Subject: About the SerDes Repeater Simulation Tool

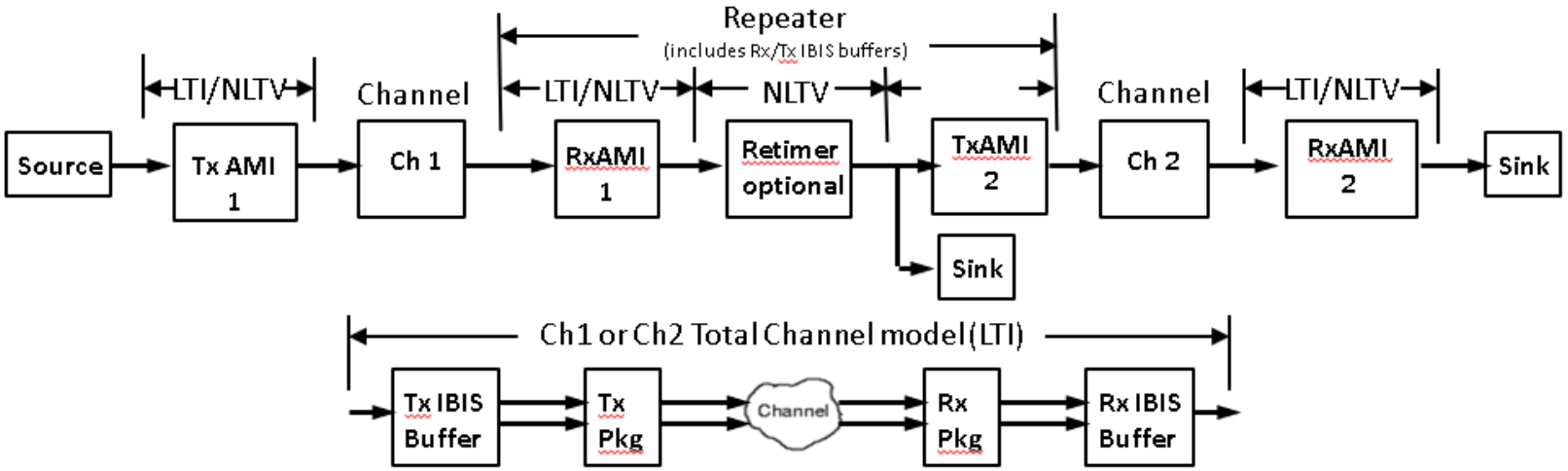
Author: John Baprawski; John Baprawski Inc. (JB)

Date: Jan 3, 2019; updated Oct. 30, 2026

This paper discusses features on the web site: **https://www.serdesdesign.com**

The SerDes Repeater Simulation Tool is located on the web at: **https://www.serdesdesign.com/home/serdes-repeater-tool/.**

The Repeater Simulation Tool analyzes a SerDes system that has a typical structure shown in this figure.



The SerDes Repeater Simulation Tool analyzes a SerDes system that has two differential channels with the repeater (Rx1+Tx2) (with optional retimer model) between the two channels. The tool displays channel characteristics in the frequency and time domains.

This tools supports the I/O Buffer Information Specification (IBIS) 7.2 ratified January 27, 2023 with exceptions noted **here**.

Copyright © 2018-2026 SerDes Design | All Rights Reserved

See these examples on using the SerDesDesign.com SerDes Repeater Simulation Tool. This tool is available for free use on the web site listed above or with a yearly lease for use on one's local Windows 65 bit PC. The following examples are based on the Windows copy. The web based and Windows based tools are the same. The web tool is slower and may have limitations in length of simulation time; the Windows copy does not have those limitations.

- Example repeater using Tx/Rx IBIS-AMI models.
- Example with repeater IBIS-AMI model generation.
- Example with repeater using back-channel training from Rx to Tx.

