

# Characterizing and Modeling of a Clamped Non-Linear CTE/AGC

***Skipper Liang***

Asian IBIS Summit

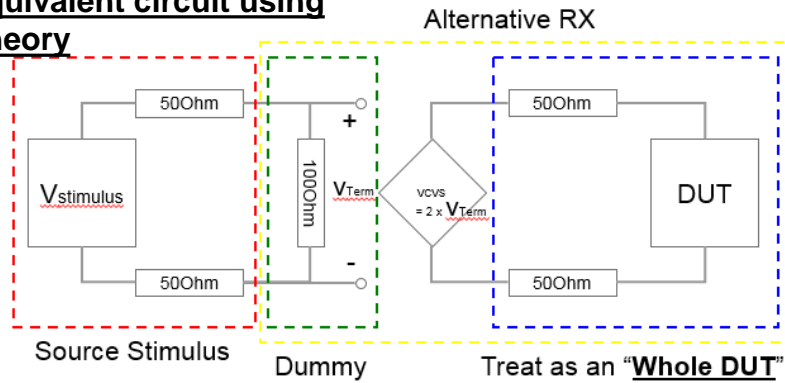
Shanghai, PRC

November 14, 2018

# Characterize a Linear CTE or CTE+AGC

In 2017 IBIS Summit, we deliver an easy but accurate methodology of characterizing a linear CTE (or even CTE+AGC, as long as the linearity is met.)

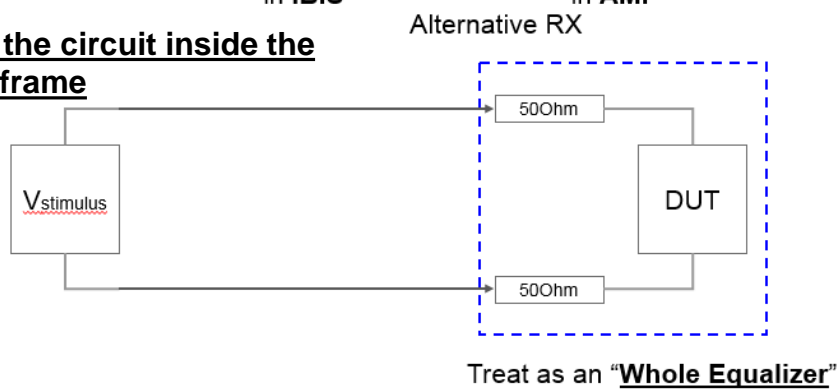
## 1. Derive an equivalent circuit using Thevenin Theory



Modeling in IBIS

Modeling in AMI

## 2. Characterize the circuit inside the blue dashed frame



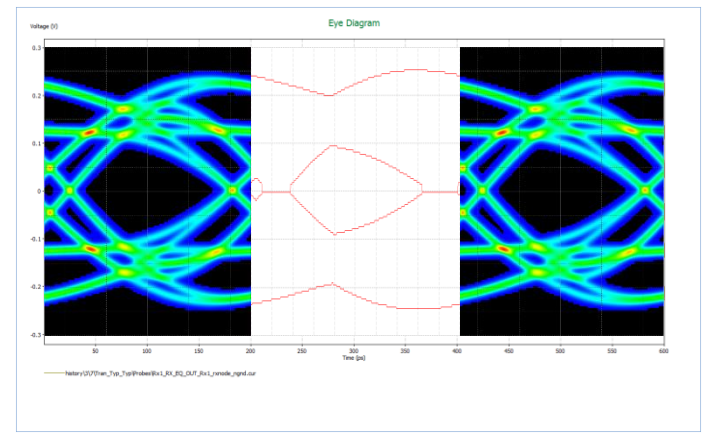
Modeling in AMI



SkippierLiang  
Asian IBIS Summit  
Shanghai, PRC  
November 13, 2017

cadence

## 3. Correlate the result of AMI in Channel analysis with the one of netlist in Transient analysis.



cadence

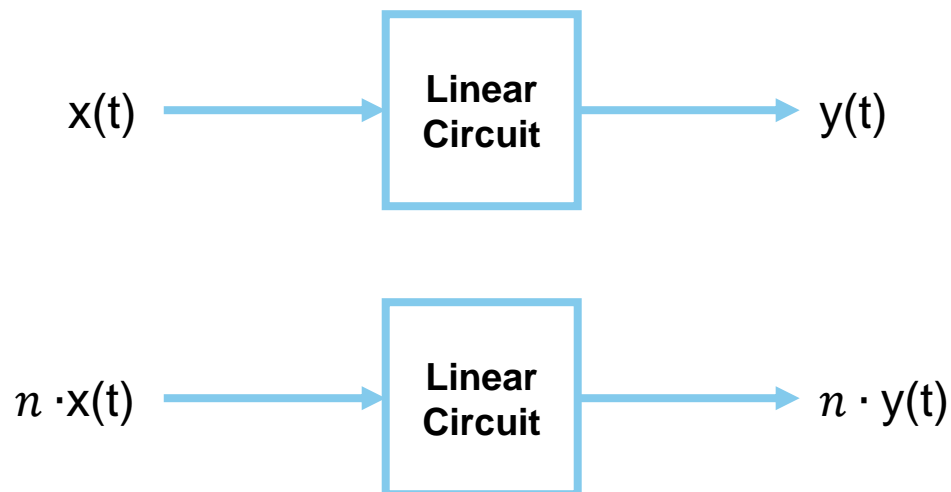
# Scale of Characterization Stimulus

Scale of Characterization Stimulus is important as it will be a normalization factor in an AMI model.

```
( RX_HyperCore
  ( CTE
    ( csvfilt
      ( file D:\Case_CDNS_20160616_IBIS_AMI\Modelize_RX\CTLE_TRAN_0.5
        ( sel 0
          ( input 0.57 )
        )
      )
    )
    ( module_off 0
      ( csvid_file csvid.txt
        ( td_filter_out cte_td_out.txt
          ( adapt_cte_sel_file cte_out.txt
        )
      )
    )
  )
)
```

```
24 .subckt TX_50OUT pos neg pwr in ngnd
25
26 E1 pos ngnd volt='0.47+0.57*v(in,ngnd)'
27 E2 neg ngnd volt='1.04-0.57*v(in,ngnd)'
28
29 .ends
```

However, for a “Linear” RX EQ, we didn’t pay too much focus on how much the scale of characterization stimulus should be because for a “Linear” RX EQ, it should meet the following:

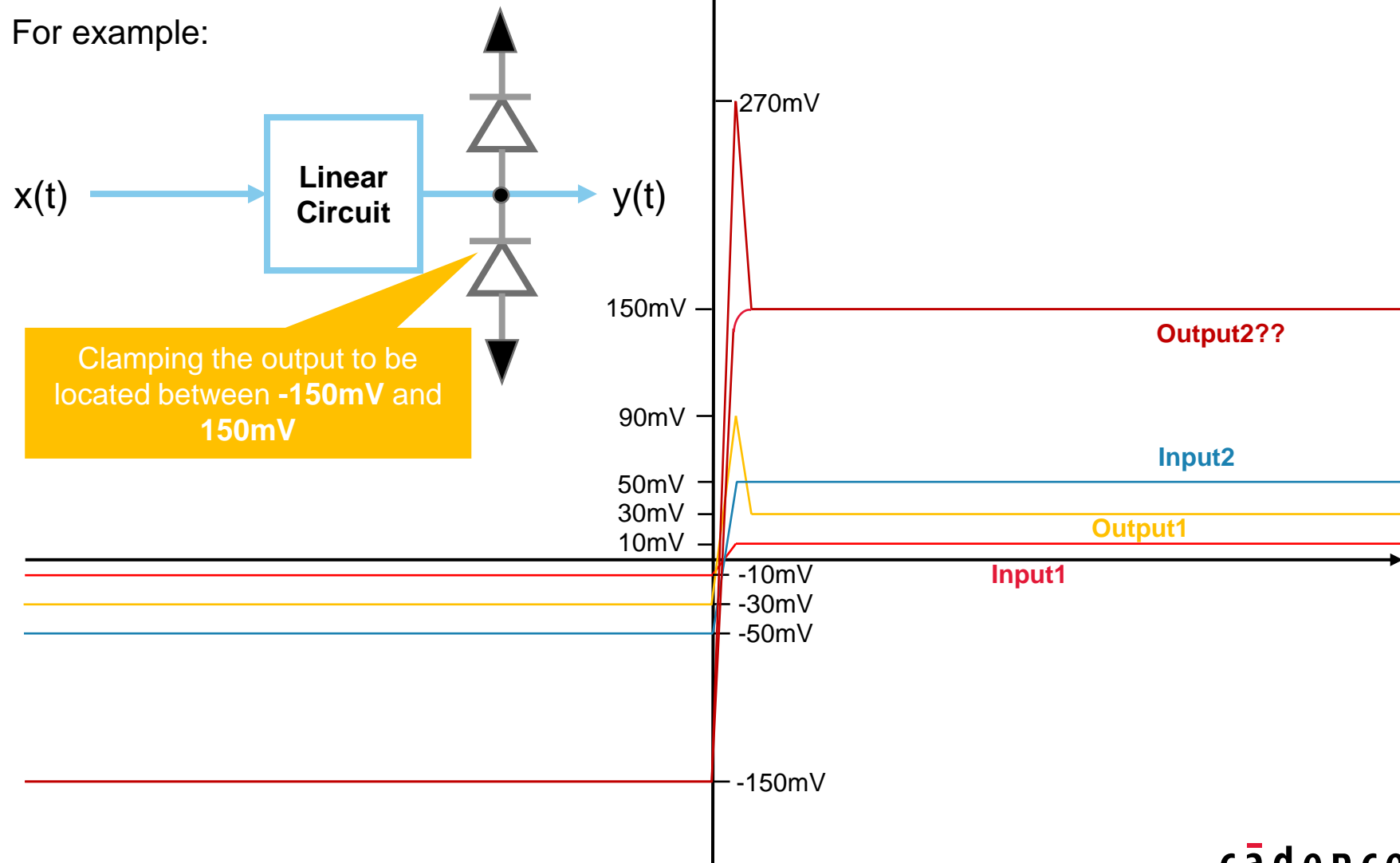


It means no matter how much the scale of characterization stimulus is, as long as the normalization factor is correct, the model will work close to the real circuit.

# Characterize of Non-linear CTE/AGC

But most circuits are not linear as there are clamping diodes to protect the circuit.

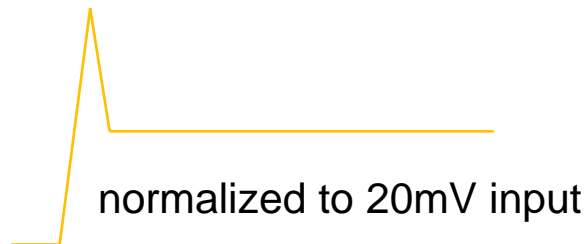
For example:



# Characterize of Non-linear CTE/AGC

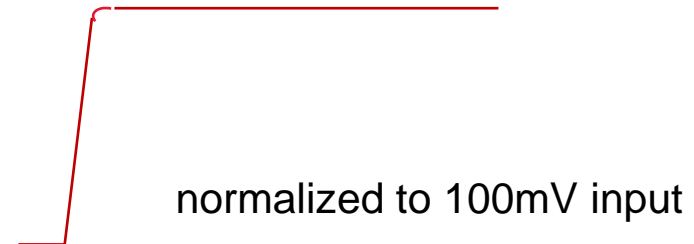
How to characterize such a non-linear circuit?

