More experiences

with IBIS-AMI models

Eckhard Lenski

European IBIS Summit

Sorrento, Italy

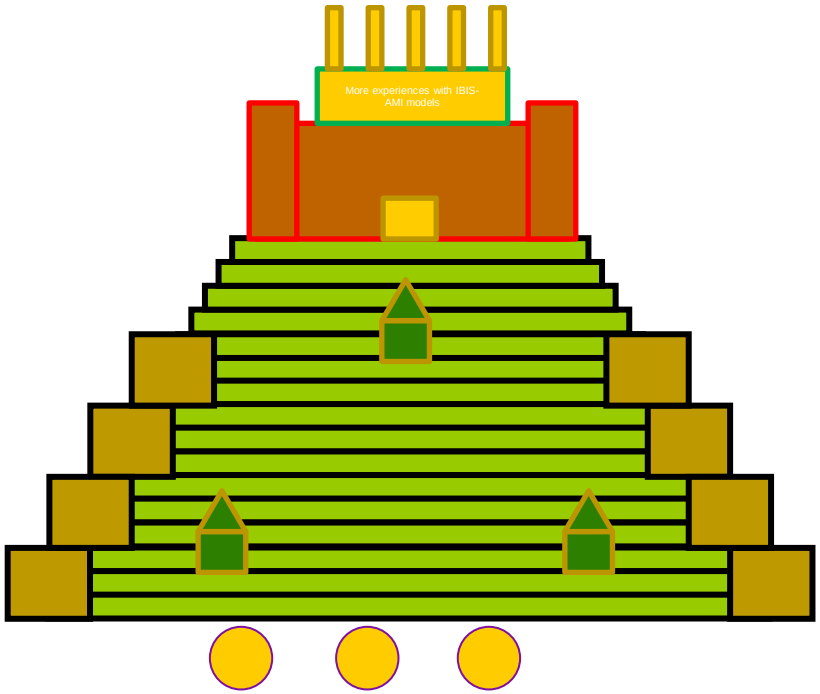
May 16, 2012

Public

More experiences with IBIS-AMI models

summary

Why IBIS-AMI



IBIS-AMI models

Differences IBIS vs IBIS-AMI

Basics IBIS-AMI

Public

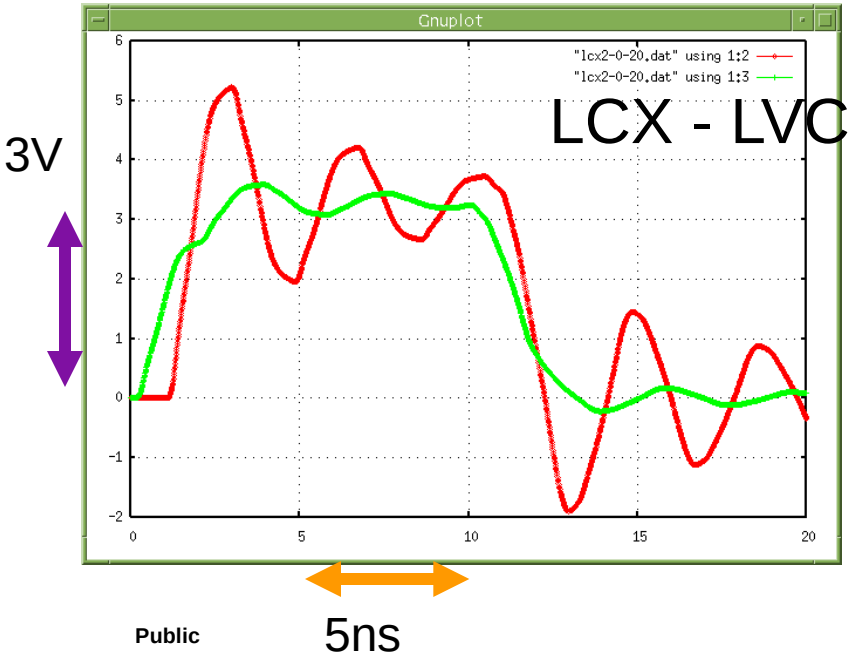
Higher datarates

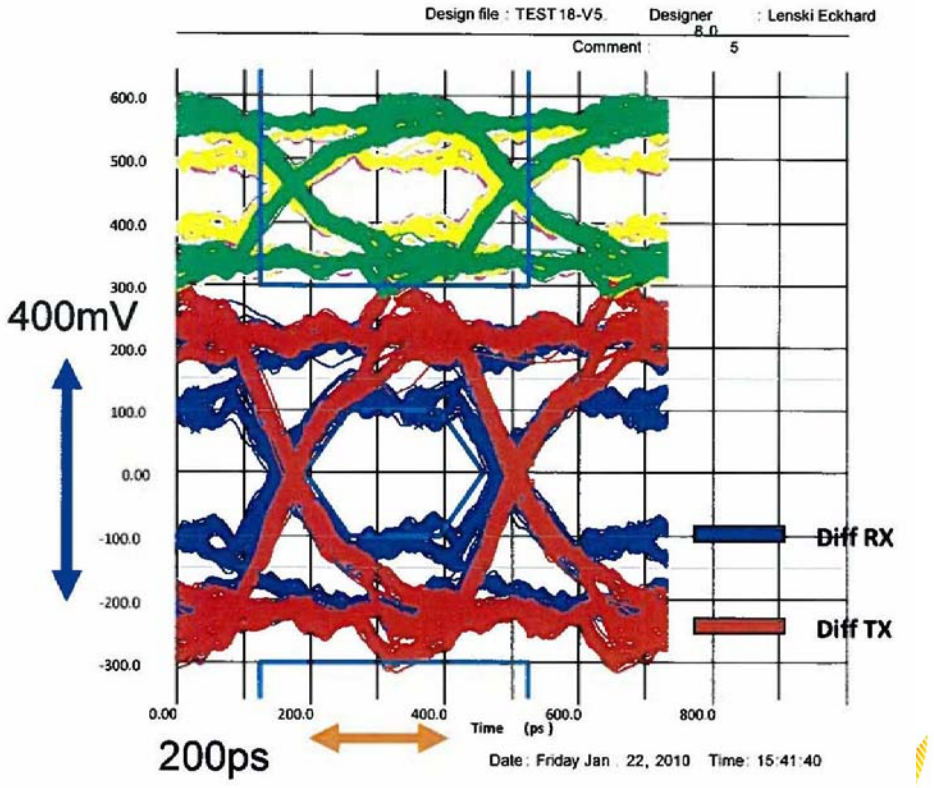
Classic SI

- Rise fall times
- Overshoots

BER, ISI

- UI
- Eye opening
- Jitter





Higher datarates

10Gbps and more

- PCI-Express 2, 3, 4
- Serial ATA 2, 3
- USB 3
- XAUI
- CEI 11G
- Emerging 25 / 40 / 100 Gbps standards

Above 1GHz

- ISI (Inter Symbol Interference)
- Attenuation
- Jitter
- Influence of vias
- Influence of connectors

