IBIS-AMI Modeling and Simulation of 56G PAM4 Link Systems
Outline

- Introduction
- IBIS-AMI Modeling for NRZ Signaling
- Brief Introduction to PAM4 Signaling
- IBIS-AMI Modeling for PAM4 Signaling
- Eye Plot and Bathtub Construction
- IBIS-AMI Model Simulation for a PAM4 Link
- Conclusions
Introduction

- PAM4 signaling becomes competitive for 40G+
- Current IBIS-AMI standard only supports NRZ
- IBIS-AMI modeling of PAM4 signaling possible with two extensions
  - Four TX input levels from simulators
  - RX slicer levels sent to simulators
- PAM4 eye diagrams and bathtub curves
- Proposal for merged NRZ-equivalent eye and bathtub curves
IBIS-AMI Modeling for NRZ Signaling

- TX DLL input is switching between 0.5V and -0.5V
- TX output is convolved with channel impulse response
- The resultant waveform is input to the RX DLL
- RX equalized signal is sampled at each clock time and compared with 0V for BER calculation
- RX data segments are processed sequentially with each AMI_GetWave() call.
IBIS-AMI Modeling for NRZ Signaling

- The simulator sends square waves to TX IBIS-AMI model
- TX output is convolved with channel impulse response
- RX sends processed waveform and clock ticks to the simulator
Brief Introduction to PAM4 Signaling

- **PAM4** – 4-level Pulse Amplitude Modulation
- Every 2 bits are mapped to one level (one symbol)
- Requires half of the bandwidth
- **SNR penalty**: $\sim 9.5$dB
- Linear mapping vs gray mapping

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>01</td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>00</td>
<td></td>
</tr>
</tbody>
</table>

Linear mapping vs gray mapping.
PAM4 Waveform and Eye Diagram

NRZ has 2 levels (normalized to 1 and -1)
Two levels form one data eye

PAM4 has 4 levels (normalized to 3, 1, -1, and -3)
Four levels form three data eyes
PAM4 Advantage Over NRZ – An Example

- Assume we need 32Gbps throughput.
- For NRZ, the Nyquist frequency is 16GHz. The loss is 42.1dB.
- For PAM4, the Nyquist frequency is half, 8GHz. The loss is 22.4dB.
- The net difference is nearly 20dB, much larger than 9.5dB penalty.
- 42dB is by itself very difficult to equalize, making NRZ extremely challenging.
- In this case PAM4 might be a viable candidate for this link system.
How PAM4 Signal is Detected?

PAM4 signaling detection

- Top eye slicer, DT
- Middle eye slicer, DM
- Bottom eye slicer, DB

Sampling phase

If \( x_k \geq DT \), then \( \hat{x}_k = 3 \)

Else if \( x_k < DT \) & \( x_k \geq DM \), then \( \hat{x}_k = 1 \)

Else if \( x_k < DM \) & \( x_k \geq DB \), then \( \hat{x}_k = -1 \)

Else \( \hat{x}_k = -3 \)

- Three data slicers are needed for detecting four signal levels
- Data slicer levels are usually adapted to achieve best system SNR
- Typically, DB = -DT, and DM=0
IBIS-AMI Modeling for PAM4 Signaling – TX

- TX DLL input needs to switch between 0.5V, 0.5/3V, -0.5/3V and -0.5V, represent the 4 normalized levels, 3, 1, -1, and -3
- There is no need to make changes to the TX DLL interface for PAM4, from NRZ
- The simulator is responsible for mapping a given NRZ bit stream into a PAM4 data stream for a given coding scheme
- Impairments, such as jitter and noise, are handled the same as that for NRZ
RX DLL passes slicing levels to the simulator through `AMI_parameters_out` in `AMI_GetWave()`.

EDA tools use these slicer levels for deriving 3 sets of bathtub curves, and for SER/BER calculations.

An NRZ equivalent merged bathtub curve and eye diagram can be formulated in the post processing.

FEC is not currently included in the AMI modeling, but could be another topic to study next.
There are three vertically stacked eyes.

Each eye is treated with respect to its own data slicer.

For ADC based architecture, there is only one sample per symbol, thus no conventional eye diagram exists.

From simulation point of a virtual eye can be constructed.
Three sets of independent bathtub curves are formulated.

Each set contains a vertical bathtub for voltage and a horizontal bathtub for timing.

Each slicer samples every symbol regardless of the expected signal level.

An error may be counted in multiple bathtub curves (e.g. a level -3 signal appears above DM).