The SerDes repeater system includes:

- SOURCE: a signal source that supports either an NRZ or PAM4 signal.
- TX1: a transmit equalizer (Tx) that is linear and time invariant (LTI) or nonlinear and/or time invariant (NLTV).
- Ch1: the repeater input (upstream) S-Parameter channel; may include Tx and/or Rx IBIS-AMI models; may include Tx and/or Rx S-Parameter package models.
- Rx1 Front End: a receiver front end equalizer that is LTI.
- Rx1 CDR/DFE: receiver clock and data recovery (CDR) unit and decision feedback equalizer (DFE) and is NLTV.
- Rx1 Regenerator: regenerator (with optional jitter) when the repeater is used as a retimer.
- TX2: a transmit equalizer (Tx) that is LTI or NLTV.
- Ch2: the repeater output (downstream) S-Parameter channel; may include Tx and/or Rx IBIS-AMI models; may include Tx and/or Rx S-Parameter package models.
- Rx2 Front End: a receive front end equalizer that is LTI.
- Rx2 CDR/DFE: receiver clock and data recovery (CDR) unit and decision feedback equalizer (DFE) and is NLTV

The repeater is defined with the above blocks from Rx1 front end through Tx2.

Per the IBIS-AMI standard, a repeater is a "redriver" when the Rx1 model has no clock_times output and the input to the Tx2 model is not a regenerated bit/symbol stream.

Per the IBIS-AMI standard, a repeater is a "retimer" when the Rx1 model does have a clock_times output and the input to the Tx2 model is a regenerated bit/symbol stream.

Copyright © 2018-2026 SerDes Design | All Rights Reserved

See below for **IBIS-AMI Notes on the Repeater**.

Each differential channel often includes a Tx buffer/package and a Rx package/buffer and is LTI.

- The differential channel represents a hardware SerDes channel and is typically characterized by measuring its N-port S-parameters and is typically a 4-port. The 4-port differential input ports are typically port 1 (+) and port 3 (-). The associated differential output ports are typically port 2 (+) and port 4 (-). The differential characteristic (Port 1 – Port 3 vs. Port 2 – Port 4) is the channel transmission characteristic and is observed versus frequency.
- See S-parameter detail in References > **S-Parameter Channel Examples**
- The S-parameters may also be obtained from various simulators. A high speed digital SerDes channel typically has substantial high frequency attenuation at and beyond the bit/symbol rate Nyquist frequency and requires compensation using equalizers at the transmit and/or receive side of the channel.

The total channel is inclusive of the Tx IBIS Buffer and Rx IBIS Buffer. Per the IBIS-AMI standard, the IBIS Buffers are to be considered LTI for use in a Channel Simulator. Thus, the total channel is LTI.

- See IBIS Buffer detail in: **IBIS Buffers used in SerDes Simulations**
- The total channel, as used in a Channel Simulator and inclusive of the S-parameters, is converted to an equivalent single ended impulse response.
- See channel impulse response detail in References > **Channel Time-Domain Response**
- The typical approach involves zero-padding the S-parameters for the time domain SampleRate (SampleRate = Bit/Symbol Rate * SamplesPer Bit/Symbol) for a maximum frequency of SampleRate/2.0 and applying the constraints for physical realizability which include meeting the mathematical aspects of the Kramers-Kronig relations applied to linear time invariant (LTI) systems. This zero-padding approach often results in high frequency aliasing.
- SerDesDesign.com uses a proprietary algorithm to obtain the causal channel impulse response which inherently does not result in any high frequency aliasing.
- See Causal S-Parameters detail in: **About the Generate Causal S-Parameters Tool**

To use the SerDes System Single Repeater Tool, follow the steps on the web page.

1. Define the analysis name

Copyright © 2018-2026 SerDes Design | All Rights Reserved

- An alpha-numeric character string; including underbar - case sensitive - start with alpha character.
- The full analysis name = Serdes_<your entered name>
- This full name is used in the Eye Analysis Tool for further processing of the SerDes system analysis output data.

