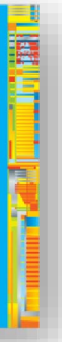
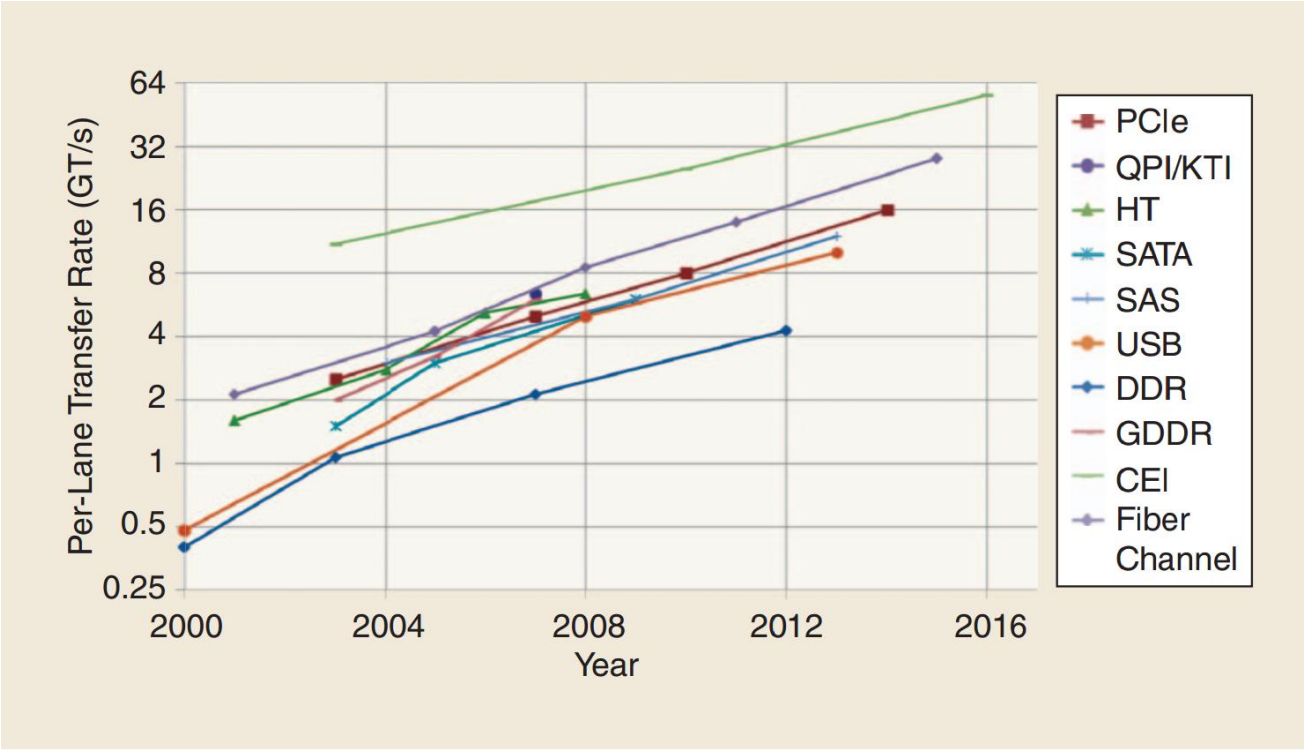
Clock and Data Recovery Systems



Fulvio Spagna – Sr. Principal Engineer (INTEL)

Introduction

Per-pin data rate versus year for a variety of common I/O standards



D. Daly et al., "Through the Looking glass – The 2017 Edition", *IEEE SOLID-STATE CIRCUITS MAGAZINE Winter 2017*

Introduction

High-Speed Serial IO (HSIO) have, more than ever, become a critical ingredient in high performance communication systems.

As the link data rate increases, the HSIO designer is tasked with optimizing Performance, Power Consumption, Reliability and Cost.

- Advanced equalization techniques, clocking and timing recovery solutions, are all fundamental ingredients in this optimization strategy.

Tutorial Objective

The focus of this tutorial is on Clock Data Recovery architectures and in particular on:

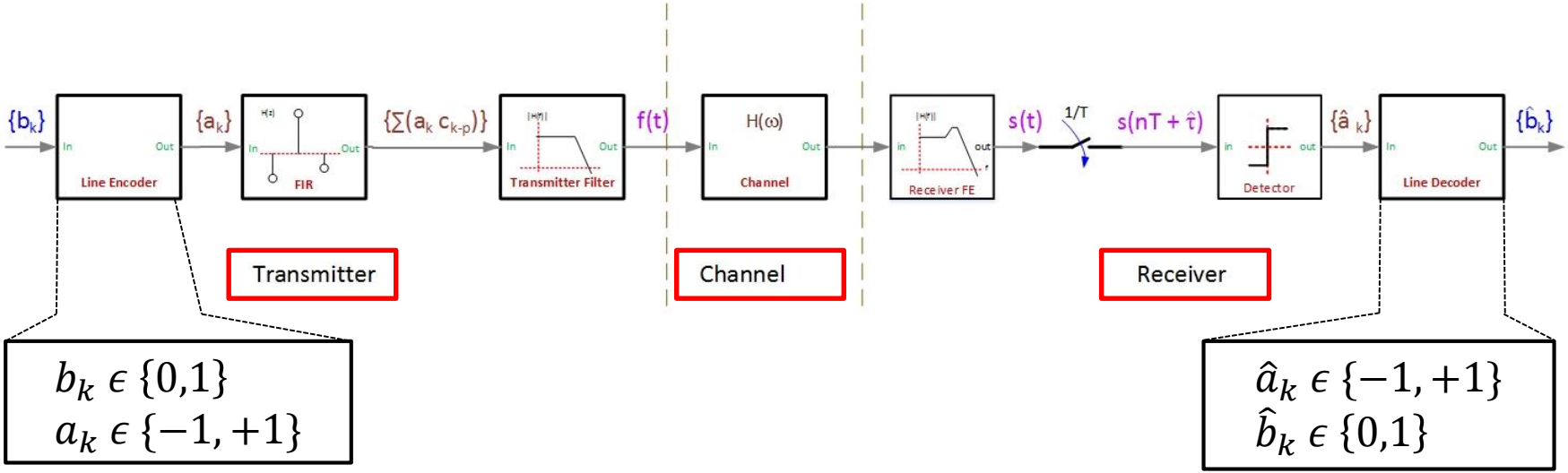
- the Phase Detector and its interaction/dependency on the link equalization strategy
- defining which circuit topologies are better suited for a given Phase Detector choice
- illustrating simple methods for analyzing the Clock Data Recovery performance

Tutorial Outline

- Basic Communication System
- Signal recovery problem
- Eye Diagram
- Timing Functions
- Sampling Point choice
- Alexander Phase detector
- Mueller - Muller Phase detector
- Summary

BASIC COMMUNICATION SYSTEM

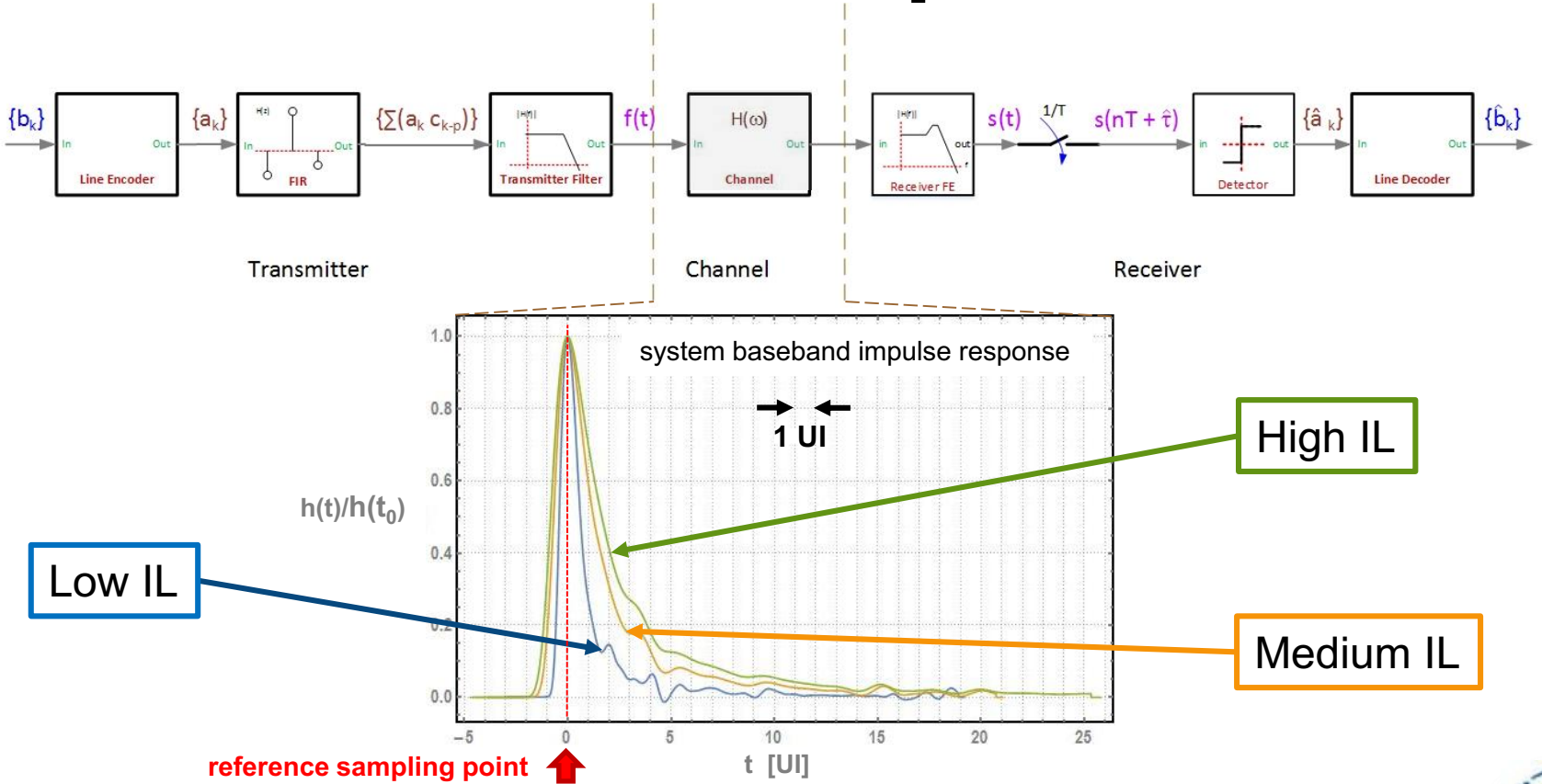
Basic Communication System



The block diagram outlines the structure of a typical baseband PAM system.

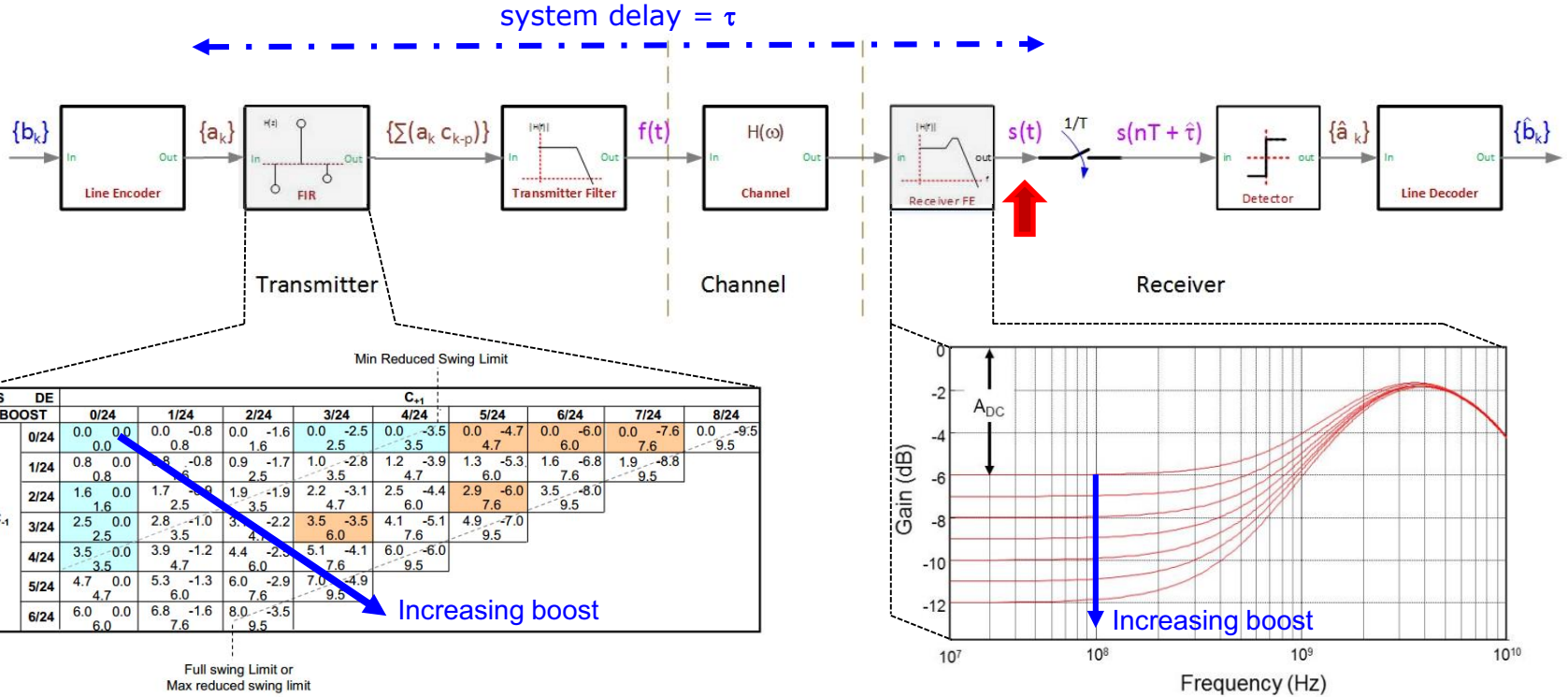
In this presentation we will focus on PAM2 systems where the channel (backplane, wireline etc.) is assumed *time invariant*.

Channel Response



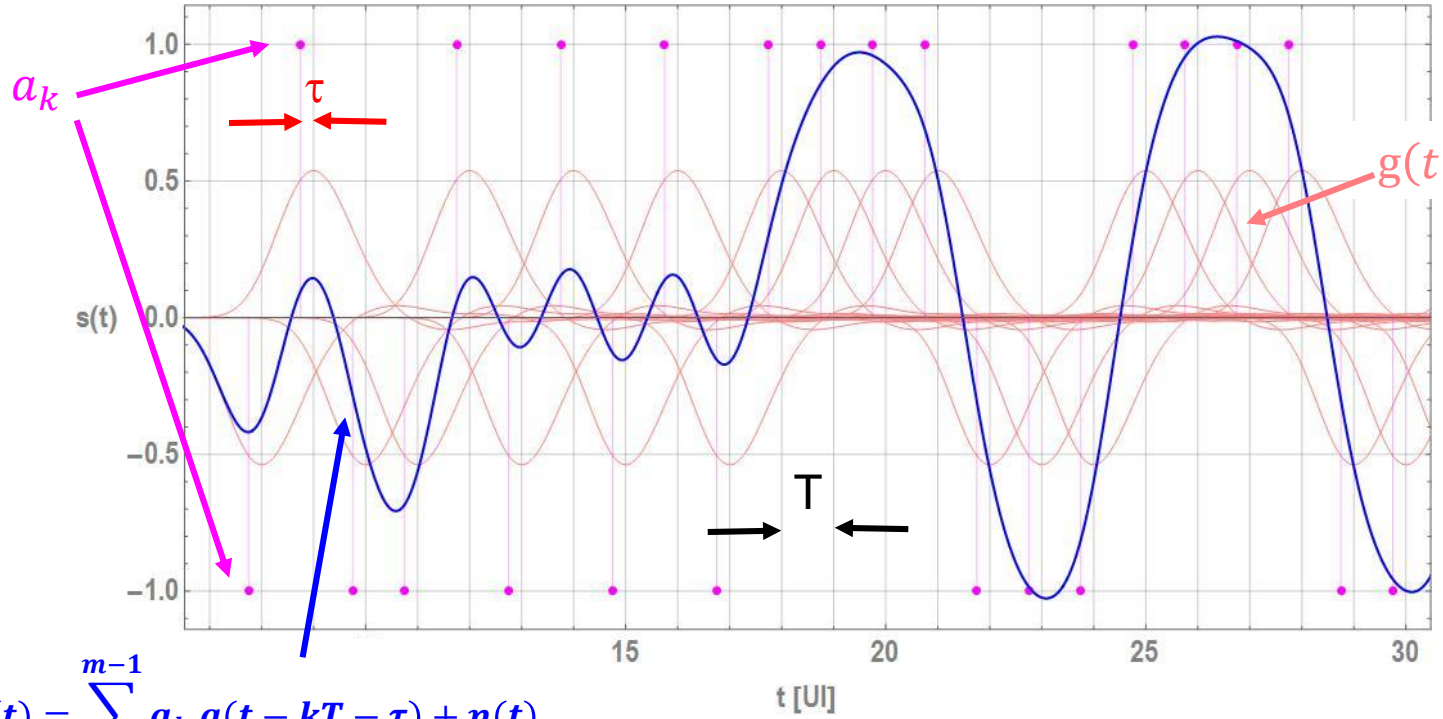
Equalization Capabilities

(an example from the PCIe standard)



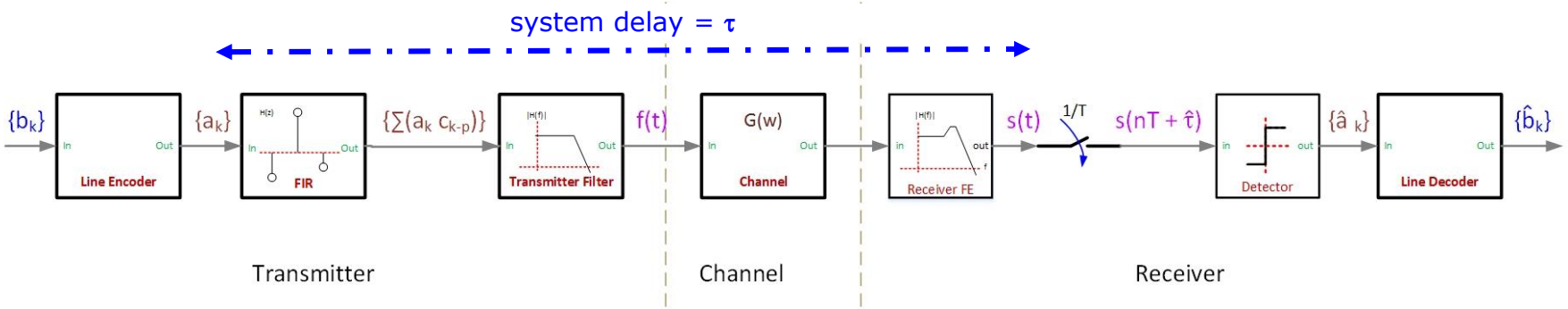
PCIe Express Base specification, Rev. 4.0

Signal at the Receiver FE Output



$$s(t) = \sum_{k=0}^{m-1} a_k \, g(t - kT - \tau) + n(t)$$

Sampled Signal - 1

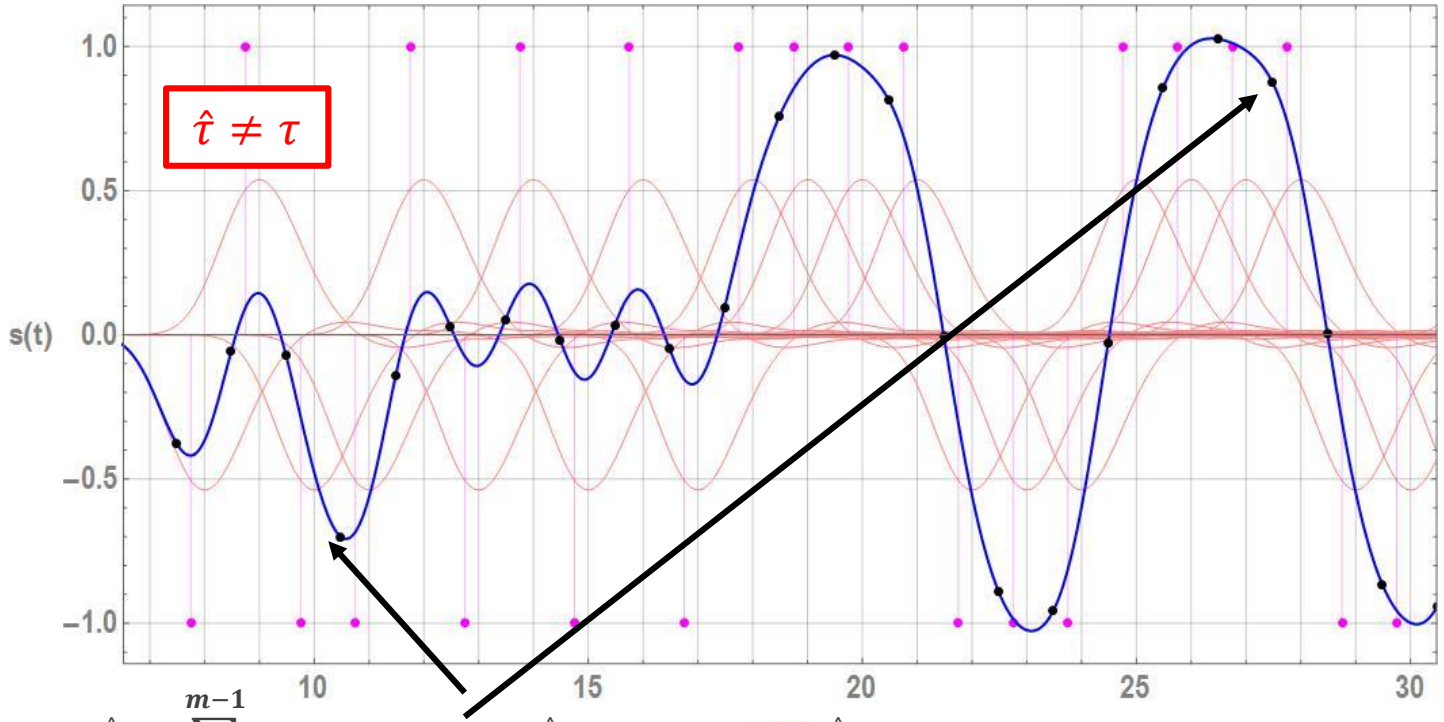


Since the symbols sequence a_k is clocked at instants $\{\boldsymbol{nT}\}$ we can sample the received signal *s(t)* at some instants $\{\boldsymbol{nT} + \hat{\boldsymbol{\tau}}\}$ to produce a *sampled version* of *s(t)*:

$$s_n = s(nT + \hat{\tau}) = \sum_{k=0}^{m-1} a_k g((n-k)T + (\hat{\tau} - \tau)) + n(kT + \hat{\tau})$$