<table>
<thead>
<tr>
<th>Slicer level</th>
<th>Logic high traces</th>
<th>Logic low traces</th>
</tr>
</thead>
<tbody>
<tr>
<td>DT</td>
<td>$v_3(t) - DT(t)$</td>
<td>$v_1(t) - DT(t)$</td>
</tr>
<tr>
<td></td>
<td>$v_{-1}(t) - DT(t)$</td>
<td>$v_{-3}(t) - DT(t)$</td>
</tr>
<tr>
<td>DM</td>
<td>$v_3(t) - DM(t)$</td>
<td>$v_{-1}(t) - DM(t)$</td>
</tr>
<tr>
<td></td>
<td>$v_1(t) - DM(t)$</td>
<td>$v_{-3}(t) - DM(t)$</td>
</tr>
<tr>
<td>DB</td>
<td>$v_3(t) - DB(t)$</td>
<td>$v_{-3}(t) - DB(t)$</td>
</tr>
<tr>
<td></td>
<td>$v_1(t) - DB(t)$</td>
<td>$v_{-1}(t) - DB(t)$</td>
</tr>
</tbody>
</table>
PAM4 Merged Eye and Bathtub Curves Construction

- Bathtub based on the following equivalent eye construction:
  \[
  v_3(t) - DT(t) \quad v_1(t) - DM(t) \quad v_{-1}(t) - DB(t) \quad \text{Logic 1}
  \]
  \[
  v_1(t) - DT(t) \quad v_{-1}(t) - DM(t) \quad v_3(t) - DB(t) \quad \text{Logic 0}
  \]

- Consolidated to only one set of bathtub curves
- No double counting of errors
An AMI model for 56G PAM4 is constructed.

Simulation conditions:
- 56G PAM4 with Gray coding
- IL = 36dB at 14 GHz
- RL = 17dB at 14 GHz
- PSXT = -54dB at 14 GHz
AMI Model for PAM4 Simulator Setup

- Test bench setup is in ADS (modified version)
- ICN is computed from the crosstalk aggressors
- ICN is then treated as noise, to simply the setup
Simulated RX Output – Waveform

- A section of waveforms at data slicers are shown to the right.
- The waveform and data slicer levels are used for post processing.
- The three data slicer levels are changing with time.
- The data slicer levels are output from the RX DLL.

[Graph showing waveforms with levels 1, -1, and -3.
X-axis: time, nsec; Y-axis: 0.3 to -0.3]
Simulated RX Output – Sampled Eye

- RX output at sampling point, the “sampled eye”, which is a function of time (symbols)
- For post processing it is important to ignore enough symbols to make sure the adaptation converged
- System SER/BER can be computed either through true comparison or statistical computation
Simulated RX Output – Eye Diagrams

- Eye diagram constructed from ADS for the case study
- Clock ticks are used same as in NRZ AMI modeling
- Note that ADS always shows two UI of the data
Three sets of PAM4 SER contours are constructed by ADS (at 1E-10, 1E-11, and 1E-12)

Each set is centered around its own data slicer

System SER is determined by the worst of the set
Simulated RX Output – Timing Bathtub Curves

- Timing bathtub curves at the top, middle and bottom eye are formulated by ADS.
- The overall performance is limited by the worst one.
Simulated RX Output – Voltage Bathtub Curves

- Voltage bathtubs at the top, middle and bottom eye are formulated in ADS

- The overall performance is limited by the worst one
Simulated RX Output – Merged NRZ Eye

- Merged NRZ-equivalent eye is constructed by ADS
- As ADS uses 2-Ul to do the construction, only center 1-Ul portion reflects an NRZ eye
- This makes the post processing similar as in NRZ
The merged NRZ-equivalent timing bathtub curve is constructed by ADS. It alone can be used to determine the link system performance margin.
Top eye data slicer, DT, from RX DLL output is plotted.

Its convergence profile provides useful information:

- Even after convergence, the slicer level is still dithering around some mean value.
- We need to ignore 10 µsec of the data, or about 280K UI.
- In this example DM is set to 0 and DB is tied to –DT for processing convenience.
Conclusions

- AMI modeling for PAM4 systems is illustrated
- TX needs to send 4 different levels at ±0.5V and ±0.5/3V
- RX needs to pass on adapted slicer levels to EDA tools using AMI_parameters_out for eye plots and BER bathtubs/contours
- Three sets of bathtub curves can be merged into one set
- NRZ and PAM4 dual mode support is both feasible and desirable
- BER/SER calculation should consider the coding scheme
- FEC is important for PAM4 systems, but not included here