Algorithmic Modeling BIRD

IBIS-ATM Group

August 24, 2007

Agenda

- Serial Link Design Trends
- IBIS-ATM Goals
- AMI Terminology
- Proposed IBIS File Structure & Syntax
- API Overview
- Interoperability tests & AMI toolboxes
- Next Steps & conclusion

Who Needs SerDes Models?

- System Designers
  - Predict end-end link BER
  - Evaluate system-level design tradeoffs
- ASIC designers
  - Evaluate different TX/RX behavior
- SerDes circuit designers
  - Validate with standard test setups
- Test Equipment Vendors
  - Model device-specific equalization & clock recovery

Serial Link Analysis Needs

- Analyze link behavior > 10^7 bits
- Model transmit / receive equalization
- Model clock recovery behavior

- Serial link analysis is best addressed through a combination of analytical methods
  1. Characterization of the analog channel
  2. Equalization & clock recovery modeling

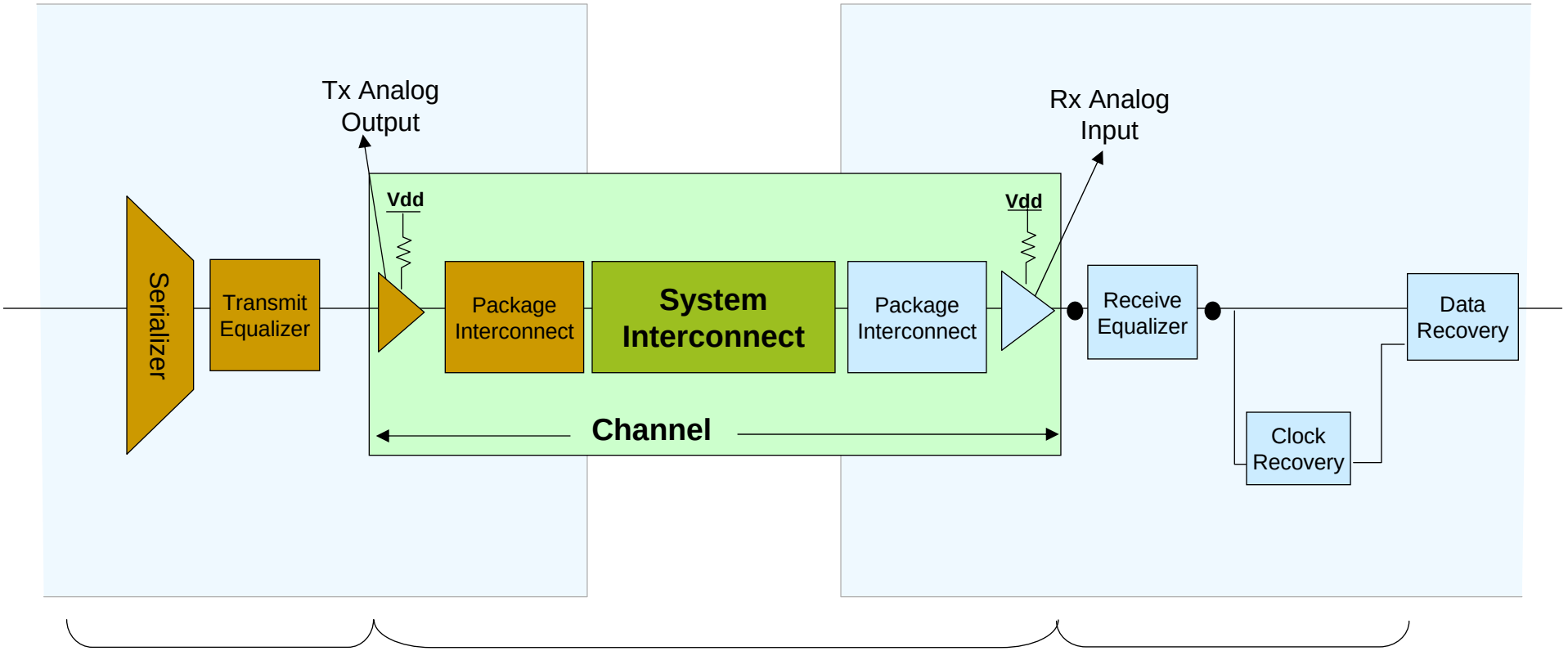
Why Use Separate Steps?