2 and 5. Define the Transmitters Tx1 and Tx2.

- By default, the transmit models are defined as LTI or NLTV Tx IBIS-AMI models associated with Ch1/Ch2.
- Alternately, behavioral LTI transmitters (Tx1 and Tx2) can be defined in a number of different ways.
  - As a feed forward equalizer (FFE) with optional filtering.
  - As an FFE with quantized values defining the pre and post cursors with optional filtering.
  - As an FFE with integer codes defining the pre and post cursors with optional filtering.
  - As an FFE based on black-box modeling.
  - As an FFE based on digital register values.
  - As a continuous time linear equalizer (CTLE) based on poles and zeros, on peaking response characteristics in the time or frequency domain, on a family of time domain step responses, or as a family of frequency domain spectral responses.
- If the Tx is defined behaviorally and a Tx IBIS-AMI model is also defined with the channel, then the Tx behavioral model is used with the Tx IBIS buffer defined with the channel.
- Transmit input source jitter can be defined from either the Ch1 Tx AMI model or behaviorally by setting the TransmitterJitterType =1 and setting its random Gaussian or deterministic jitter values.
- See detail: **Define Transmitter**
- See detail: **IBIS_AMI_Jitter_Summary.pdf**

3 and 6. Define the Channels (Ch1 and Ch2).

- The channel is defined here which consists of a basic differential channel defined with an S-parameter file with at least 4 ports plus any of the following options.
- Transmitter AMI portion as part of a Tx IBIS-AMI model which may be LTI or NLTV.
- Transmitter IBIS output buffer. See detail: **IBIS Buffers used in SerDes Simulations**
- Transmitter differential channel packaging S-parameter file with at least 4 ports.
- Receiver differential channel packaging S-parameter file with at least 4 ports.

Copyright © 2018-2026 SerDes Design | All Rights Reserved

- Receiver IBIS input buffer. See detail: **IBIS Buffers used in SerDes Simulations**
- Receiver AMI portion as part of a Rx IBIS-AMI model which may be LTI or NLTV.
- Any S-parameter file used is automatically adjusted as needed to conform to the physical realizability constraints of passivity, reciprocity, and causality, as well as reduction of noise in the S-parameters.
- See detail: **Channel_wTxRx_IBIS_Pkg**

4 and 7. Define the Receiver (Rx1, Rx2).

- By default, the receiver models are defined as LTI or NLTV Rx IBIS-AMI models associated with Ch1/Ch2.
- Alternately, a behavioral LTI receiver front ends (RxFE1 and RxFE2) can be defined in a number of different ways.
    - As a 1 section continuous time linear equalizer (CTLE) defined with optional automatic gain control (AGC) and based on poles and zeros, on peaking response characteristics in the time or frequency domain, on a family of time domain step responses, or as a family of frequency domain spectral responses.
    - As a 2 section CTLE each defined with a family of step responses and with optional AGC.
    - As a 3 section CTLE each defined with a family of step responses and with optional AGC.
    - As a 4 section CTLE each defined with a family of step responses and with optional AGC.
    - As an FFE defined as similarly defined for the Tx LTI behavioral models and with optional AGC.
- Optionally, a behavioral NLTV receiver back end (RxCDRDFE1 and RxCDRDFE2) can be defined in a number of different ways.
    - As a model with a CDR and decision feedback equalizer (DFE).
    - As a model with a CDR and DFE with DFE taps that are quantized.
    - As a model with a CDR and DFE with DFE taps that are integer codes.
- If the Rx is defined behaviorally and a Rx IBIS-AMI model is also defined with the channel, then the Rx behavioral model is used with the Rx IBIS buffer defined with the channel.
- For Rx1, the repeater may be used as a retimer when the Rx1 Regenerator is enabled (with optional jitter).
- For Rx2, receiver output jitter can be defined from either the Ch2 Rx AMI model or behaviorally by setting the ReceiverJitterType =1 and setting its random Gaussian or deterministic jitter values.
- See detail: **RxFE1 - 1 section CTLE with AGC**
- See detail: **RxFFE - FFE with AGC**
- See detail: **RxCDRDFE - Rx CDR with DFE**

