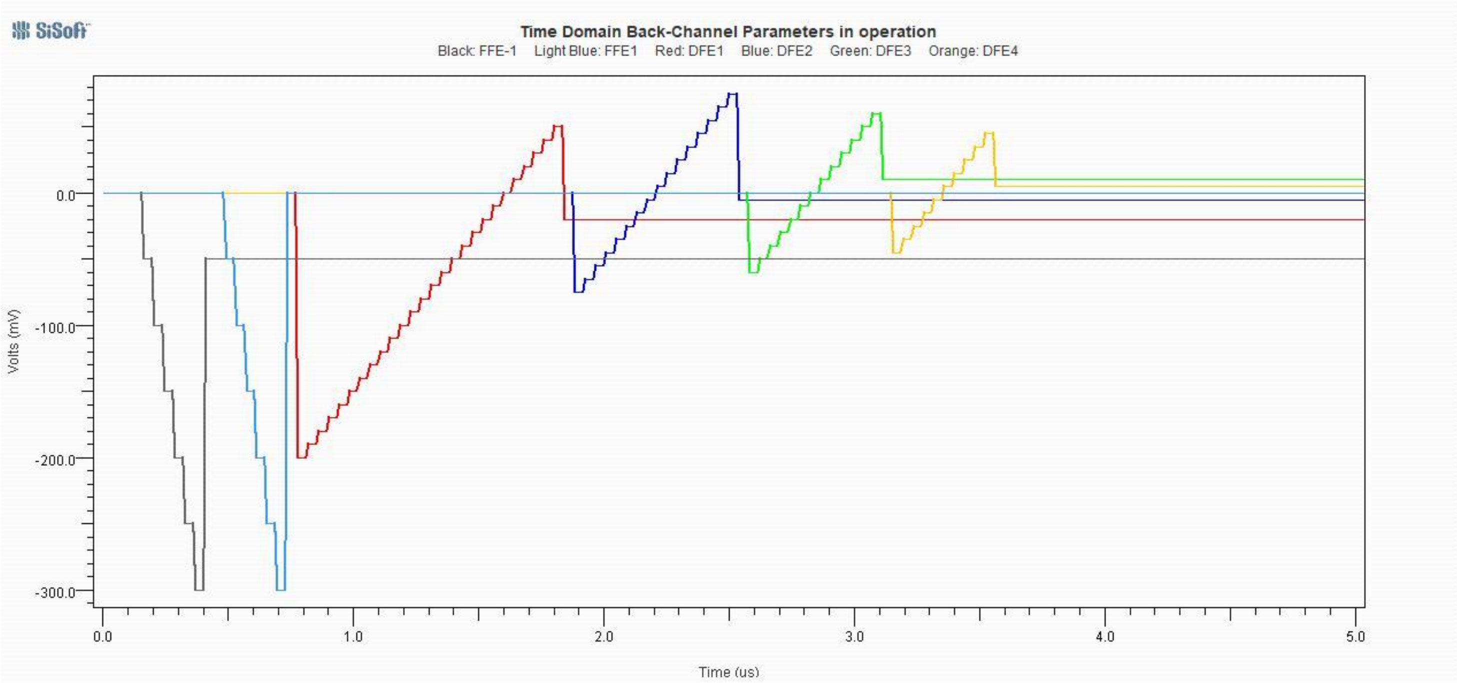
(BCI (BCI_State “Training”) (BER 1e-16) (Eye_Height .3) (Eye_Width 50e-12) (Eye_Area 30e-12))

