56g pam4 ibis ami modeling

MIG SIPI
Intel Corporation
Jonggab Kil
PCIe Express link bandwidth increased exponentially!

- **PCIe Gen6** (64Gb/s)
- **PCIe Gen5** (32Gb/s)
- **PCIe Gen4** (16Gb/s)
- **PCIe Gen3** (8Gb/s)
- **PCIe Gen2** (4Gb/s)
- **PCIe Gen1** (2Gb/s)

**Bandwidth (Gbps)** vs **Time**

- **TX De-emphasis**: 8b/10b
- **RX CTLE/DFE**: 128b/130b
- **TX pre-emphasis**: 8b/10b

Intel® PAM4
IBIS AMI Modeling Challenges

High speed IO design is extremely complicated.
- Heavy equalization scheme is adopted as interfaces get faster.
- Prefabricated EQ blocks had been utilized from serdes tool box.

Customer demands qualified models in an early design stage.
- IBIS AMI model is the best known method to capture buffer characteristics.
- Projected design parameters are easily tunable through serdes tool box prior to the final design release.

No standard or solid IBIS modeling methods.
- Achieving both IBIS model accuracy and efficiency is crucial.
- This paper demonstrates the cost efficient way to address this issue by using serdes tool box templates.

Long simulation time
- Long simulation time due to design and adaptation complexity.
- IBIS AMI Model should capture circuit characteristics accurately and improve run time significantly.
Highlevel Pam4 architecture

Our goal is to achieve a high quality of AMI model from the complex design!
- Two pre-taps and one post-tap
  - C-2 & C0 are positive.
  - C-2 & C1 are negative.

- Range/Graduality are modeled
  - Tap(C-2) = $C_{-2} \times \text{step}$ ($C_{-2}$ is integer)
  - Tap(C-1) = $C_{-1} \times \text{step}$ ($C_{-1}$ is integer)
  - Tap(C1) = $C_{1} \times \text{step}$ ($C_{1}$ is integer)
  - Tap(C0) = $1 - \text{abs}($Tap(C-2)$) - \text{abs}($Tap(C-1)$) - \text{abs}($Tap(C1)$)

- Works for get_wave & init.
- Excellent correlation up to the 3rd harmonic
  - Most energy is concentrated up to the 3rd harmonic.

- MATLAB functions
  - `Rationalfit`, `Freqresp` (frequency response of a rational function)
  - `residue_rat_fit`, `getGPZ` (Gain, pole and zero format) – custom scripts

- A significant amount of CTLE characteristics.
  - ~900 CTLE curves, # of boosting stages.
  - ~32 or 64 controls on each stage, 6 corner cases
Non-linearity MODELING

- Large signal response of amplification stage.
- CTLE TF is a small signal analysis.
- Swing is limited by amplifier headroom clamping.
- Critical to capture the circuit limitation.

- A significant amount of NL characteristics.
- ~300 non-linearity curves, # of boosting stages.
- Around 3 bit (8 selects) controls on each stage, 6 corner cases.
DFE/CDR MODELING

BangBang Clock Data Recovery
- Decision is made based on early or late clock to input data.
- \( BW = \text{Data}_{-rate} \times \frac{\text{CDR}_{-Step}}{\text{Sample}_{-rate}} \)

DFE NL behavior
- Slow loop response causes settling errors.
- Settling and offset errors are modeled in respect to each tap.

DFE step size/ range
- Each tap is modeled with a different step size and different tap weight range.
ADAPTATION

Global adaptation algorithm
- Use local optimal eq setting as an initial value.
- Tuning each EQ value based on the final SNR.

Statistical mode adaptation.
- Initial CTLE/DFE adaptation is done through init mode.
- Optimal EQ values are passed to bit-by-bit mode for further optimization for DFE.

COM metrics used
- Signal to noise ratio to optimize EQ adaptation.

Non Linearity aware adaptation.
- Applied to CTLEs and DFEs
Model analysis

Fast runtime

- This architectural abstract model runs more than x20 times faster than the detailed structural model.
- Fast adaptation with statistical mode optimization.

Excellent Correlation

- Statistical mode correlates well with bit-by-bit mode

Balanced eye opening

- Optimal EQ settings found.
- High speed IO modeling is extremely challenging due to the design complexity.
  - A significant amount of time and effort is a MUST to capture the high quality design behavior.

- High quality IBIS AMI model generation is critical with faster run time.
  - Help user to analyze their platform to identify any issues in a timely manner.

- Serdes Toolbox templates are used to make the modeling flow efficient.
  - A huge amount of data is processed (# of corners, # of stages, # of configuration)
  - Nice automation feature that helps to build a complicated model in an hour.
Q&A