Copyright © 2018-2026 SerDes Design | All Rights Reserved

8. Setup the analysis.

- Define the symbol rate per second (for NRZ bits or PAM4 symbols).
- Define the number of samples per symbol.
- Set up analysis options: **Setup Options**
- Set GenerateModel to generate a Rx/Tx Repeater IBIS-AMI model from the Ch1 Rx and Ch2 Tx behavioral data (GenerateModels = 3). Set the IBIS_AMI_ModelName to give your generated model a name.

9. Run the analysis.

- After the Analysis is Run, the Analysis Log file is displayed.
- Look at the bottom of the file to see that the analysis was successful:

```
******************** End runtime anaysis ********************

Go to the Eye Analysis Tool for detail eye analysis for this SerDes system (set ChAnalysisName =
Serdes_RepeaterTest or Serdes_RepeaterTest_Ch2).
Exiting SerDes System Analysis with success; run time = 20 sec.
```

10 and 11. Display results for each channel and each Rx output.

- Observe the channel frequency domain characteristic, its equivalent impulse response, its eye diagram, and its BER bathtub curve.
- See detail: **Typical SerDes System Characteristics and Displays**

Reuse the generated channel data files.

- The total channel impulse response (inclusive of the Tx/Rx IBIS models) is defined with name: Serdes_<AnalysisName>.csv and Serdes_<AnalysisName>_Ch2.csv.
  - Located in directory: C:\SerDesDesign\user-<number>\files\channel_data
  - This total channel impulse response file can be used when the ChannelType = 1 in the SerDes System Simulation Tool.

Copyright © 2018-2026 SerDes Design | All Rights Reserved

- If the Tx IBIS or Rx IBIS was defined with a channel facing S2P impedance data file, <filename>.s2p, then that S2P data was converted to the required IBIS S4P data with name: <filename>.s2p.s4p.
  - Located in directory: C:\SerDesDesign\user-<number>\files\channel_data
  - This S4P data file can be used instead of the original S2P data file.

If GenerateModel = 3, then one can use the generated IBIS-AMI models back in the simulation.

- Use the generated Rx IBIS-AMI model in the Ch1 setup with EnableRxBuffer = 2.
- Set the Ch2 Rx Regenerater (ReceiverRegenType = 1) if the Repeater was a Retimer
- Use the generated Tx IBIS-AMI model in the Ch2 setup with EnableTxBuffer = 2.

If GenerateModel = 3, then one can complete the generation of the portable IBIS-AMI models.

- See detail as: **Generate Portable IBIS-AMI Models**

Open the **Eye Analysis Tool** for detail eye and BER analysis of the SerDes system results by entering as the ChAnalysisName for the Ch1 results or the Ch2 results as noted in the log file message.

- See detail discussion at: **About_the_Eye_Analysis_Tool**

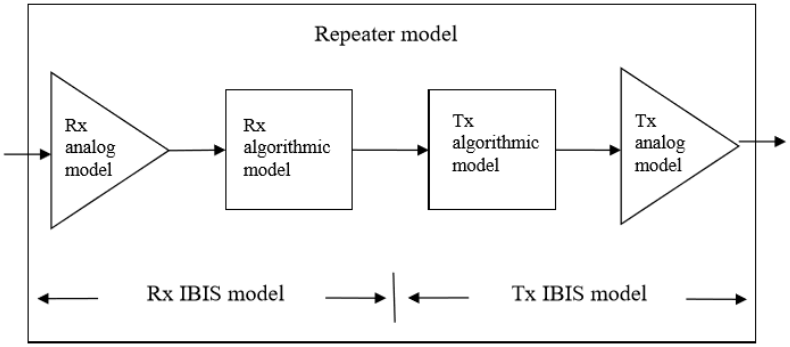
IBIS-AMI Notes on the Repeater

The following is a reprint from IBIS.org IBIS-AMI 6.0 specification. The SerDesDesign.com Repeater simulation tool only supports the bit-by-bit time domain simulation flow.