Nokia Siemens Networks

Analysis time

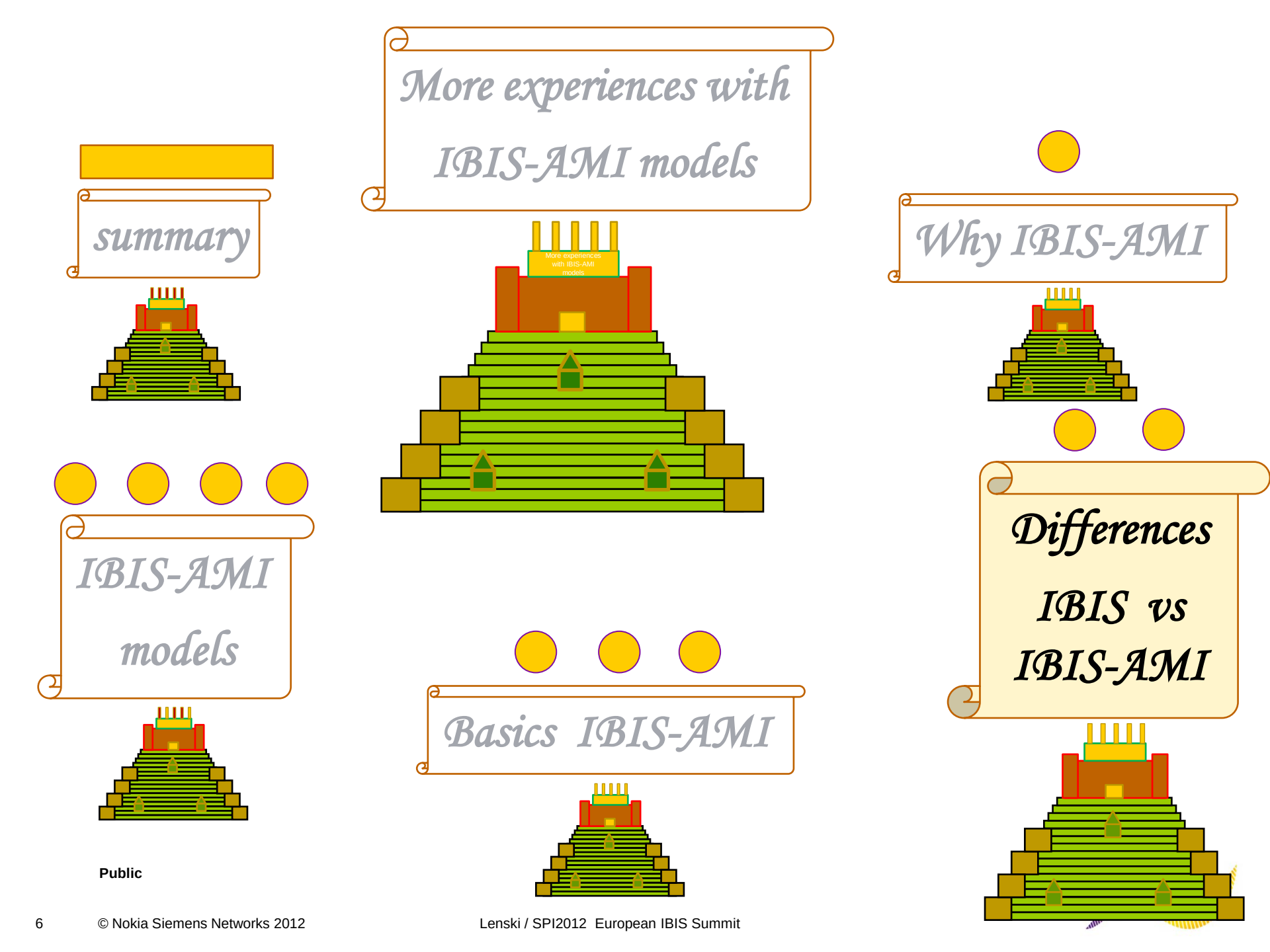
Estimated time for determination of BER

Eg 6.375Gbps

BER	Measurement	Spice	Ibis-ami
3e-12	1 minute	hour	seconds
1e-15	2 days	days	a few minutes
6e-18	1 year	more days	More minutes

Source: measurement data from ***Bogatin*** **Altera high speed seminar 2006**

Nokia Siemens Networks

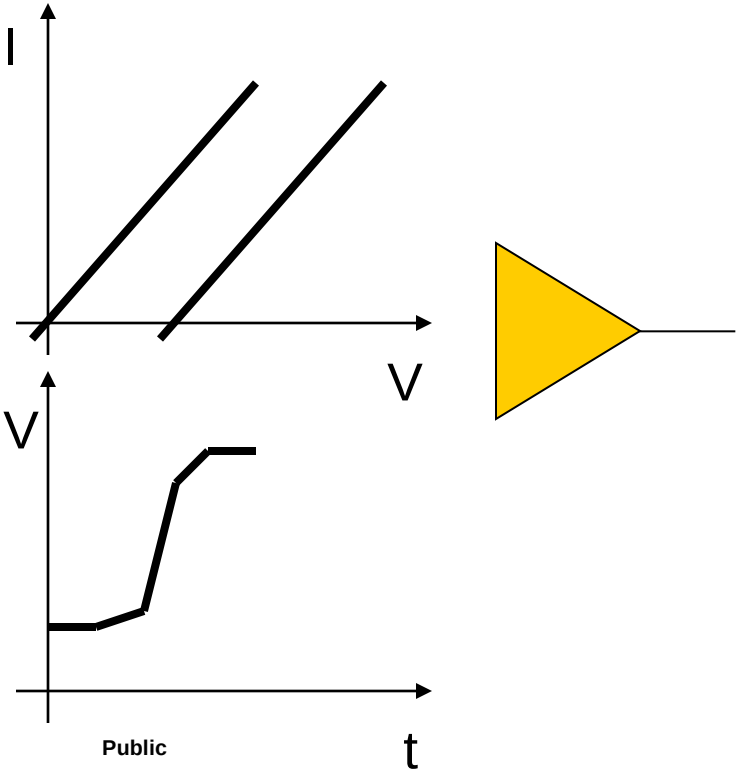


Public

IBIS

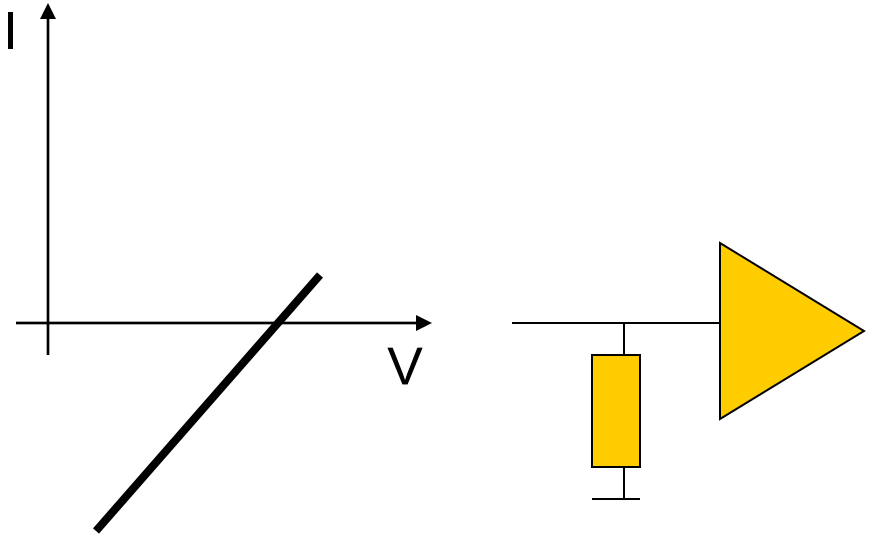
Driver

- U-I curves
- U-t curves



Receiver

- U-I curves
- Params (vinh,vinl)



Modeling of high speed features

Tx & Rx with	In IBIS	IBIS-AMI
equalization	-	X
Pre-emphasis		
— 2 tap	— X	X
— 3 tap	— -	X
FFE	-	X
DFE	-	X
jitter	-	X

