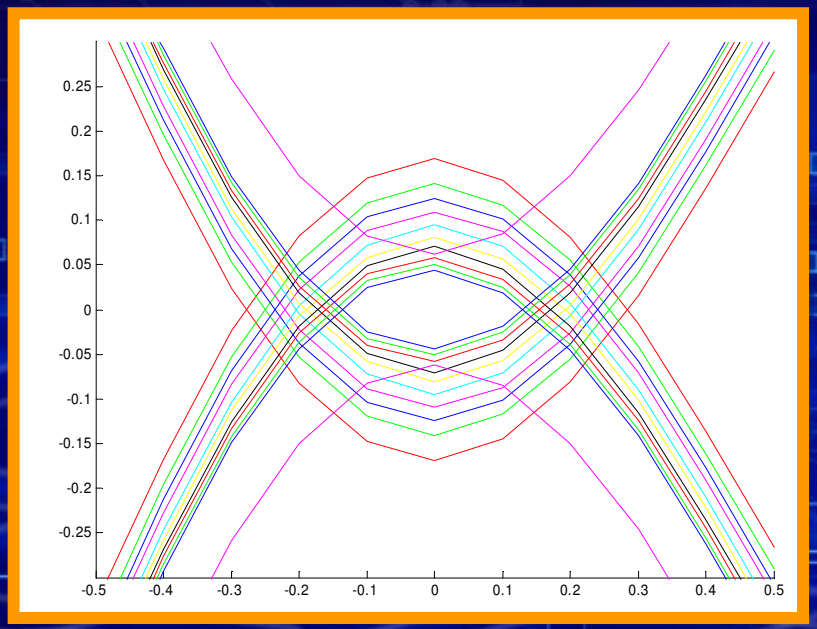
Comparing Walter's and Kumar's IBIS-AMI Proposals

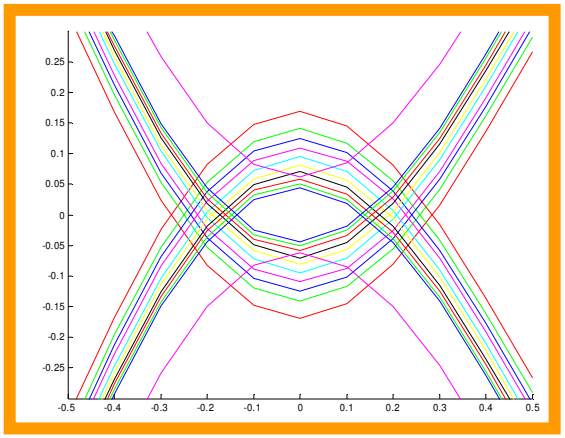
IBIS ATM Teleconference, September 8, 2009



Arpad Muranyi

Mentor Graphics®

Comparing Walter's and Kumar's IBIS-AMI Proposals



IBIS ATM Teleconference, September 8, 2009

1. Purpose of this presentation
2. First disagreement
3. Where is the real difference
4. Kumar's justification
5. Conclusion

Purpose of this presentation

- This presentation compares Walter's presentation given in the IBIS-ATM teleconference on August 18, 2009:
  - "IBIS 5.0 does not support Non-LTI transmitter models! Solution: Input to Tx GetWave should be a digital stimulus waveform"
    http://www.vhdl.org/pub/ibis/macromodel_wip/archive/20090818/walterkatz/Tx_GetWave%20Digital%20Input%20Proposal/Digital_input_Tx_GetWave.pdf
- with Kumar's response presentation given on the IBIS-ATM teleconference on September 8, 2009:
  - "IBIS ATM: txgetwave"
    http://www.vhdl.org/pub/ibis/macromodel_wip/archive/20090901/kumarkeshavan/IBIS%20ATM:%20txgetwave/atm_txgetwave.pdf
- with the purpose of finding a solution to the apparent disagreement

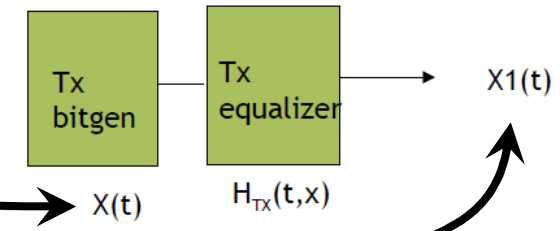
First disagreement (?)