Sampled Signal – 2

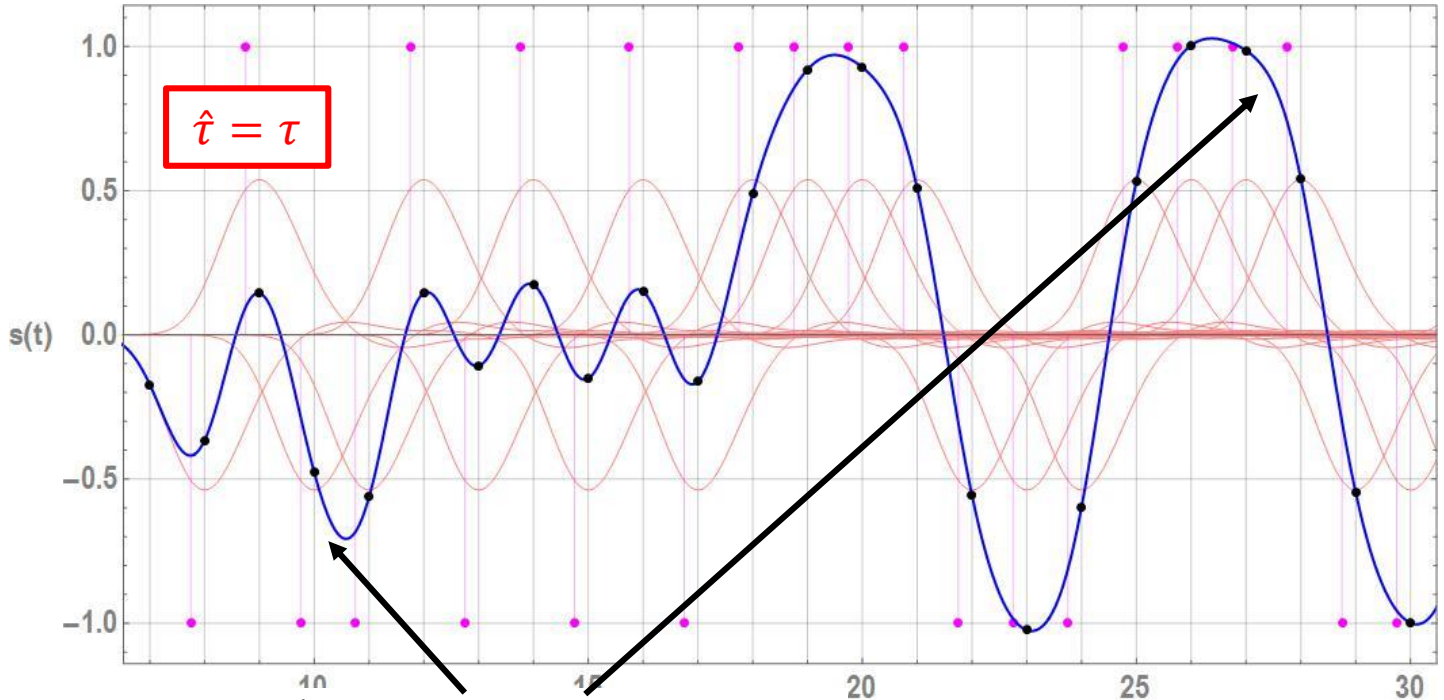
In the absence of an estimate for the system delay, $\hat{\tau}$, the sampling position of the signal s(t) is arbitrary!



$$s_n = s(nT + \hat{\tau}) = \sum_{k=0}^{m-1} a_k g((n-k)T + (\hat{\tau} - \tau)) + n(kT + \hat{\tau})$$

Sampled Signal – 3

If we correctly estimate the system delay then $\hat{\tau} = \tau$



$$s_n = s(nT + \tau) = \sum_{k=0}^{m-1} a_k g((n-k)T) + n(kT + \tau)$$

Cursor Amplitude

The average value of the received signal at the sampling times is referred to as the ***cursor amplitude*** *and can be written as*:

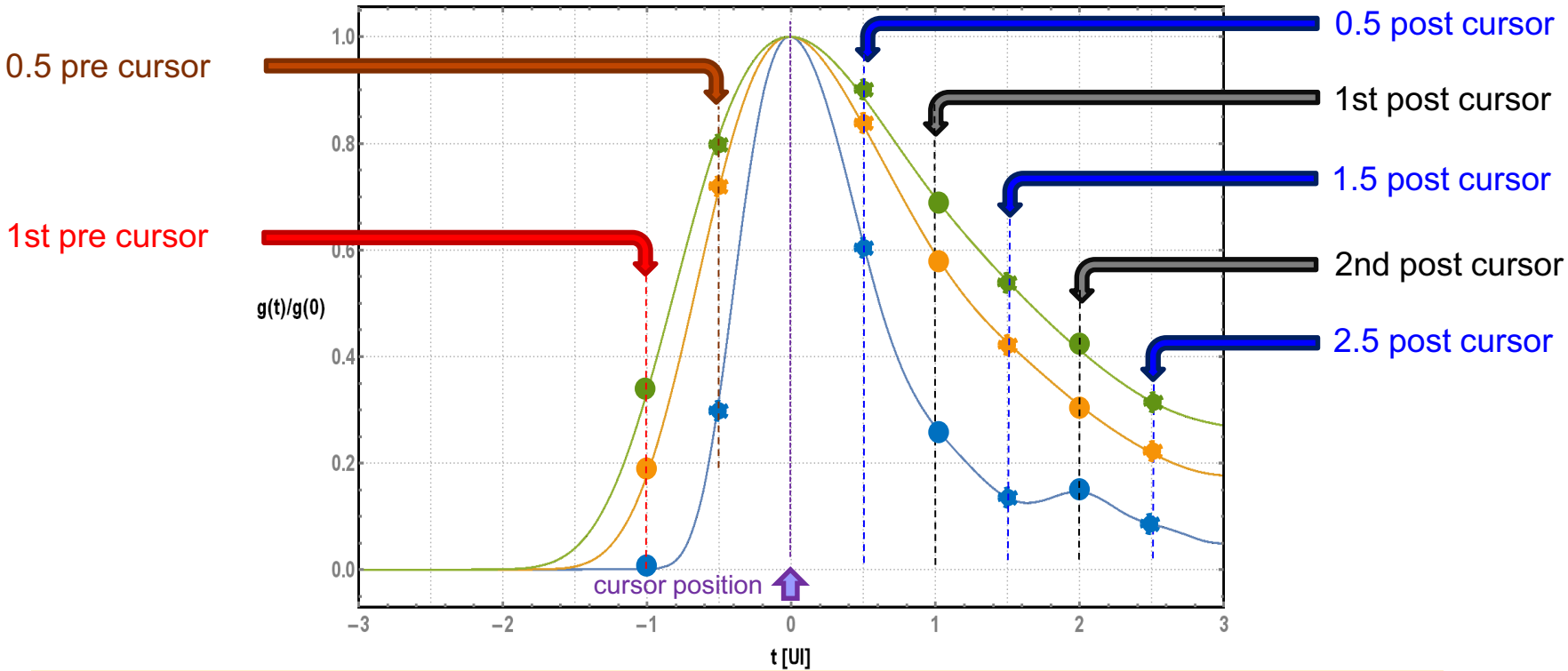
$$\langle s_n | a_n = 1 \rangle = \sum_{k=-q}^{-1} g(kT)\,\langle a_{n-k} \rangle + \boldsymbol{g(0)} + \sum_{k=1}^{m-q-1} g(kT)\,\langle a_{n-k} \rangle + \langle n(kT+\tau) \rangle$$

which reduces to:

$$\langle s_n | a_n = 1 \rangle = \boldsymbol{g(0)}$$

under the assumptions that the symbols $\boldsymbol{a_n}$ assume the values {-1,+1} with equal probability and the noise has zero mean.

Pre/Post Cursor Components



- Any point in the pulse response, *g(t)*, can be reference to in terms of its distance from the cursor position.

THE SIGNAL RECOVERY PROBLEM

The Signal Recovery Problem - 1

To recover the original symbol sequence a_k we need to:

1. correctly estimate the system delay τ
2. remove, thru a combination of TX and RX equalization, the ISI introduced by the channel

so that:

$$\boxed{g\big((n-k)T\big) = \begin{cases} 0 & n \neq k \\ 1 & n = k \end{cases}}$$

and:

$$s_n = \sum_{k=0}^{m-1} a_k g\big((n-k)T\big) + n(kT+\tau) = a_n g(0) + n(kT+\tau)$$

intel

The Signal Recovery Problem - 2

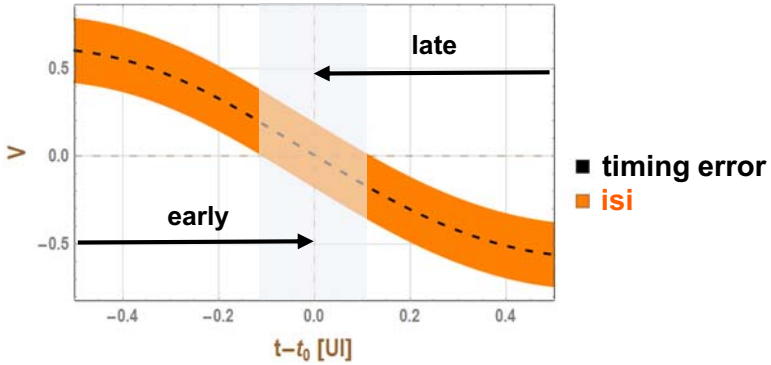
To properly estimate τ, the receiver needs to have some means to evaluate the difference between the delay τ and its current estimate $\hat{\tau}_k$ at each sampling time kT.

- The task of the Timing Recovery (or Clock Data Recovery) circuit is to provide an estimate of this timing error and continuously adjust the sampling phase to minimize this timing error.

The Signal Recovery Problem - 3

In practice the timing error will contain two terms:

- a useful components $S(\tau - \hat{\tau}_k)$ which passes thru the origin with a slope of known sign
- a zero mean noise process, N_k ,which combines the effects of imperfect equalization and additive noise



➢ In the next slides we will be mainly concerned with the effects of imperfect equalization and not concern ourselves with the effects of noise

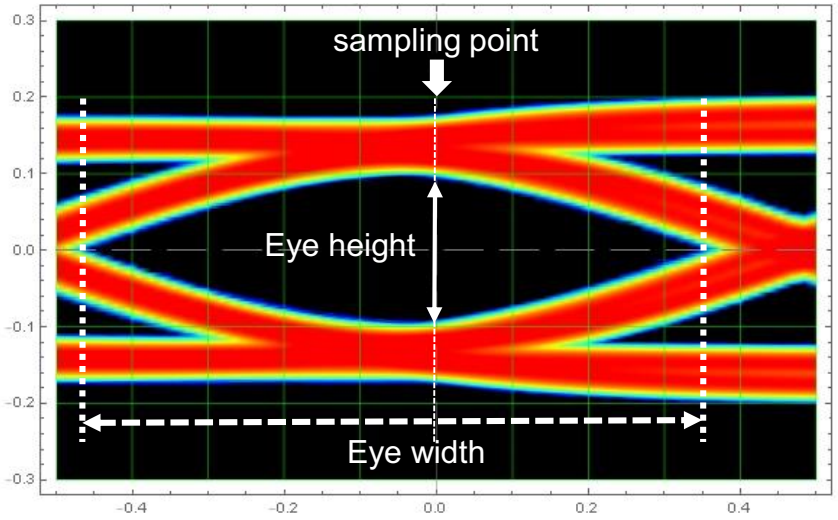
The Signal Recovery Problem - 4

In most cases the requirement:

$$g\big((n-k)T\big) = \begin{cases} 0 & n \neq k \\ 1 & n = k \end{cases}$$

is replaced by a weaker criteria of the type:

$$|g(0)| - \sum_{k \neq n} \left| a_k \mathrm{g}\big((n-k)T\big) \right| > 0$$



which guarantees a positive *margin* at the sampling point. The *margin* is typically expressed in terms of ***Eye width*** and ***Eye height***.

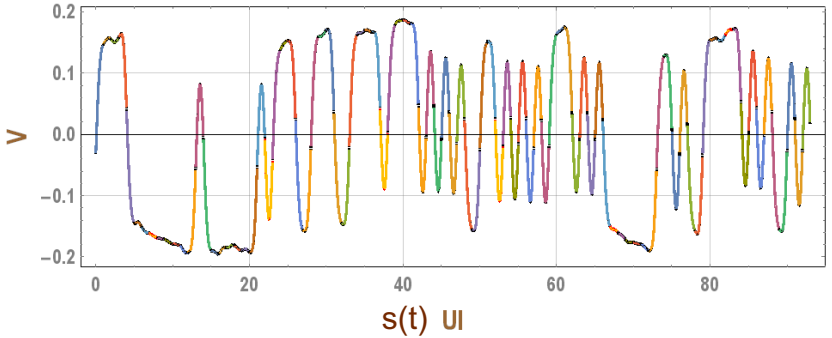
EYE DIAGRAM

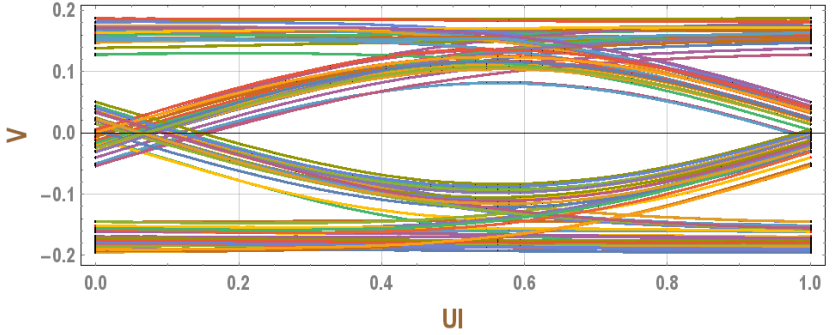
Eye Diagram - 1

The *eye diagram* is built by folding the input signal *s(t)* on a time segment of length equal to the symbol period.

The eye diagram visualizes information related to:

- the *noise margin* at the sampling point (*eye height, EH*)
- the *timing margin (eye width, EW)*





Eye Diagram - 2

The generation of a *statistical eye diagram* requires capturing a large amount of data.

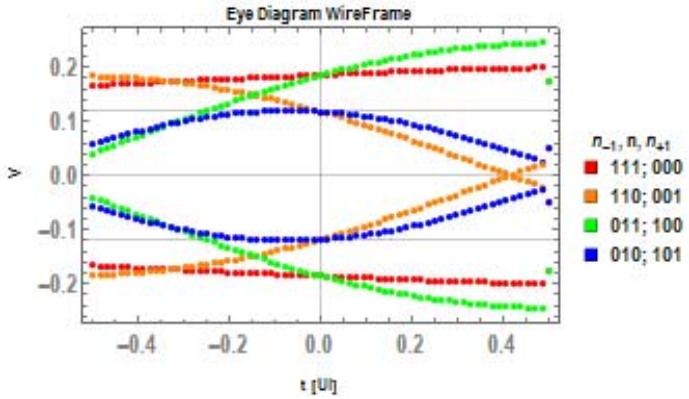
A fair amount of information is however obtainable with simple manipulations of the system baseband pulse response, *g(t)*, and comparatively little computational effort.

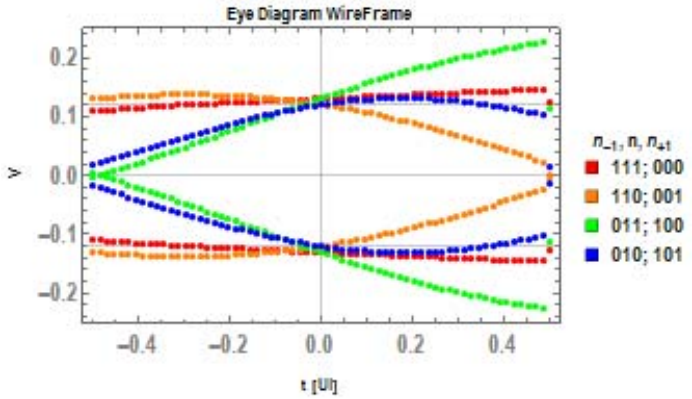
- The idea is to generate an eye diagram based on all possible three bit sequences (**tri-bit** eye diagram).

Eye Diagram – 3

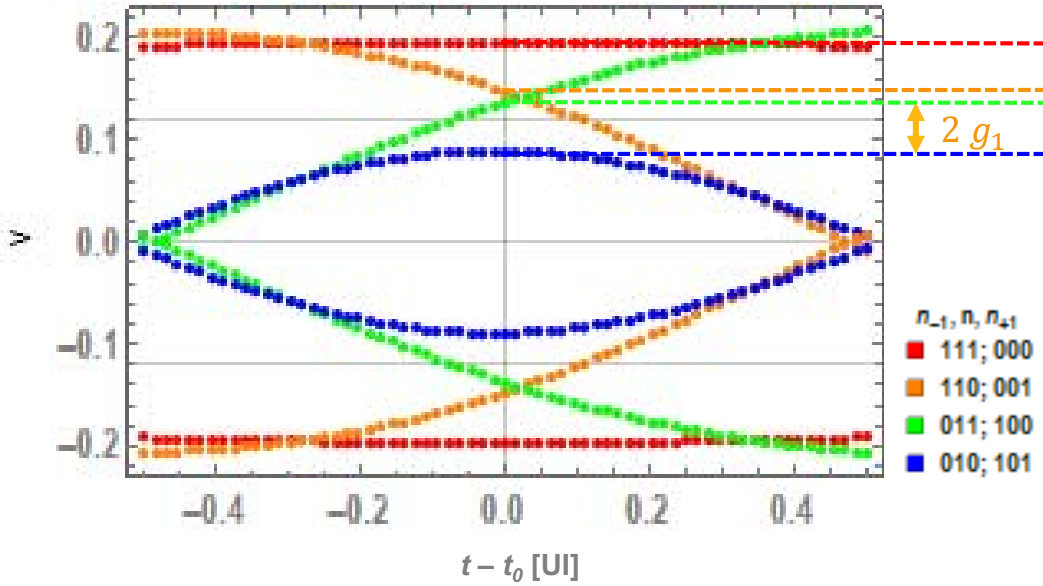
The examples below shows the results obtained for two different equalization profiles.

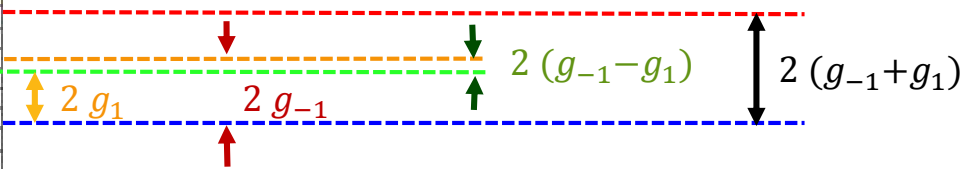
The eye diagram on the right appears qualitatively better than the one on the left because it yields a more symmetrical eye opening as well as more closely grouped trajectories.

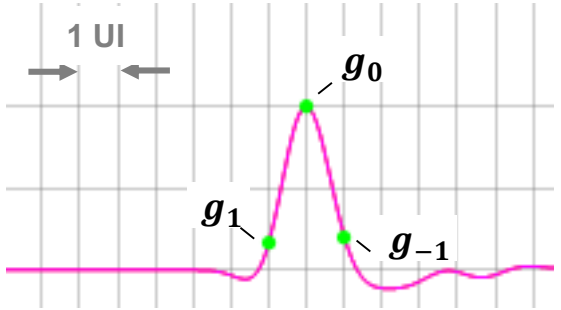




Eye Diagram - 4

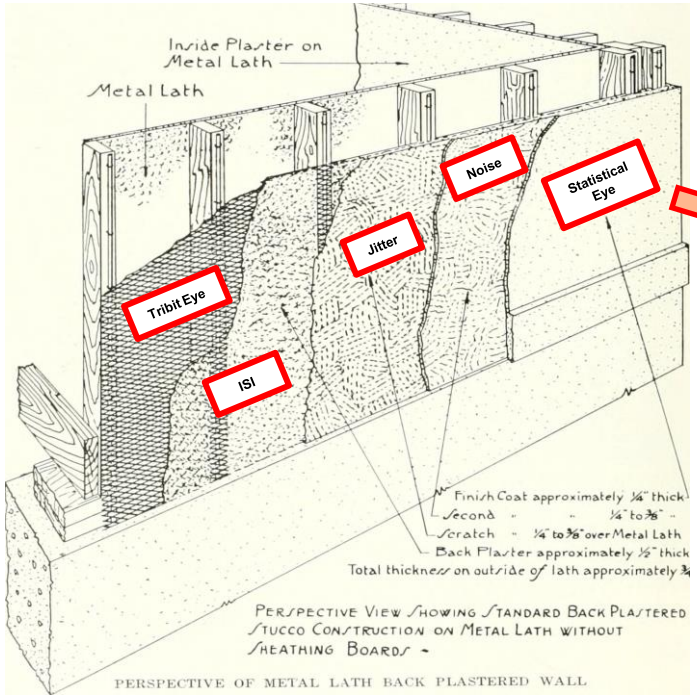




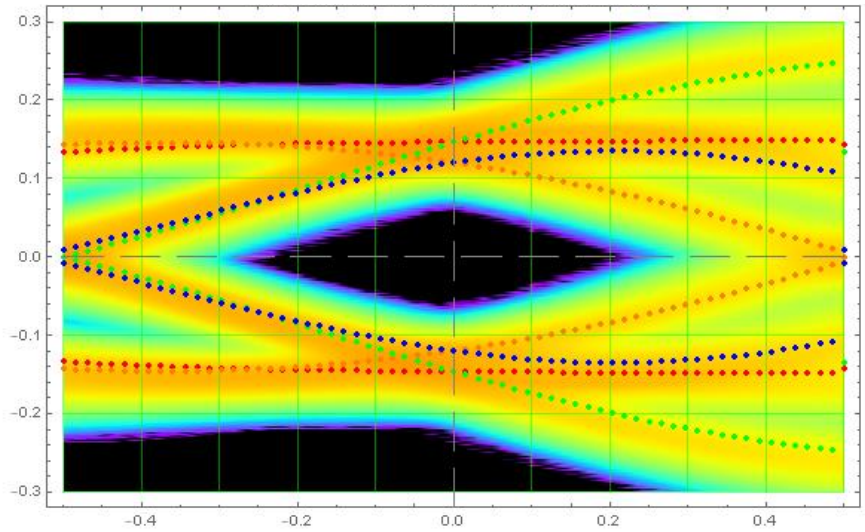


- The tri-bit eye diagram is useful in quantifying the interaction of the cursor with first pre and post cursor ISI components!