## Approach 1.

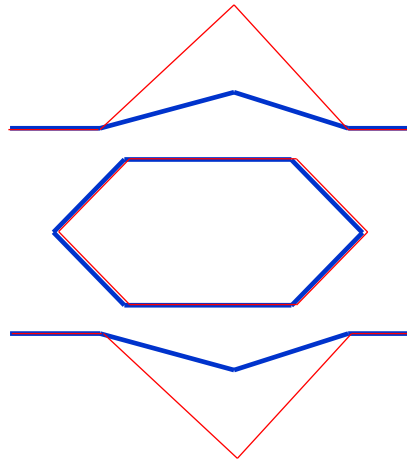


or

## Approach 2.



If we characterize such a non-linear circuit with Approach 1.



**Blue: Generated by SPICE netlist under Transient Analysis**

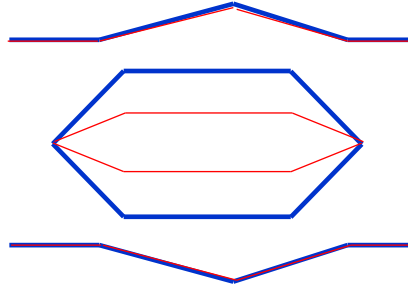
**Red: Generated by AMI model under Channel Analysis**

**“Small Signal”** to characterize the CTLE:

1. You can capture the HF response of the CTLE
2. But you will miss the DC behavior of the stable logic high and low

# Characterize of Non-linear CTE/AGC

If we characterize such a non-linear circuit with Approach 2.



Blue: Generated by SPICE netlist under Transient Analysis

Red: Generated by AMI model under Channel Analysis

“Large Signal” to characterize the CTLE:

1. You can capture the correct DC behavior of the stable logic high and low of the CTLE
2. But you will miss the HF response

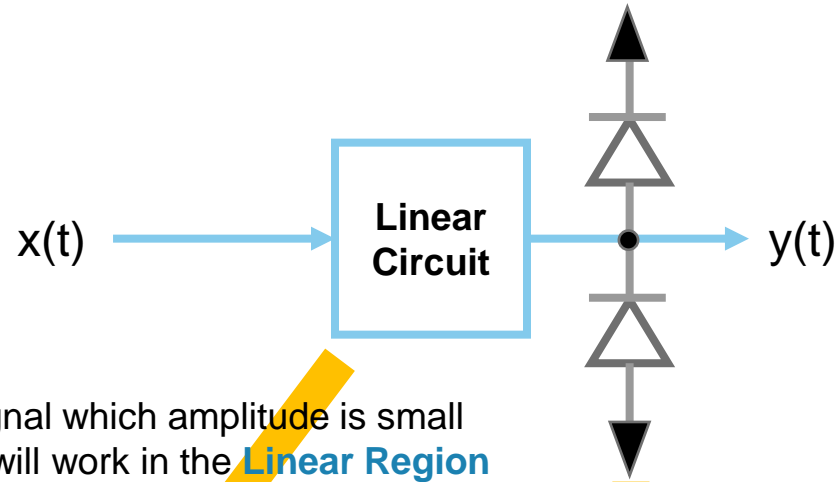
For Approach 1, since the characterization can successfully capture the circuit’s response at High Frequency range but miss the DC behavior, the model’s simulation result can be well-correlated with SPICE transient analysis’ result while a **Lossy Channel** is applied, which decays much more at High Frequency range.

For Approach 2, since the characterization can successfully capture the circuit’s DC behavior but miss the response at High Frequency range, the model’s simulation result can be well-correlated with SPICE transient analysis’ result while a **Lossless Channel** is applied, which decays much less at High Frequency range.

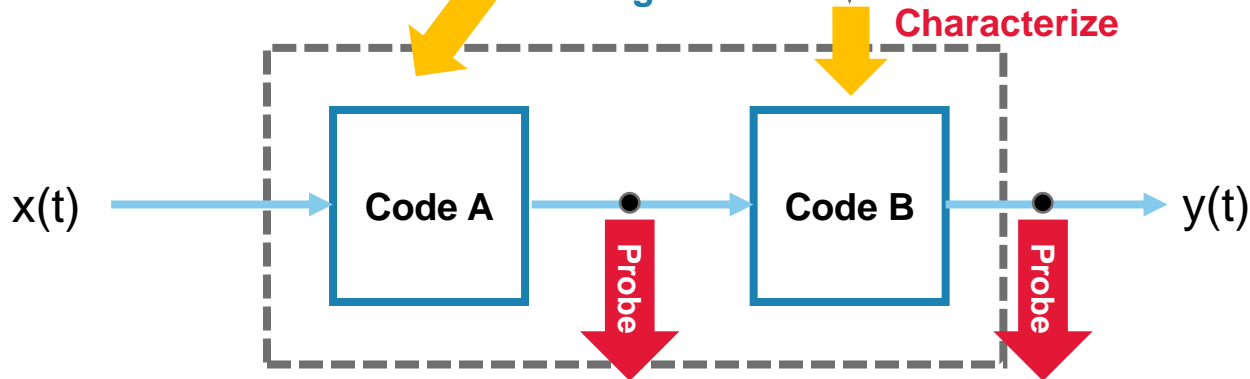
How to have a model which can accommodate all kinds of channels?

# Characterize of Non-linear CTE/AGC

The answer is:

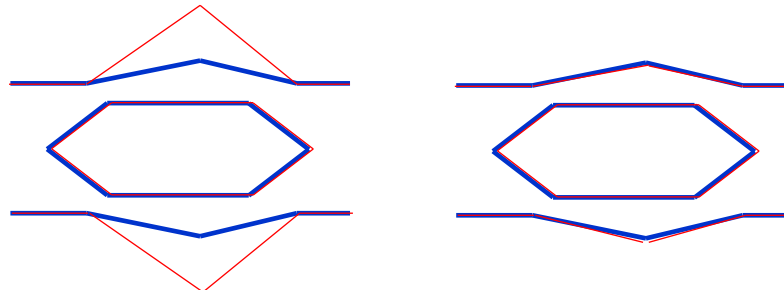


**Characterize** with a signal which amplitude is small enough that the circuit will work in the **Linear Region**

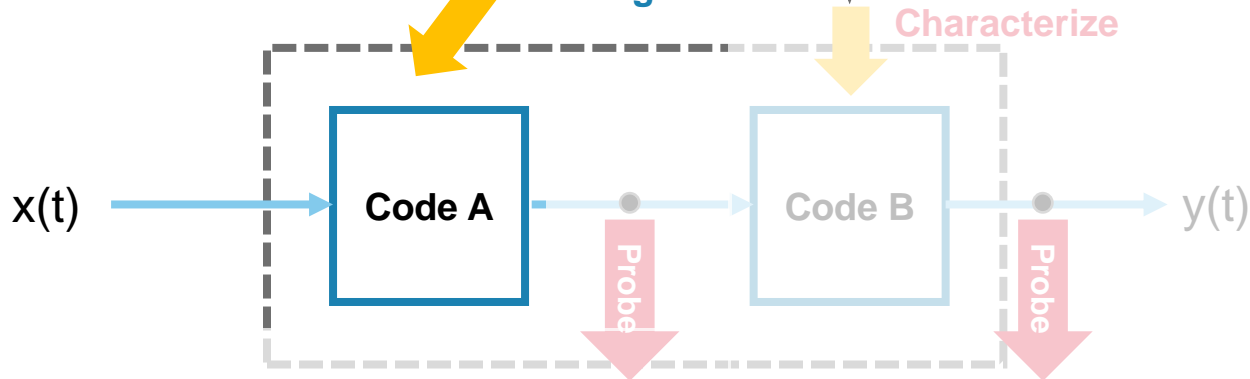
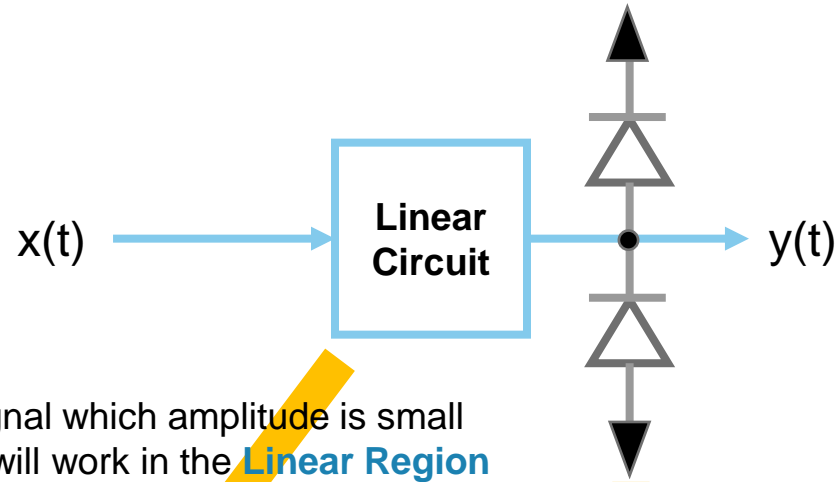


**Blue:** Generated by SPICE netlist under Transient Analysis

**Red:** Generated by AMI model under Channel Analysis

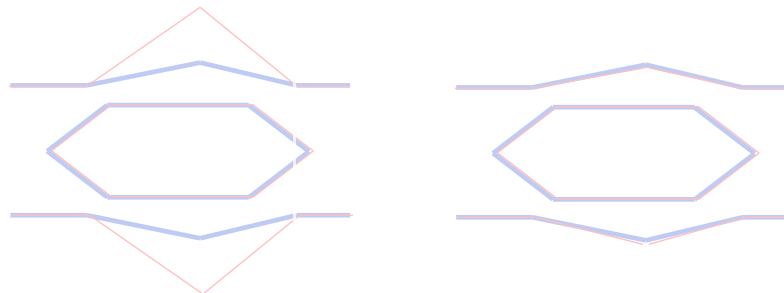


# Characterize the Linear Part - Linear Region



Blue: Generated by SPICE netlist under Transient Analysis

Red: Generated by AMI model under Channel Analysis





# Linear Region

How to know if I'm characterizing in the Linear Region?

