**BUFFER ISSUE RESOLUTION DOCUMENT (BIRD)**

**BIRD NUMBER:** 211

**ISSUE TITLE:** New Redriver Flow

**REQUESTOR:**  Walter Katz, The MathWorks, Inc

**DATE SUBMITTED:** March 23, 2021

**DATE REVISED:**

**DATE ACCEPTED:**

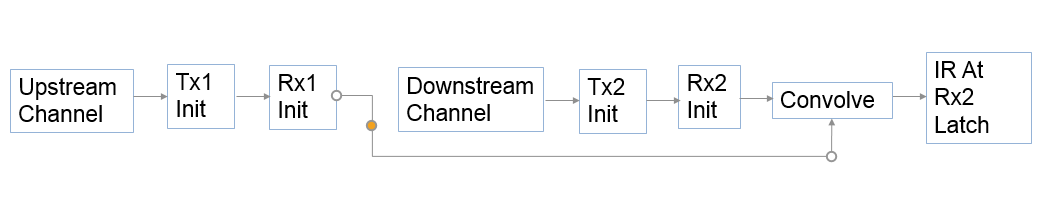
**DEFINITION OF THE ISSUE:**

The current Repeater flow is known to have following issues.

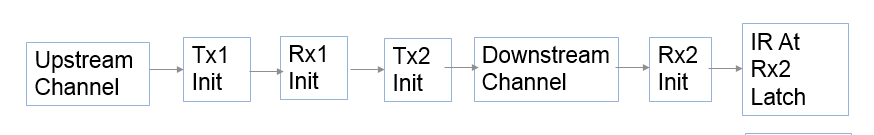
1. The cumulative upstream impulse response of the Redriver channel is not provided to the terminal Rx (including Retimer Rx) in AMI\_Init. As a result, when the terminal Rx has DFE, the end-to-end cumulative impulse response of the Redriver channel needed in statistical simulations is not available.
2. The cumulative upstream impulse response of the Redriver channel is not provided to either Tx or Rx in AMI\_Init. As a result, the AMI\_Init function cannot perform optimization on the upstream signal.
3. The combination of Tx GetWave model and Rx Init-only model leads to deconvolution in time domain simulations.

This BIRD proposes a new repeater flow, in which certain requirements are imposed on models, to address these issues. A new reserved parameter and new data in the impulse matrix are introduced to support the proposed flow.

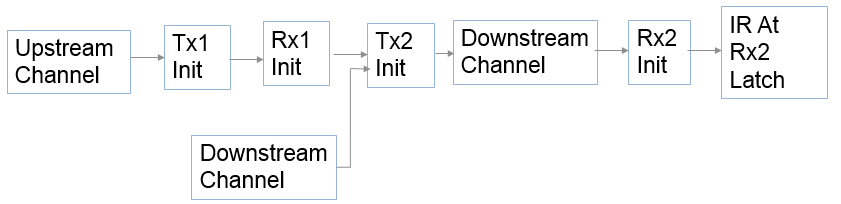
Existing Flow



If Tx\_Requires\_Downstream\_Channel is not in the Redriver Tx .ami file, or it is set to False, then the Redriver will now be:



If Tx\_Requires\_Downstream\_Channel is set to True, then the Redriver flow is:



**SOLUTION REQUIREMENTS:**

The IBIS specification must meet these requirements:

Table 1: Solution Requirements

|  |  |
| --- | --- |
| Requirement | Notes |
| * Support statistical simulations on Redriver channels whose terminal Rx (including Retimer Rx) has DFE. |  |
| * Allow Redriver Tx AMI\_Init to perform optimization on the upstream signal. |  |
| * Allow Redriver Tx AMI\_Init to perform optimization on the upstream signal and the downstream. |  |
| * Eliminate the need for deconvolution in simulations. |  |

**SUMMARY OF PROPOSED CHANGES:**

Add one column at the end of impulse\_matrix in AMI\_Init.

Add new Boolean Reserved Parameter Tx\_Requires\_Downstream\_Channel.

In the new flow only Tx redriver models are required to support parameter Tx\_Requires\_Downstream\_Channel in the AMI\_Init function.