- Exploit linear behavior of serial channel analog components for computational efficiency
- Leverage existing techniques for modeling equalization and clock recovery
  - Communication systems theory
- Enable multiple analytical approaches
  - Commercial EDA tools
  - Matlab / StatEye
  - IP vendor tools

IBIS-ATM Goals

- Establish a modeling standard for SerDes devices that describes
  - Analog I/O & electrical package characteristics } IBIS
  - Transmit / receive equalization
  - Clock recovery behavior
- The standard must:
  - Support prediction of link Bit Error Rate (BER)
  - Enable EDA & SerDes model interoperability
  - Protect Semiconductor vendor IP
  - Support design optimization

Analytical Methodology

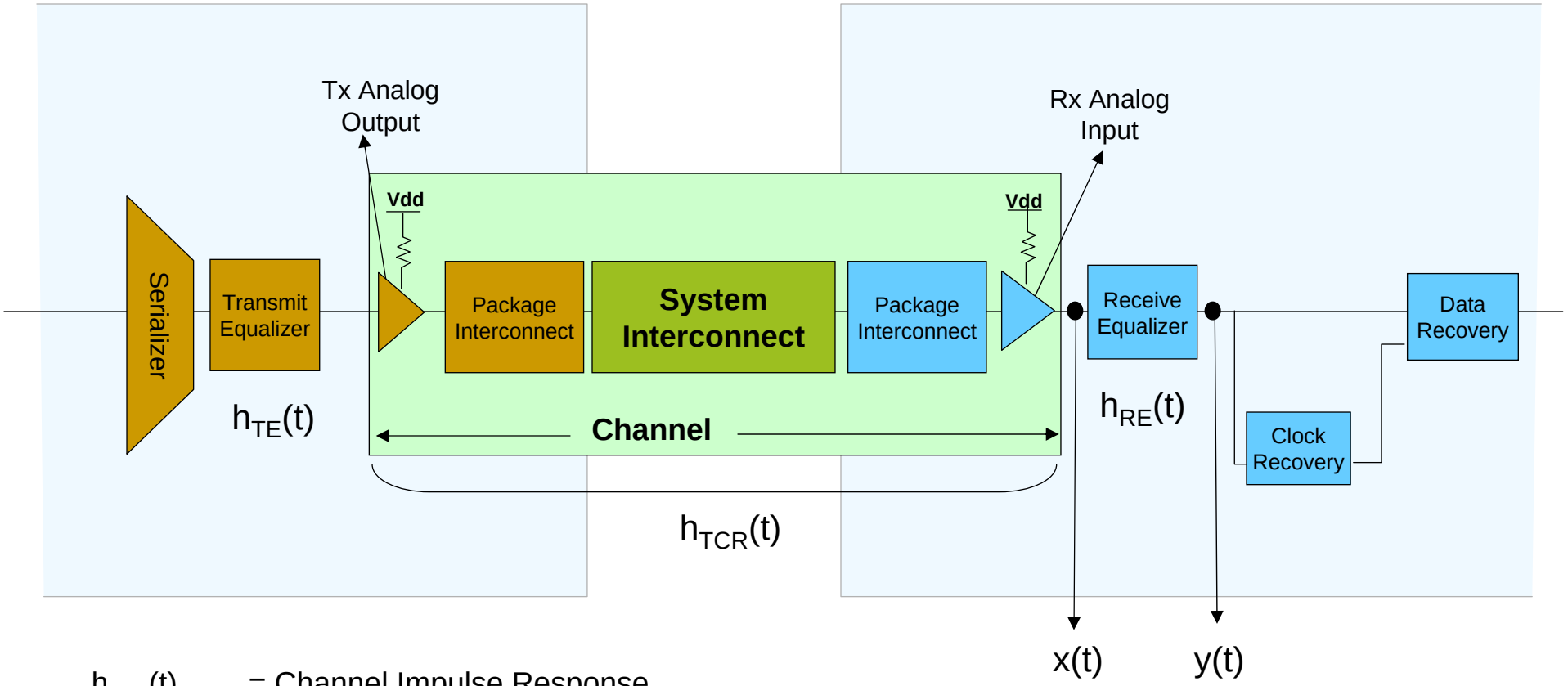


TX EQ
Signal processing

Channel & Analog I/O
Linear, Time-Invariant
Impulse response derived
from circuit simulation/analysis