Eye Diagram - 5



Residual ISI, noise and jitter builds on the tri-bit eye diagram as stucco on a chicken wire (metal lath).

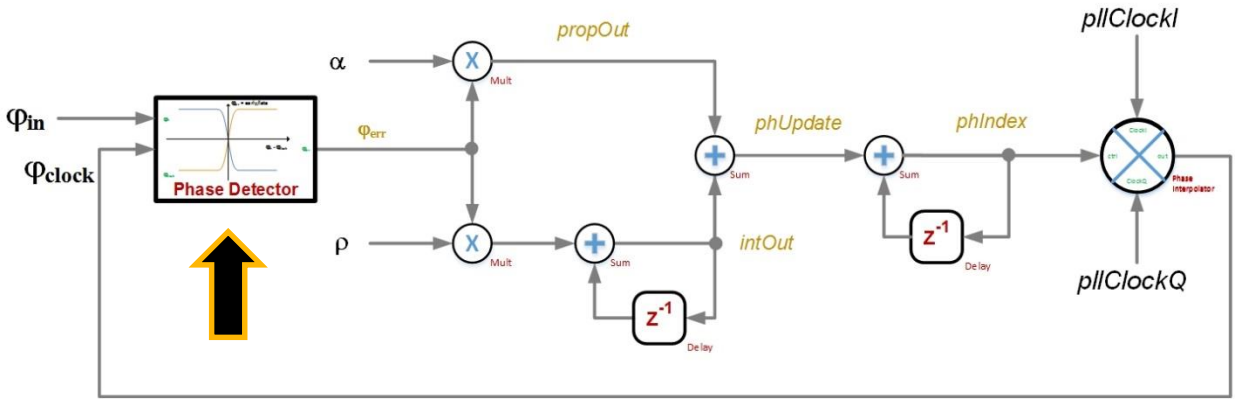


https://en.wikipedia.org/wiki/Stucco#/media/File:Architect_and_engineer_(1922)_(14594575098).jpg

- A tri-bit eye diagram superimposed on eye diagram generated by time domain simulation is a useful diagnostic tool to check the system response against expectations.

SAMPLING POINT CHOICE

Timing Recovery Block Diagram



The estimation of the error between the current sampling point and the *reference sampling point* is in practice carried out by the *phase (or timing error) detector*.

The transfer characteristics of the phase detector (PD) is sometimes referred to as *timing function*.

Timing Functions - 1

Broadly speaking, the timing functions used in High Speed Serial IO fall into two categories:

- Zero Crossing Timing Error Detectors
  - Also referred to as Alexander Phase Detectors
- Baud Rate Timing Error Detectors
  - Also referred to as Mueller and Müller Phase Detector

In the second part of this tutorial we will examine in detail the properties of these *timing functions* and the implications (constraints) that they introduce in a practical design.

Note: in this presentation the terms *timing error detector* and *phase detector* are used interchangeably.

Timing Functions - 2

Depending on how the timing function information is processed we can define the phase detector type to be:

- bang-bang (or early/late)
  - if the phase detector generates an output based on the *sign* of the timing error
- linear
  - If the phase detector generates an output which is a linear function of the timing error

In both Alexander and Mueller-Muller phase detector cases the implementation type can be *bang-bang* or *linear* (Appendix C).

Sampling Point Choice - 1

In writing the expression for an 'open eye':

$$|g(0)| - \sum_{k \neq n} \left| a_k g\left((n-k)T\right) \right| > 0$$

we have implicitly assumed that the overall solution is optimal when we are sampling at the peak of the pulse response (i.e. $\tau - \hat{\tau} = 0$)

- Because the ISI terms are not completely cancelled this choice may not necessarily the best.

Sampling Point Choice - 2

In practical situations, the CDR error signal is generated by estimating the difference between the current sampling point and a *reference sampling point,* t_0, such that $(\hat{\tau} - \tau) = t_0$ and:

$$|g(t_0)| - \sum_{k \neq n} |a_k g(t_0 + (n-k)T)| \; > \; 0$$

The choice of the reference sampling point is determined by the system timing function and should be considered as one of the receiver architecture ingredients.

ALEXANDER PHASE DETECTOR

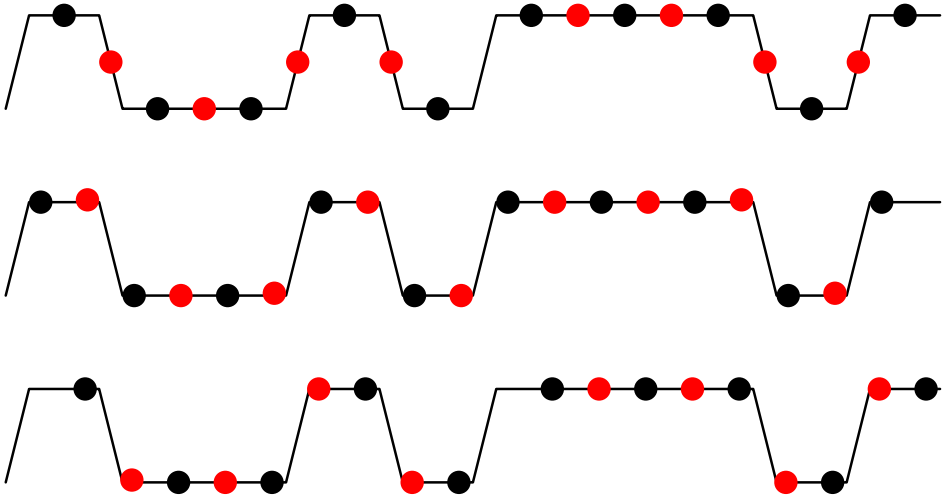
Background

- The Alexander Phase detector (or Zero Crossing time detector) extracts the timing information from the zero crossings of the signal s(t).

- Because a waveform zero crossings is associated with a data transition, it is implicit in the operation of the Alexander Phase detector the fact that timing information can (and will) only be extracted in the presence of a data transitions.

- Also implicit in the operation of such a system is the fact that the signal needs to be sampled twice per UI, once at the center of the symbol interval (*data sample*) and once at some later time (*edge sample*).

Alexander Phase Detector



Data Sample Values

The sampled signal associated with the data clock can be written as:

$$s_n(t) = \underbrace{\sum_{k=-q}^{-1} g(t+kT)a_{n-k}}_{\text{pre-cursor}} + \underbrace{g(t)a_n}_{\text{cursor}} + \underbrace{\sum_{k=1}^{m-q-1} g(t+kT)a_{n-k}}_{\text{post-cursor}}$$

which clearly identifies the ISI components related to the pre and post cursor portions of the pulse response, $g(t)$.

Edge Sample Values

An equivalent expression for the edge samples can be written as:

$$s_{n+\frac{1}{2}} = \underbrace{\sum_{k=-q}^{-2} g(t+kT+\varepsilon T)a_{n-k}}_{\text{pre-cursor}} \pm \underbrace{g(t-T+\varepsilon T)a_{n+1}}_{next\ bit} \mp \underbrace{g(t+\varepsilon T)a_n}_{current\ bit} + \underbrace{\sum_{k=1}^{m-q-1} g(t+kT+\varepsilon T)a_{n-k}}_{\text{post-cursor}}$$

where:

- the signs of the *current bit* and the *next bit* have been chosen to indicate a data transition, i.e. $a_n a_{n+1} = -1$.
- εT represents the delay between the *data clock* and the *edge clock*.

Alexander Timing Function - 1

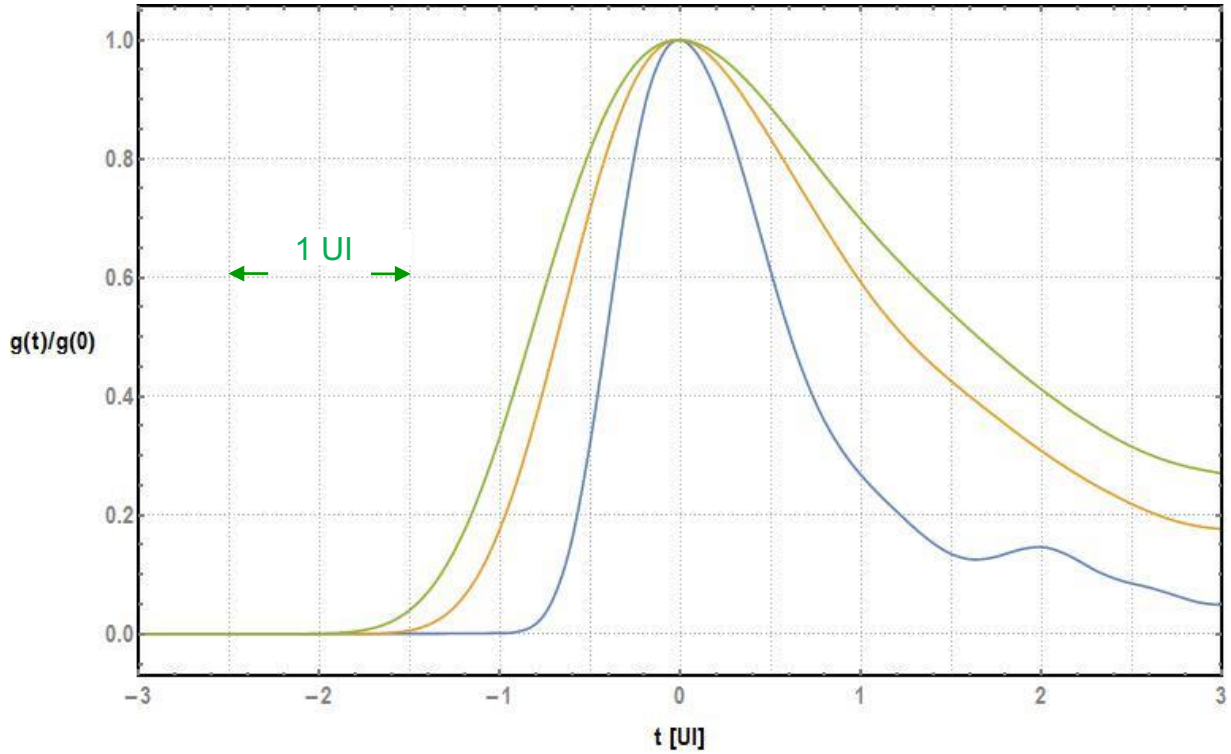
By definition the *edge sample* will be zero at a zero crossing:

$$0 = \left[g(t - T + \varepsilon T) - g(t + \varepsilon T)\right] a_n + \underbrace{\sum_{k \neq n,\ k \neq n+1} g(t - kT + \varepsilon T) a_{n-k}}_{ISI\ terms}$$

Under the assumption that the ISI terms can be neglected (as, for example, when g(t) decreases rapidly after the peak), this reduces to:

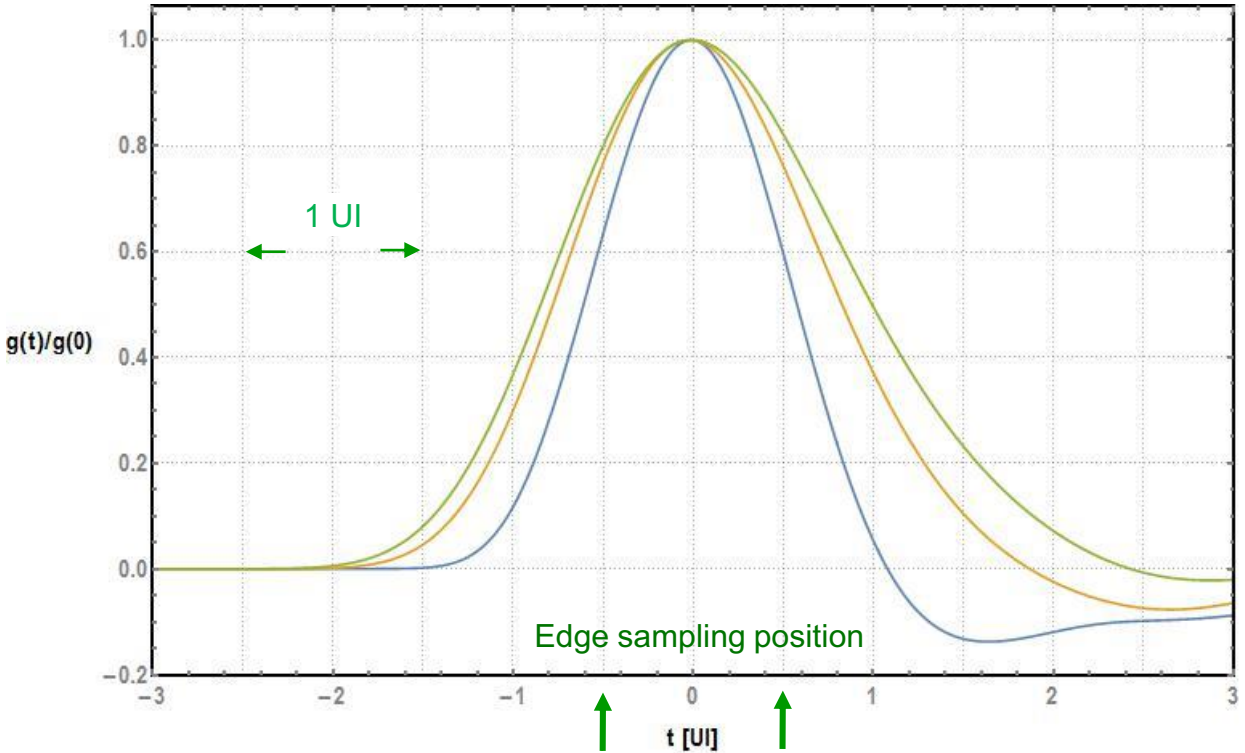
$$g(t - T + \varepsilon T) = g(t + \varepsilon T)$$

Un-equalized Channel Response



In general, an un-equalized system response does not exhibit even symmetry.

Equalized Channel Response



By proper equalization choice, the pulse response may approximate even symmetry.

Alexander Timing Function - 2

For a pulse response with even symmetry the equation:

$$g(t - T + \varepsilon T) = g(t + \varepsilon T)$$

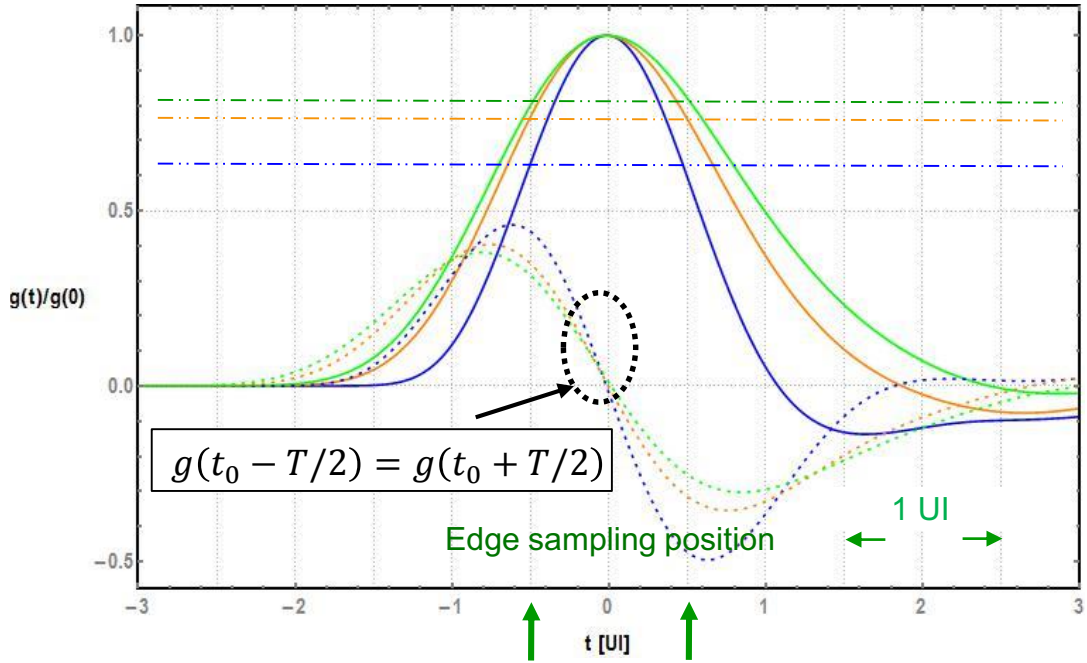
implies that:

- $\varepsilon = \frac{1}{2}$, i.e. the *edge* to *data* delay is half UI
- the *reference point*, t_0, is solution to the equation:

$$\boxed{g(t_0 - T/2) = g(t_0 + T/2)}$$

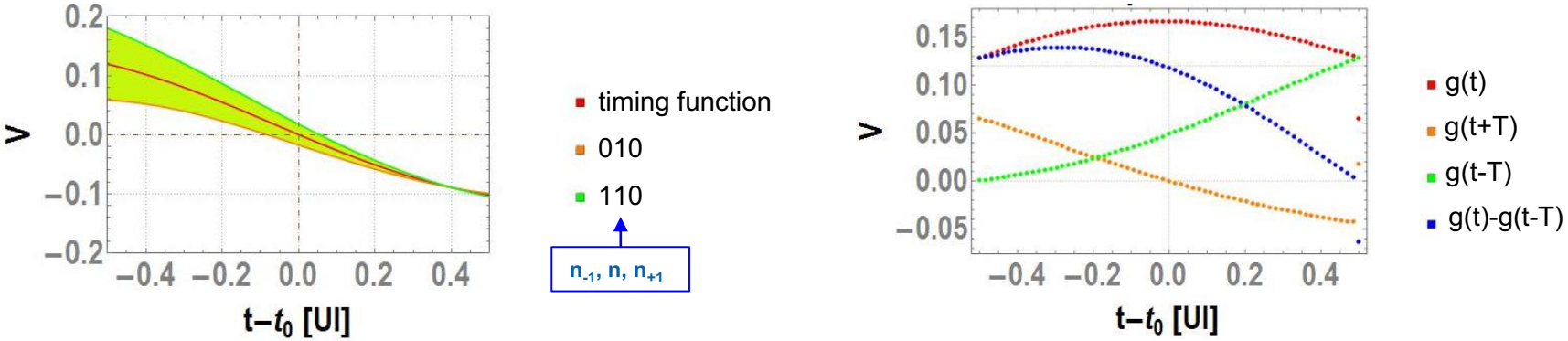
referred to as the Alexander Phase Detector *timing function*.

Alexander Timing Function - 3



- For an even symmetry case, the Alexander Timing Function represents a discrete time *approximation* of the pulse response derivative

Timing Function decomposition

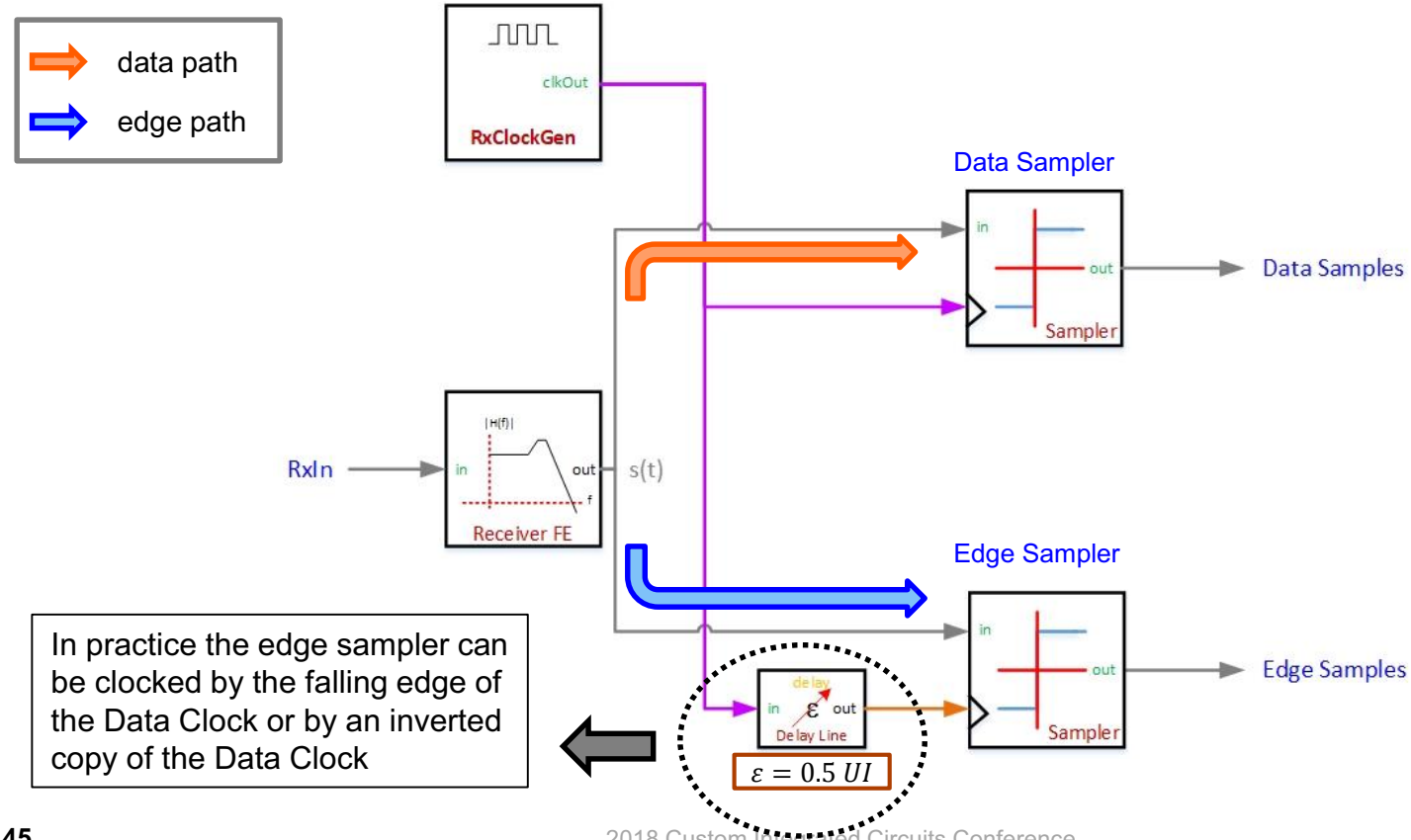


- The influence of the first *post-cursor* on the timing function can be quantified by plotting the Timing Functions for the 010 and 110 patterns.
- By plotting g(t) and g(t-T) on the same interval, we can visualize differences in the slopes of g(t) at the edge points.