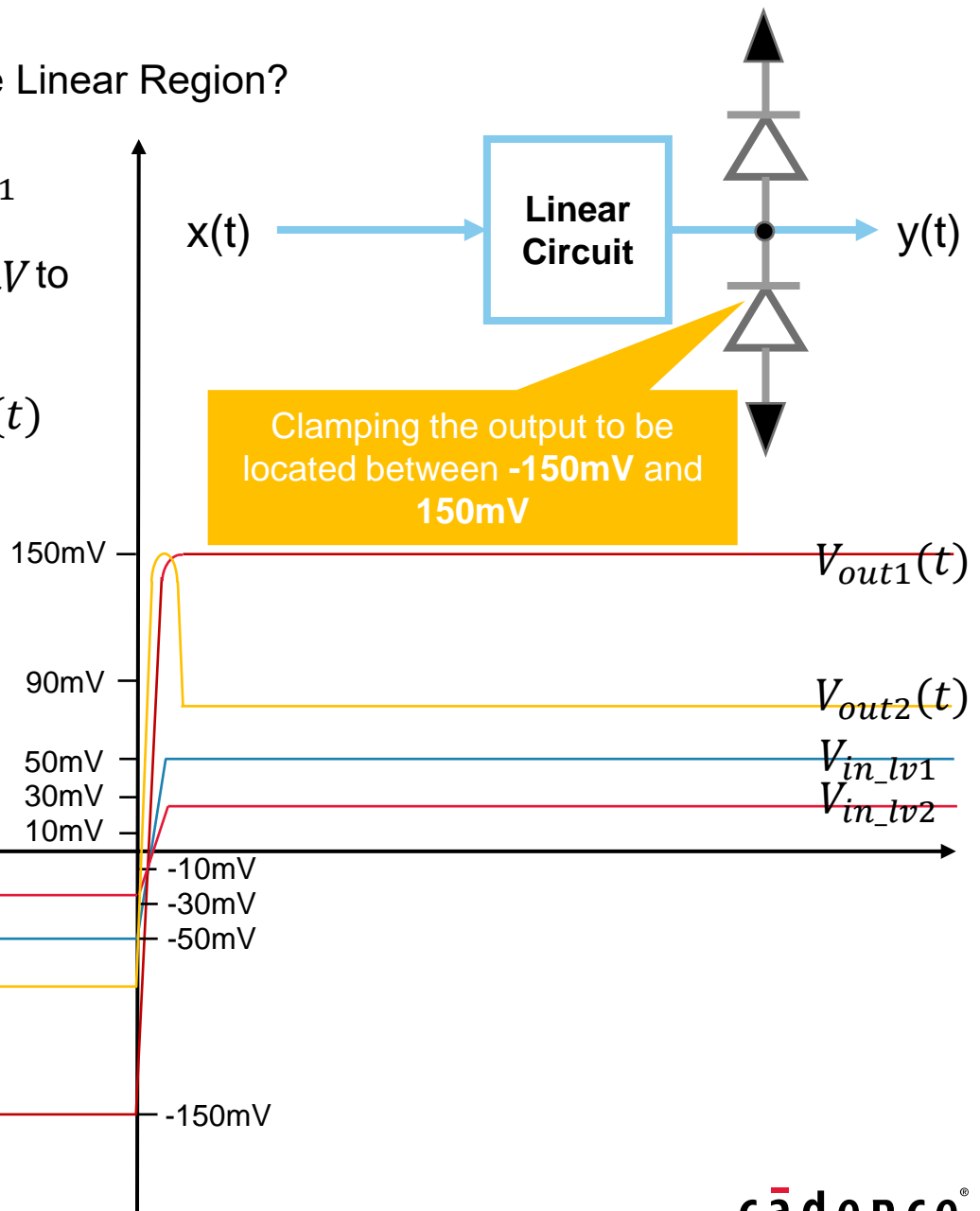
1. Choose a input voltage level  $V_{in\_lv1}$

For example: Apply  $V_{in\_lv1} = 50mV$  to the circuit at the right.

2. Record the circuit response.  $V_{out1}(t)$
3. Choose another input voltage level

$$V_{in\_lv2} = 0.5 \times V_{in\_lv1}$$

For example:  
Apply  $V_{in\_lv2} = 25mV$  to the same circuit.

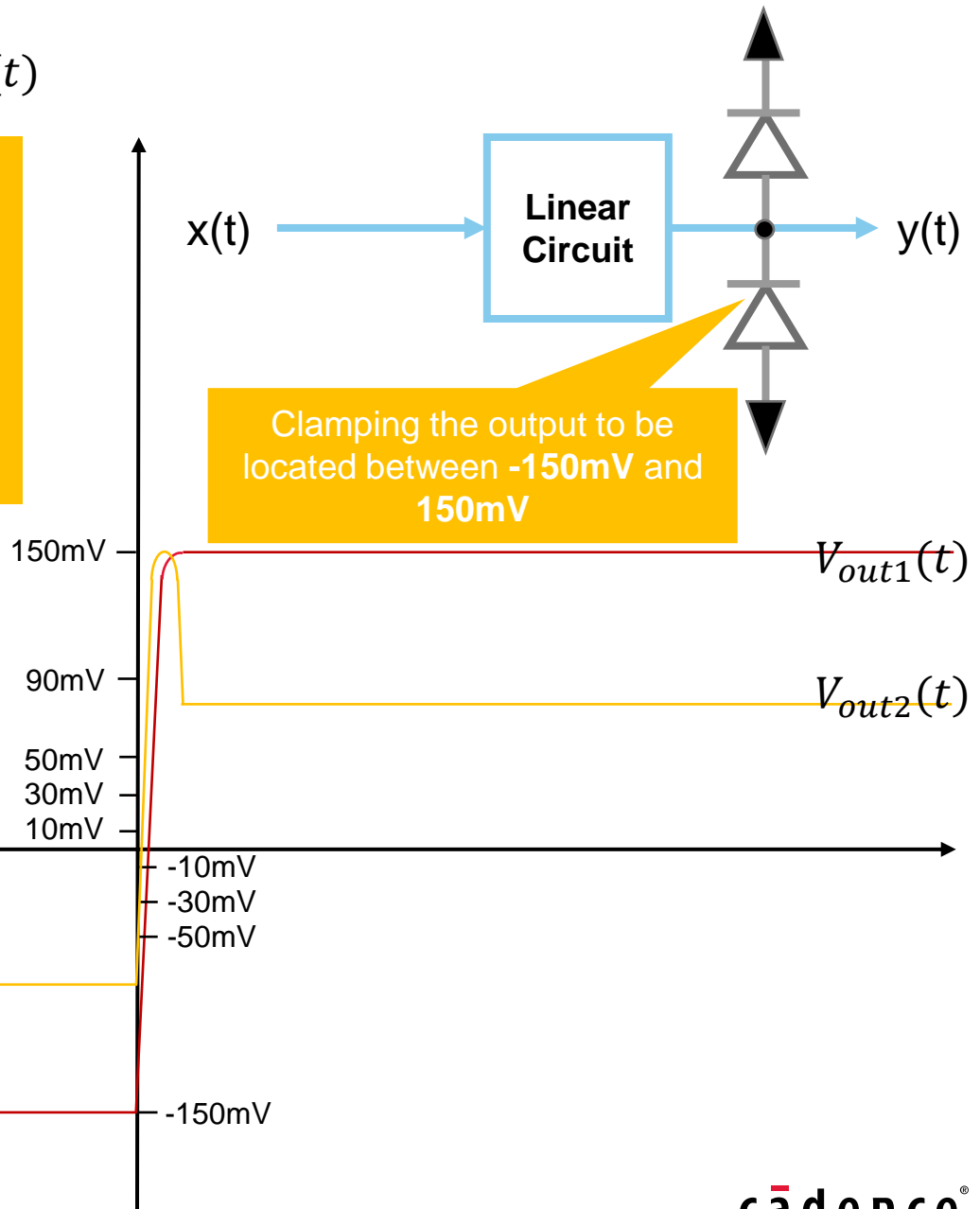


# Linear Region

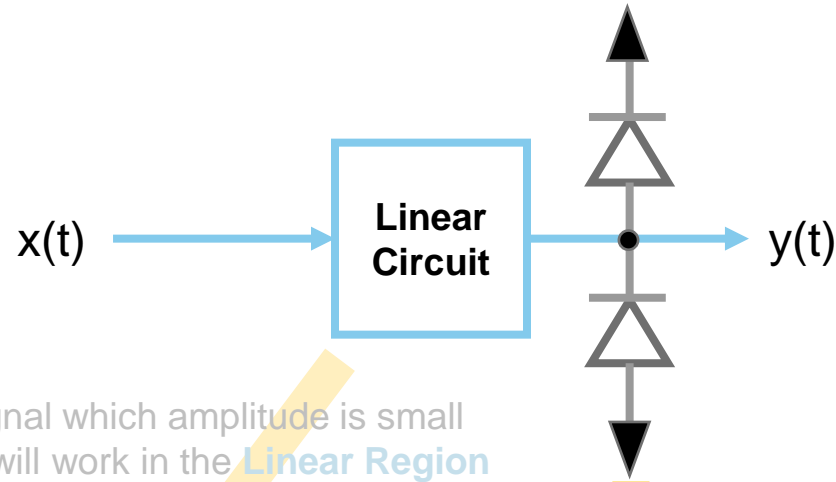
4. Record the circuit response.  $V_{out2}(t)$

If  $V_{out2}(t) = 0.5 \times V_{out1}(t)$  along the time axis, we can use  $V_{in\_lv1}$  to characterize this circuit, that is, we can take  $V_{out1}(t)$  as the circuit's response.

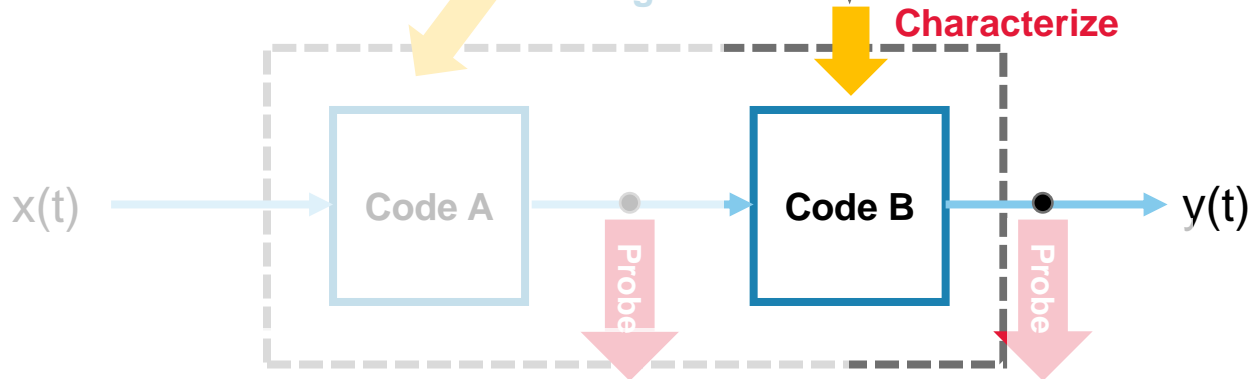
In this example, according to the rule detailed above, we **can't** use a  $\pm 50mV$  step function to characterize the circuit.



# Characterize the Non-Linear Part

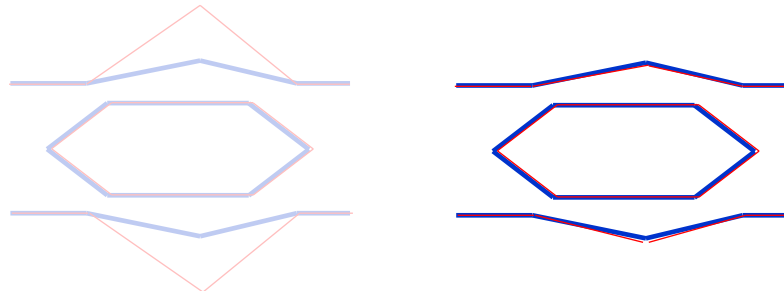


Characterize with a signal which amplitude is small enough that the circuit will work in the **Linear Region**