Algorithmic Modeling Interface

Nokia Siemens Networks

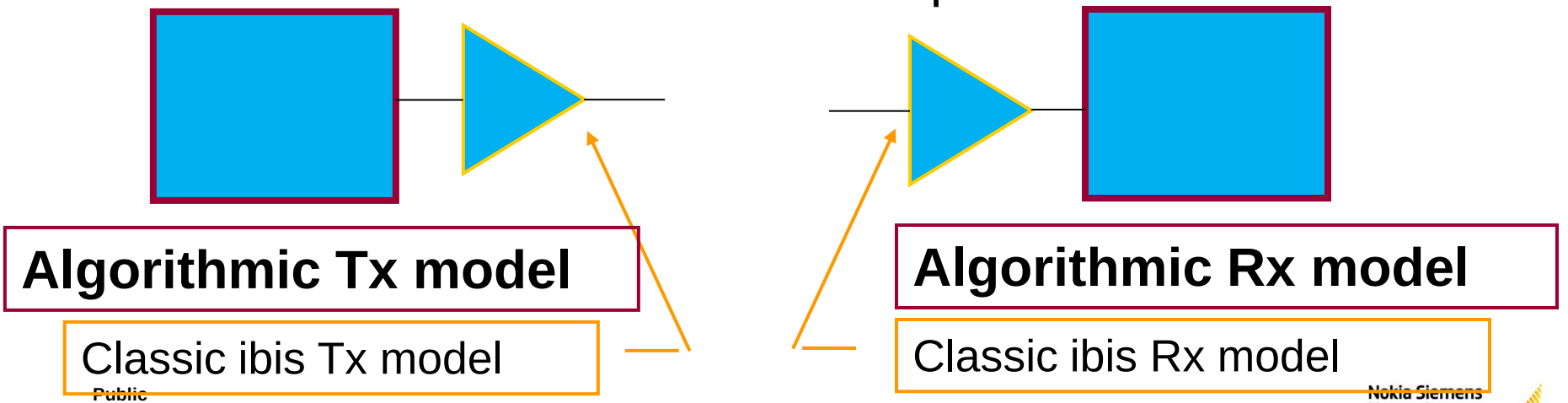
IBIS-Ami (Algorithmic modeling interface)

Driver

- FFE
- Swing
- Taps / Pre-emphasis

Receiver

- DFE
- Taps
- Jitter
- CDR
- Equalization



Public

platform

IBIS

- Runs on every platform
- Same model

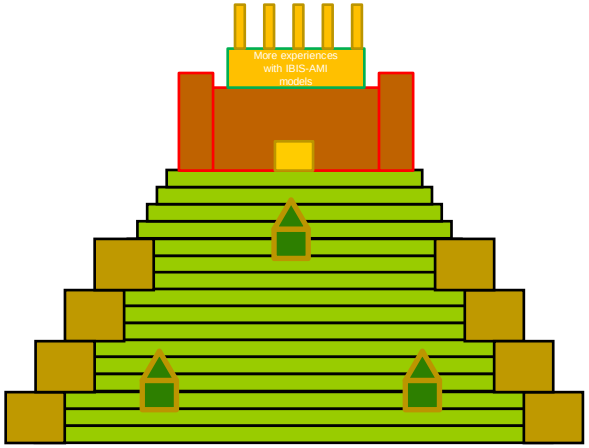
IBIS-AMI

- Platform dependant
  - Linux
  - windows
- 32 / 64 bit
- different model
  - Model.so (linux)
  - Model.dll (windows)
- documentation file necessary

summary

More experiences with IBIS-AMI models

Why IBIS-AMI



IBIS-AMI models

Basics IBIS-AMI

Differences IBIS vs IBIS-AMI

Public

IBIS-AMI keyword

[Model] Tx

- Model_type Open_sink
- [Voltage Range] 1.2 1.14 1.26
- C_comp 1.00p 0.95p 1.05p
- [Pullup]

- **[Algorithmic Model]**
- **Executable Windows_VisualStudio_32 NSNTx.dll NSNTx.ami**
- **Executable Linux_gcc3.2.3_32 NSNlibTx.so NSNlibTx.ami**
- **[End Algorithmic Model]**

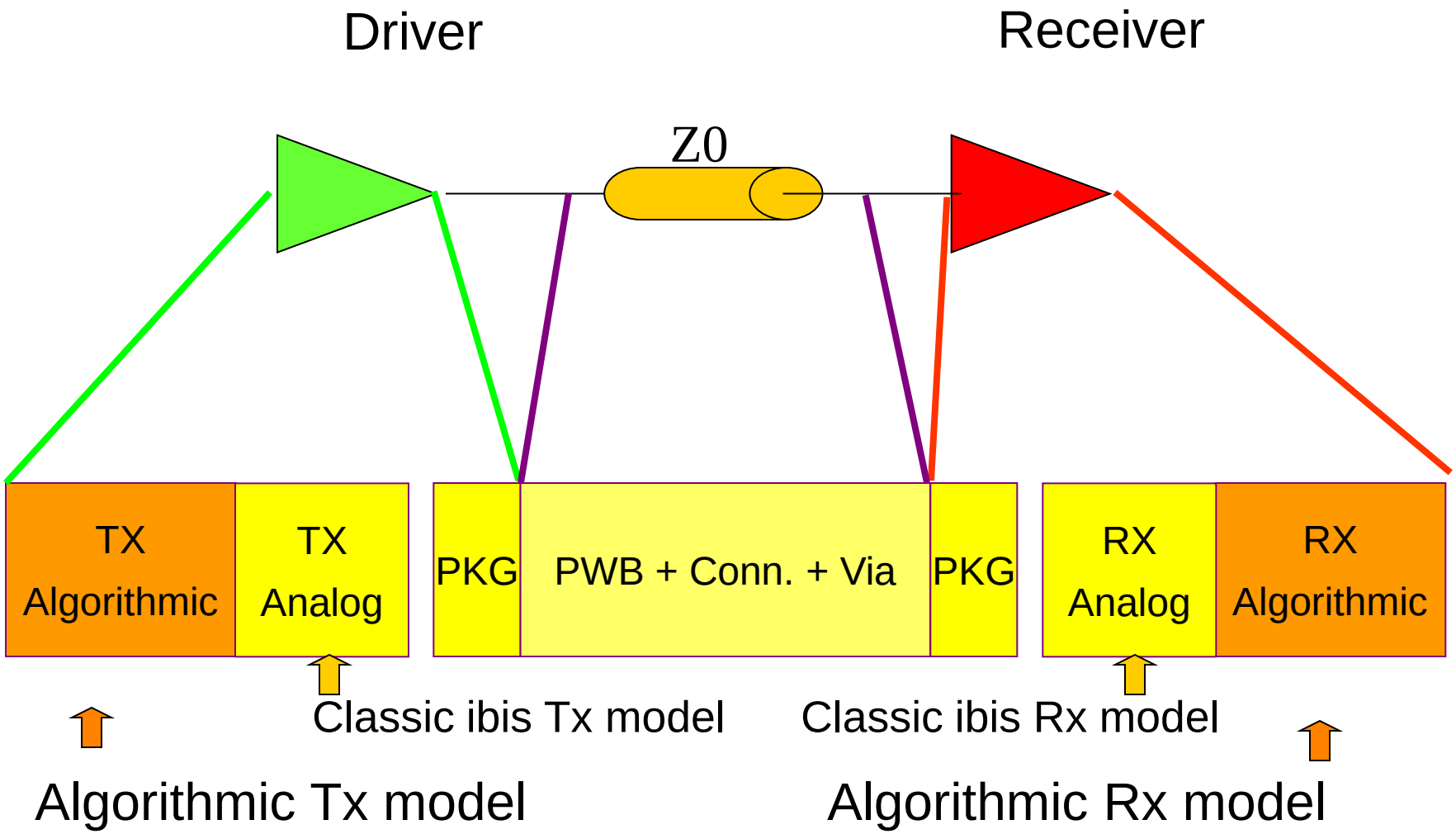
Nokia Siemens Networks

Executable
Windows_VisualStudio_32
NSNTx.dll
NSNTx.ami

- Executable → Subparameter
- Windows_VisualStudio_32 → Platform_Compiler_Bits
- NSNTx.dll → name of shared library file
- NSNTx.ami → Description file for params