Preliminary Observations

- A timing recovery built around an Alexander Phase Detector will require two clocks, one for sampling data, one for edge sampling.
- The delay between these two clocks needs to be adjusted on a case by case basis to compensate for ISI effects in the edge sample.
- Both data and edge samples remain subject to ISI terms which need to be compensated as they will, if left untouched, degrade the noise and timing margins

Alexander – Ctle



In practice the edge sampler can be clocked by the falling edge of the Data Clock or by an inverted copy of the Data Clock

Hardware Complexity

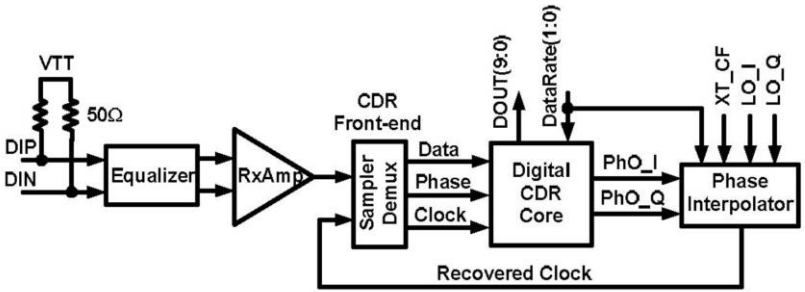
	Summers	Clock Phases	Data Slicers	Edge Slicers	Error Slicers	
Alexander PD + Ctle	0	2N	N	N	0	

This is possibly the simplest topology that can support equalization on the receive side and does not entirely rely on the transmitter to equalize the link.

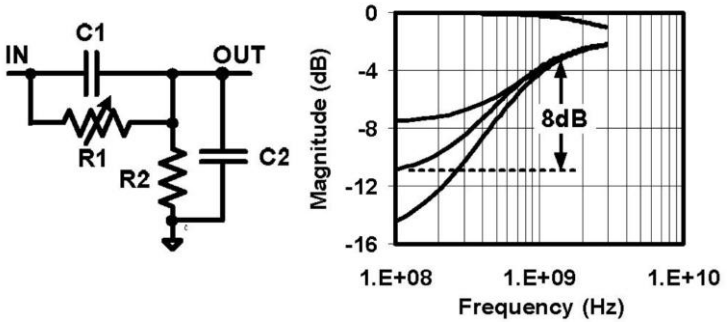
Note: In the table above, **N** indicates the number of interleaves.

Ctle Examples

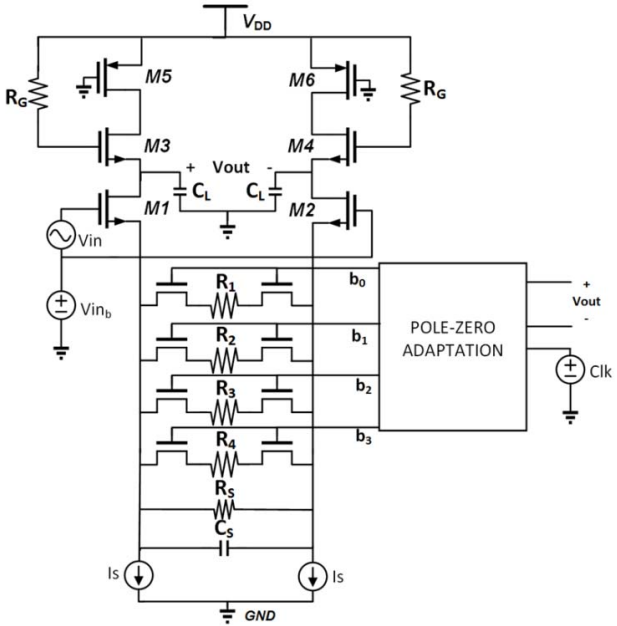
passive filter



Chen et al.. "A 9Onm 1-4.25-Gb/s Multi Data Rate Receiver for High Speed Serial Links", 2006 IEEE Asian Solid-State Circuits Conference

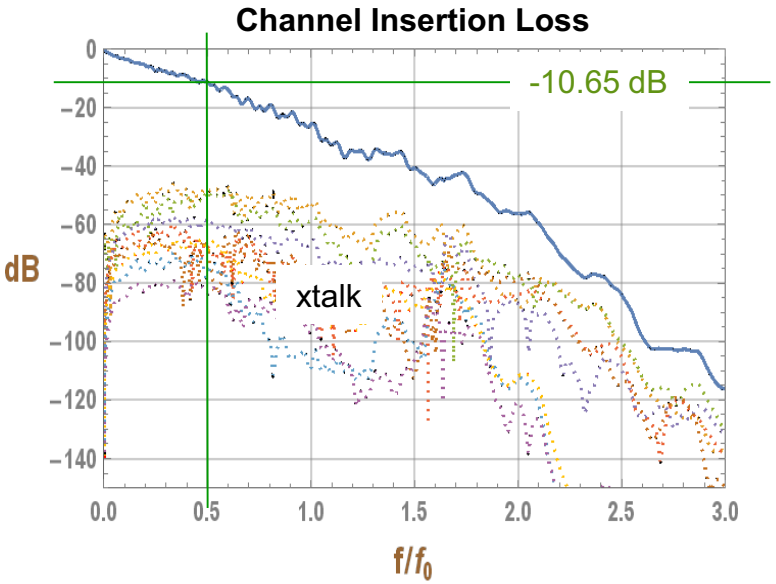


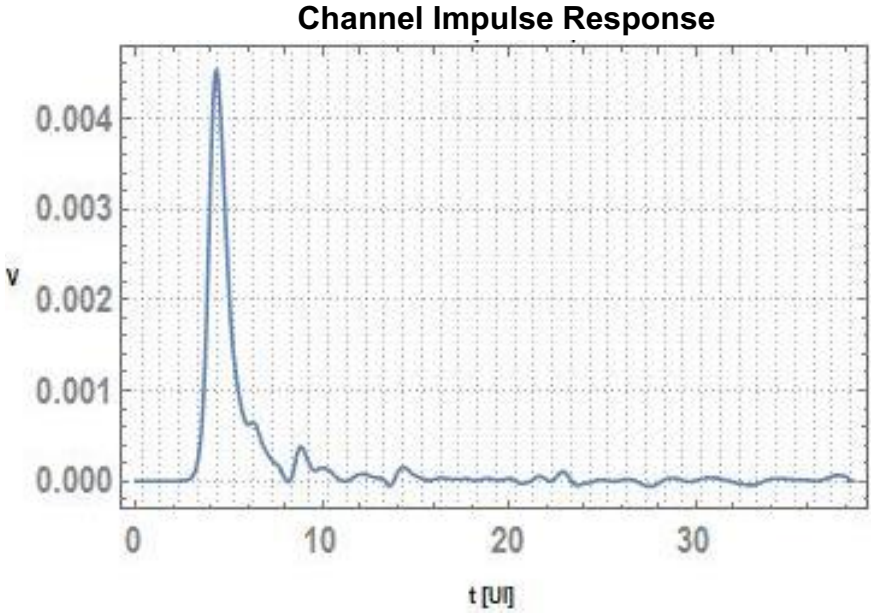
active filter



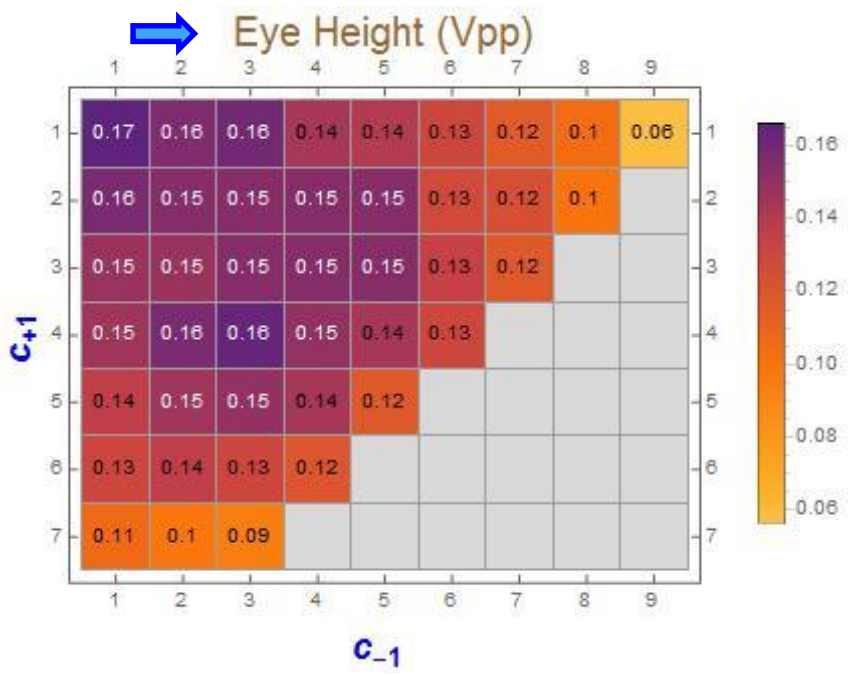
S. Agarwal et al. "A 5-Gb/s Adaptive Ctle with Eye-Monitoring for Multi-Drop Bus Applications", 2014 IEEE 57th International Midwest Symposium on Circuits and Systems (MWSCAS)

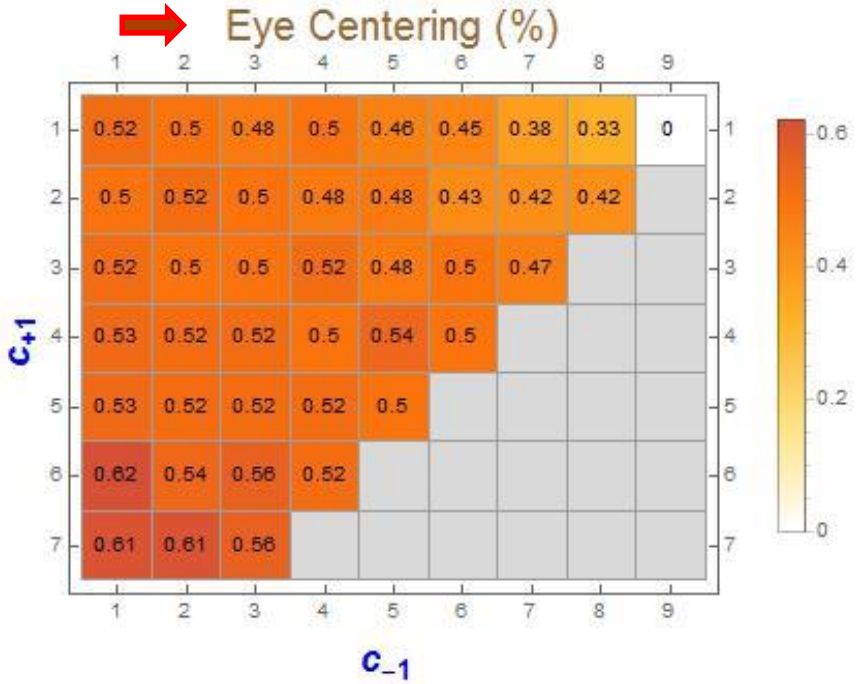
Low IL Channel





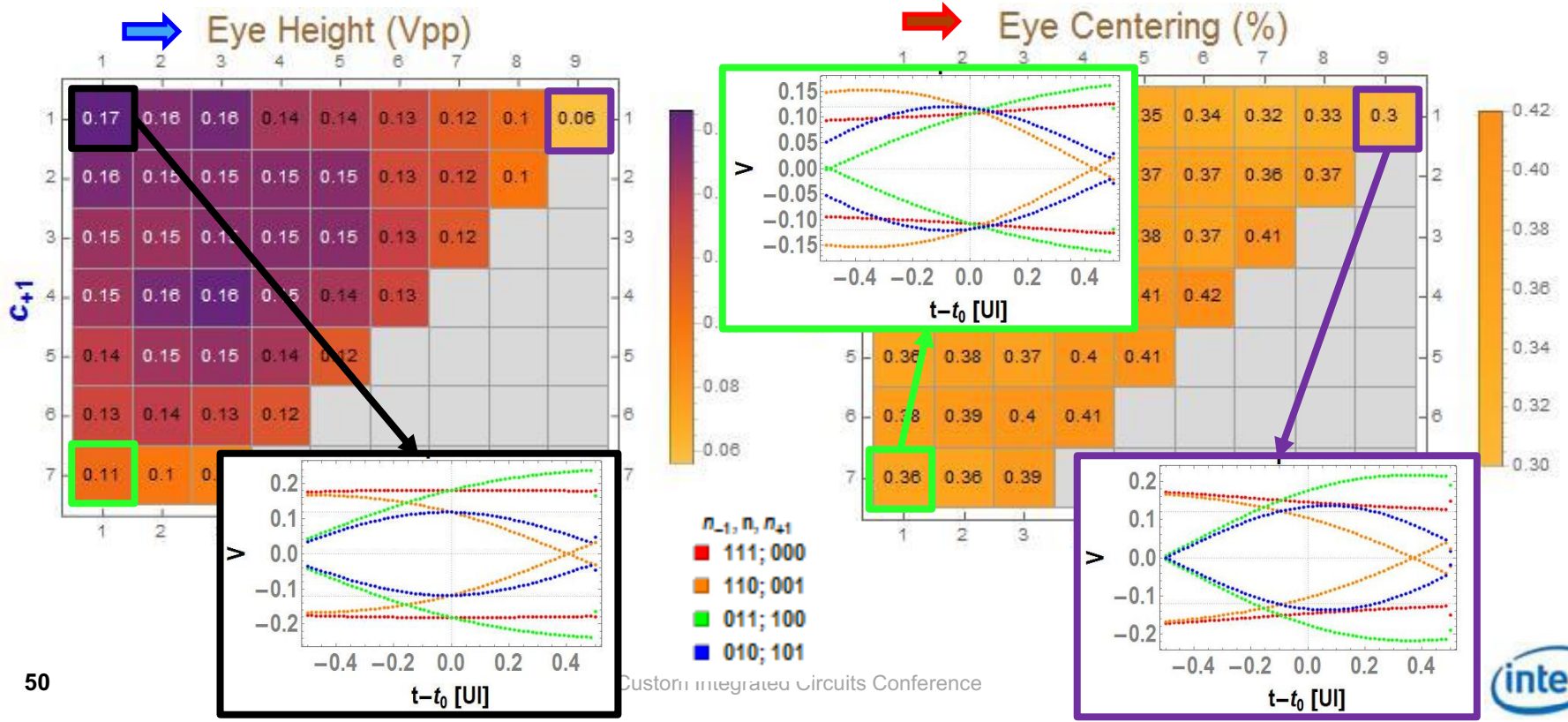
Margin Maps
(Low IL)



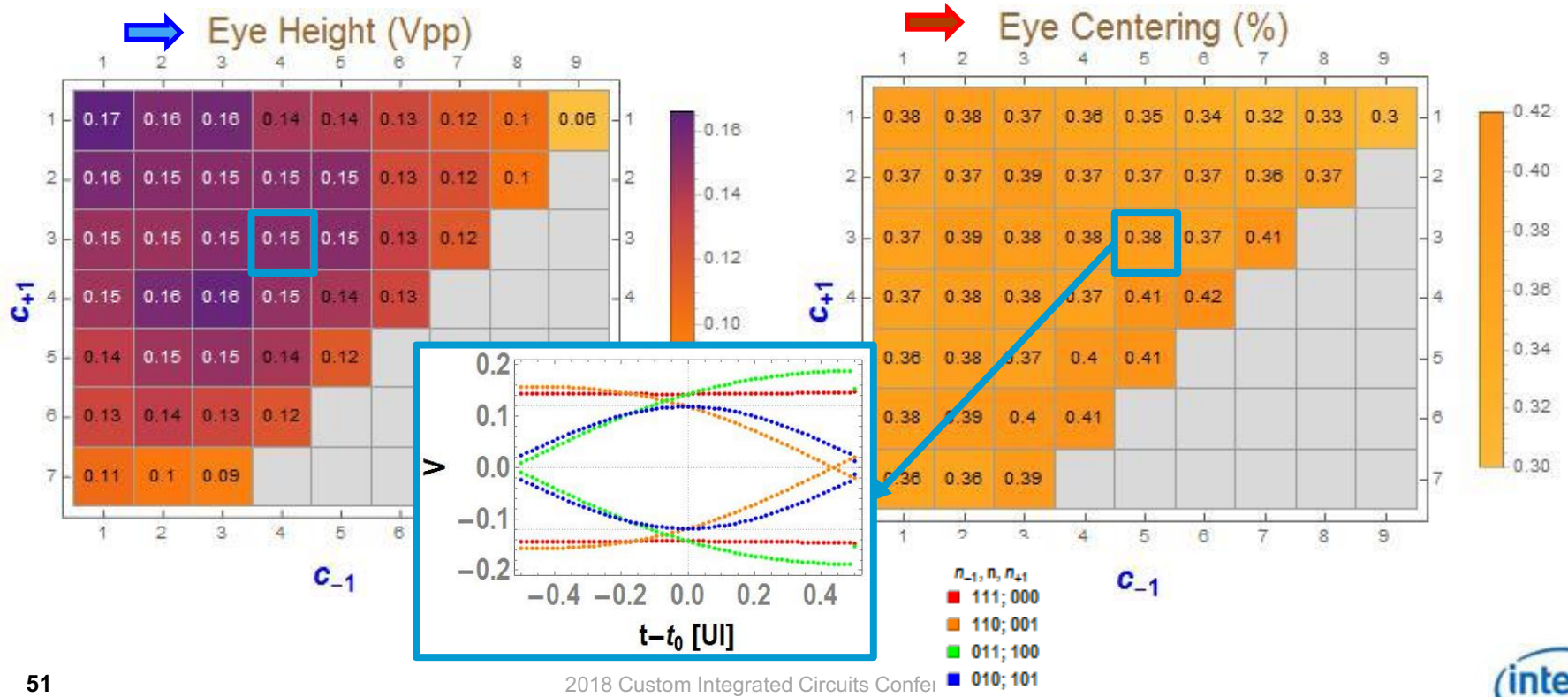


- Margin Maps are a useful diagnostic tool to assess the system sensitivity to different TxEq choices.

Tri-bit Eye Diagram - 1



Tri-bit Eye Diagram - 2



Observations

Increasing *post-cursor* equalization affects the trailing edge of the pulse response

- In an over-equalized condition this causes the right part of the eye to shrink resulting in asymmetric timing margin

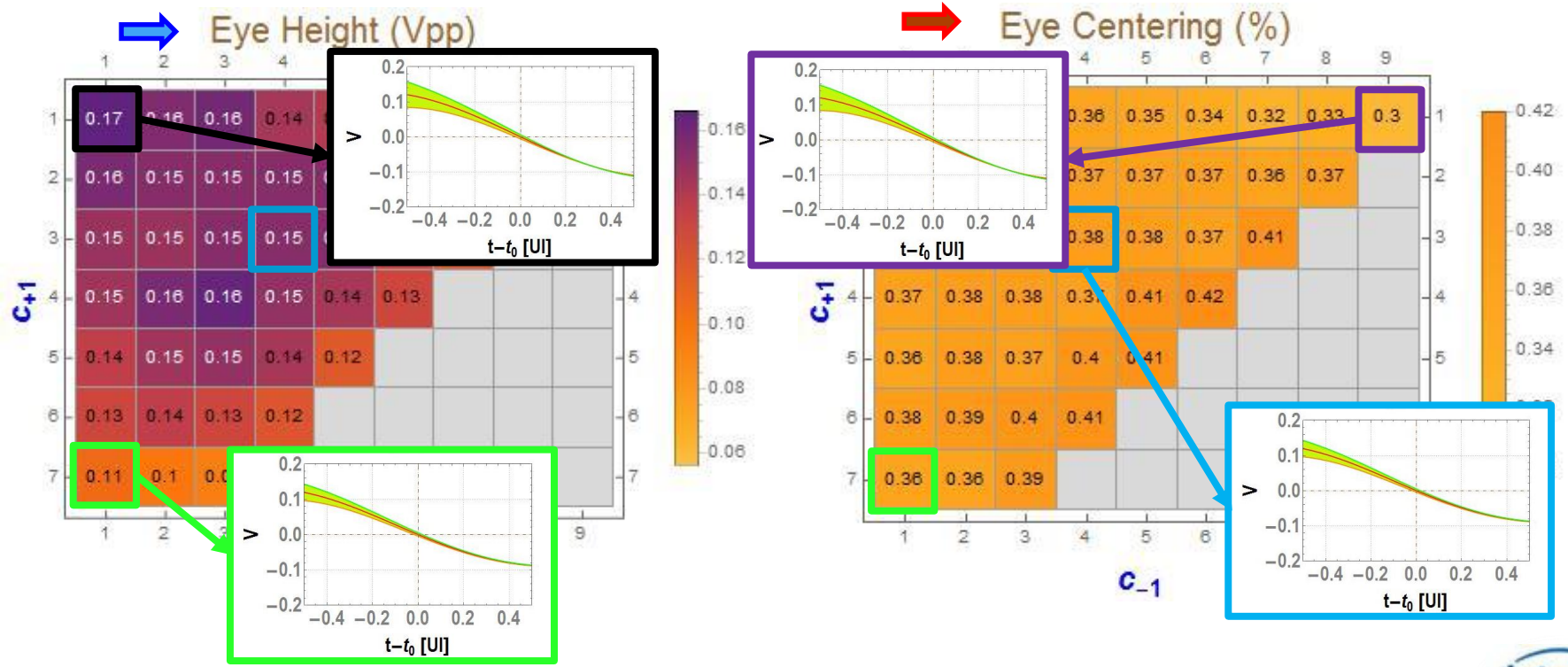
Increasing *pre-cursor* equalization affects the leading edge of the pulse response

- In an over-equalized condition, this causes the left part of the eye to shrink resulting in asymmetric timing margin

Increasing pre and post cursor equalization in a balanced way tends to yield a more symmetrical eye.