A Repeater is a type of device that is placed in the middle of the channel to compensate channel loss. Repeaters consist of two categories, Redrivers and Retimers. A Redriver equalizes the upstream channel signal and retransmits it to the downstream channel. The output signal is continuously driven by the input signal. **A Redriver has no retiming performed when the Redriver retransmits the signal**. **A Retimer equalizes the upstream channel signal, recovers the clock using a CDR and generates a digital stimulus that is transmitted to the downstream channel**.

Copyright © 2018-2026 SerDes Design | All Rights Reserved

A Repeater is modeled by two back-to-back input-output IBIS-AMI models as shown here.

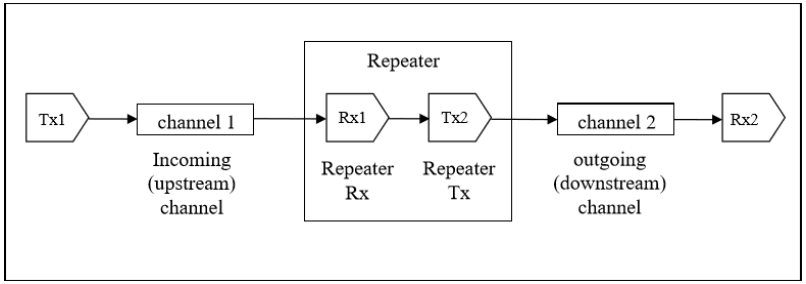


The analog part of the Rx model represents the input termination at the device input. The analog part of the Tx model represents the output impedance at the device output. The two algorithmic models represent equalizers, clock data recovery or CDR circuits (if they exist) and/or preemphasis inside the devices. In a Redriver, both algorithmic models can optionally implement the AMI_GetWave function. In a Retimer, the Rx algorithmic model must implement AMI_GetWave and the function must return clock times. The Retimer Tx algorithmic model can optionally implement AMI_GetWave. The order of signal flow in a Repeater model is from Rx analog to Rx algorithmic to Tx algorithmic to Tx analog. Looking from the Rx analog portion, the Rx algorithmic block is assumed to have infinite input impedance. Looking from the Tx analog portion, the Tx algorithmic block is assumed to have an output of an ideal voltage source. A Repeater model is specified in a single .ibs file that includes both input and output models.

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 is shown here.

Copyright © 2018-2026 SerDes Design | All Rights Reserved



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).

The **bit-by-bit time domain simulation flow** for the Repeater link shown in the above figure is defined here.

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 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.

Copyright © 2018-2026 SerDes Design | All Rights Reserved

Step 8a. **Redriver**: The EDA tool uses the signal waveform at the output end of Rx1's algorithmic model in step 7, regardless whether Rx1's AMI_GetWave exists or not, as the stimulus of 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 spec for channels without Redrivers.

Step 8b. **Retimer**: 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 spec 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_Reciver_Sensitivity, the logic level is unchanged from the previous bit. The digital stimulus have values of -½ volt for logic 0 and +½ volt for logic 1.

Step 9. The EDA tool calls the AMI_Close function of each algorithmic model in Tx1, Rx1, Tx2 and Rx2.

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. Since the Retimer output signal is driven by a digital stimulus as described above in step 8b, jitter and noise parameters specified in Retimer .ami files are applied according to the specification for channels without Repeaters.

Terms & Conditions | Privacy Policy

Copyright © 2018-2026 SerDes Design | All Rights Reserved