Nokia Siemens Networks

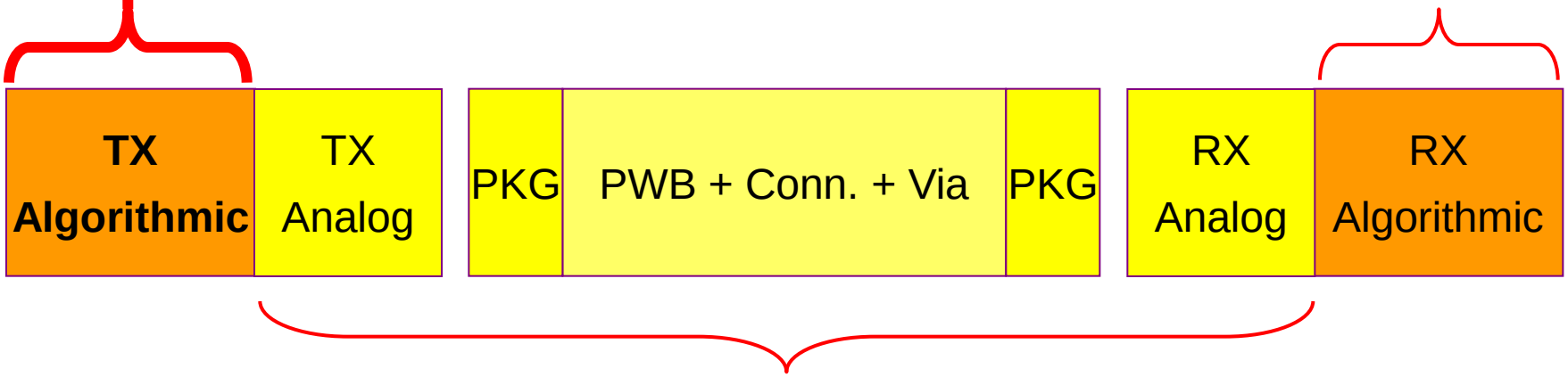
IBIS-AMI principle



IBIS-Ami driver

Signal processing in the driver before the final output stage

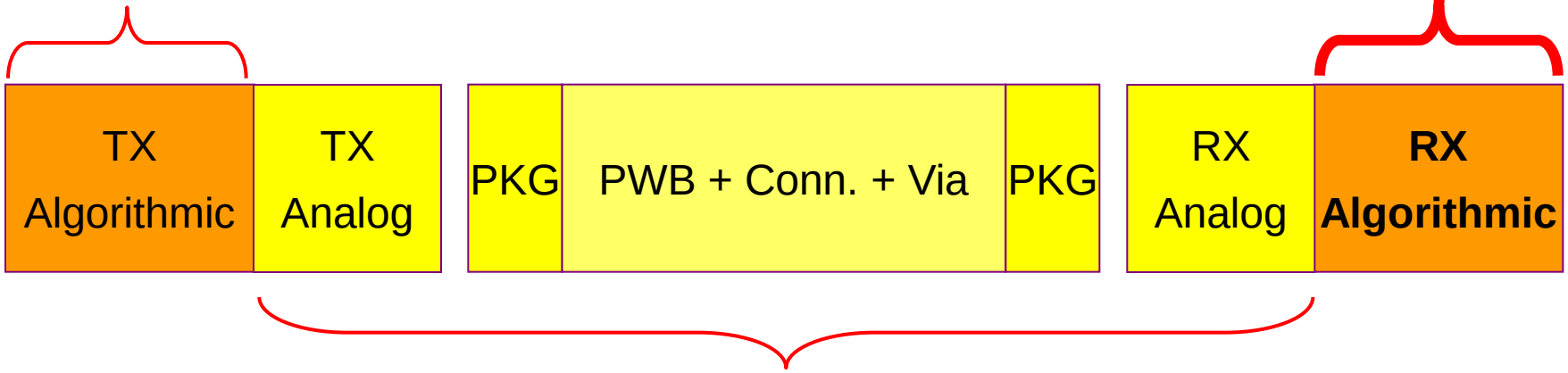
- equalization/pre-emphasis (FFE feed forward equalizer)
- encoding (8b/10b)