Timing Function Decomposition

(post-cursor impact)



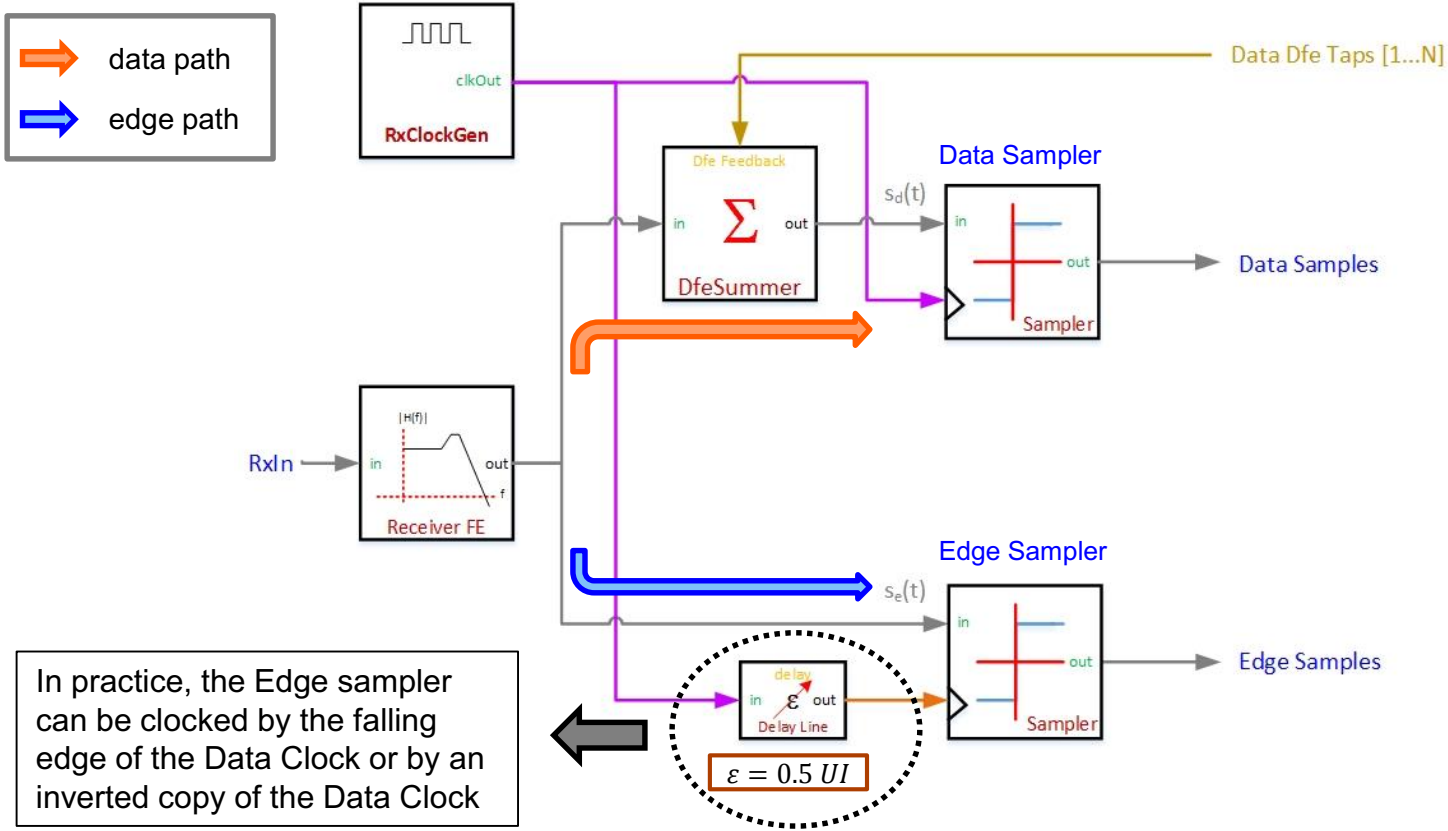
Ctle limitations

In practice, a receiver equalizer built exclusively around a Ctle, is effective only at moderate IL level.

- The data and edge samples are obtained from the same Ctle output signal so the overall equalization is a compromise between the need for good vertical eye opening and good jitter characteristics
- Higher IL levels require large amounts of high frequency boost (noise amplification problem)
- It is difficult to apply large level of high frequency boost w/o compromising the pulse response symmetry

Decoupling the Data and Edge paths

Dfe on Data path



In practice, the Edge sampler can be clocked by the falling edge of the Data Clock or by an inverted copy of the Data Clock

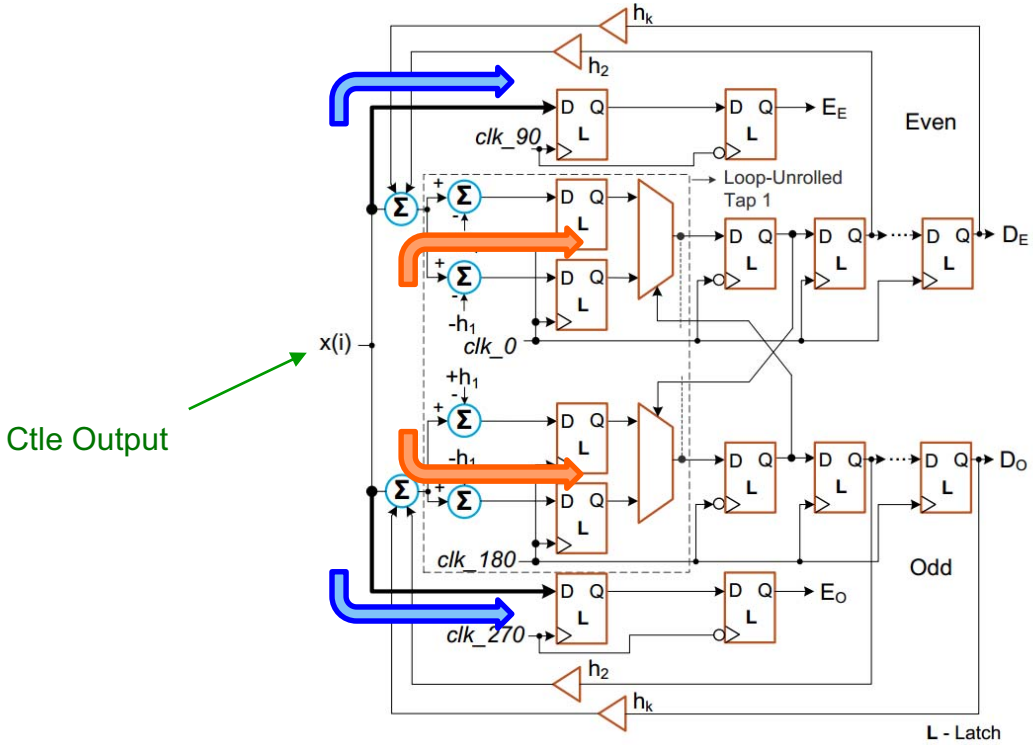
Hardware Complexity

	Summers	Clock Phases	Data Slicers	Edge Slicers	Error Slicers	
Alexander PD + Ctle	0	2N	N	N	0	
Alexander PD + Ctle + Data Dfe	**N**	**2N**	**N**	**N**	**N**	

The flexibility of the new topology comes at the price of higher hardware complexity due to the need to generate both *data* and *error* to support gain and dfe adaptation.

Note: In the table above, **N** indicates the number of interleaves.

Example



- Half-rate design
  - 2 Data interleaves
  - 2 Edge interleaves
  - 4 clock phases
- Speculative Data Dfe
  - 1 tap speculation
- No Dfe on Edge path

data path

edge path

C. Holdenried et al. “Performance of Edge Tap Decision Feedback Equalization methods for Wireline receivers”, 2014 IEEE 12th International New Circuits and Systems Conference (NEWCAS)

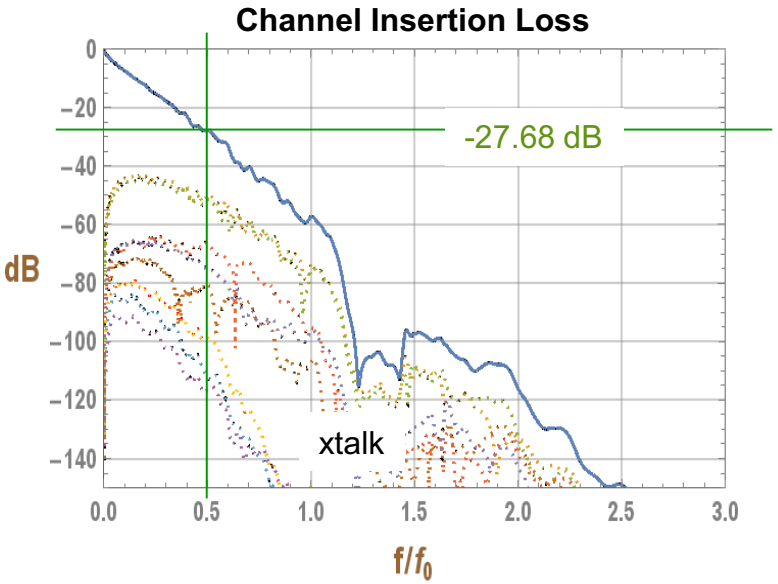
Observations

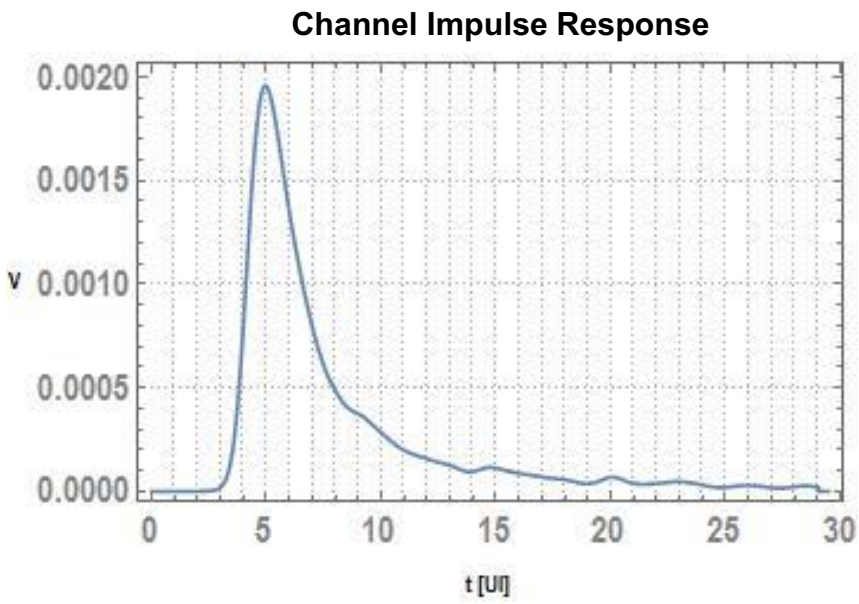
- Because the edge samples are still obtained from the Ctle output, the *timing function* and *sampling point* are not going to be different than in the previous case.

- However, the fact that the Dfe provides additional equalization on the data path, offers, in principle, the ability to tailor the Ctle to optimize the timing function and the Dfe to optimize the vertical opening

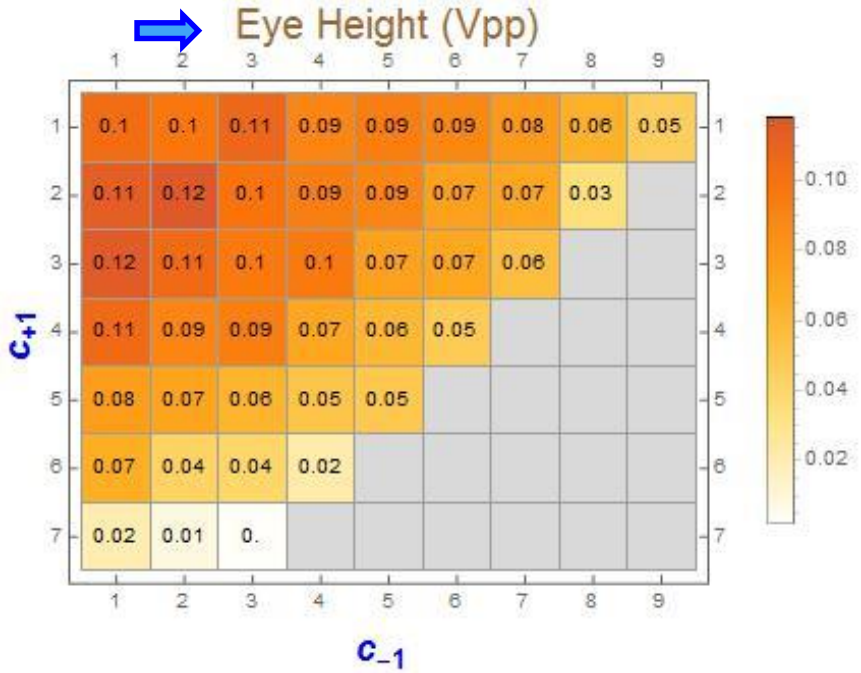
- This comes at the cost of additional hardware in the form of extra samplers and adaptation engines to tune the Dfe taps

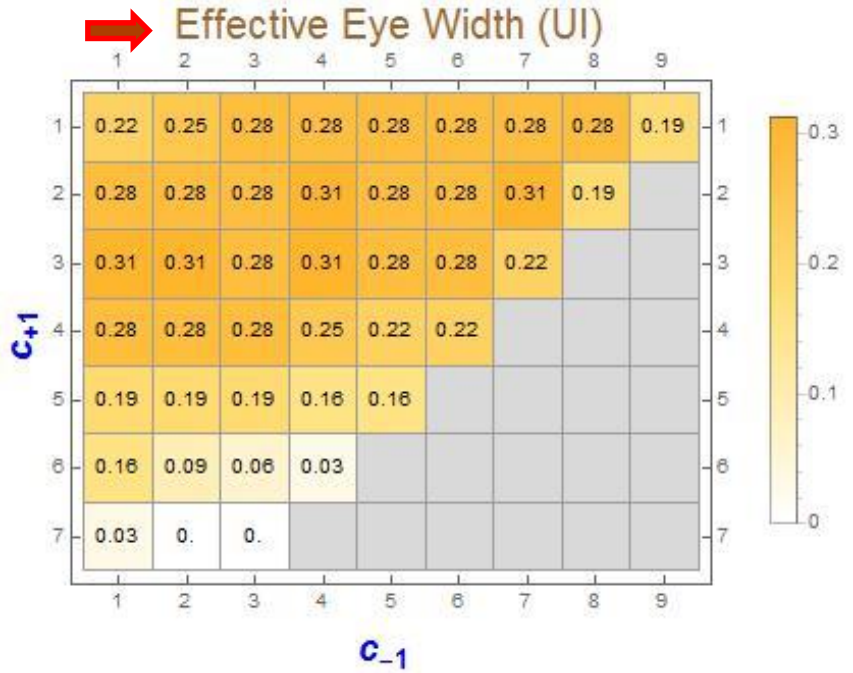
High IL


Channel Insertion Loss
0
−20
−40
−60
−80
−100
−120
−140
dB
-27.68 dB
xtalk
0.0
0.5
1.0
1.5
2.0
2.5
3.0
f/f0


Channel Impulse Response
0.0020
0.0015
0.0010
0.0005
0.0000
v
0
5
10
15
20
25
30
t [UI]

Margin Maps
(High IL)

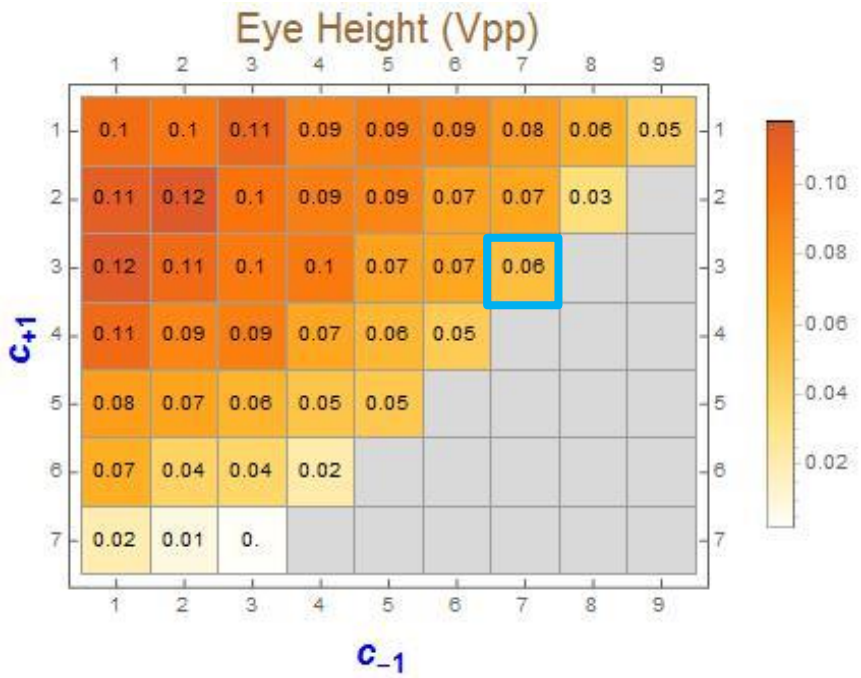


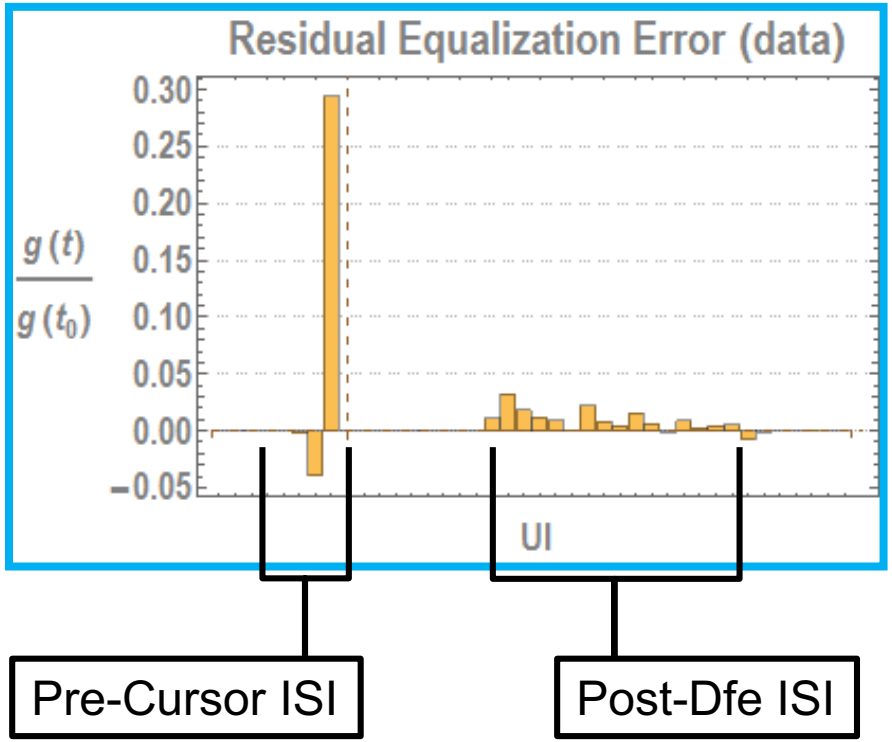


Further Improvements

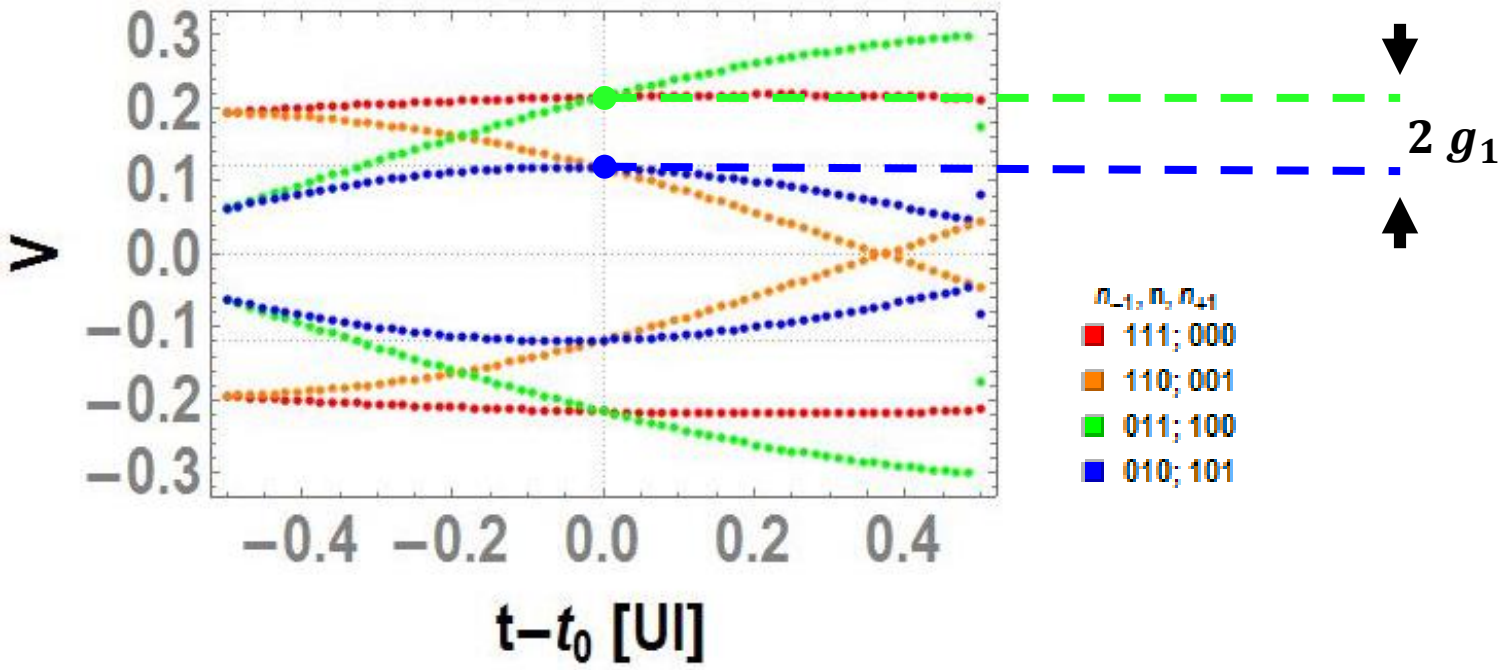
To identify possible improvements on the architecture it is necessary to look at what are the factors that limit the performance of the current topology.