- Walter says that in the presence non-LTI effects (DCD, jitter, n-tap table lookup, etc.) the AMI flow is broken
  - (Walter's slides 7, 9)
- Kumar says that "bit distortion like DCD does not make Tx equalizer non LTI"
  - (Kumar's slide 6)
  - Kumar explains: "The reason for this is that the Tx equalizer is independent of the bit generator"
- Walter's examples (3-tap table lookup and DCD) involve non-LTI transfer functions
  - (Walter's slides 8, 12)
  - the table lookup in the equalizer creates a non-linear transfer function
  - the 75% DCD bit pattern is time variant (as shown in the example) and the otherwise LTI Tx equalizer will yield a non-LTI transfer function because it is bit pattern dependent
- For case 1 Kumar states that the Tx equalizer and channel transfer functions are time and bit pattern independent
  - (Kumar's slides 4-5)
  - "$H_{tx}(\tau)$ is the equalizer transfer function. In case1 it is independent of x(t)"
  - "…$H_{Tx}(\tau)$ and Hc(τ) is LTI (i.e. they have time independent characteristic transfer functions)…"

The first problem is a non issue

- **Kumar says (slide 7):**
  - You need getwave only in the case where
    - $H_{tx} == H_{tx}(\tau, x(t))$
    - the equalizer depends on its input (x(t)) and is 'time varying'
  - **Which is in agreement with Walter's DCD example**
    - **Walter's Tx equalizer does depend on the input pattern (which is what makes it time varying)**
  - **Kumar's equation on the bottom of slide 7 seems to agree with Walter's proposed equations on slide 4**

- Waveform input to TX AMI_Getwave ("Analog Stimulus")
  - $p(t) \otimes b(t) \otimes h_{AC}(t)$
  - Should be $p(t) \otimes b(t)$
- Waveform input to RX AMI_Getwave
  - $g_{TEG}(p(t) \otimes b(t) \otimes h_{AC}(t))$
  - Should be $g_{TEG}(p(t) \otimes b(t)) \otimes h_{AC}(t)$



Where is the real difference?

- **Although not explicitly stated, Walter seems to want the channel response to be convolved inside the Tx GetWave**
  - **(Walter's slide 4)**
- **Kumar wants the channel response be convolved in the EDA tool instead**
  - **(Kumar's slide 8)**

- Waveform input to RX AMI_Getwave
  - $g_{TEG}(p(t) \otimes b(t) \otimes h_{AC}(t))$
  - Should be $g_{TEG}(p(t) \otimes b(t)) \otimes h_{AC}(t)$

- The eda tool can produce an wave form y(t) at the rx input in a flexible manner.
- This division (i.e tx ouputs only x1(t) and does not go out of its domain to produce rx input) is more natural and should be the only one supported for txgetwave

