Algorithmic Modeling Interface (AMI) Proposed Changes to IBIS

C. Kumar, Architect
Cadence Design Systems, Inc.
October 17, 2006

Proposed changes to IBIS

- Introduce a new section (“AMI”) with a unique name that is parallel to External Model construct
- AMI section sits on top of and leverages the circuit simulation infrastructure
  - Algorithmic model requires existing IBIS structure to represent the Tx and Rx load models
  - These Tx, Rx models along with the channel constitute a Linear Time Invariant (LTI) system
- AMI section introduces
  - Three API calls: AMI_init, AMI_GetWave, AMI_Close
  - Each call provides a means for model developer to pass algorithmic model specific parameters: # of filter taps, filter tap spacing, etc
    - Model developer provides documentation on parameters to model consumer
  - An AMI section can have multiple algorithmic models: for example one for Amplifier (eye opener) and another for DFE/CDR
    - Simulation platform expected to call each AMI section in the order it appears in the AMI section

Syntax Structure

```
|-- [Model]                          Model_type, Polarity, Enable,
|   -------                          Vinl, Vinh, C_comp, C_comp_pullup,
|      |                             C_comp_pulldown,
|      |                             C_comp_power_clamp,
|      |                             C_comp_gnd_clamp
|      |                             Vmeas, Cref, Rref, Vref
|      |                             Rref_diff, Cref_diff
|      |
|      |-- [Model Spec]              Vinh, Vinl, Vinh+, Vinh-, Vinl+,
|      |                             Vinl-, S_overshoot_high,
~~~
|      |                             Receiver_model_inv, R_diff_near,