Residual Equalization Error



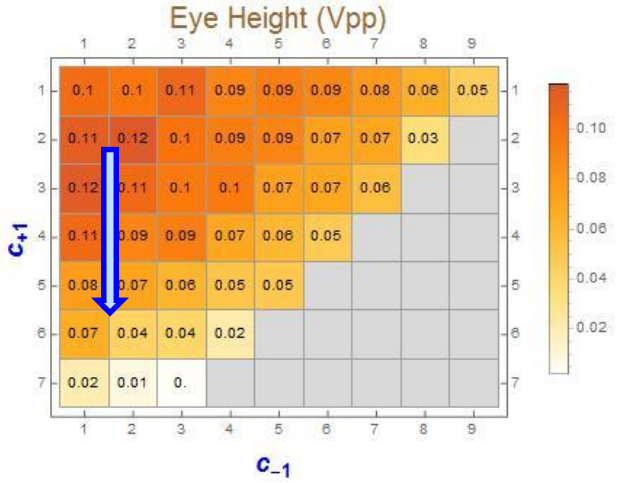


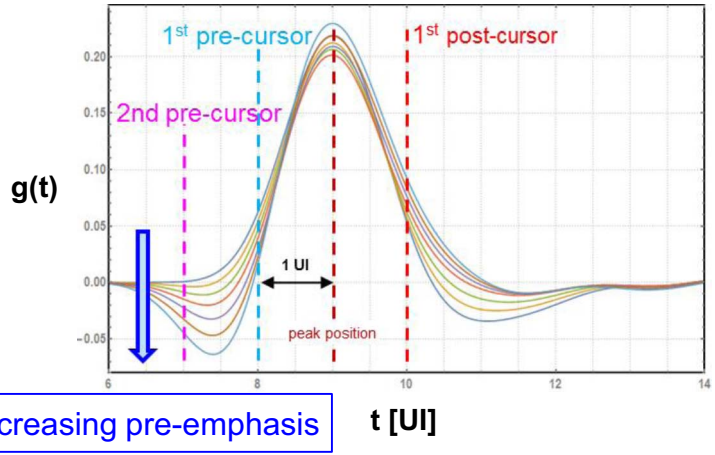
TxEq = {7,3}



- The magnitude of the first precursor ISI can be observed directly from the *tri-bit* eye diagram!

Pre-Cursor Impact



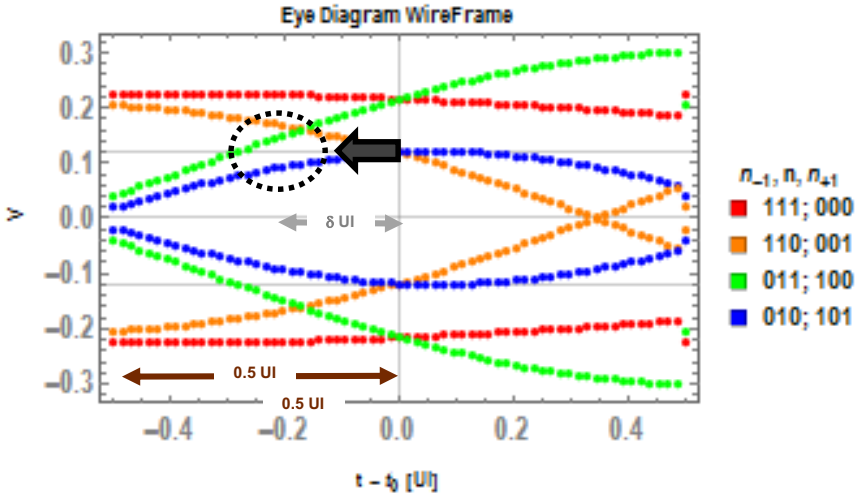


Increasing pre-emphasis

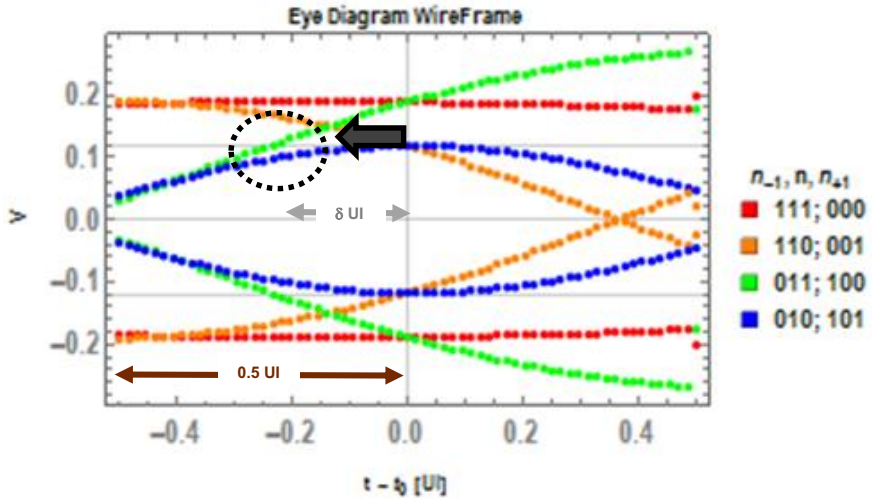
- The obvious remedy, increasing the TX pre-emphasis, is not optimal because increasing the TX pre-emphasis causes:
  - a reduction in the TX output
  - increase in 2nd pre-cursor caused by strong undershoot in the pulse response

Minimizing Pre-Cursor Impact

TxEq = {9,1}



TxEq = {7,3}

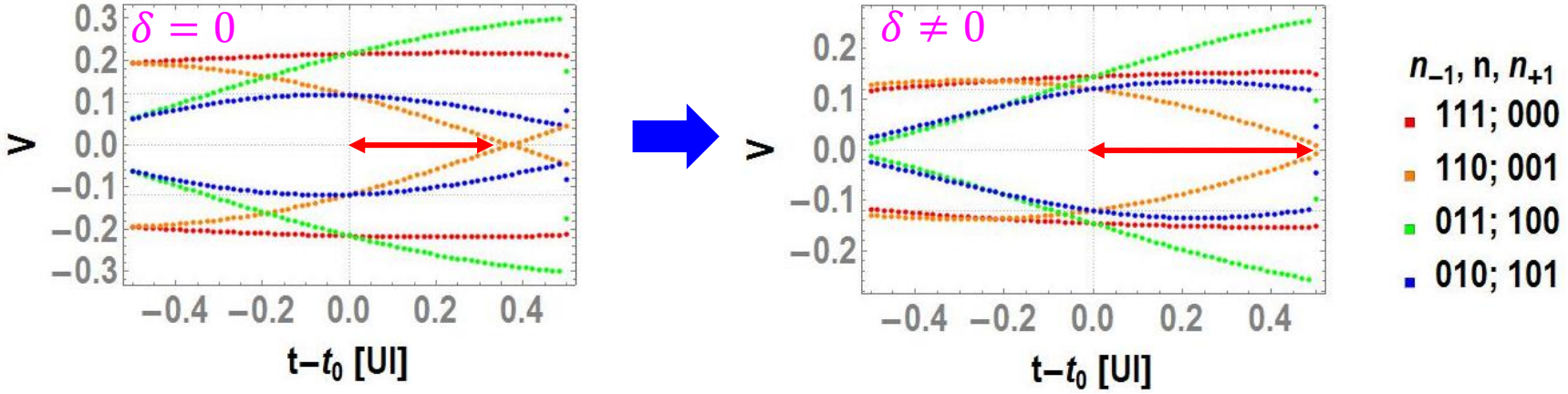


The first pre-cursor impact could be minimized if, somehow, we were able to shift the sampling point to the left so as to reduce its magnitude.

- We refer to this arrangement as the 'Modified Alexander' timing function.

Minimizing Pre-Cursor Impact

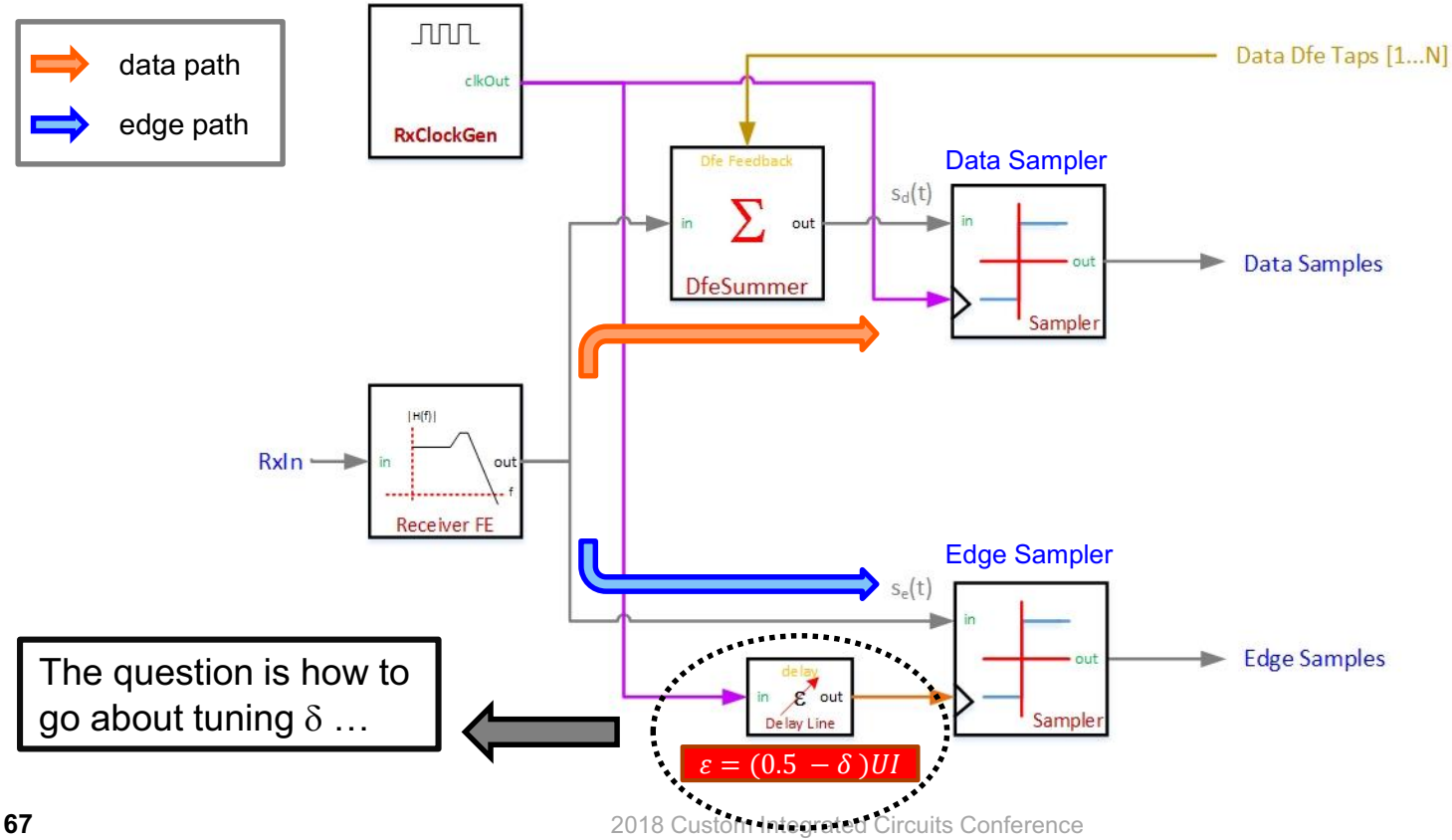
TxEq = {7,3}



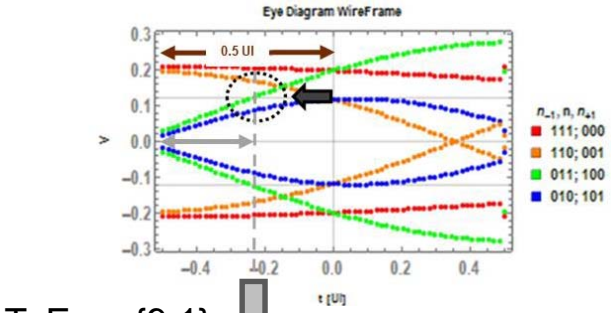
Besides reducing the pre-cursor effect, a sampling point shift to the left has the benefit of improving the horizontal eye opening symmetry.

Modified Alexander Timing Function

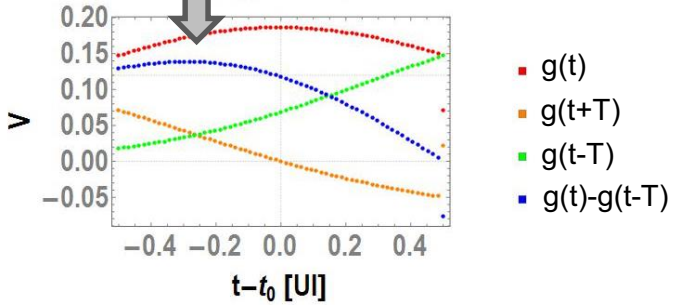
Dfe on Data path

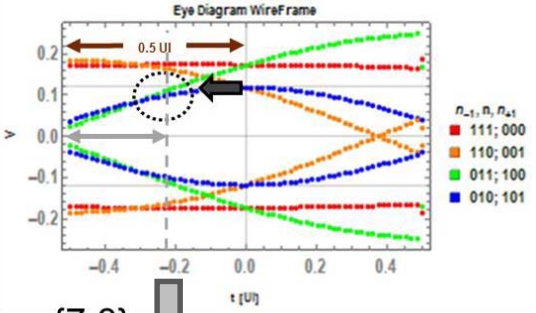


A Strategy for Tuning δ

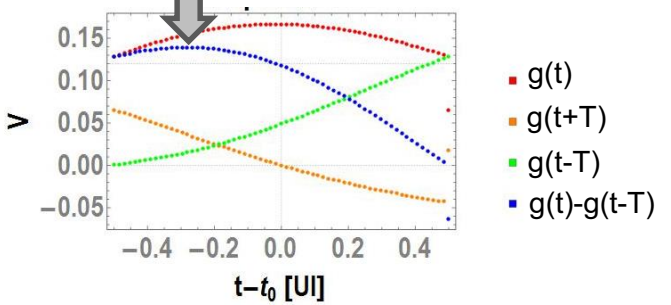


TxEq = {9,1}



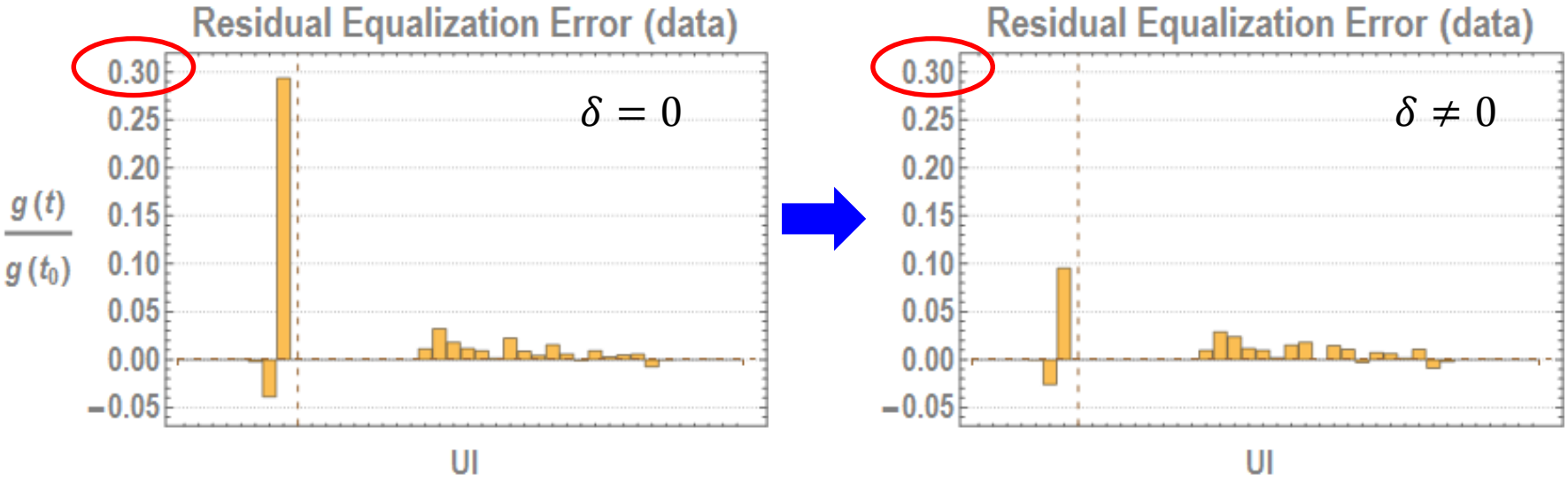


TxEq = {7,3}



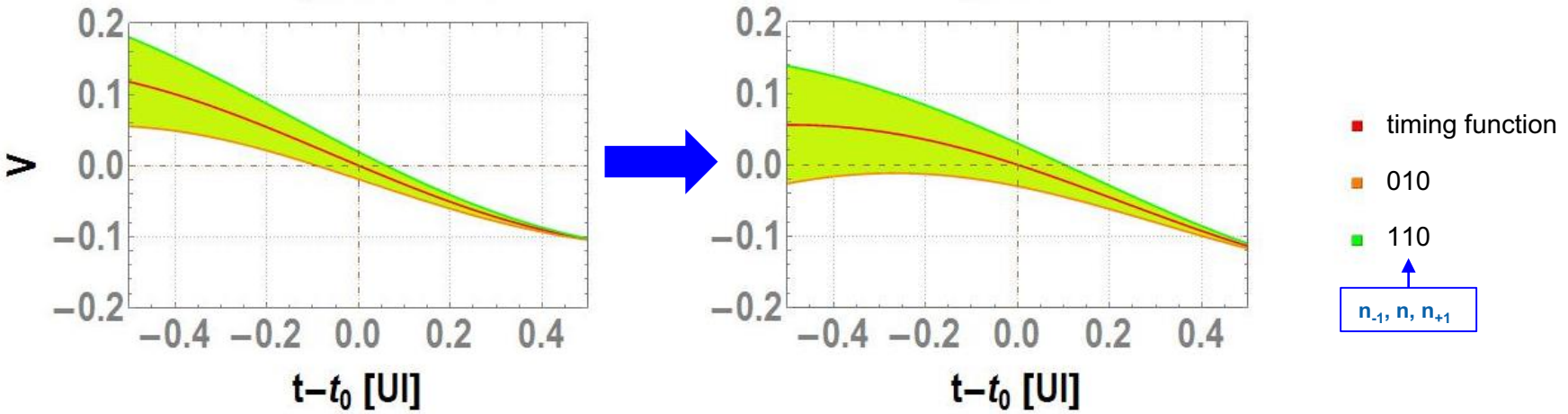
➢ The 'cursor'-'pre-cursor' value is a good optimization metric for tuning δ.

First Pre-Cursor ISI Reduction



- Tuning the data/edge delay can significantly reduce the impact of the first pre-cursor ISI.