Blue: Generated by SPICE netlist under Transient Analysis

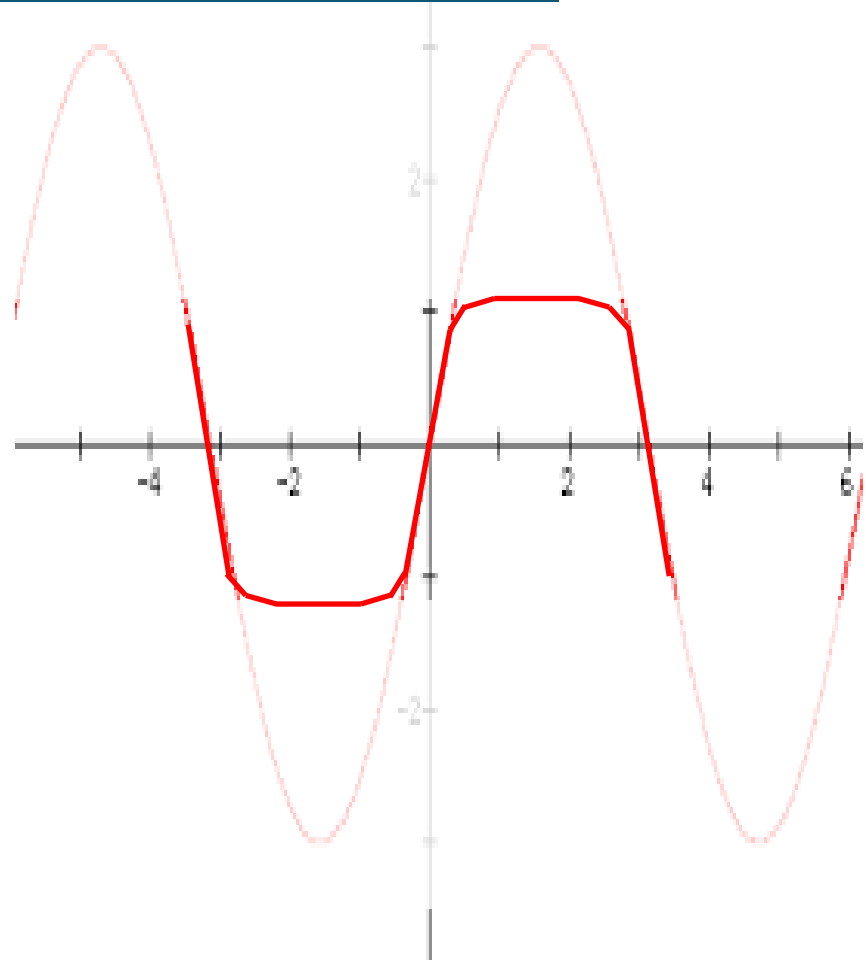
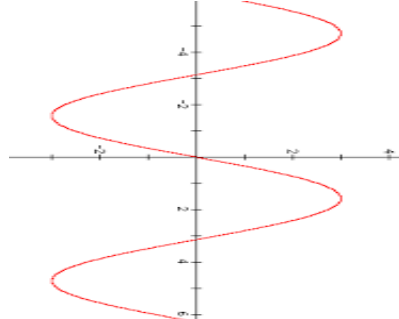
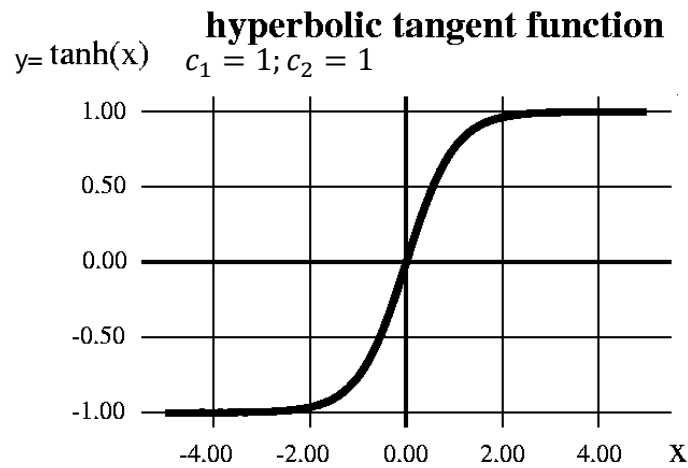
Red: Generated by AMI model under Channel Analysis



# Hyperbolic Tangent Function

We use a hyperbolic tangent function to model the clamping behavior

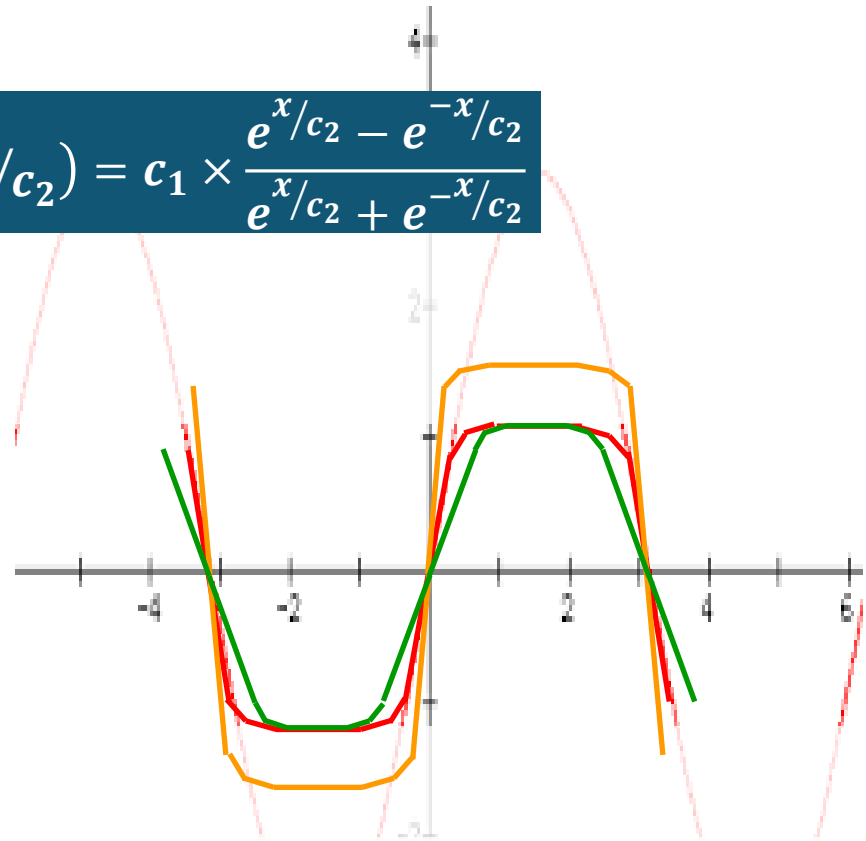
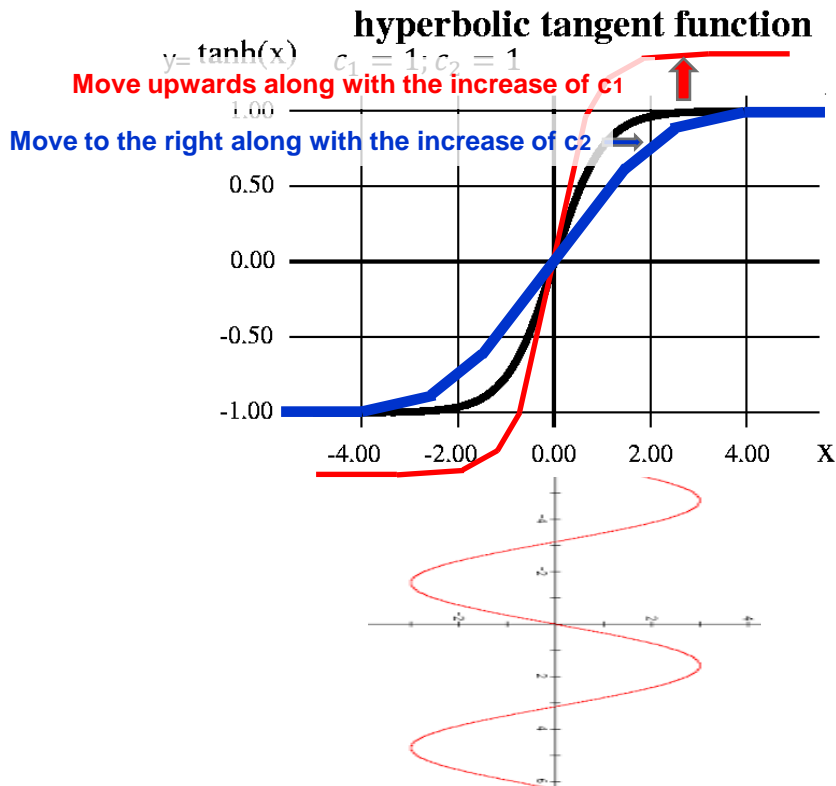
$$y = c_1 \times \tanh\left(\frac{x}{c_2}\right) = c_1 \times \frac{e^{x/c_2} - e^{-x/c_2}}{e^{x/c_2} + e^{-x/c_2}}$$



# Hyperbolic Tangent Function

By adjusting parameters C1 and C2, we can customize the Hyperbolic Tangent function as close to the circuit's behavior as we want.

$$y = c_1 \times \tanh\left(\frac{x}{c_2}\right) = c_1 \times \frac{e^{x/c_2} - e^{-x/c_2}}{e^{x/c_2} + e^{-x/c_2}}$$

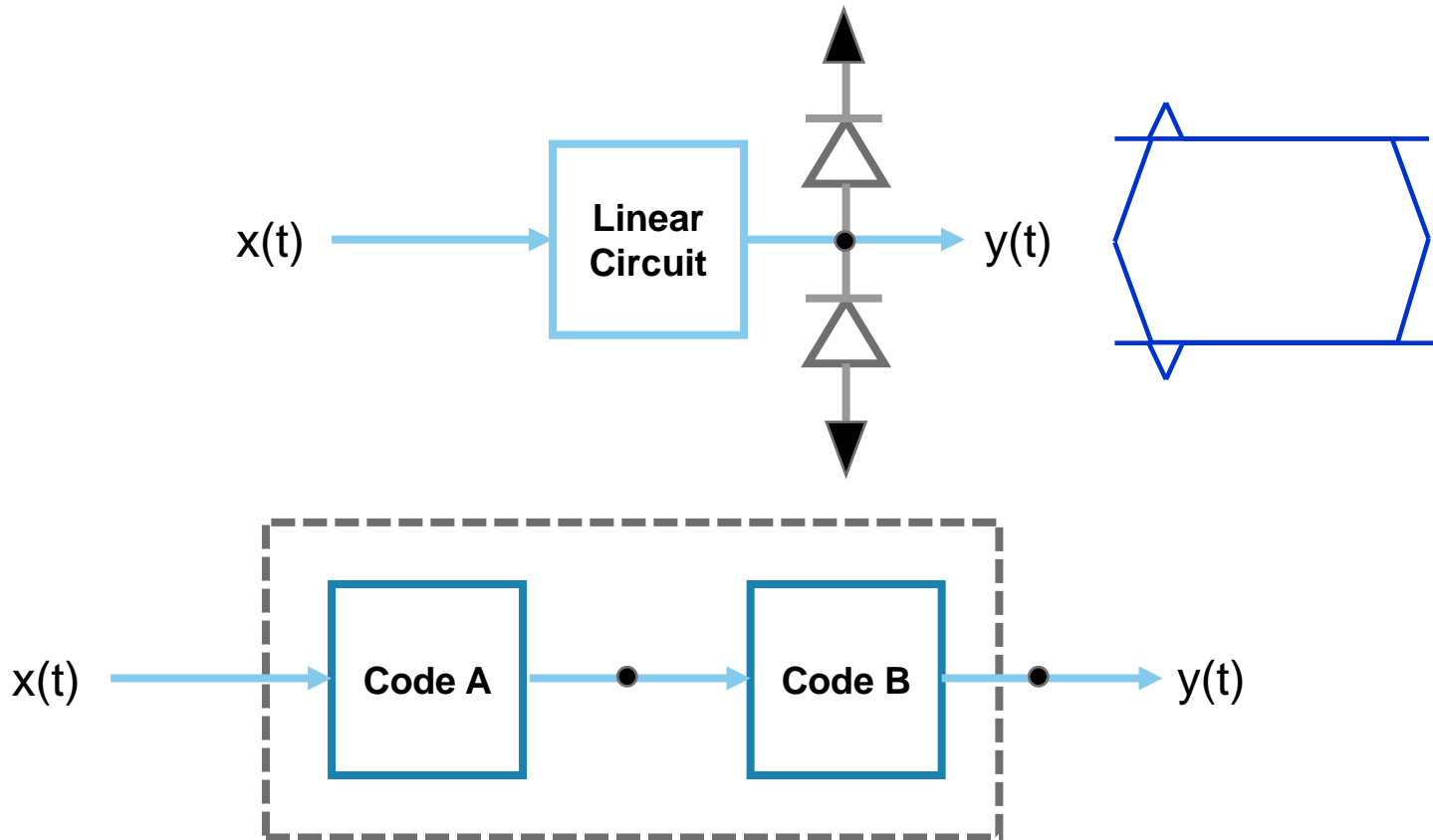


Increase of  $c_1$ : Rising/Falling slew rate increase, upper/lower increase

Increase of  $c_2$ : Rising/Falling slew rate decrease, upper/lower remains the same

The ratio  $C1/C2$  represent the slope of the linear region of the hyperbolic tangent function and could be deemed as the amplification scale.