RX EQ, CDR
Signal processing

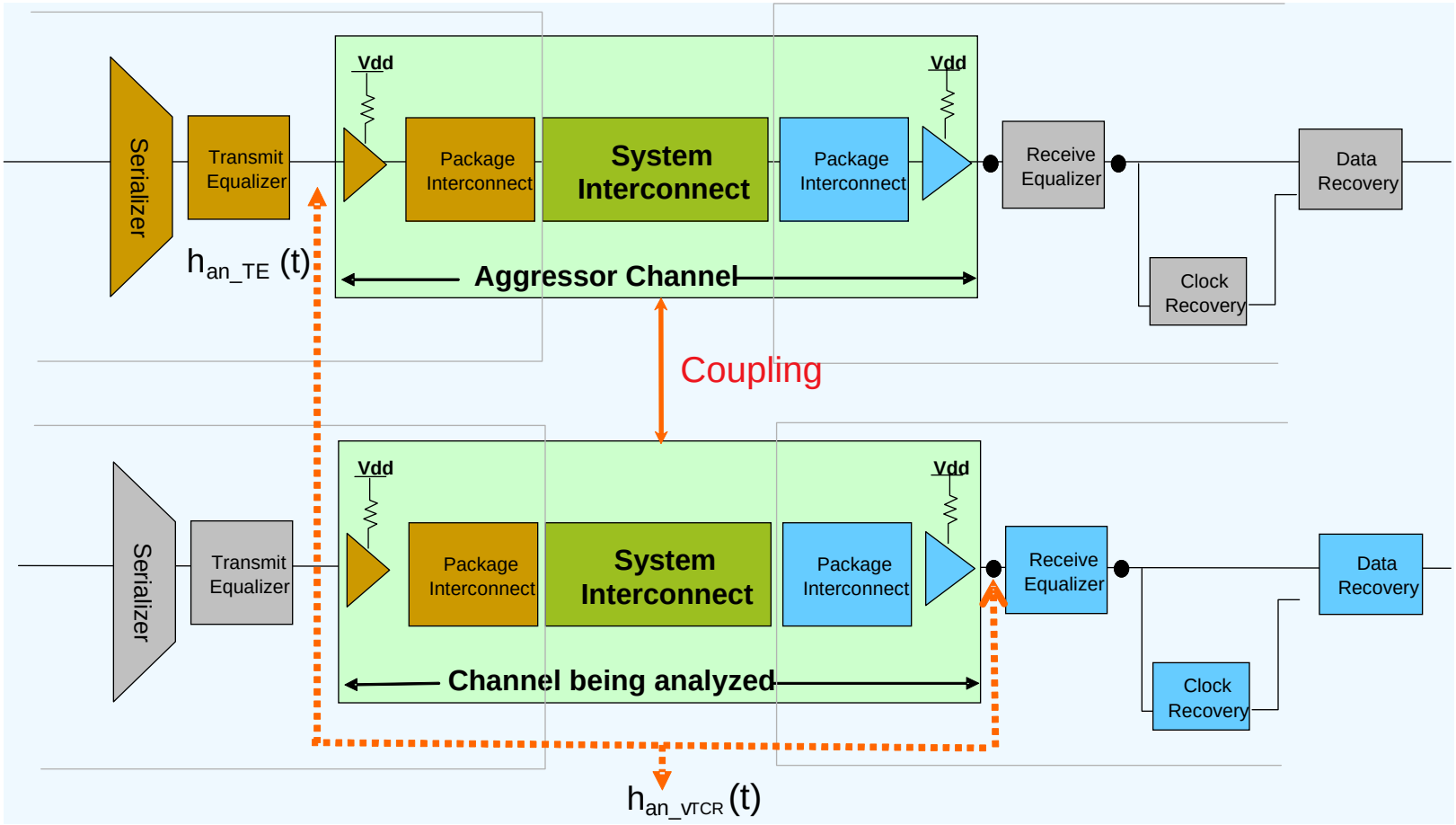
Terminology



$h_{TCR}(t)$ = Channel Impulse Response
$h_{TE}(t)$ = Impulse Response of TX Equalization
$h_{RE}(t)$ = Impulse Response of RX Equalization
x(t) = Signal at receiver input pad
y(t) = Signal at receiver decision point