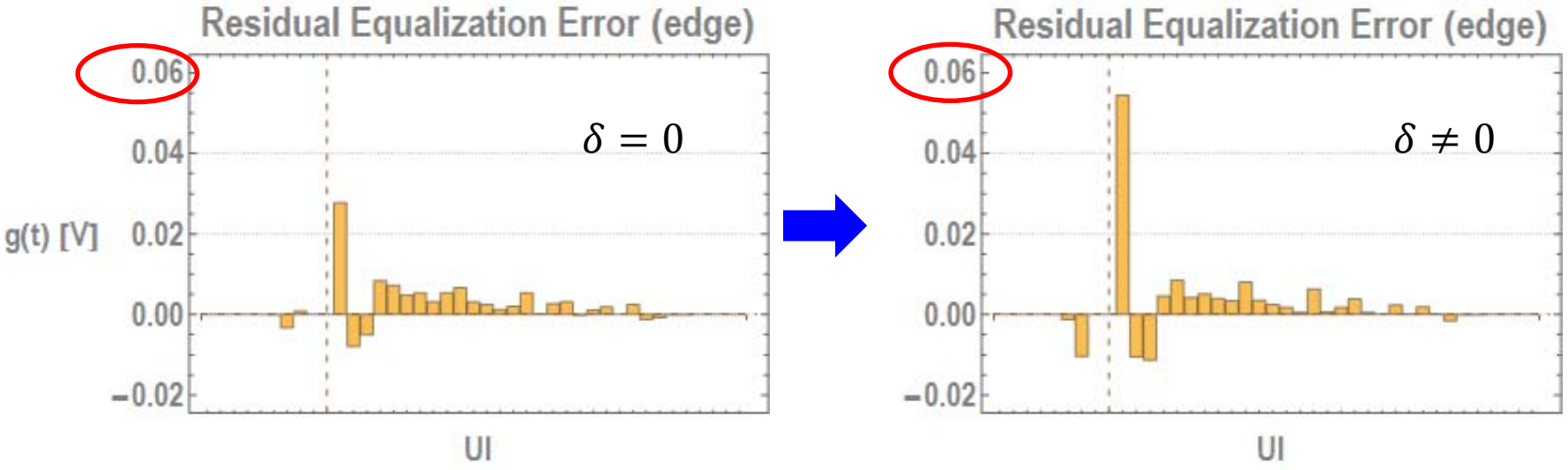
Side Effects



While the shifting of the sampling point to the left of its original position helps in reducing the impact of the first pre-cursor, it is accompanied by two undesired effects:

- Generalized increase in the post cursor ISI terms
- The effectiveness of the timing function is reduced because of the increased ISI on the edge

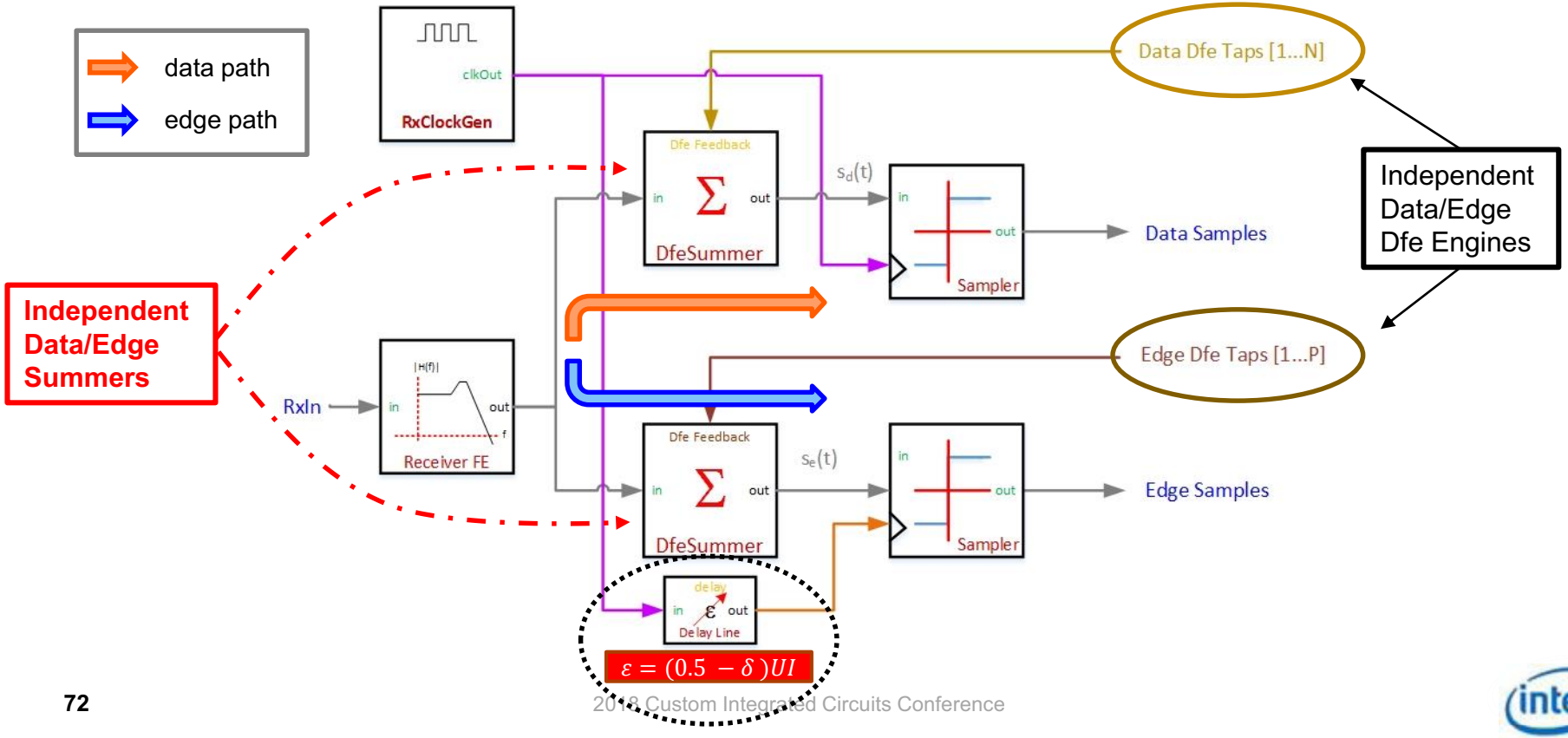
Residual Edge Equalization Error



- The best way to counteract the ISI increase on the edge samples is to *equalize* the edge path.

Modified Alexander Timing Function

Dfe on Data and Edge paths



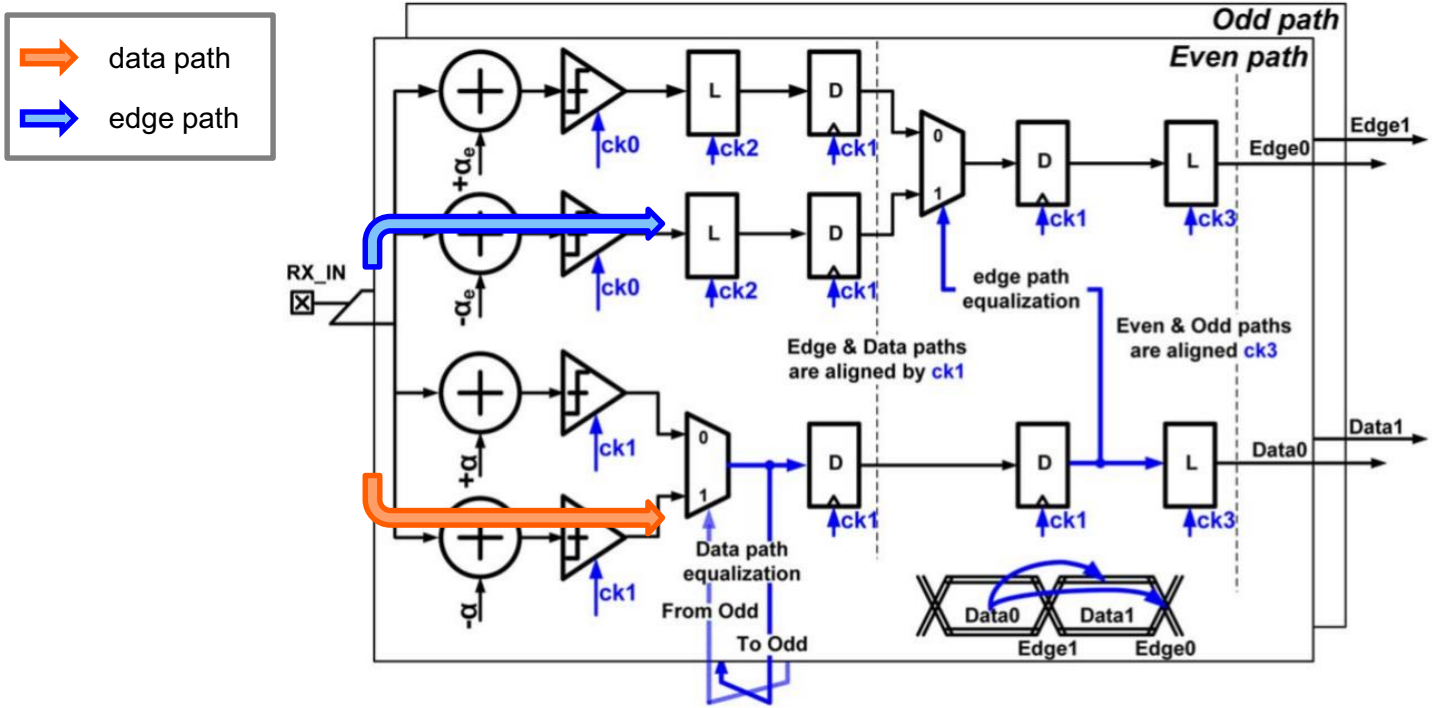
Hardware Complexity

	Summers	Clock Phases	Data Slicers	Edge Slicers	Error Slicers	
Alexander PD + Ctle	0	2N	N	N	0	
Alexander PD + Ctle + Data Dfe	N	2N	N	N	N	
Modified Alexander PD + Ctle + Data Dfe	N	2N	N	N	N	δ-tuning circuitry
Modified Alexander PD + Ctle + Data and Edge Dfe	**2 N**	**2N**	**N**	**N**	**N**	**δ-tuning circuitry**

- Performance refinements are unavoidably accompanied by an increase in hardware complexity.

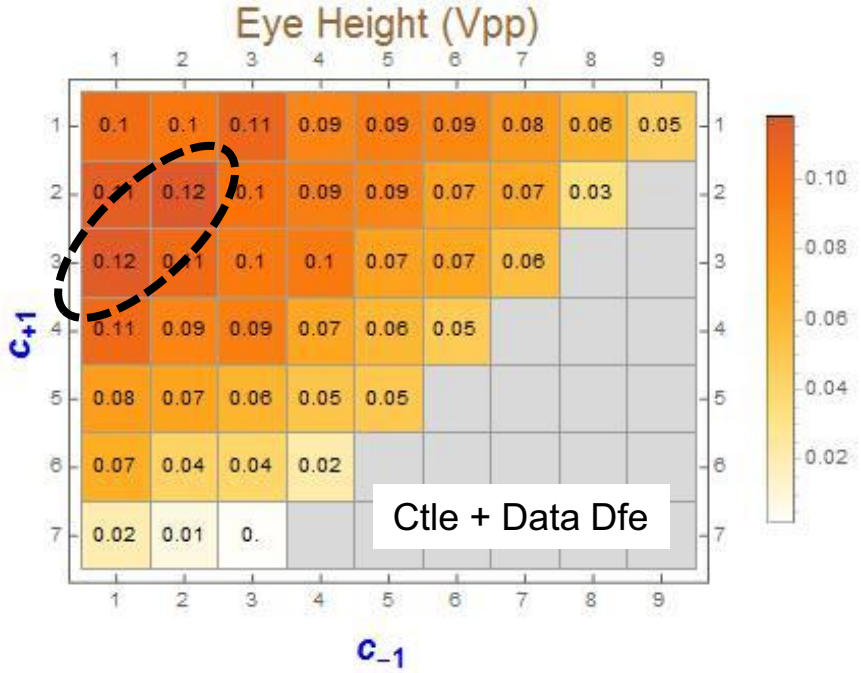
Note: In the table above, **N** indicates the number of interleaves.

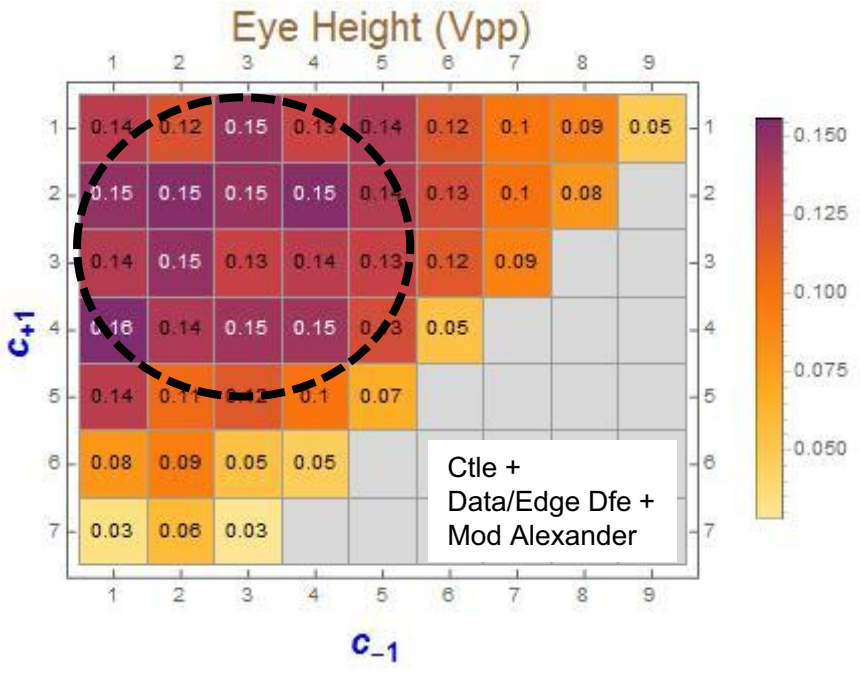
Example



KIM et al.: "VOLTAGE-MODE NEAR-GND RECEIVER WITH ONE-TAP DATA AND EDGE Dfe", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS, VOL. 61, NO. 6, JUNE 2014

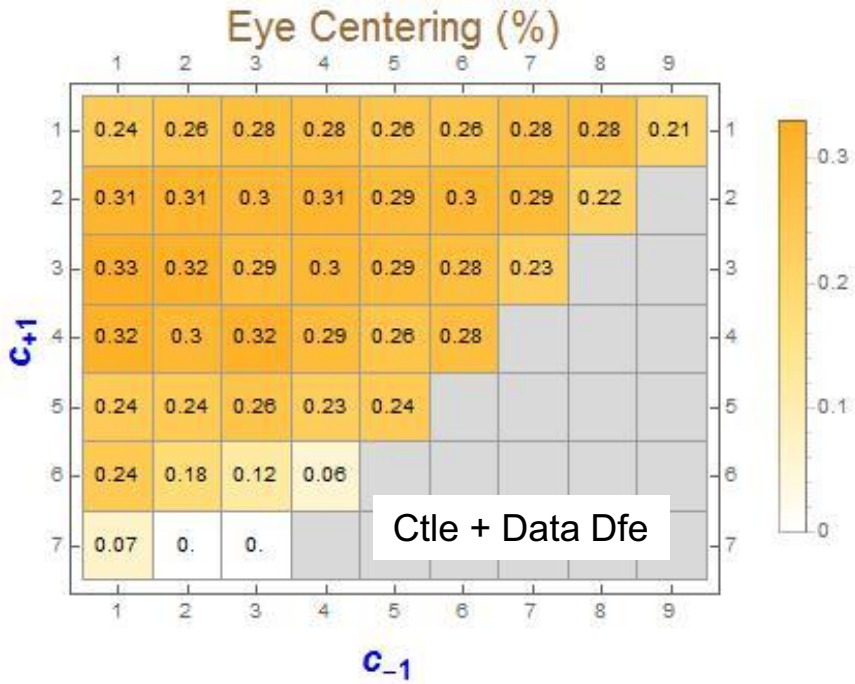
Margin Maps
(High IL)

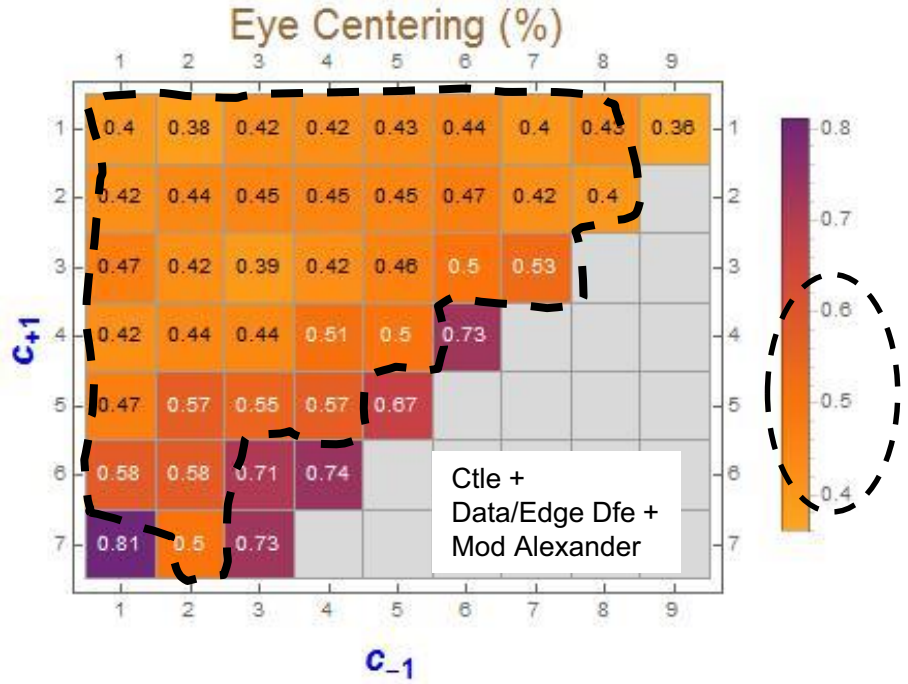




- The vertical eye opening greatly benefits form the use of the Modified Alexander timing function and Edge equalization (right figure).

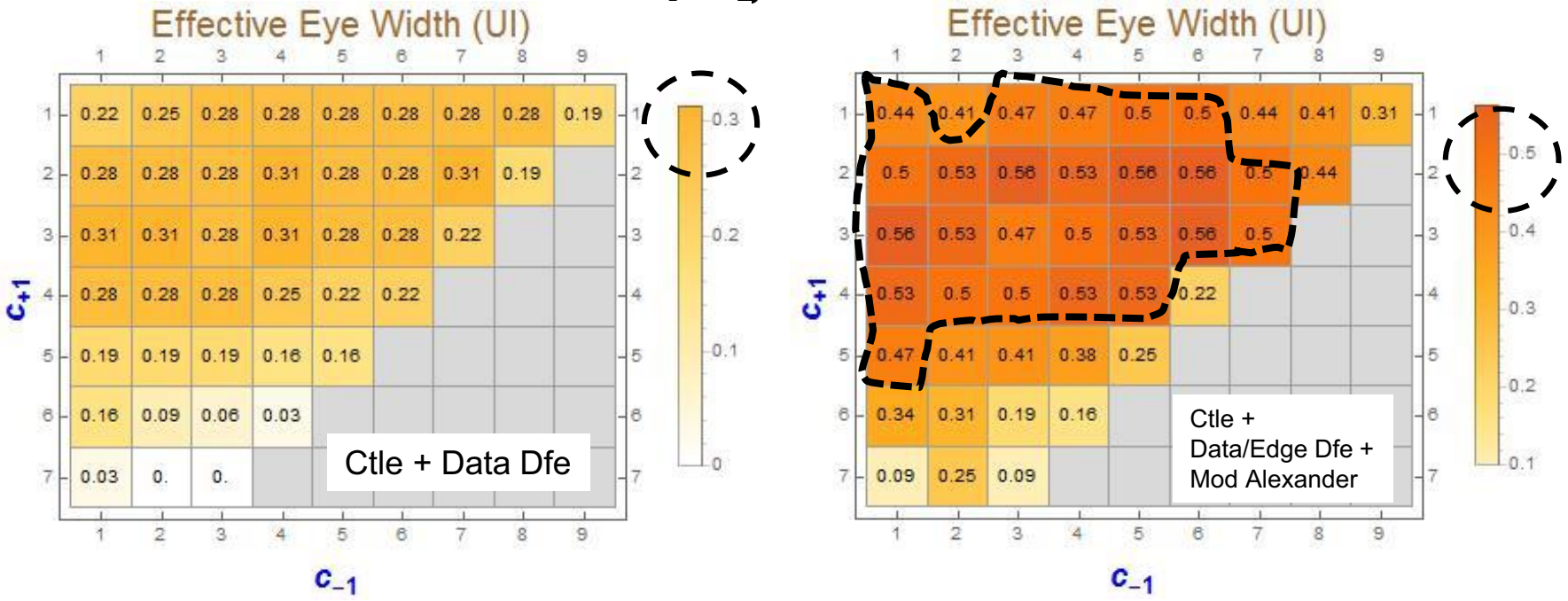
Margin Maps
(High IL)





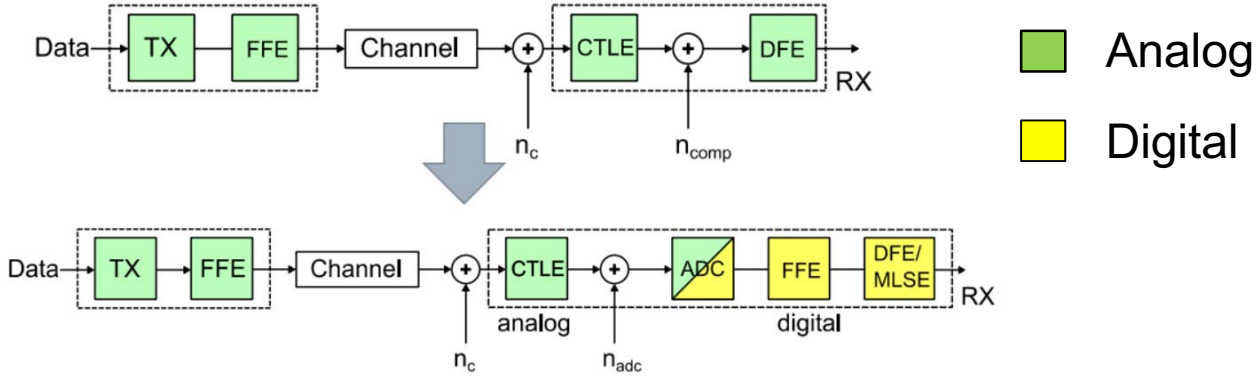
- The Modified Alexander timing function and Edge equalization (right figure) yield an horizontal eye opening that is inherently more symmetric.

Margin Maps
(High IL)



- The combined Modified Alexander timing function and Edge equalization (right figure) yield a noticeably wider horizontal eye.

Hardware Complexity



Toifl et al.: " Design Considerations for 50G+ Backplane Links",ESSCIRC Conference 2016: 42nd European Solid-State Circuits Conference

Hardware complexity, and its impact on area and power efficiency, is an important issue as HSIOs transition to higher data rates which require higher PAM orders or adopt an A/D topology.

With this in mind, we want to look at the possibility of eliminating the need for 2x oversampling required in an Alexander CDR.

MUELLER-MÜLLER PHASE DETECTOR

Background - 1

The Mueller-Müller Phase detector is an example of Baud Rate time detectors which extract the timing information directly from the samples of the signal s(t).

As a consequence the receiver requires a *single clock phase* to generate data samples and timing information.

This is particularly convenient in receiver architectures where the samples are processed by an A/D where the cost of an edge sampling clock would be prohibitive.

Background - 2

The Alexander Phase Detector timing function represents a discrete time *approximation* of the pulse response derivative based on samples of signal s(t) spaced 1 UI apart.

As it turns out, it is possible to generate, thru an appropriate linear combination of samples s_n, an approximation of the pulse response derivative. This time, however, the approximation is based on samples spaced **2 UI** apart.