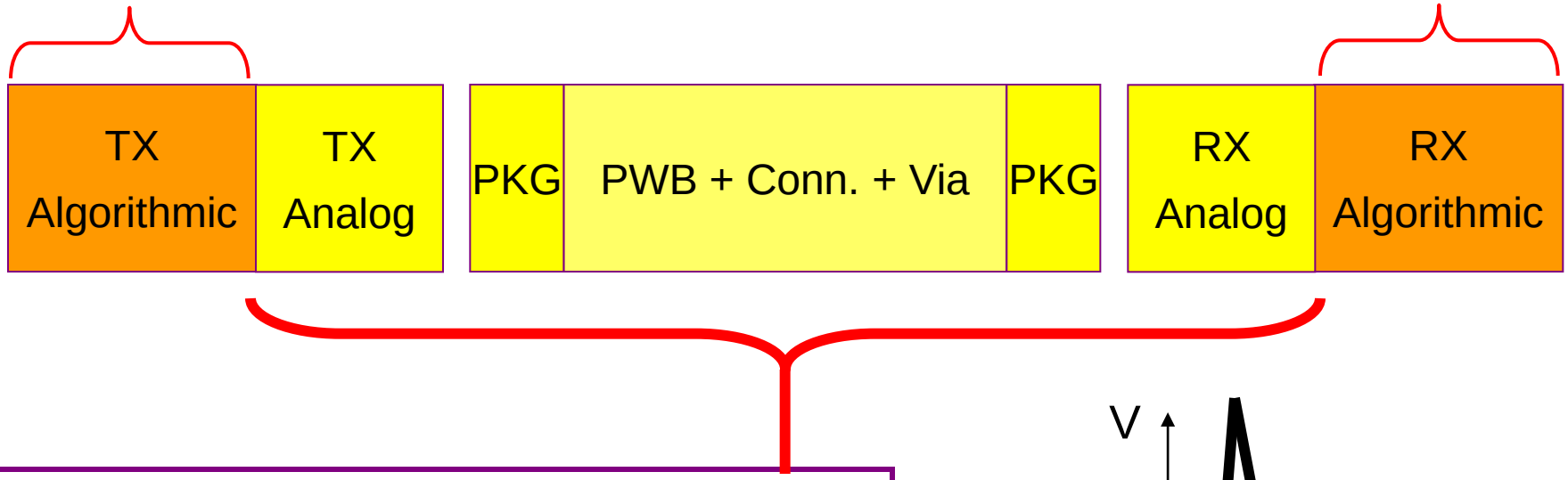
IBIS-Ami receiver

Signal processing in the receiver after the first input stage

- equalization (e.g. DFE decision feedback equalizer)
- clock recovery

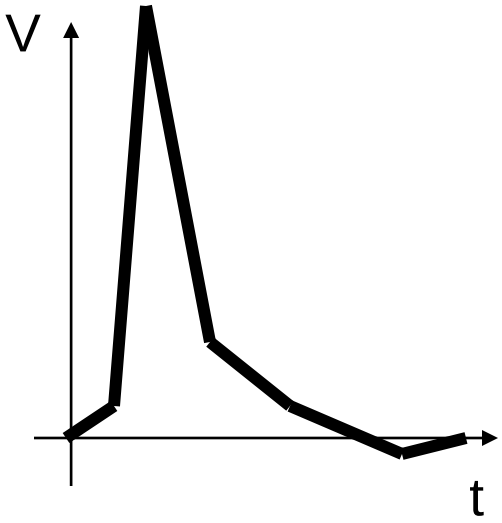


IBIS-Ami channel



Impulse response representing the analog channel

- Driver (TX) final output stage
- Channel
- Receiver (RX) input stage



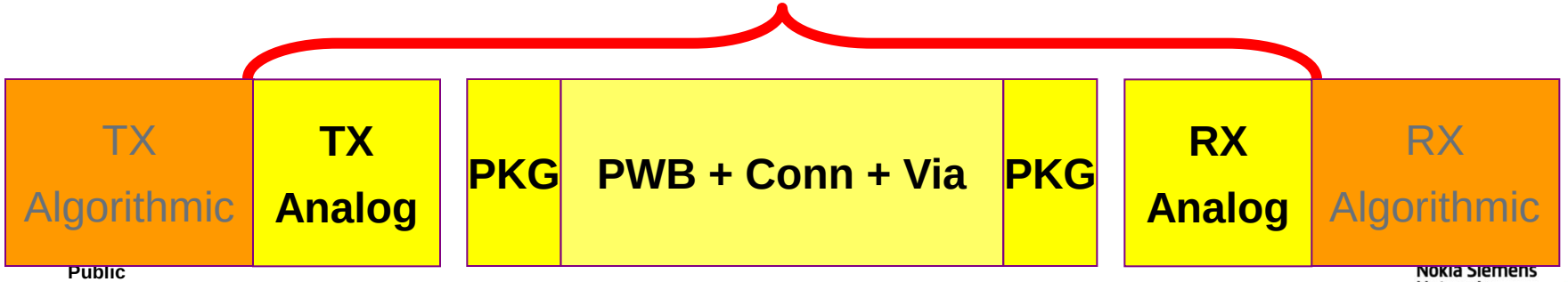
Channel must be LTI (Linear and TimeInvariant)

Public

Nokia Siemens Networks

IBIS-AMI channel elements

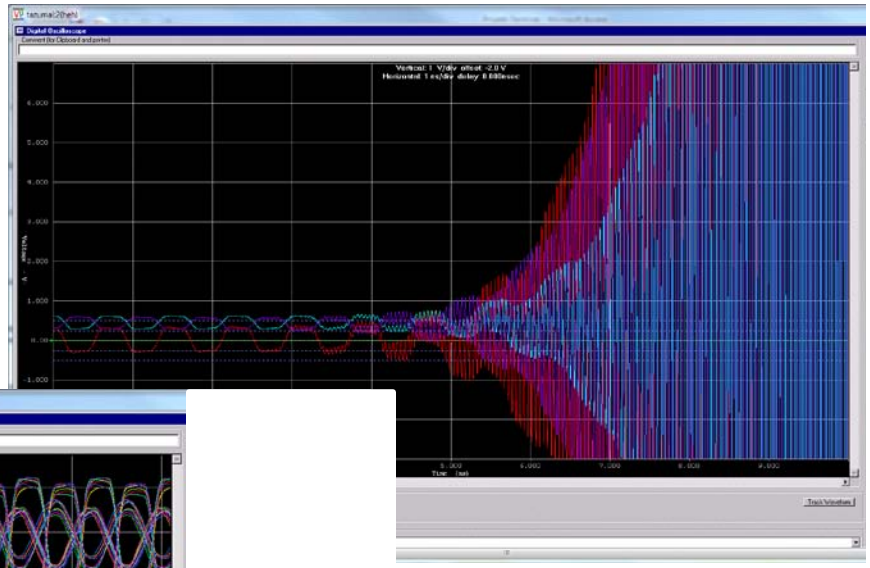
Elements	models
Tx analog	Classic ibis driver
Rx analog	Classic ibis receiver
Pkg	S-params
Channel, T-line	W-element , S-params
Via	S-params
connector	S-params

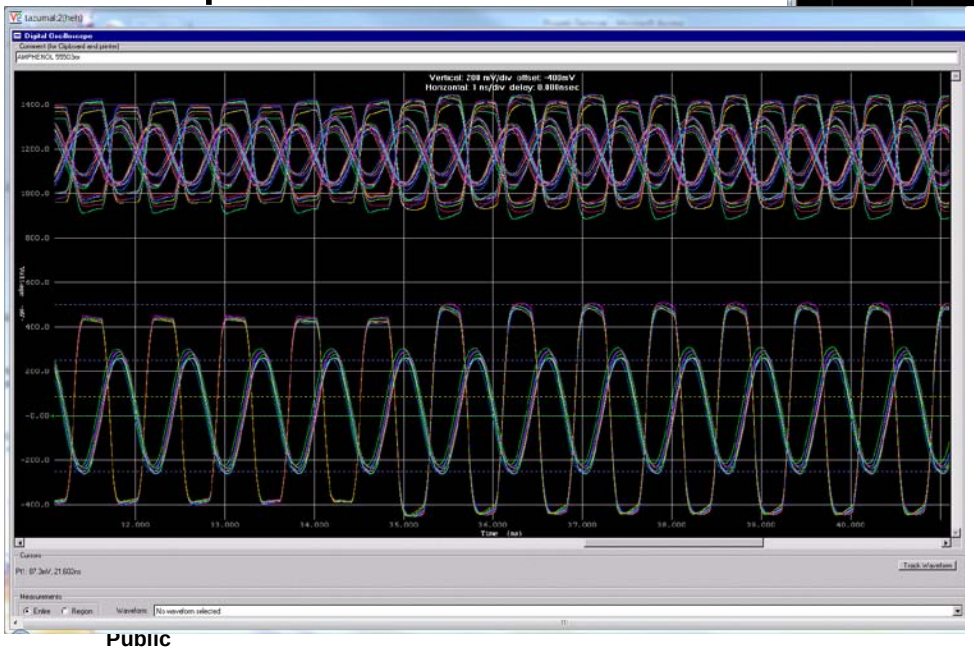


Modeling of high speed features

Question

- Which simulation
  - used the good
  - used the bad
- S parameter model ?





Nokia Siemens Networks

IBIS-AMI driver / receiver elements

TX

- AMI_INIT
- AMI_Getwave
- AMI_close

RX

- AMI_INIT
- AMI_Getwave
- AMI_close

TX/RX Ami

- AMI_INIT
  - Used for statistical analysis, setting for getwave parameters, LTI
- AMI_Getwave
  - Timing analysis, non LTI behavior
- AMI_close
  - Clear memory, etc.