# Characterization of $y=C1*\tanh(x/C2)$ – C1 and C2

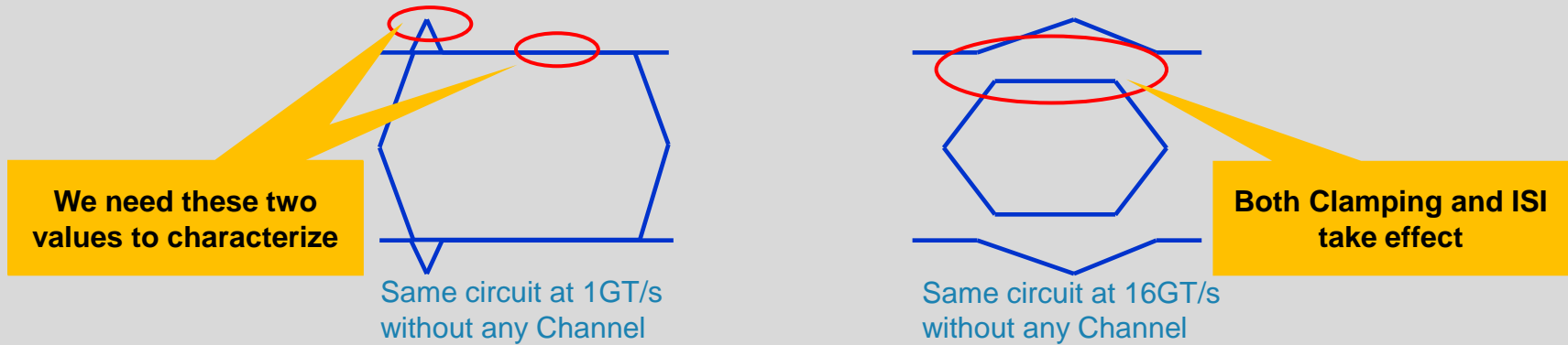


Step 1: Transient analysis over the transistor netlist.

Note: Set the amplitude of  $x(t)$  to be the regular input voltage level of the RX circuit. Don't use small signals.

Set the bit rate slow enough that almost no ISI will happen, no matter how much bit rate the RX circuit will be applied to in practical usage. For example: 1GT/s

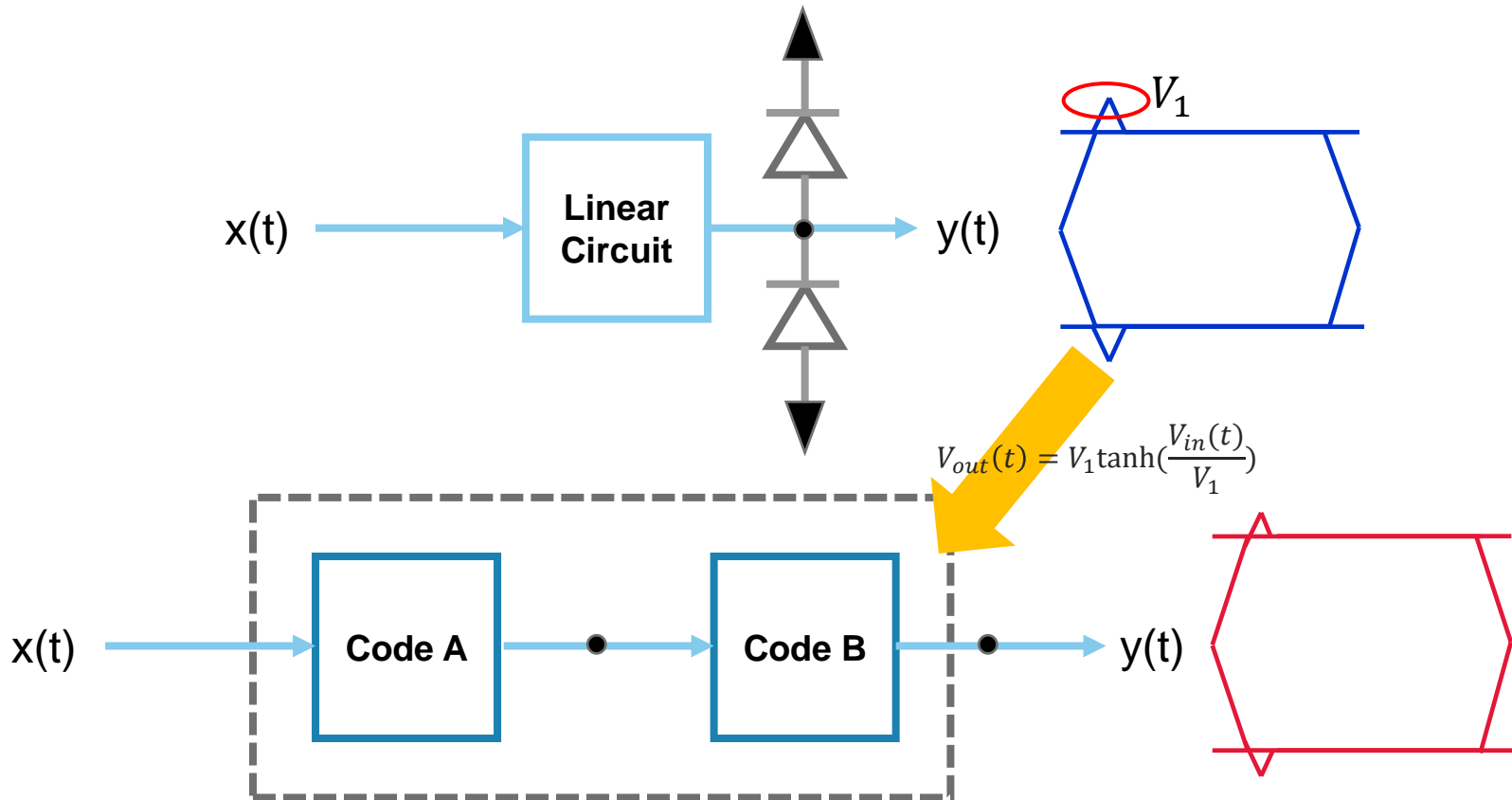
# Characterization of $y=C1*\tanh(x/C2)$ – C1 and C2



The reason to set the bit rate slow enough, no matter how much bit rate the RX circuit will be applied is:

1. We need the outer and inner contour of the eye which can tell us how much the clamping takes effect and only the clamping takes effect.
2. An eye folded from a slow transition waveform can guarantee the amplitude of the outer and inner contour of the eye is only affected by the clamping but **free from ISI**.
3. Even without any Channel applied, it's impossible to get rid of ISI effect once the circuit is operated under a fast transmission rate

# Characterization of $y=C1*\tanh(x/C2) - C1$ and $C2$



Step 2: Start from the value of the outer contour of the eye generated by folding the waveform of transient analysis over the transistor netlist -> Take the value  $V_1$  to replace the  $C_1$  and  $C_2$  in the hyperbolic tangent function, that is,

$$V_{out}(t) = V_1 \tanh\left(\frac{V_{in}(t)}{V_1}\right)$$



# Characterization of $y=C1*\tanh(x/C2)$ – C1 and C2

Step 3: Overlap the two eye diagrams or record the value of the inner contours of these two eye diagrams.

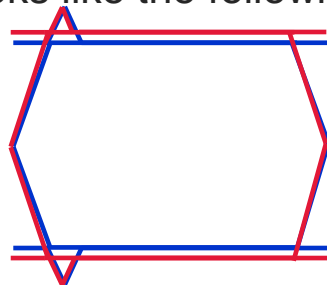
If the eye diagram of **the model under channel analysis** has **higher inner** contour than the eye diagram of **the transistor netlist under transient analysis** -> **Increase  $C_2$**  with **increment = 0.25** until the two eyes' inner contours meet each other.

$$V_{out}(t) = V_1 \tanh\left(\frac{V_{in}(t)}{V_1}\right) \uparrow$$

If the eye diagram of **the model under channel analysis** has **lower inner** contour than the eye diagram of **the transistor netlist under transient analysis** -> **Decrease  $C_2$**  with **increment = 0.25** until the two eyes' inner contours meet each other.

$$V_{out}(t) = V_1 \tanh\left(\frac{V_{in}(t)}{V_1}\right) \downarrow$$

For example, if the comparison looks like the following, you should **Increase  $C_2$**

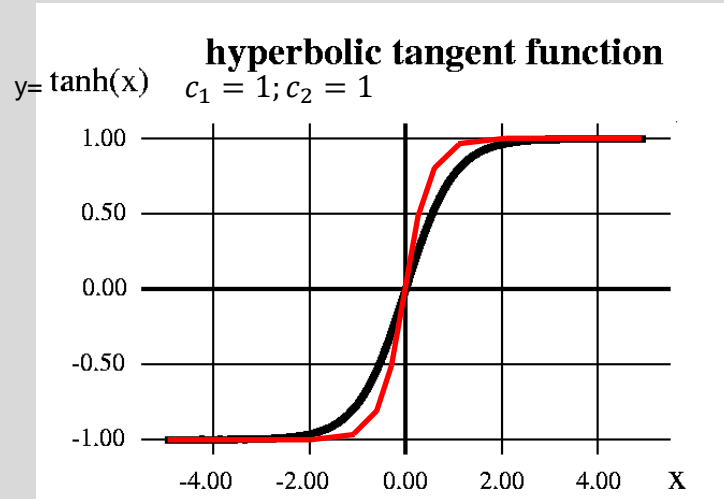


**Blue:** Generated by SPICE netlist under Transient Analysis

**Red:** Generated by AMI model under Channel Analysis

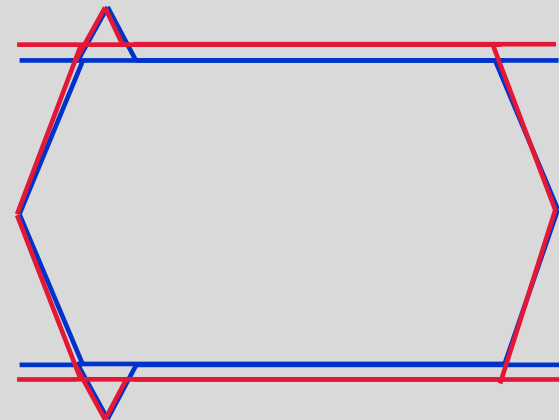
# Characterization of $y=C1*\tanh(x/C2) - C1$ and $C2$

The reason to cause “the model under channel analysis has higher inner contour than the eye diagram of the transistor netlist under transient analysis” is:



**Black:** The characteristics of the transistor netlist


**Red:** The characteristics of the model while applying  $V_1$  to be  $C_1$  and  $C_2$  of the hyperbolic tangent function




# Characterization of $y=C1*\tanh(x/C2)$ – C1 and C2

Step 4: Overlap the two eye diagram or record the value of the outer contour of these two eye diagram.

