BIRD 166 / BIRD190 Summary

Walter Katz

SiSoft

August 1, 2017

Summary of the Problem

1. An Rx model does not know if it is the terminus of a simple Tx/Rx channel, or a Tx/Redriver/Rx model.
2. Init-Only Rx models can only optimize themselves in AMI_Init
3. Dual Rx model can optimize themselves in AMI_Init and communicate their equalization settings to their AMI_GetWave
    1. This AMI_GetWave may further adapt the equalizations setting based on the initial equalization from their AMI_Init
    2. “3b” Optimizing modern Rx models with large number of DFE taps can take excessively long number of UI
        a. A hardware adaptation may take only one second in the hardware, at 25Gbps, this is 25e9 UI
        b. Typically, AMI_GetWave time domain simulations are 1e6 UI per minute
        c. Thus a software implementation of this hardware adaptation can take 25e9/1e6 =25,000 minutes

SiSoft Redriver Time Domain Flow

- The AMI_Init flow is the same as defined in BIRD 166 statistical flow (Slide 8)
- If both the Tx2 and Rx2 AMI models do have an AMI_GetWave
  - The EDA tool simply uses the waveform output of Rx2 AMI_GetWave
- If both the Tx2 and Rx2 AMI models do not have an AMI_GetWave
  - The tool can convolve the IR output with the Tx1 input digital stimulus to get the waveform at the Rx2 latch
  - The tool can use deconvolution techniques to generate an IR of the Tx2/Rx2 channel to convolve with the waveform output of Rx1 to get the waveform at the Rx2 latch
    - Fangyi says that this technique gives the wrong answer
    - SiSoft says that this technique give the right answer, particularly predicting the eye height at the Rx2 latch. .
  - Similar techniques can be used when either the Tx2 or Rx2 (but not both) do not have an AMI_GetWave

Cadence/Keysight Redriver Time Domain Flow

- The AMI_Init flow is the same as defined in IBIS 6.1 time domain flow (Slide 9)
- Since the Rx2 AMI_Init will not have a complete IR input to the Rx2 AMI_Init, any adaptation that the Rx2 could do will be incorrect. The IBIS 6.1 flow says Rx2 adaptive equalization should be turned off and the equalization set by the User or EDA tool. BIRD 190 re-affirms this IBIS 6.1 flow.
- If both the Tx2 and Rx2 AMI models do have an AMI_GetWave
  - The Rx2 AMI_GetWave function can do adaptation.
- If both the Tx2 and Rx2 AMI models do not have an AMI_GetWave
  - The tool can convolve the IR output of Rx2, IR output of Rx1 and the Tx1 input digital stimulus to get the waveform at the Rx2 latch
  - The tool can use deconvolution techniques to generate an IR of the Tx2/Rx2 channel to convolve with the waveform output of Rx1 to get the waveform at the Rx2 latch
  - Similar techniques can be used when either the Tx2 or Rx2 (but not both) do not have an AMI_GetWave

Bob Miller’s Comments

"3b" in the first section is the screaming issue for modelers of sophisticated serdes. Though the issue goes well beyond DFE. Some AFE taps, especially in PAM4, take >>1e9 of dwell to converge in the hardware. This can often be synthetically reduced if the tap is analyzed with specific patterns (not required in the actual hardware optimization since UI are cheap there), but this imposes an additional awkward dance in EDA-managed bit-by-bit (GetWave) simulation between the model, the user, and *every* supported EDA tool to get everything synced up. (How stimulus is organized, selected, and controlled is EDA-specific.) It becomes a big mess.

When the upstream model/channel behavior is (or nearly is) linear, an accurate impulse in becomes a marvelously convenient key to eliminating this traffic jam. The model can use it to generate it's own stimulus.

That IBIS doesn't support this when it can, and 190 seems to want to cast this limitation in stone, is hugely disappointing to me.

What the Industry is Doing

- Some IC Venders have decided to generate Init-Only models.
  - Init may be doing all of the heavy lifting such as optimizing the CTLE, AGC, DFE taps, AFE taps …
  - It is straightforward to add a GetWave which uses the equalization from the Init call
    - David Banas has published models that show you how to do this
    - Adding an AMI_GetWave to an Init model is easy, and eliminates the need for EDA tool to use deconvolution. Therefore it is strongly recommended to IC Vendors to add an AMI_GetWave to insure that their customers can get consistent and correct results using any EDA Vendor software.
  - This is a convince to EDA tools that have problems with Init-Only models when used in time domain flows.
- Some companies have decided to use only Statistical simulation for SerDes and DDR5 channels.

Summary

- I have no problem with BIRD 190 documenting the problems and limitations of the existing IBIS 6.1 flow, but only if the SiSoft flow was added as an alternative time domain flow. (Slide 10)
- EDA tools can then indicate which flow their tool is using.

BIRD 166.3 Fix to Redriver Statistical Flow

Make the changes indicated below on page 244:

Change:

Step 6. The output of step 5 is presented to Rx2's AMI_Init function and Rx2's AMI_Init function is executed.

Step 7a. Redriver: The EDA tool convolves impulse responses returned by Rx1's AMI_Init in step 3 and by Rx2's AMI_Init in step 6 to obtained the full channel impulse response and uses it to perform statistical simulation.

To

Step 6a. Redriver: The simulation platform convolves the impulse response returned with Rx1's AMI_Init in step 3 by the output of step 5 and presents the results to Rx2's AMI_Init function and Rx2's AMI_Init function is executed.

Step 7a. Redriver: The simulation platform uses the impulse responses returned by Rx2's AMI_Init in step 6a to obtain the full channel impulse response and uses it to perform statistical simulation.

WMK> Note that this change moves the output of Rx1' AMI_Init function from the output of Rx2 AMI_Init to the Input of Rx2 AMI_Init.

IBIS 6.1 Redriver Time Domain Flow

<u>On Page 243</u>

Step 5. The output of step 4 is presented to Tx2's AMI_Init function and Tx2's AMI_Init function is executed.

Step 6. The output of step 5 is presented to Rx2's AMI_Init function and Rx2's AMI_Init function is executed.

WMK> Note that the input to the Rx2's AMI_Init function does not include the equalization in the upstream channel (output of Rx1 AMI_Init)

Alternative Redriver Time Domain Flow

On Page 243

Change

Step 6. The output of step 5 is presented to Rx2's AMI_Init function and Rx2's AMI_Init function is executed.

To

Step 6. The output of step 5 is presented to Rx2's AMI_Init function and Rx2's AMI_Init function is executed.

> Note: The Rx2 executable model file writer for the downstream channels with Redrivers should keep in mind that the impulse response that is presented to the Rx2 AMI_Init function does not include the effects of the upstream equalization. Therefore, the Rx AMI_Init function will not be able to perform accurate optimization in the absence of the upstream channel characteristics and/or equalization effects. For this reason, the parameters of the Rx AMI_Init function should always default to valid values or have a mechanism to accept user-defined coefficients and allow the user to turn off any automatic optimization routines to ensure successful simulations.

Step 6 (alternative for Redrivers). The simulation platform convolves the impulse response returned with Rx1's AMI_Init in step 3 by the output of step 5 and presents the results to Rx2's AMI_Init function and Rx2's AMI_Init function is executed.