IBIS-AMI parameters

General parameters

- Init_Returns_Impulse
- Getwave_Exists
- (Use_Init_Outputs)
- Ignore_Bits
- Max_Init_Aggressors
- ……

Specific model parameters

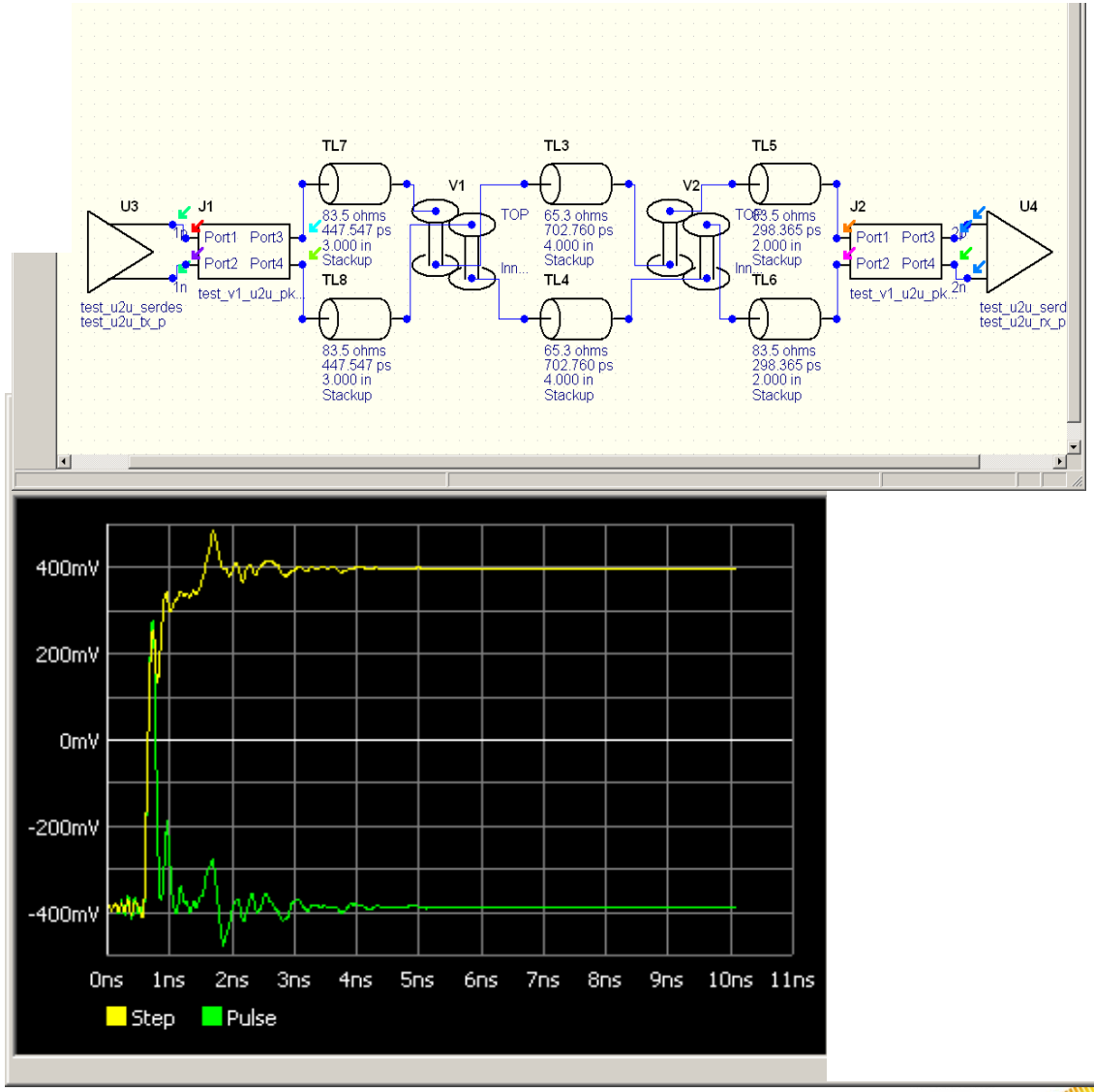
- Tap settings
- Equalization settings
- Driver strength
- PVT conditions
- Gain
- ……

IBIS-AMI topologie

Impulse response

- Step response
- Impulse response

- Each topologie has its own impulse response



Public

Selection of model specific parameters

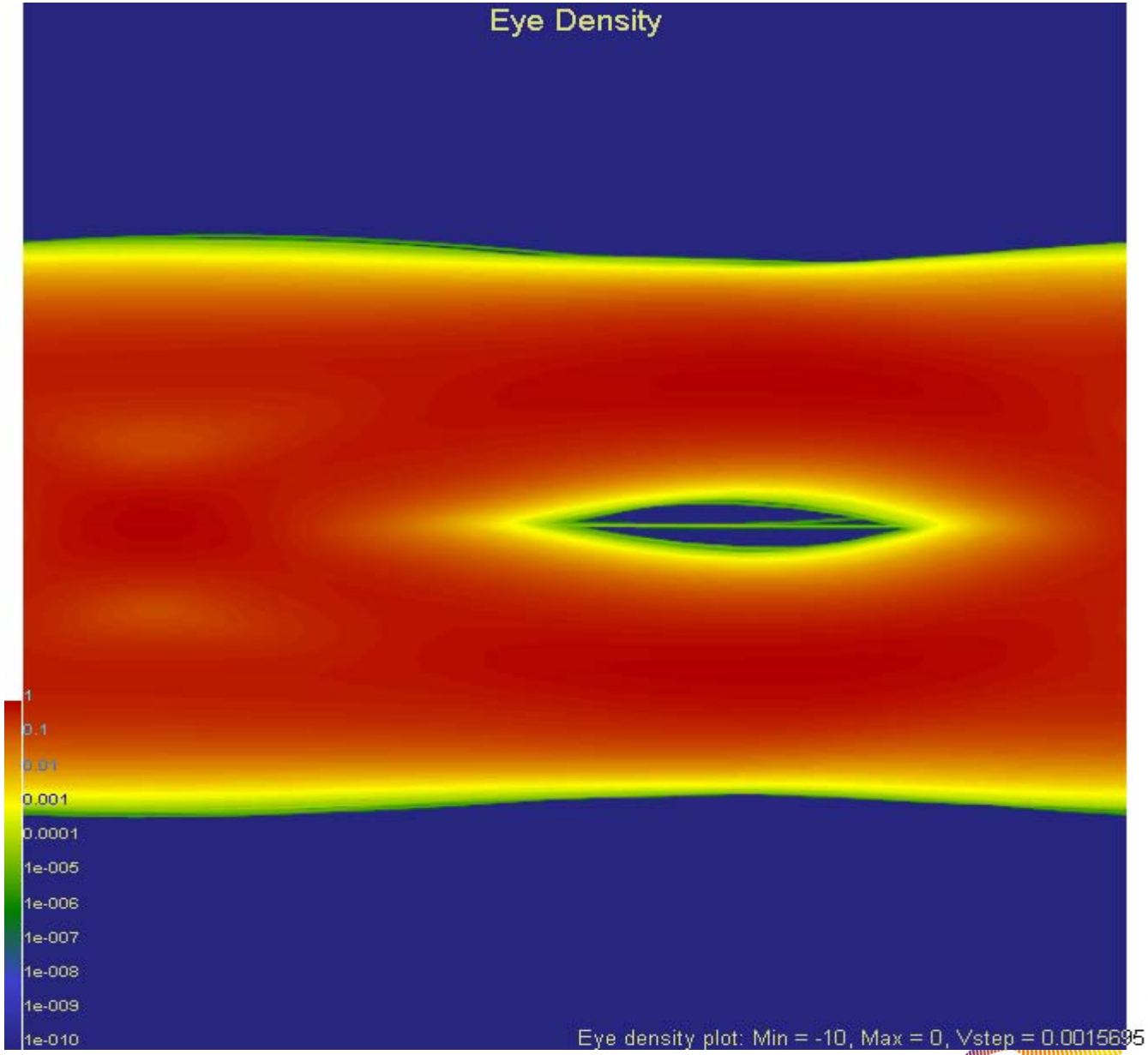
Specific model parameters

- Tap 0
  - Value 1
- Tap 1
  - Increment 0 -2 3 0.2
- Equalization settings
  - Range 0 , 0.5 , 1
- Driver strength
  - list 200mv , 400mv , 600mv , 800mv