|      |                             R_diff_far
|      |
|      |-- [External Model]          Language, Corner, Parameters,
|      |   ----------------          Ports, D_to_A, A_to_D
|      |     |-- [End External Model]
|      |
|      |-- [Add Submodel]
```

Add [AMI] here

[AMI] Syntax

[AMI] AMI_name

| Initialization function API
AMI_Initial()
Parameter a=5
Parameter b=10
……

| GetWave function API
AMI_GetWave()

| Clear and Close function API
AMI_Close()

[End AMI]

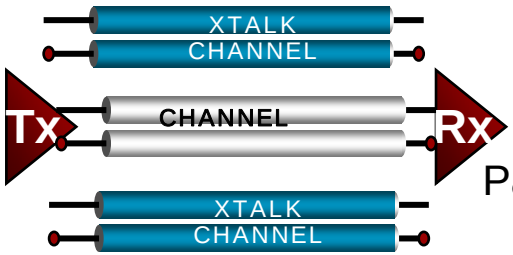
Simple API

- Init
  - Initialize and optimize channel with Tx / Rx Model
  - This is where the IC DSP decides how to drive the system: e.g., filter coefficients, channel compensation, …
  - Input: Channel Characterization, system and dll specific parameters from config file
    - bit period, sampling intervals, # of forward/backward coefficients, …
  - Output: Modified Channel Characterization, status
- GetWave
  - Modify continuous time domain waveform [CDR, Post Processing]
  - Input: Voltage at Rx input at specific times
  - Output: Modified Voltage, Clock tics, status
- Close
  - Clean up, exit

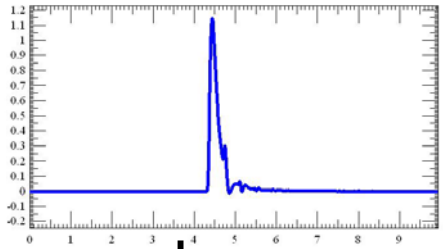
Parameters passed by the system simulation platform are in red

AMI_Init

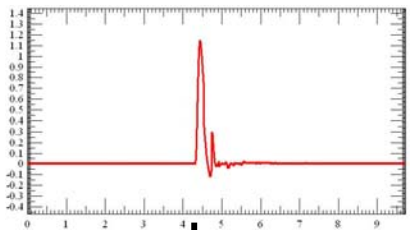
Simulation Platform
Generates and consume
Impulse response matrix



Pass characterization
in matrix



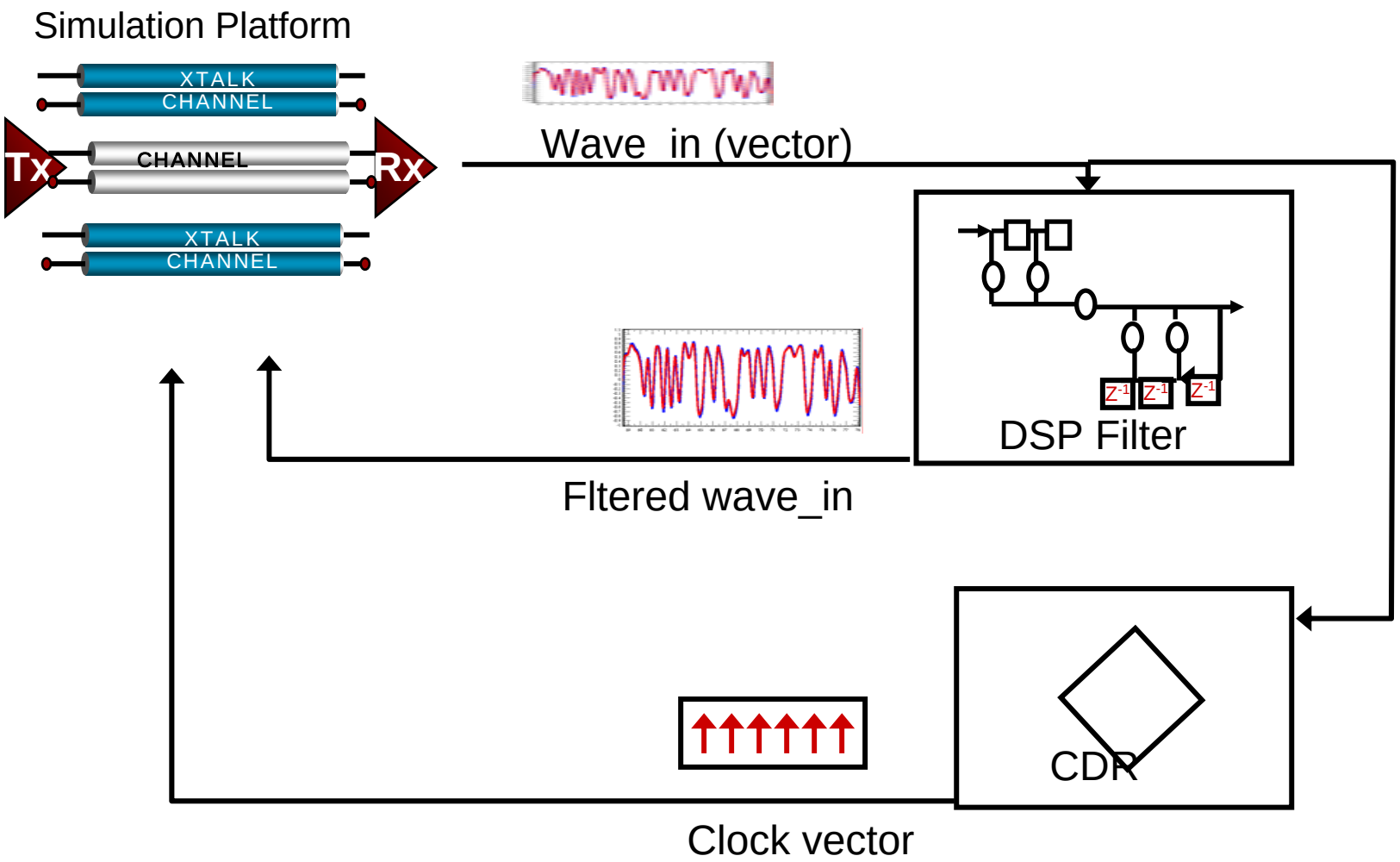
DSP algorithms
modify characterization



Send modified char back
(modified matrix)

Internal storage

AMI_GetWave


Simulation Platform
XTALK
CHANNEL
Tx
CHANNEL
Rx
XTALK
CHANNEL
Wave_in (vector)
DSP Filter
Fltered wave_in
CDR
Clock vector