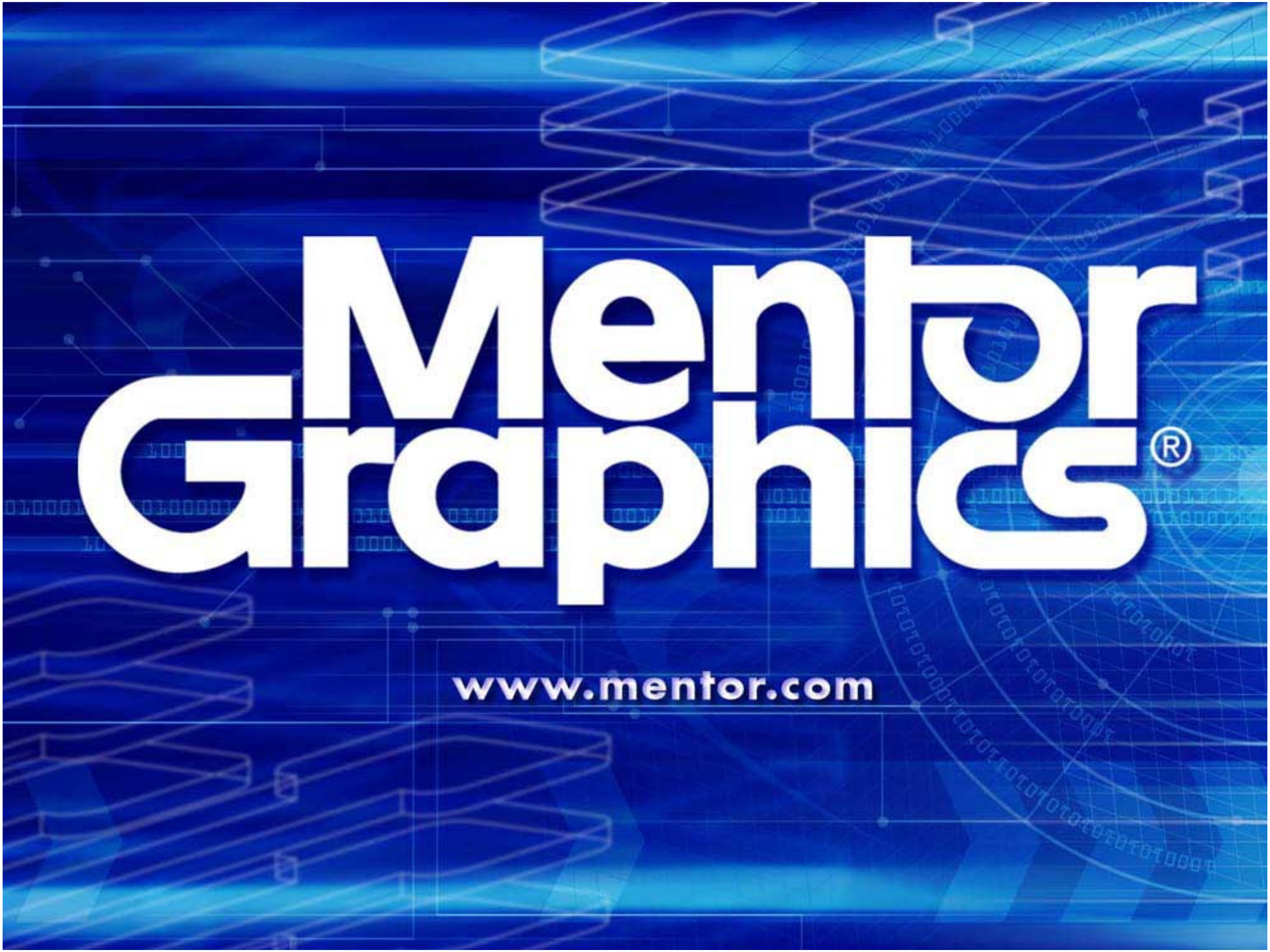
Kumar's justification

- **The channel and its corresponding waveforms are in the "analog domain" and belong to the EDA tool**
- **The Tx filter and its associated waveforms are in the "digital domain" and belong to the DLL (Tx GetWave)**
  - **(Kumar's slide 8)**

    - The analog front end channel is in the eda domain; x1(t) is from the tx device and is neatly separated
    - The eda tool can produce an wave form y(t) at the rx input in a flexible manner.
      - For example if it is deemed that front end channel non linearity has to be taken into account it can be done
    - Removes cross talk drive complication
    - This division (i.e tx ouputs only x1(t) and does not go out of its domain to produce rx input) is more natural and should be the only one supported for txgetwave

Conclusion

- Walter and Kumar seems to be in agreement regarding the technical aspects of this discussion
- The only difference seems to be the question of where the channel response is convolved with the Tx filter's output
  - EDA tool
  - Tx GetWave function
- At the end of the teleconference Walter indicated verbally that he is in agreement with Kumar's recommendation that the EDA tool does the channel response
- With that, there seems to be no differences to be resolved on this topic!


Mentor Graphics®
www.mentor.com