**PROPOSED CHANGES:**

Add the following paragraph to Section 10.2.3.

IBIS Version 7.1 introduces modification to the Redriver simulation flow and a new simulation flow. The modification to the existing Redriver flow feeds the output of the Redriver Rx directly into the Redriver Tx. The downstream channel is combined with the Redriver Tx output and is the input to the terminal Rx. In the new flow (indicated by Tx\_Requires\_Downstream\_Channel=True), the EDA tool is responsible to add one additional column at the end of the input impulse\_matrix and to fill this additional column with the impulse response of the downstream channel from the redriver Tx to the terminal Rx algorithmic model’s input node. The Redriver Tx model’s AMI\_Init function is responsible to:

1. Modify the through channel column of impulse\_matrix in place by applying its gain and equalization to the first column of the impulse\_matrix
2. Modify the crosstalk channel columns of impulse\_matrix in place by applying its gain and equalization to the aggressor columns
3. The model shall not modify the additional column that contains the downstream channel.

Add the following new parameter:

*Parameter:* **Tx\_Requires\_Downstream\_Channel**

*Required:* No, and illegal before AMI\_Version 7.1

*Direction:* Tx

*Descriptors:*

Usage: In

Type:                     Boolean

Format: Value

Default:                 <Boolean\_literal>

Description:*<*string>

*Definition:* The Value must be True. The EDA tool is responsible to set the value True or False and pass it to the model in the AMI\_Init function call to inform the model whether the new flow or the old flow is being executed, respectively. The EDA tool can set the value to either True or False in the AMI\_Init function call.

*Usage Rules:* This parameter can only be used in Redriver Tx models. A model that specifies this parameter must also specify Init\_Returns\_Impulse=True. When Tx\_Requires\_Downstream\_Channel is False, the model’s AMI\_Init function behaves as specified in this 7.1 Specification. When Tx\_Requires\_Downstream\_Channel is True, the model’s AMI\_Init function modifies impulse\_matrix as specified in Section 10.2.3. If this parameter is not present in the AMI\_parameters\_in string, the model assumes that Tx\_Requires\_Downstream\_Channel=False.

*Other Notes:* In Repeater AMI simulations, both Repeater analog models are treated as if they are linear and time-invariant. The incoming (upstream) analog channel of the Redriver, including the upstream Tx analog model, the physical channel, and the Repeater Rx analog model, is represented by an impulse response. The outgoing (downstream) analog channel of the Repeater, including the Repeater Tx analog model, the physical channel, and the downstream Rx analog model, is represented by another impulse response.

The time domain simulation flow for a Repeater link shown in Figure 41 is defined below.

Figure 41 – Repeater Link

Repeater

Rx

Tx1

Rx1

Tx2

Rx2

channel 1

channel 2

Repeater

Repeater Tx

Incoming

(upstream)

channel

outgoing

(downstream)

channel

Here Tx1 denotes the Repeater upstream channel (channel 1) Tx AMI model (including analog and algorithmic models), Rx1 the Repeater Rx AMI model (including analog and algorithmic models), Tx2 the Repeater Tx AMI model (including analog and algorithmic models), and Rx2 the Repeater downstream channel (channel 2) Rx AMI model (including analog and algorithmic models).

**Retimer Flow**

Step 1. The EDA tool obtains the impulse response of the upstream analog channel, which represents the combined impulse response of Tx1’s analog model, physical channel 1, and Rx1’s analog model.

Step 2. The output of step 1 is presented to Tx1’s AMI\_Init function and Tx1’s AMI\_Init function is executed.

Step 3. The output of step 2 is presented to Rx1’s AMI\_Init function and Rx1’s AMI\_Init function is executed.

Step 4. The EDA tool obtains the impulse response of the downstream analog channel, which represents the combined impulse response of Tx2’s analog model, physical channel 2, and Rx2’s analog model.

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.

Step 7. The EDA tool uses the impulse response returned by Rx1’s AMI\_Init in step 3 to perform a statistical simulation of channel 1. The EDA tool uses the impulse response returned by Rx2’s AMI\_Init in step 6 to perform a statistical simulation of channel 2.