Channel description includes return paths etc which are not shown for the sake of clarity

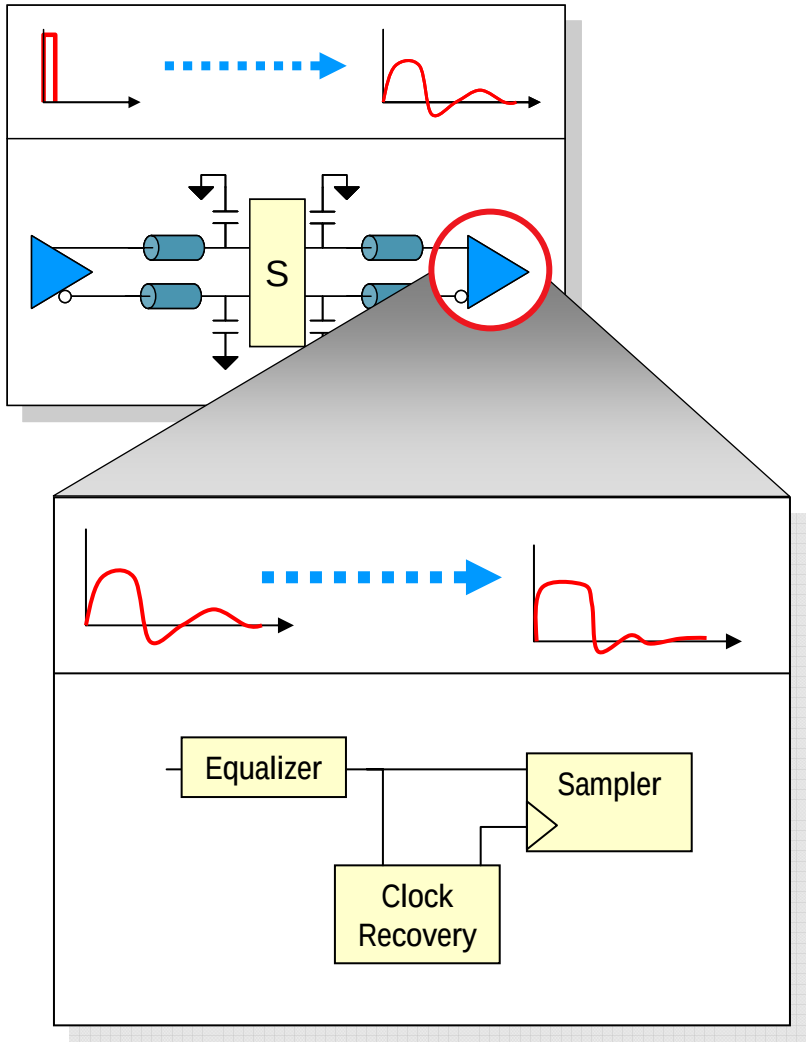
Crosstalk Terminology



Proposed IBIS File Structure

Circuit Simulation

Signal Processing



[Component] MY_COMP

…

[Model] ModelName

Model_Type Output

- Existing IBIS buffer syntax

- [Algorithmic Model]
  …
- [End Algorithmic Model]

[Algorithmic Model] Syntax

- [Algorithmic Model] section points to
  - Executable model code
  - Model-specific parameter file containing
    - A set of “Reserved” Parameters that allow EDA tools to understand what functions the model implements and adapt their flow accordingly
      - e.g., Ignore_bits, Init_Returns_Impulse
    - Model-specific parameter declarations
      - Tell the EDA tool what additional data needs to be passed to the model, what the legal values are and how to format it
    - BIRD covers both parameter types and data exchange formats
    - IBISCHK can parse and report on the parameter file, because formats are well-defined