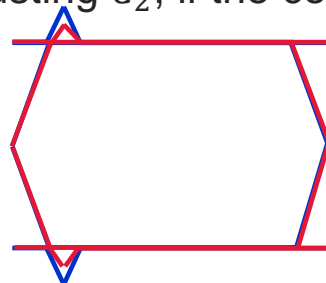
If the eye diagram of **the model under channel analysis** has **higher outer** contour than the eye diagram of **the transistor netlist under transient analysis** -> **Decrease  $C_1$**  with **increment = 0.25** until the two eyes' **outer** contour meet each other.

$$V_{out}(t) = V_1 \tanh\left(\frac{V_{in}(t)}{V_2}\right)$$


If the eye diagram of **the model under channel analysis** has **lower outer** contour than the eye diagram of **the transistor netlist under transient analysis** -> **Increase  $C_1$**  with **increment = 0.25** until the two eyes' **outer** contour meet each other.

$$V_{out}(t) = V_1 \tanh\left(\frac{V_{in}(t)}{V_2}\right)$$


In our previous example, after adjusting  $C_2$ , if the comparison looks like the following, you should **Increase  $C_1$**

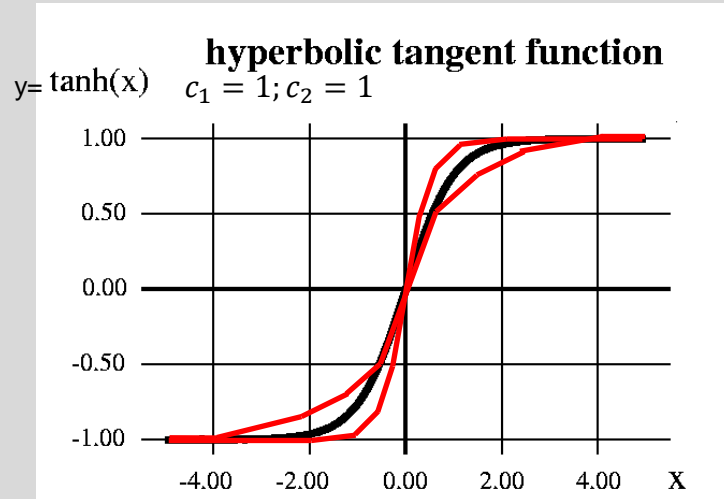


**Blue:** Generated by SPICE netlist under Transient Analysis

**Red:** Generated by AMI model under Channel Analysis

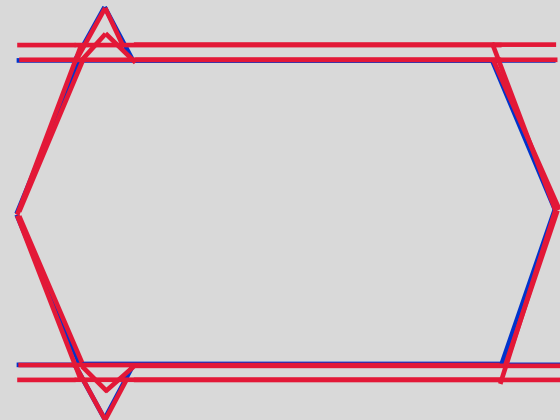
# Characterization of $y=C1*\tanh(x/C2) - C1$ and $C2$

The reason to cause “the model under channel analysis has **Lower outer** contour than the eye diagram of **the transistor netlist under transient analysis**” **after adjusting  $C_2$**  is:



**Black:** The characteristics of the transistor netlist

**Red:** The characteristics of the model after adjusting  $C_2$  to make the inner contour of the model meet with the inner contour of the transistor netlist

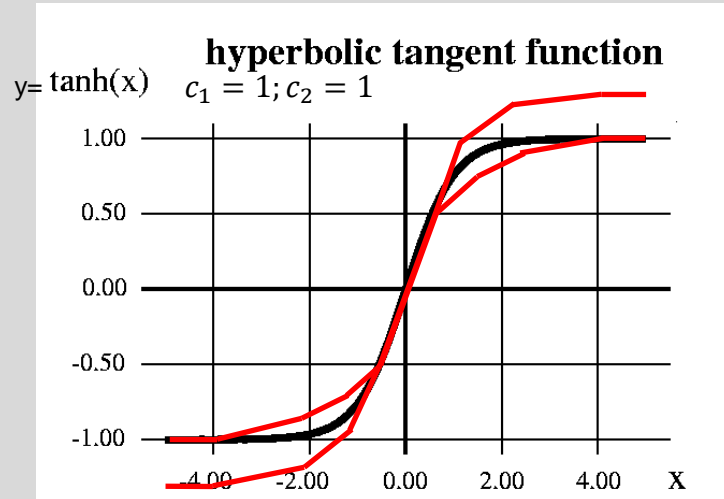


**Blue:** Generated by SPICE netlist under Transient Analysis

**Red:** Generated by AMI model under Channel Analysis

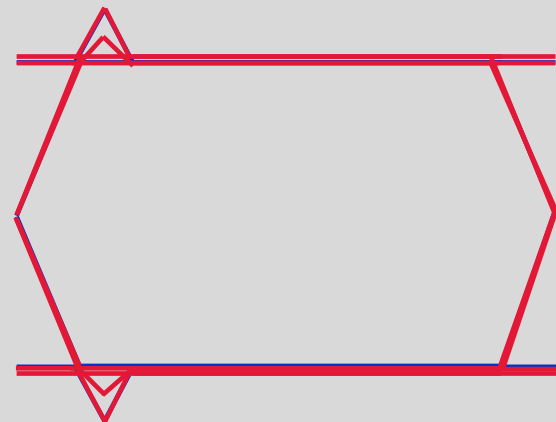
# Characterization of $y=C1*\tanh(x/C2) - C1$ and $C2$

And **after adjusting  $C_1$**  to make the outer contour of the model meet with the outer contour of the transistor netlist:



**Black:** The characteristics of the transistor netlist

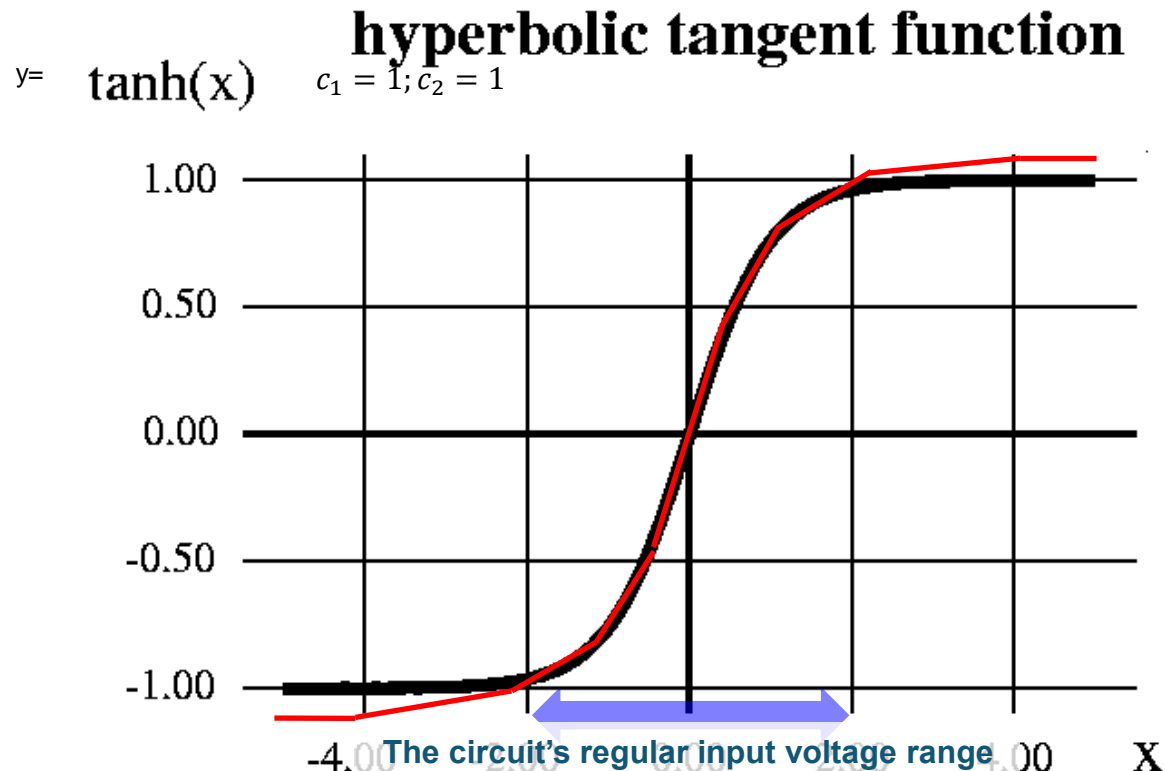
**Red:** The characteristics of the model after adjusting  $C_1$  to make the outer contour of the model meet with the outer contour of the transistor netlist



# Characterization of $y=C1*\tanh(x/C2) - C1$ – C1 and C2

Step 5: Go back to Step 3 & Step 4 and keep iterating until you reach a satisfied result.

At the end when you reach a satisfied result, it doesn't imply that we have a hyperbolic tangent function which perfectly overlaps with the circuit's characteristics but means that in the circuit's regular input voltage range, we have a hyperbolic tangent function which gets as close to the circuit's characteristics as we wish.



# AMI\_GetWave()

- Does a Hyperbolic Tangent Function have a corresponding frequency response?

Ans: Almost impossible because one of the criteria for a function to be Fourier transformable is

$$\int_{-\infty}^{\infty} |f(x)| dx < \infty$$

(Absolutely Integrable)

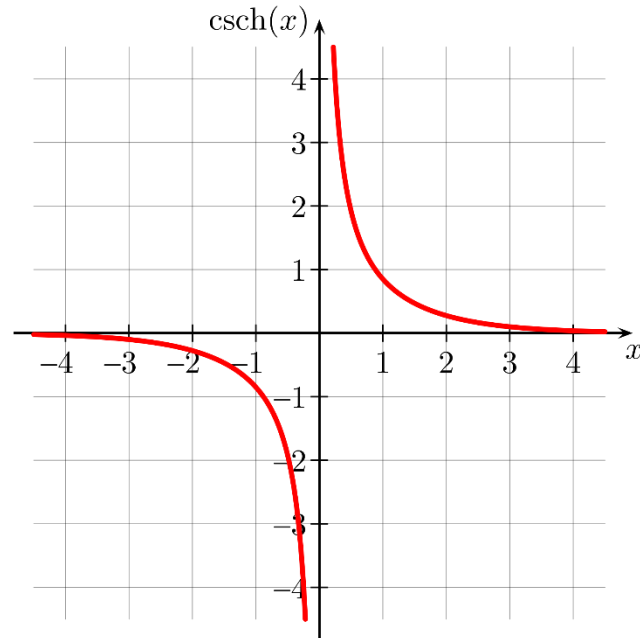
And obviously, Hyperbolic Tangent function fails this criteria.

However, just like we can mathematically deduce the Fourier Transform of a unit step function ( $F\{u(t)\} = \frac{1}{j\omega} + \pi\delta(\omega)$ ), we can also mathematically have Hyperbolic Tangent function Fourier Transformed as:

- $$F\{\tanh(t)\} = j\sqrt{\frac{\pi}{2}} \cdot \operatorname{csch}\left(\frac{\pi\omega}{2}\right)$$

But what does a Hyperbolic Cosecant function look like?

# AMI\_GetWave()



=> What's the value at DC and LF range?