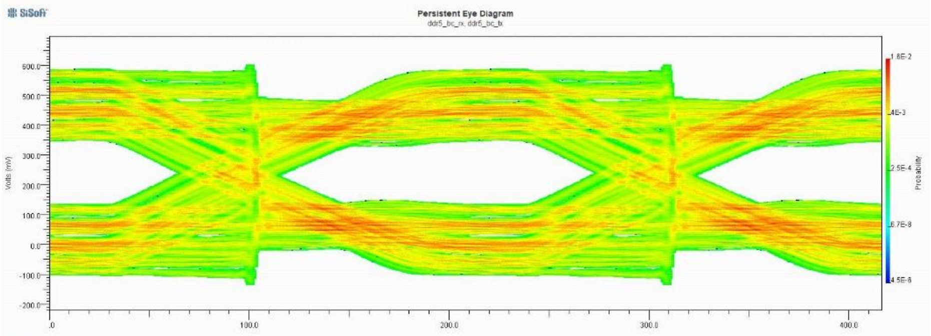
This is a Preliminary DDR5 DQ Write Protocol

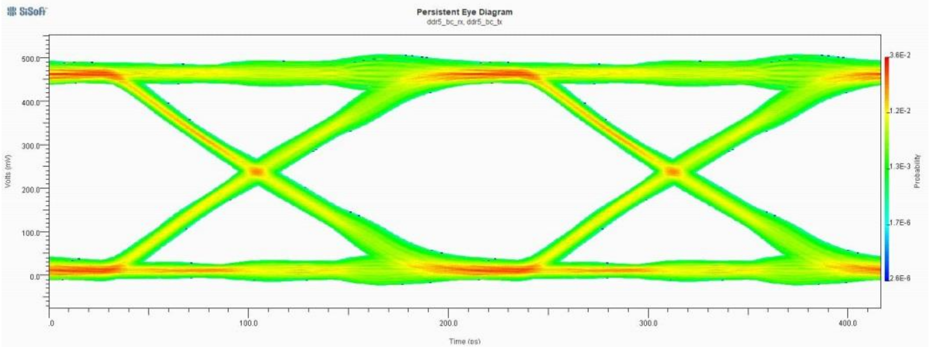
- The protocol will include a detailed method for the Rx to calculate the Eye Height and Eye Area Metrics from an Impulse Response
- Memory and Controller vendors may want to add
  - Jitter and noise impairments to the protocol
  - VrefDQ register value to the protocol
  - Additional or alternative metrics
- We would like DDR5 Memory and Controller vendors to contribute to this specification by attending or following the work we will be doing on this protocol in the IBIS-ATM (Advanced Technology Modeling) sub committee (http://ibis.org/atm_wip/).

BCI Time Domain and Statistical Optimization Converge to Same Equalization Solution

- Time Domain optimization (5 minutes) is ~100 times slower than Statistical optimization (5 seconds).
- The higher performance of BCI Statistical optimization is required to enable new Artificial Intelligence methods including Machine Learning, Genetic Optimization, Deep Learning and Reinforcement Learning.







56G PAM4 SerDes Statistical Training Extends the Rx Training Algorithm Presented Earlier this Week

DesignCon paper “DfA (Design for AMI) – A New Integrated Workflow for Modeling 56G PAM4 SerDes Systems” describes an adaptation algorithm for a 56G PAM4 Rx. This Rx has multiple CTLE stages, AGCs DFEs. The Rx adaptation process takes just 30 seconds for each FFE configuration. This did not include Tx FFE optimization.

Considering the number of CTLE/AGC/DFE/Process Corners, this is excellent performance and is a significant improvement from the GetWave adaptation which could potentially take many hours to converge depending on the design.

We have implemented a prototype BIRD 201 solution with a script that intelligently sweeps the Tx FFE taps combined with the Rx optimization described above. Compute time to co-optimize the full channel is ~10 minutes.

Our next step is to use Machine Learning to generate a Machine Learning Model. We expect to reduce the co-optimization time by at least one order of magnitude.

Implications of this Shift Left

- Evaluation of different equalization strategies in the hardware requires that channel equalization be optimized very early in the design process.
- The skill sets to write FFE, CTLE, AGC and DFE filters is different then the skill sets to implement optimization and training algorithms.

We are Now Prototyping Machine Learning, Artificial Intelligence and Genetic Optimization using the Statistical Back Channel Flow

- We are currently learning how to optimize the Machine Learning process.
- Optimization using the Machine Learning Model is very fast.