Step 8. The EDA tool performs simulation on the upstream channel, which consists of Tx1, physical channel 1, and Rx1, according to the AMI flow defined in the specification for channels without Repeaters.

Step 9. The EDA tool samples the output waveform of Retimer Rx AMI\_GetWave at ½ UI after each clock tick returned by the function, generates a digital stimulus as the input to Tx2’s algorithmic model, regardless whether Tx2’s AMI\_GetWave exists or not, and performs simulation on the downstream channel, which consists of Tx2, physical channel 2, and Rx2, according to the AMI flow defined in the specification for channels without Redriver. The logic level of the digital stimulus is 1 if sampled value >= Rx1’s Rx\_Receiver\_Sensitivity and 0 if sampled value <= Rx1’s Rx\_Receiver\_Sensitivity. If –Rx1’s Rx\_Receiver\_Sensitivity < sampled value < Rx1’s Rx\_Receiver\_Sensitivity, the logic level is unchanged from the previous bit. The digital stimulus shall have values of -½ volt for logic 0 and +½ volt for logic 1.

Step 10. The EDA tool calls the AMI\_Close function of each algorithmic model in Tx1, Rx1, Tx2 and Rx2.

If just doing a statistical simulation, the flow is terminated after step 7. If doing a time domain simulation, step 7 may be skipped.

Since the Retimer output signal is driven by a digital stimulus as described above in step 9, jitter and noise parameters specified in Retimer .ami files are applied according to the specification for channels without Repeaters.

**Redriver Flow**

All models, including the Primary Tx, Redriver Rx, Redriver Tx, and Terminal Rx should specify Init\_Returns\_Impulse=True.

Step 1. The EDA tool obtains the impulse response of the upstream analog channel, which represents the combined impulse response of Tx1’s analog model, physical channel 1, and Rx1’s analog model.

Step 2. The output of step 1 is presented to Tx1’s AMI\_Init function and Tx1’s AMI\_Init function is executed.

Step 3. The output of step 2 is presented to Rx1’s AMI\_Init function and Rx1’s AMI\_Init function is executed.

Step 4. The EDA tool obtains the impulse response of the downstream analog channel, which represents the combined impulse response of Tx2’s analog model, physical channel 2, and Rx2’s analog model.

Step 5a. If Tx\_Requires\_Downstream\_Channel is False, or not present: The output of step 3 is presented to Tx2’s AMI\_Init and Tx2’s AMI\_Init function is executed.

Step 5b. If Tx\_Requires\_Downstream\_Channel is True: The output of step 3 and the output of step 4 is presented to Tx2’s AMI\_Init and Tx2’s AMI\_Init function is executed.

Step 6. The output of step 4 is convolved with the output of step 5.

Step 7. The output of step 6 is presented to Rx2’s AMI\_Init function and Rx2’s AMI\_Init function is executed.

Step 8. The EDA tool uses the impulse response returned by Rx2’s AMI\_Init in step 7 to perform a statistical simulation.

Step 9. The EDA tool performs simulation on the upstream channel, which consists of Tx1, physical channel 1, and Rx1, according to the AMI flow defined in the specification for channels without Repeaters.

Step 10. The EDA tool uses the signal waveform at the output end of Rx1’s algorithmic model in step 8, regardless of whether Rx1’s AMI\_GetWave exists or not, as the stimulus of Tx2’s algorithmic model, regardless of whether Tx2’s AMI\_GetWave exists or not, and performs simulation on the downstream channel, which consists of Tx2, physical channel 2, and Rx2, according to the AMI flow defined in the specification for channels without Redrivers.

Step 11. The EDA tool calls the AMI\_Close function of each algorithmic model in Tx1, Rx1, Tx2, and Rx2.

If just doing a statistical simulation, the flow is terminated after step 8. If doing a time domain simulation step 8 may be skipped.

Since the Redriver output signal is driven continuously by the input analog signal and does not have a sampling latch, clock times, if returned by a Redriver model, jitter parameters, and the Rx\_Noise parameter specified in Redriver .ami files are ignored by the EDA tool.

*Example:*

(Tx\_Requires\_Downstream\_Channel (Usage In) (Type Boolean) (Value True)

(Description "The model supports the new Redriver flow"))