IBIS-Ami first results

Model settings :

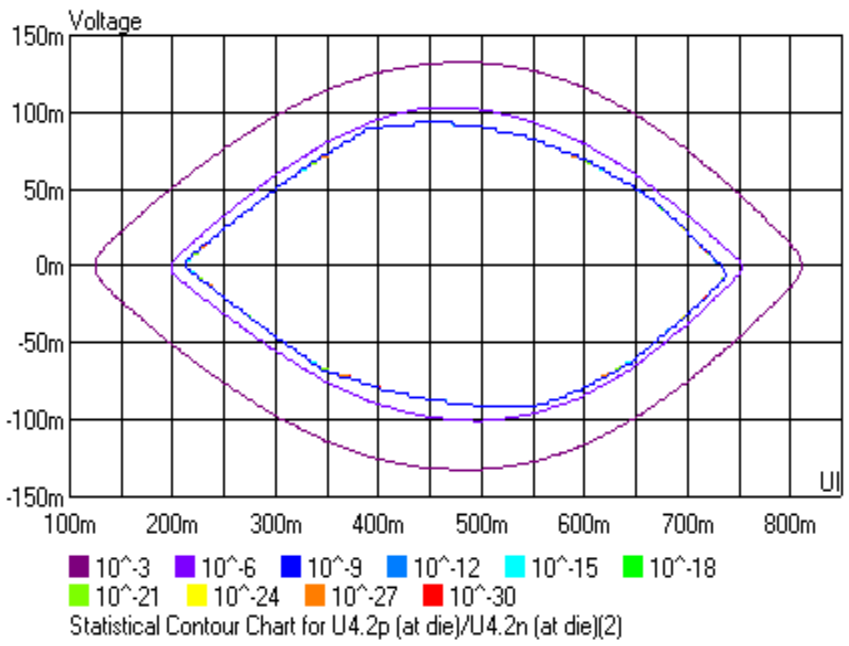
- Swing
  - — 0 … 63
- pretap
  - — 0 … 32
- posttap
  - — 0 … 63



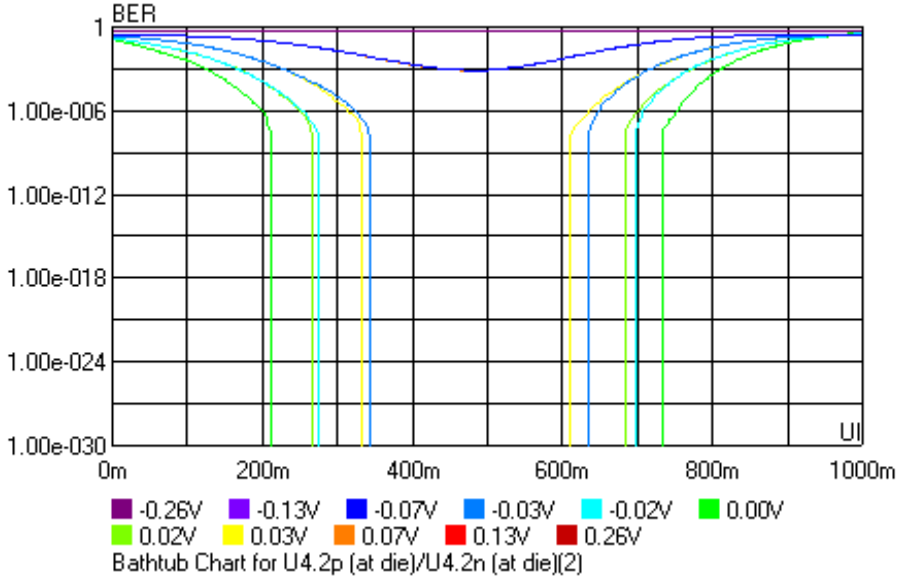
Public

IBIS-AMI first results

Eye diagram

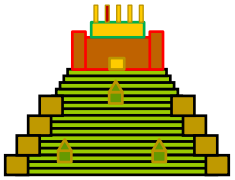


BER

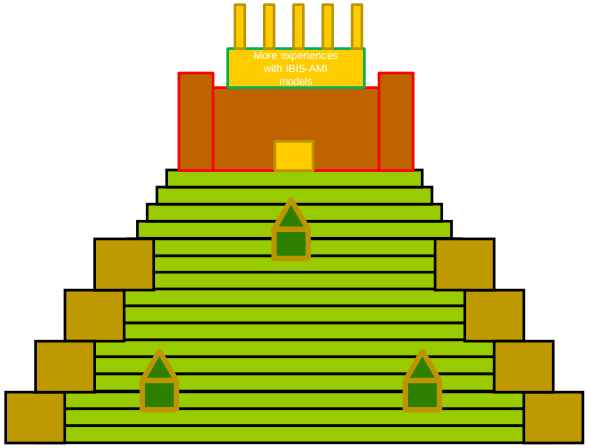


Nokia Siemens Networks

summary

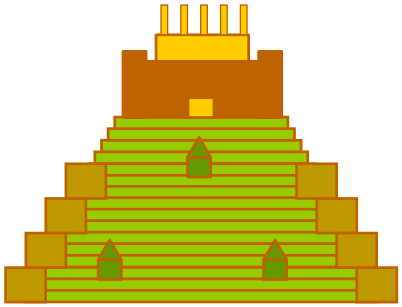


More experiences with IBIS-AMI models

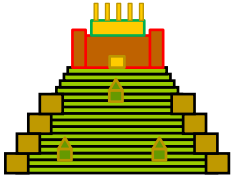


Why IBIS-AMI

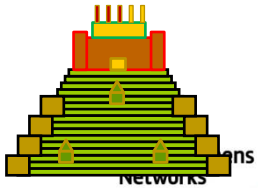
IBIS-AMI models



Basics IBIS-AMI



Differences IBIS vs IBIS-AMI



[Model] ALPHA

- Model_type Open_sink
- [Voltage Range] 0.03V 0.02V 0.04V
- C_comp 1.00p 0.95p 1.05p
- [Pullup]

- **[Algorithmic Model]**
- **Executable Windows_VisualStudio_32 NSNTx.dll NSNTx.ami**
- **Executable Linux_gcc3.2.3_32 NSNlibTx.so NSNTx.ami**
- **[End Algorithmic Model]**

Nokia Siemens Networks

[Model] BETA

- Model_type Output
- [Voltage Range] 1.8V 1.75V 1.85V
- C_comp 0.80p 0.65p 0.95p
- [Pullup]

- **[Algorithmic Model]**
- **Executable Windows_VisualStudio_32 NSNTx.dll NSNTx.ami**
- **Executable Linux_gcc3.2.3_32 NSNlibTx.so NSNTx.ami**
- **[End Algorithmic Model]**
- **Model needs mathematical program ,but with specific release**
- **Model runs on windows XP, but not on Win7**