Examples of IBIS Algorithmic Model Section

```
[Model] Tx_SerDes
Model_type Output
[Voltage_Range] 1.2 1.1 1.3
…
|
[Algorithmic Model]
|
Executable Windows32visualStudio tx_serdes.dll tx_serdes_params.ami
Executable Linuxgcc tx_serdes.dll tx_serdes_params.ami
Executable Solaris tx_serdes.so tx_serdes_params.ami
|
[End Algorithmic Model]
```

```
[Model] Rx_SerDes
Model_type Input
[Voltage_Range] 1.2 1.1 1.3
…
|
[Algorithmic Model]
|
Executable Windows32visualStudio rx_serdes.dll rx_serdes_params.ami
Executable Linuxgcc rx_serdes.dll rx_serdes_params.ami
Executable Solaris rx_serdes.so rx_serdes_params.ami
|
[End Algorithmic Model]
```

Example AMI parameter file

```
(sampleAMI
 (Reserved_Parameters |Reserved Keyword
   | all the parameters under this section are
    defined as part of the standard.  They are
    used by the EDA tool to direct the analysis
    flow
  (Ignore_Bits (Type Integer) (Default 21))
  (Max_Init_Aggressors (Type Integer)(Default 25))
  (Init_Returns_Impulse (Type Bool)(Default True))
  (GetWave_Exists (Type Bool) (Default True))
 ) | End Reserved
```

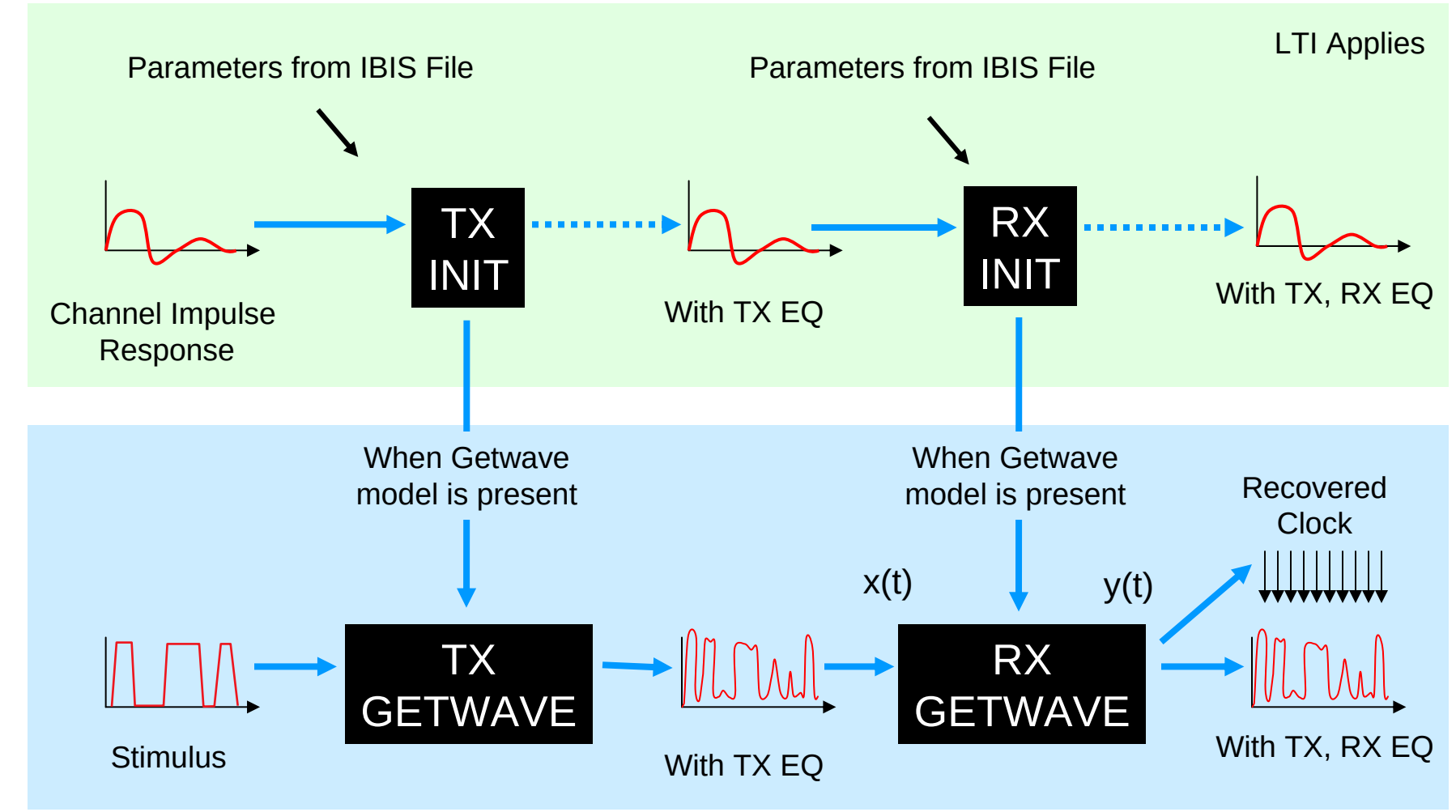
```
(Model_specific |Reserved Keyword
  | txtaps is a structure
  (txtaps
    (tapid (Range -1 3) (type int) )
    (txtapcoeff
        (-1 (Range -0.1 0.1 0.1) (Type float)
    (Default 0))
      (0 (Range 1 1 0) (Type float) (Default 1))
      (1 (List -0.1 0.1) (Type float)(Default  0))
      (2 (Range -0.01 0.02 0.005) (Type float)
      (Default 0))
      (3 (Range -0.01 0.02 0.005) (Type float)
      (Default 0))
    )  | tapcoeff
    )  | txtaps
  (tx_freq_offset (Range 0 150 1) (Type ppm)
    (Default 0))
 ) | End User_Defined
  | here is an example of txtaps instance
  | (txtapcoeff (-1 -0.05) (0 1) (1 0) (2 0) (3 0))
)  | End SampleAMI
```