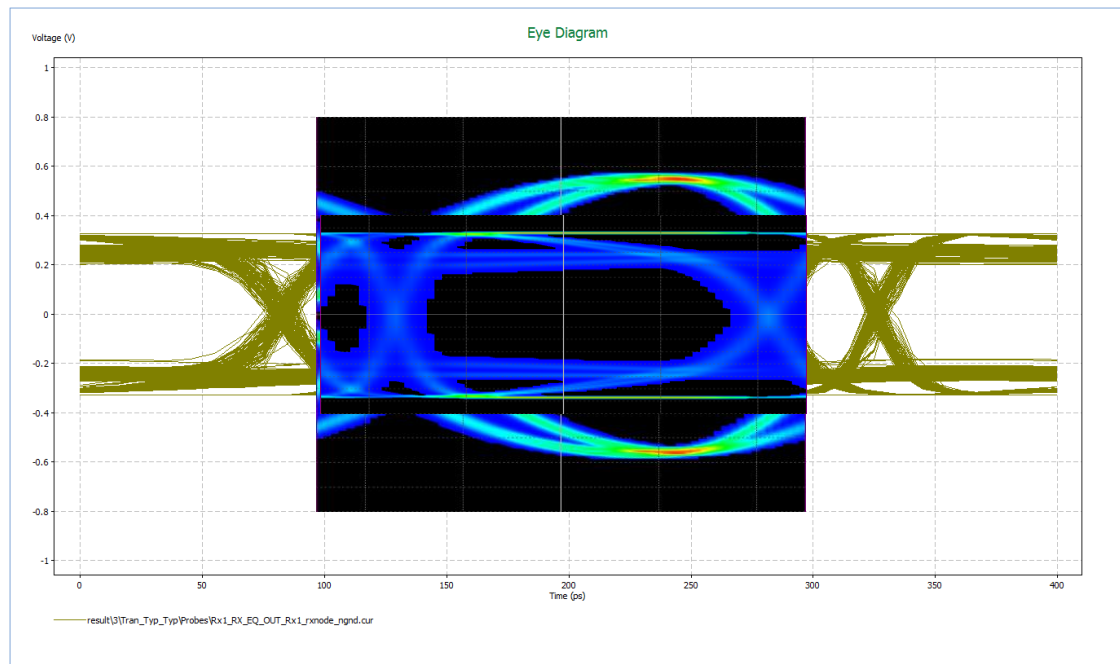
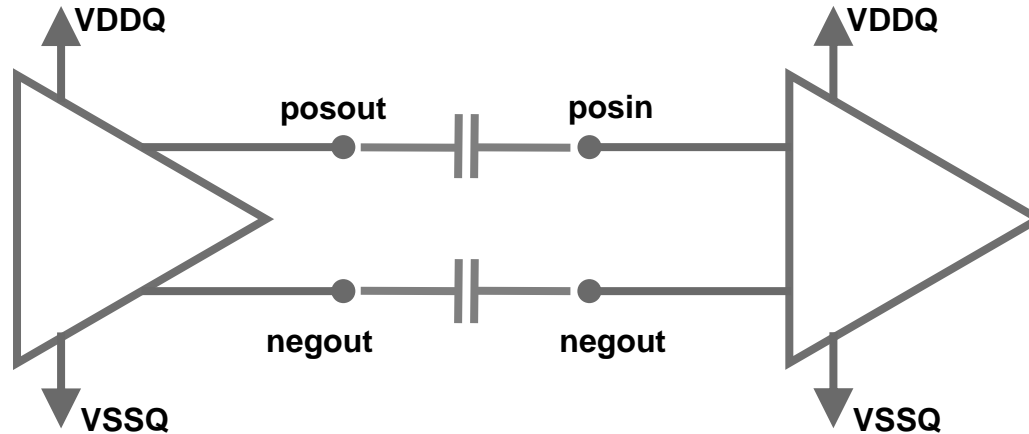
- The most intuitive way to model a Hyperbolic Tangent function is to implement it in [AMI\\_GetWave\(\)](#)

```
for (t=0; t<end_time; t++)  
{  
    Vout[t] = C1*tanh(Vin[t]/C2);  
}
```



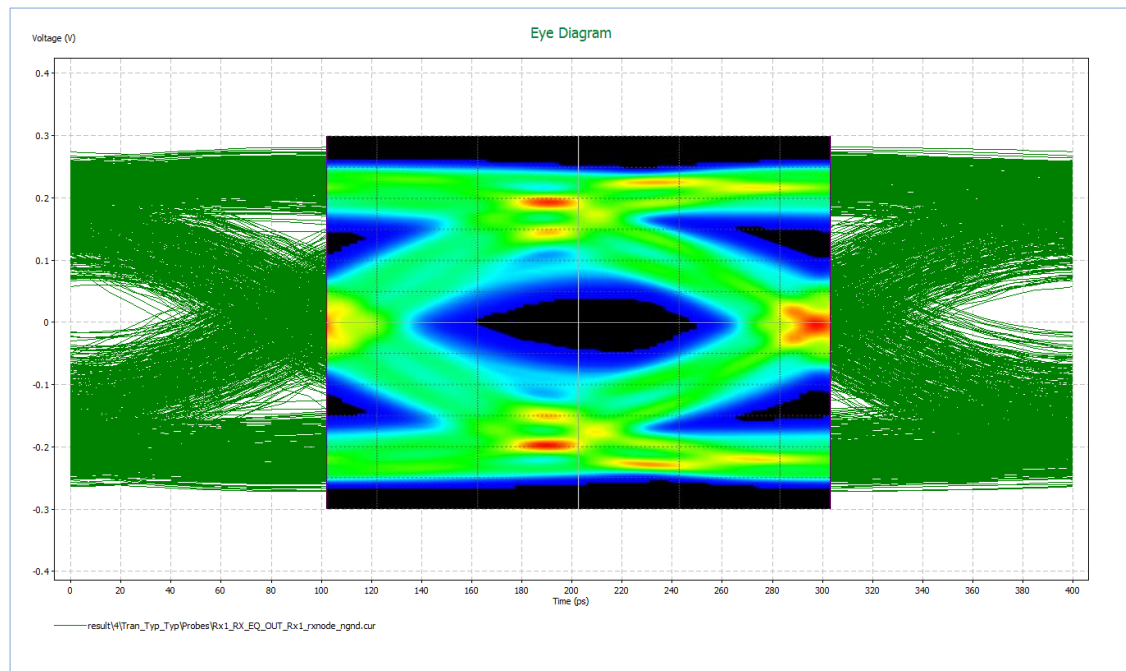
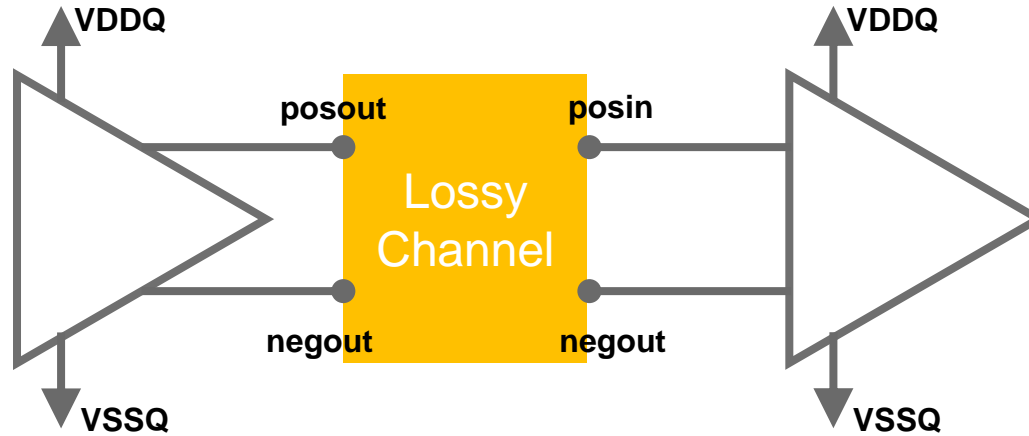
# Example 1

- An USB 3.0 IP, Transmission Rate = 5Gbps, No Channel between Tx and Rx but only a pair of AC Caps:



# Example 1 (Cont'd)

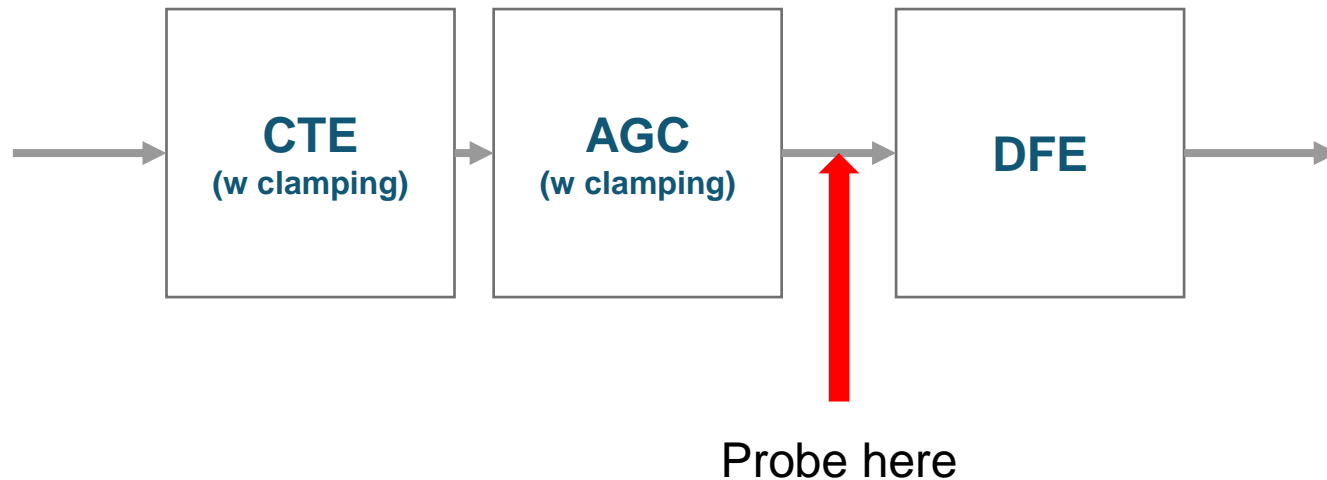
- An USB 3.0 IP, Transmission Rate = 5Gbps, Lossy Channel:



## Example 2

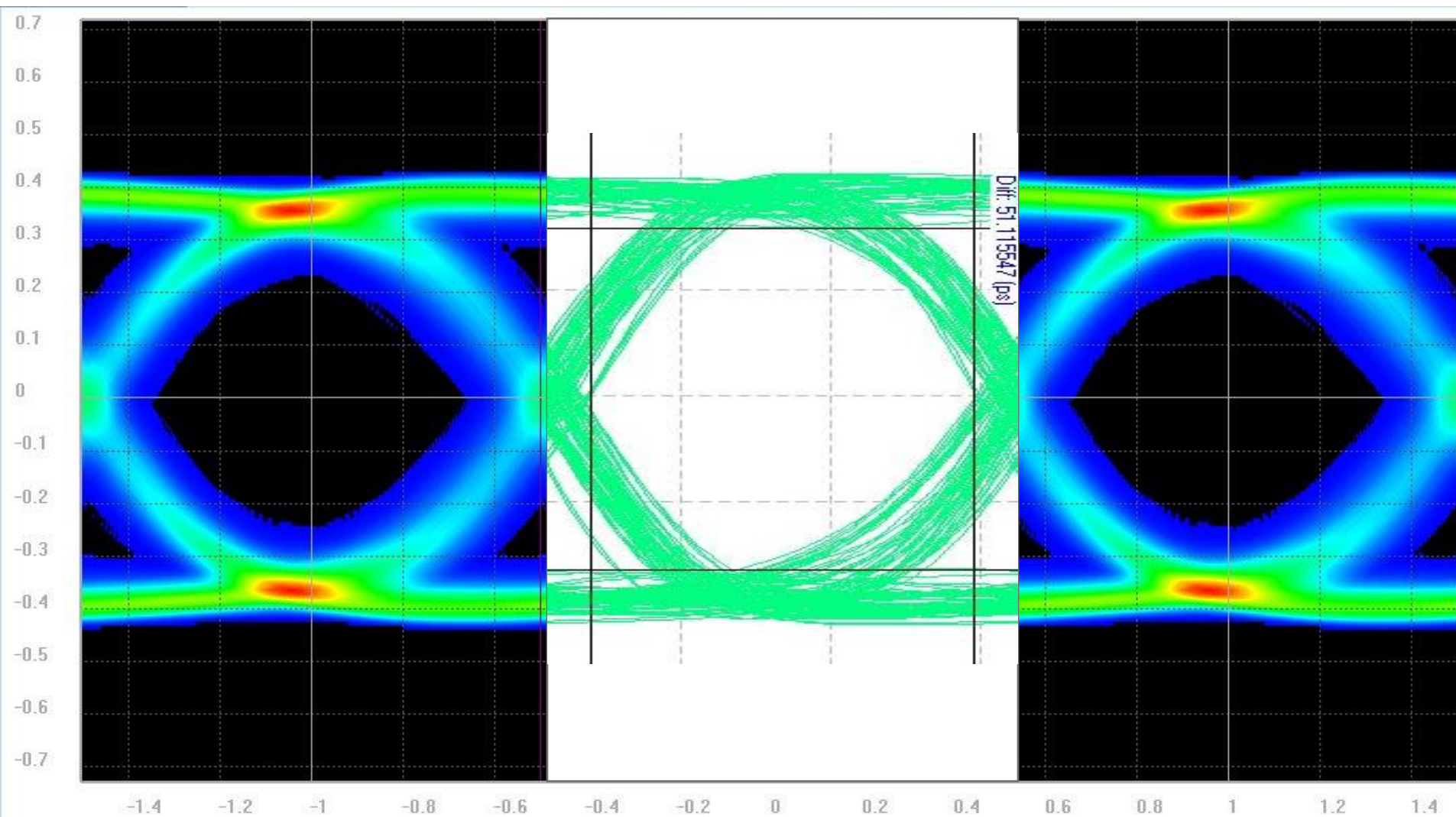
- A PCIe Gen 4.0 IP, Transmission Rate = 16Gbps, M31 published on CDNLive Taiwan 2018

### Rough Block Diagram of RX EQ:



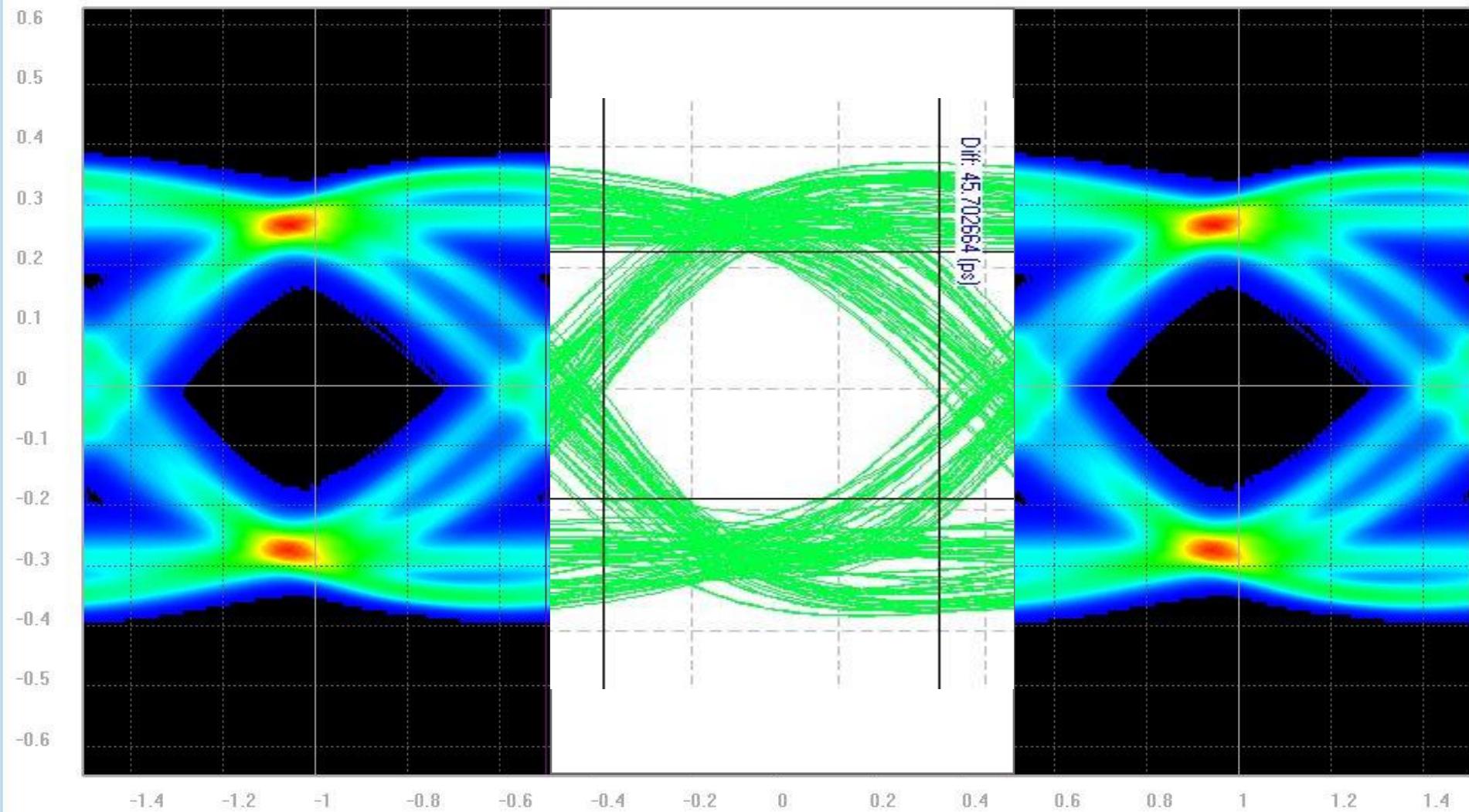
## Example 2 (Cont'd)

- Short Channel – Loss= -10dB



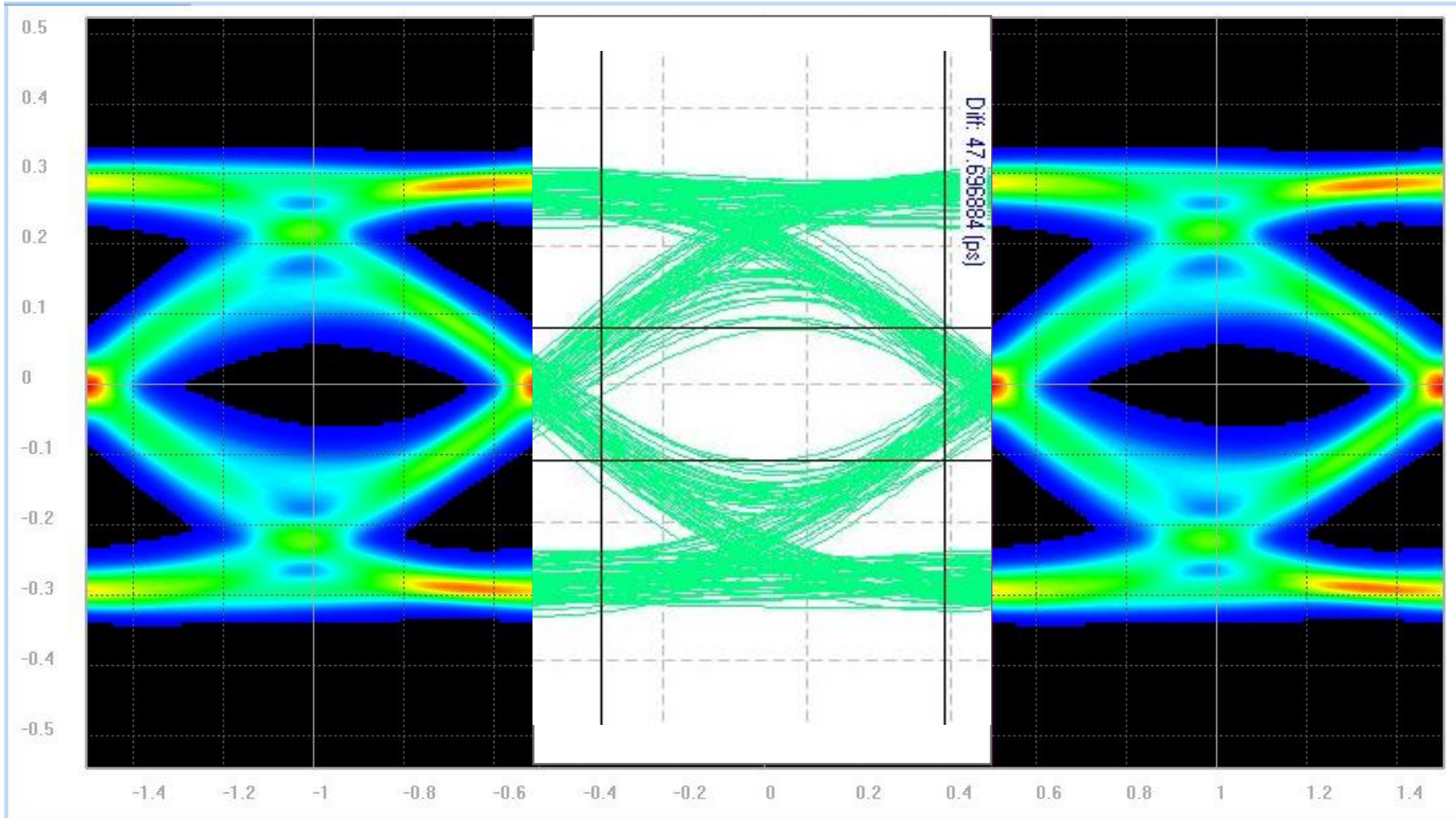
## Example 2 (Cont'd)

- Mid Channel – Loss= -18dB



## Example 2 (Cont'd)

- Long Channel – Loss= -28dB



# Conclusion

- Clamping diodes or circuits with similar behaviors are deemed as protection means and so common to be in most designs which makes most designs to be **Non-Linear**.
- We suggest **Hyperbolic Tangent Function** to be the optimal choice to describe such nonlinearity of circuits.
- We suggest a methodology with which model engineer can approach a nonlinear clamping behavior ultimately.
- Hyperbolic Tangent Function is hard to be implemented in `AMI_Init()` due to its nature but can be easily and intuitively implemented in `AMI_GetWave()`, for this we even suggest a simplified code.
- According to the description above, this implementation will limit the so-compiled AMI model to work properly in a “Time Domain Analysis” channel simulator but **fail to behave as we desire in a “Statistical” channel simulator**.
- Correlations against SPICE transient analysis are provided and it proves models generated by the methodology we proposed here can accommodate all kinds of channels.

*See you on IBIS Summit 2019*

**cā d e n c e<sup>®</sup>**

©2018 Cadence Design Systems, Inc. All rights reserved worldwide. Cadence, the Cadence logo and Spectre are registered trademarks and Sigrity, SystemSI, and T2B are trademarks of Cadence Design Systems, Inc. in the United States and other countries.. All rights reserved. All other trademarks are the property of their respective owners.