[Model] GAMMA

- Model_type Open_sink
- [Voltage Range] 1.0V 0.92V 1.08V
- C_comp 0.50p 0.45p 0.55p
- [Pullup]

- **[Algorithmic Model]**
- **Executable Windows_VisualStudio_32 NSNTx.dll NSNTx.ami**
- **Executable Linux_gcc3.2.3_32 NSNlibTx.so NSNTx.ami**
- **[End Algorithmic Model]**

- **Request for package models**
- **Answer : package model is inside dll,**

- **According IBIS-AMI spec this is not valid**
- **Package is a part of the channel**

Nokia Siemens Networks

[Model] DELTA “not completed” .ami file

```
Model_type   Open_sink
[Voltage Range]     0.03V   0.02V   0.04V
C_comp 1.00p  0.95p  1.05p
[Pullup]

[Algorithmic Model]
Executable   Windows_VisualStudio_32   NSNTx.dll   NSNTx.ami
Executable   Linux_gcc3.2.3_32        NSNlibTx.so      NSNTx.ami
```

IBIS-Ami file model10G_file.ami

(Model_Specific

(DRIVE (Usage In) (Type Integer)

(**Format Value 0**) (Default 0) (Description “DRIVE"))

IBIS-AMI description file model10G-ami.pdf

(DRIVE)	TX Swing	TX Pre-emphasis
0	***200mv***	***3db***
10	***400mv***	***5db***
.....		
80	***800mv***	***8db***

Public

[Model] DELTA upgraded .ami file

```
Model_type   Open_sink
[Voltage Range]     0.03V   0.02V   0.04V
C_comp 1.00p  0.95p  1.05p
[Pullup]

[Algorithmic Model]
Executable   Windows_VisualStudio_32   NSNTx.dll   NSNTx.ami
Executable   Linux_gcc3.2.3_32        NSNlibTx.so      NSNTx.ami
```

IBIS-Ami file model10G_file.ami

(Model_Specific

(DRIVE (Usage In) (Type Integer)

(**Format List 0 10 …..80**) (Default 0) (Description “DRIVE"))

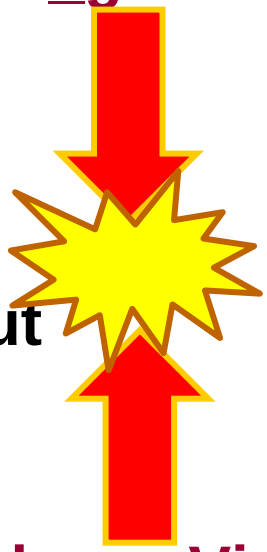
IBIS-AMI description file model10G-ami.pdf

(DRIVE)	TX Swing	TX Pre-emphasis
0	***200mv***	***3db***
10	***400mv***	***5db***
……		
80	***800mv***	***8db***

Nokia Siemens Networks

[Model] EPSILON & [Model] OMEGA

- **Model_type Open_sink**
  - [Algorithmic Model]
  - Executable Linux_gcc3.2.3_32 NSNlibTx.so NSNTx.ami
  - [End Algorithmic Model]
- **Model_type Input**
  - [Algorithmic Model]
  - Executable Windows_VisualStudio_32 NSNlibRx.dll NSNRx.ami
  - [End Algorithmic Model]



Nokia Siemens Networks

[Model] ZETA

- Model_type Open_sink
- [Voltage Range] 0.03V 0.02V 0.04V
- C_comp 1.00p 0.95p 1.05p

- [Model Selector]
  - **Swing10**
  - **…….**
  - **Swing630**

ibis file
Nsn.ibs

ibis ami file
NsnTx.ami

- **[Algorithmic Model]**
- **Executable Windows_VisualStudio_32 NSNTx.dll NSNTx.ami**

(TX_swing **(Range 310 10 630)** (Usage In)(Type Integer)(Default 310)

(Description "Transmitter swing"))

- **Executable Linux_gcc3.2.3_32 NSNlibTx.so NSNTx.ami**
- **[End Algorithmic Model]**

Nokia Siemens Networks

[Model] ETA

- Executable Windows_VisualStudio_32 NSNTx.dll NSNTx.ami

```
(Reserved_Parameters
(Rx_Clock_PDF (Usage Info) (Type Float) (Format Table
        (Labels Row_No Time_UI Density)
        (-3 -1.2                0.1)
        ( 3  1.17E-001          0.01)
        ) | End Table
    ) | End Rx_Clock_PDF
  ) | End Reserved_Parameters
```

- IBISCHK5 results in 2 errors :
- E4630 - Parameter Format is mandatory for Rx_Clock_PDF
- E4630 - Parameter Description is mandatory for Rx_Clock_PDF

[Model] ETA

Executable Windows_VisualStudio_32 NSNTx.dll NSNTx.ami

- (Reserved_Parameters

(Rx_Clock_PDF (Usage Info) (Type Float) **(Format Table**

(Labels Row_No Time_UI Density)
(-30 -1.17E-001 0)
(30 1.17E-001 0)
) | End Table

(Description "Receiver clock pdf is in table format.")

) | End Rx_Clock_PDF

) | End Reserved_Parameters

Bug in ibis parser

Error in ami file

- **E4630 - Parameter Format is mandatory for Rx_Clock_PDF**
- E4630 - Parameter Description is mandatory for Rx_Clock_PDF

[Model] THETA no documentation supported

Model_type Open_sink

IBIS-Ami file model8G_file.ami

(Model_Specific

(EQUAL (Usage In) (Type Integer) (Format List 0 ….. 8) (Default 0)

Labels " 0 001 0.0db "

…….

" 8 110 9.5db : max ")

(Description "EQUAL post tap settings ")

IBIS-AMI description file model8G-ami.pdf

(EQUAL) post

0 2db 001

……..

8 8db 110

Data sheet model8G.txt

Equalization post-tap :

0001 no equalization

……..

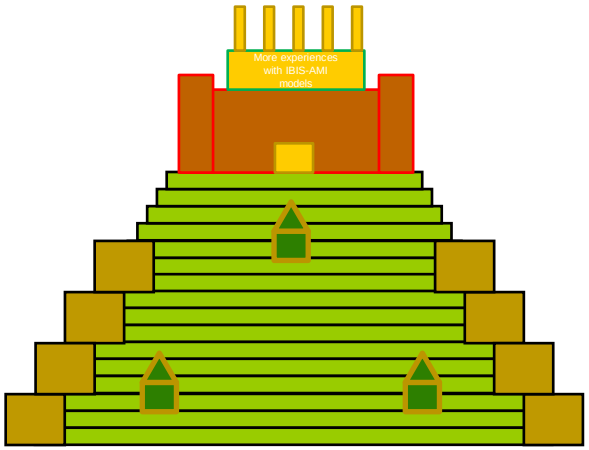
0110 75%

Public

Nokia Siemens Networks

summary

More experiences with IBIS-AMI models



Why IBIS-AMI

IBIS-AMI models

Basics IBIS-AMI

Differences IBIS vs IBIS-AMI

Summary

- For > 5Gbps , IBIS,SPICE is too slow
- IBIS-AMI is a good solution

  for BER < 1e-12
- Increasing model support from
  - IC-vendors
  - Tool vendors
- IBIS forum is working on enhancements
- IBIS-AMI is challenging

Nokia Siemens Networks

Thank you

Questions ?