API Introduction

- Executable computer code models signal processing functions inside the SerDes transmitter / receiver

- **Init:** Impulse response processing
  - Initializes TX / RX model and optimizes device settings
    - Filter coefficients / channel compensation
  - Optionally returns modified input impulse response & equalization settings to EDA platform
- **GetWave:** Continuous time-domain waveform processing
  - Applies equalization & passes waveform to EDA platform
  - Recovers and passes “clock ticks” back to EDA platform
  - Performs any proprietary Post Processing

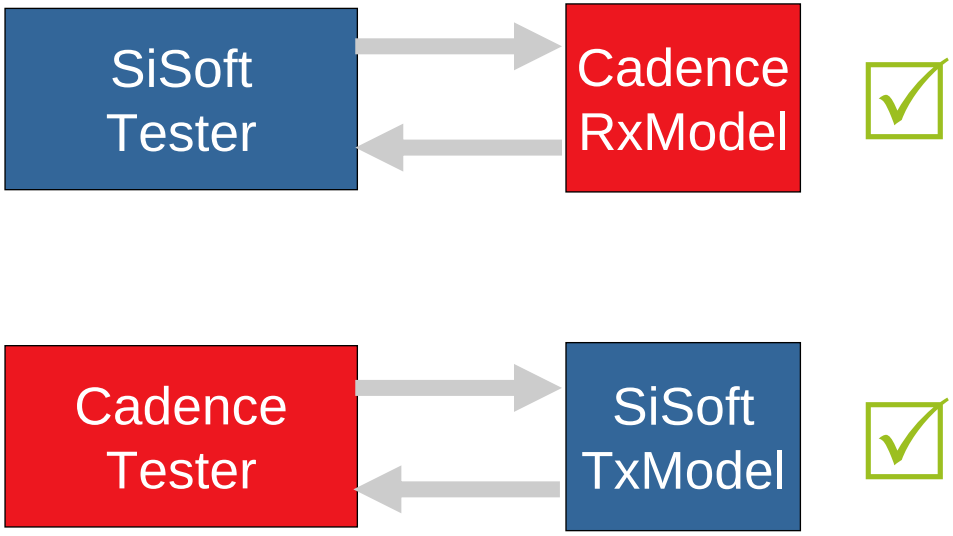
IBIS-ATM Algorithmic Models



Interoperability & AMI Toolboxes

- Interoperability tests performed successfully with sample AMI models
  - Cadence Rx AMI Model worked with SiSoft Tester program
  - SiSoft Tx AMI Model worked with Cadence Tester program

- Toolboxes, toolkits availability
  - SiSoft AMI toolkit is available on IBIS-ATM work archive
  - Cadence AMI toolbox will be available shortly



Next Steps & conclusion

- Formally submit the BIRD

- We look forward to feedback on AMI toolboxes from IBIS Open Forum