Data Sample Values

As before, the sampled signal associated with the data clock can be written as:

$$s_n(t) = \underbrace{\sum_{k=-q}^{-1} g(t+kT)a_{n-k}}_{\text{pre-cursor}} + \underbrace{g(t)a_n}_{\text{cursor}} + \underbrace{\sum_{k=1}^{m-q-1} g(t+kT)a_{n-k}}_{\text{post-cursor}}$$

and let's consider the linear combination:

$$\boxed{f_n(t) = s_n(t)\,\hat{a}_{n-1} - s_{n-1}(t)\,\hat{a}_n}$$

where $\hat{a}_n$ represents the detector (slicer) output.

Timing Function

Assuming that the decisions are correct then $\hat{a}_n \equiv a_n$:

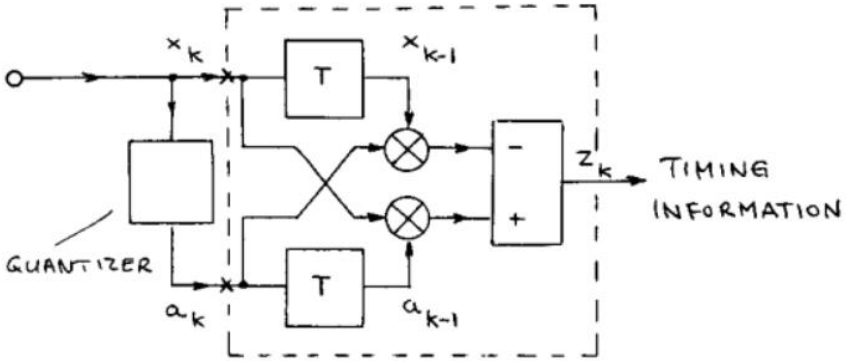
$$\boxed{f_n(t) = s_n\, a_{n-1} - s_{n-1}\, a_n}$$

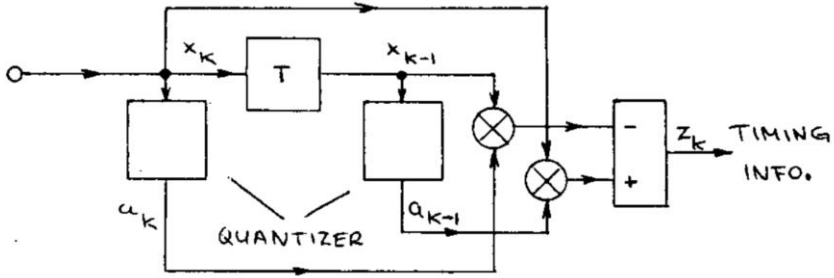
After some manipulations this reduces to:

$$\boxed{f(t) = \underbrace{g(t+T)}_{post-cursor} - \underbrace{g(t-T)}_{pre-cursor}}$$

- This is referred to as a Mueller-Müller **Type A** timing function.

Mueller-Müller Phase Detector



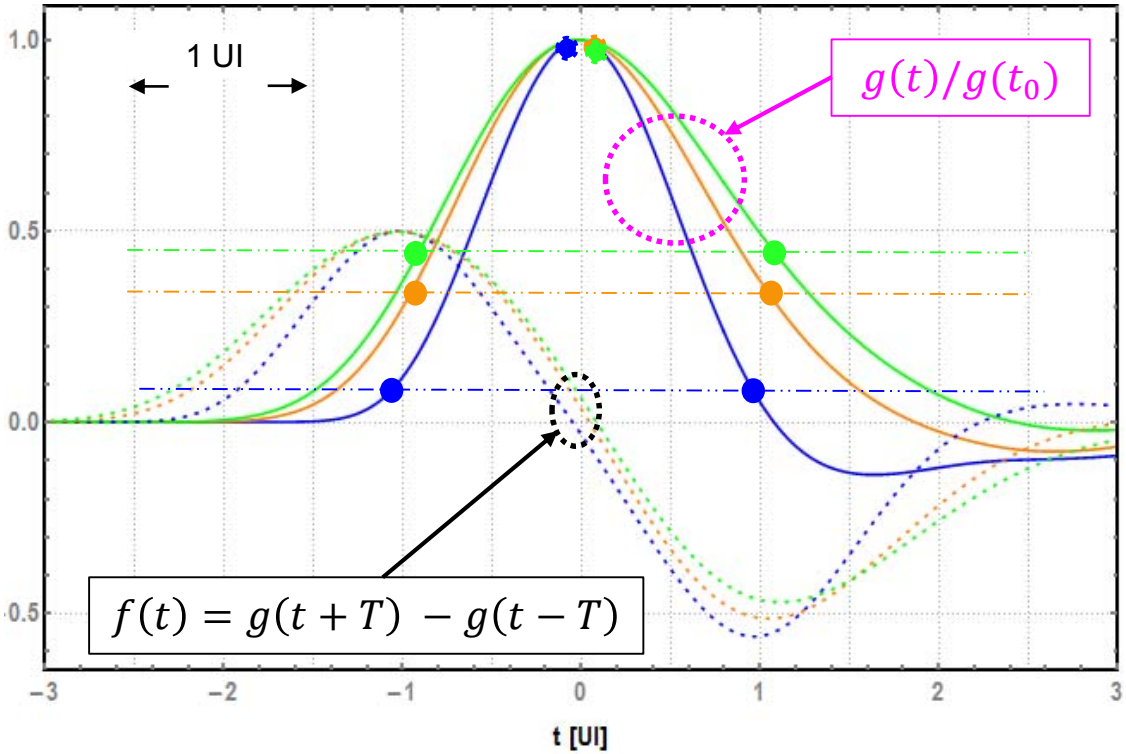


In practice, the analog sample x_k is replaced by the *sign,* $\boldsymbol{\eta}$*,* of the difference $x_k - g(0)$:

$$f_n(t) = \eta_n \, a_{n-1} - \eta_{n-1} \, a_n$$

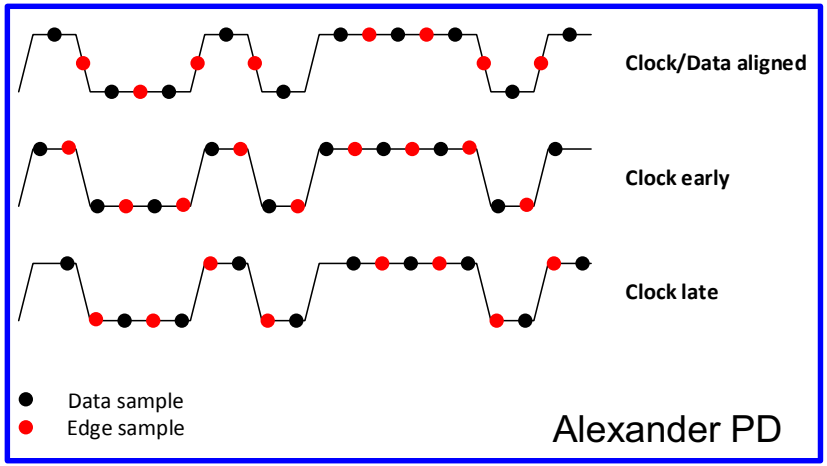
K. Mueller and M. Müller, “”Timing Recovery in Digital Synchronous Data Receivers”, IEEE Transactions on Communications

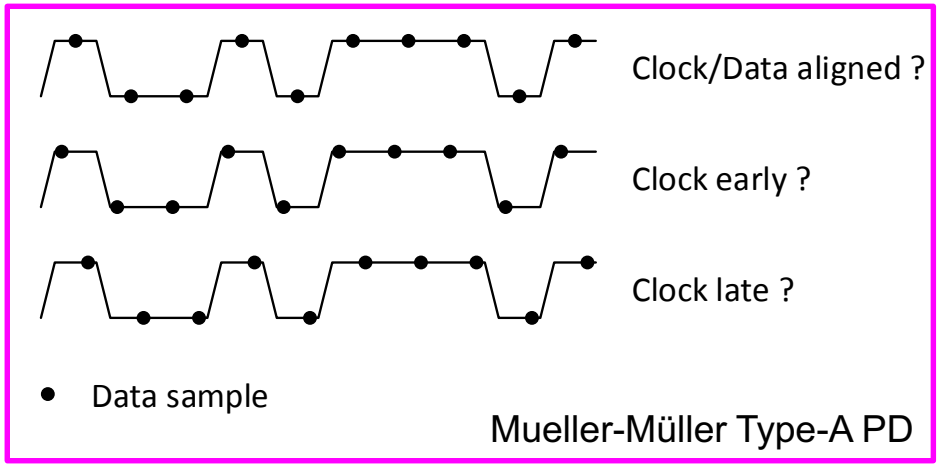
Mueller-Müller Type-A Timing Function



Symbol Rate Timing Recovery Limitations

Low ISI

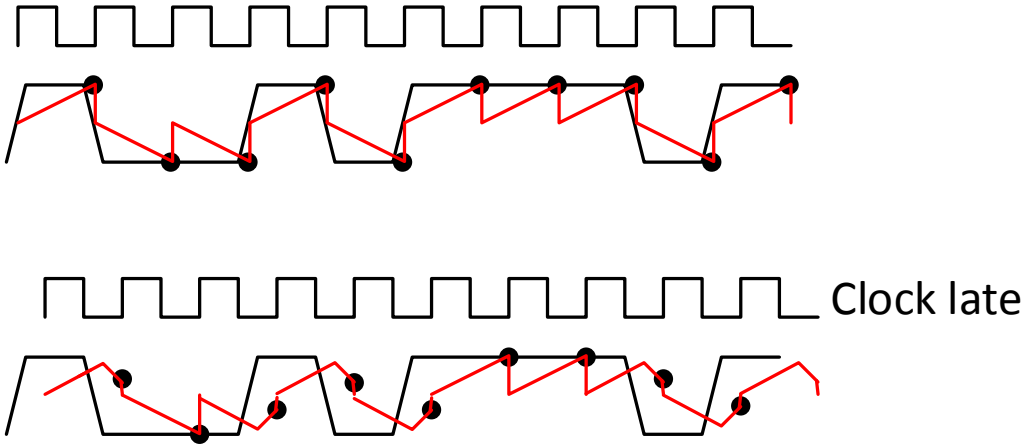




In a low ISI case, the only place where there is an amplitude variation which can be used to discriminate between sampling points is at the edge of the eye which is not desirable!

- Proper operation of the Mueller-Müller PD requires a certain amount of ISI in the channel to avoid locking at the edge of the eye.

Overcoming Symbol Rate Timing Recovery Limitations: ISI



- Data sample

- By *integrating the input signal* we establish a relationship between sample amplitude and sampling phase which we can use to estimate the phase error.

- This approach has the advantage to provide a solution which is more power efficient than one based on a linear summer but at the cost of additional ISI.

Symbol Rate Timing Recovery Limitations
Patterns

Since the timing estimate processed by the PD is defined by:

$$\boxed{f_n(t) = s_n(t)\,\hat{a}_{n-1} - s_{n-1}(t)\,\hat{a}_n}$$

it is apparent than a *constant* pattern or a 1T clock pattern will have little, if any, useful timing information.

The clock pattern limitation is particularly important in those cases where the protocol relies on clock pattern for in-band signaling since these cannot be eliminated by scrambling.

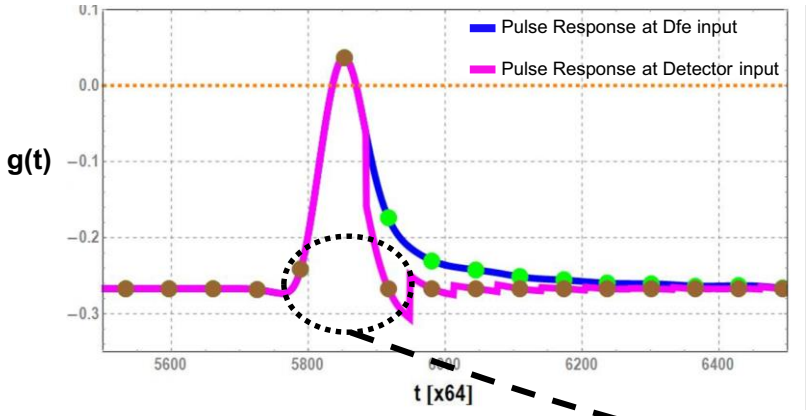
Mueller-Müller Type-A Timing Function

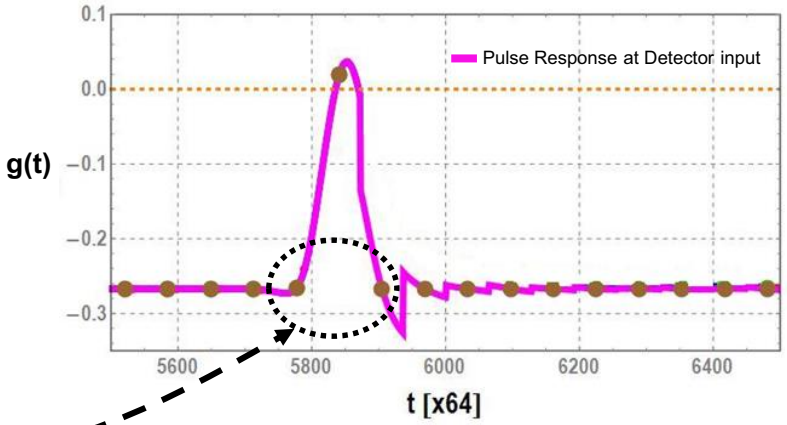
Dfe implications

Because the Dfe zero-forces the first post-cursor tap to zero (i.e. $g(t+T)=0$) the timing function reduces to:

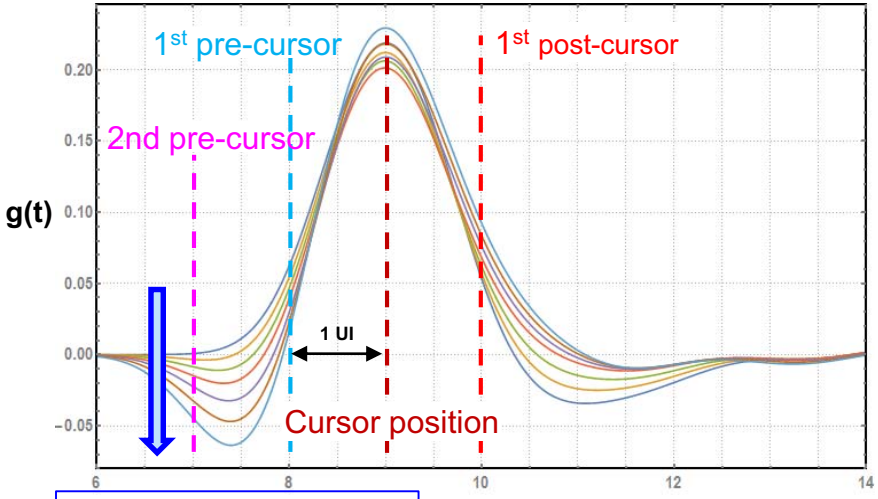
$$\boxed{0 = g(t-T)}$$

- This is referred to as the **Zero First Pre-cursor** criteria.

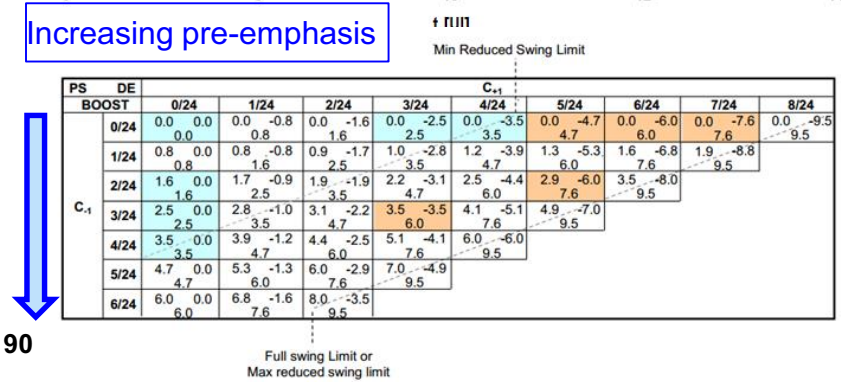




Zero First Pre-Cursor Implications



Increasing pre-emphasis



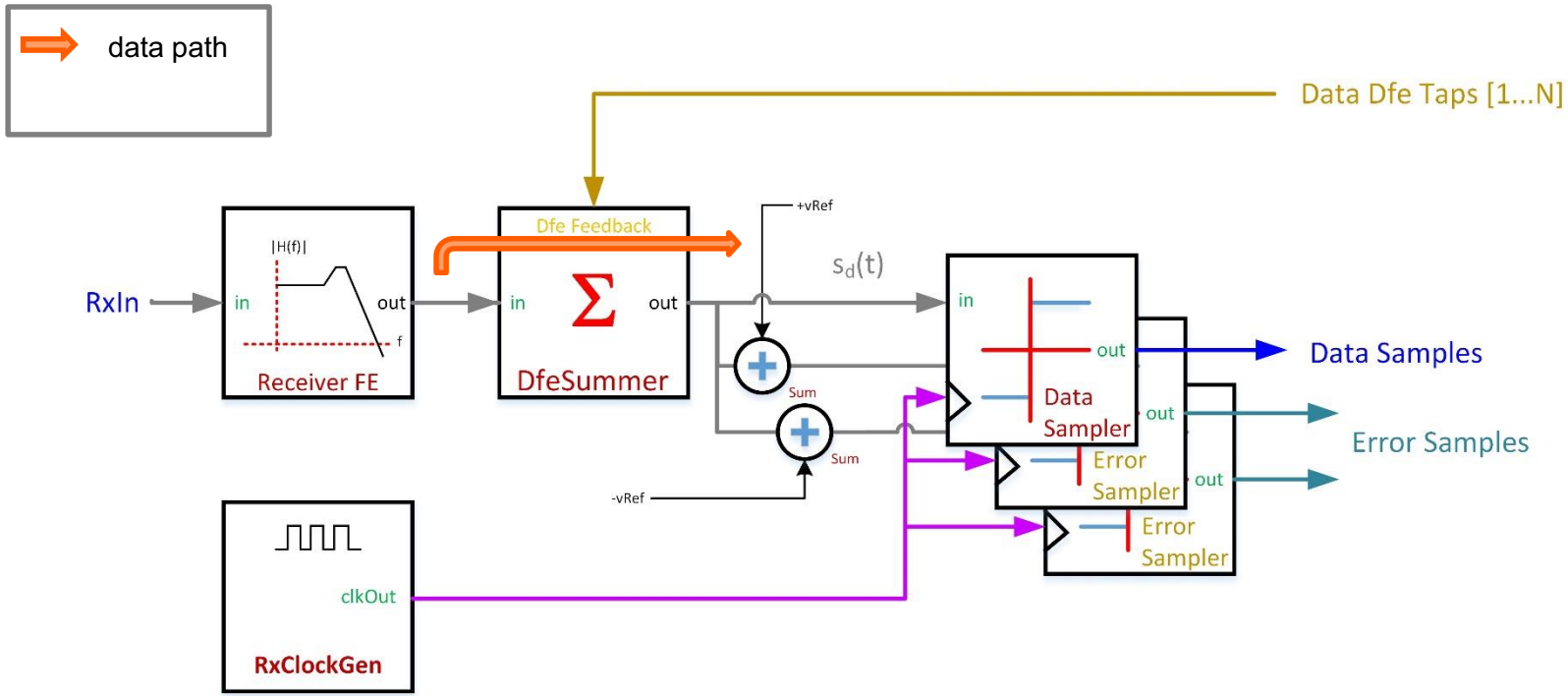
90

The zero first pre-cursor shifts the sampling position to the left of the pulse response peak

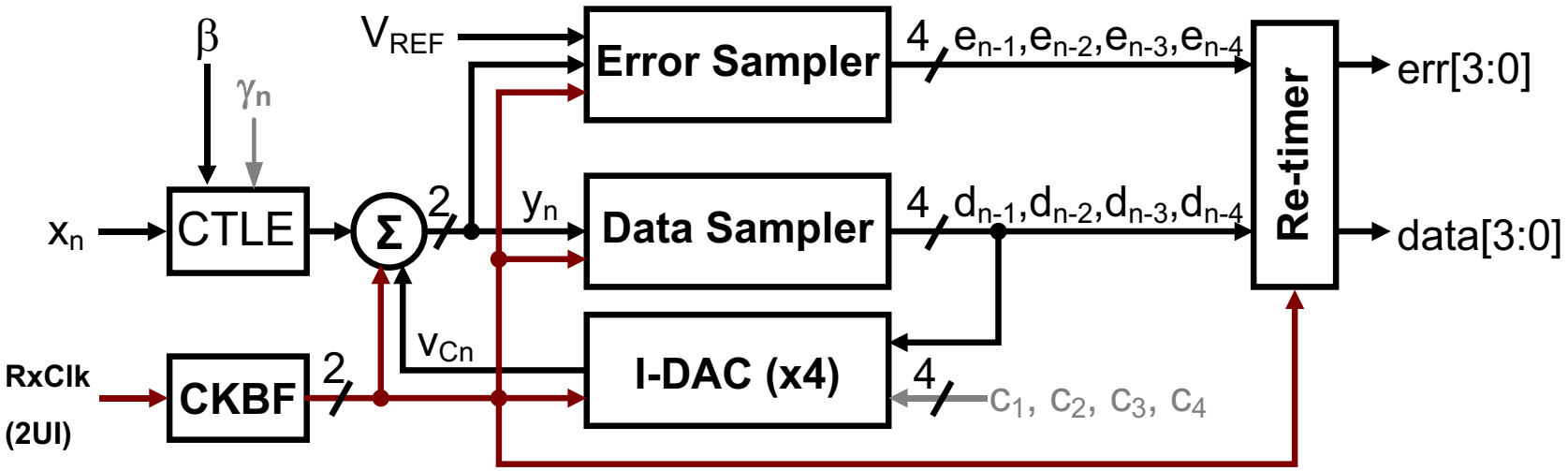
The shift is smaller for higher pre-emphasis but comes at a price of a 2nd pre-cursor ISI

- In conjunction with DFE, a stable sampling point requires a well defined zero crossing on the leading edge of the pulse response.

Mueller-Müller Phase Detector
With Dfe



Example



Spagna et al., "A 78mW 11.8-Gb/s Serial Link Transceiver with Adaptive RX Equalization and Baud-rate CDR in 32nm CMOS", ISSCC 2010

Hardware Complexity

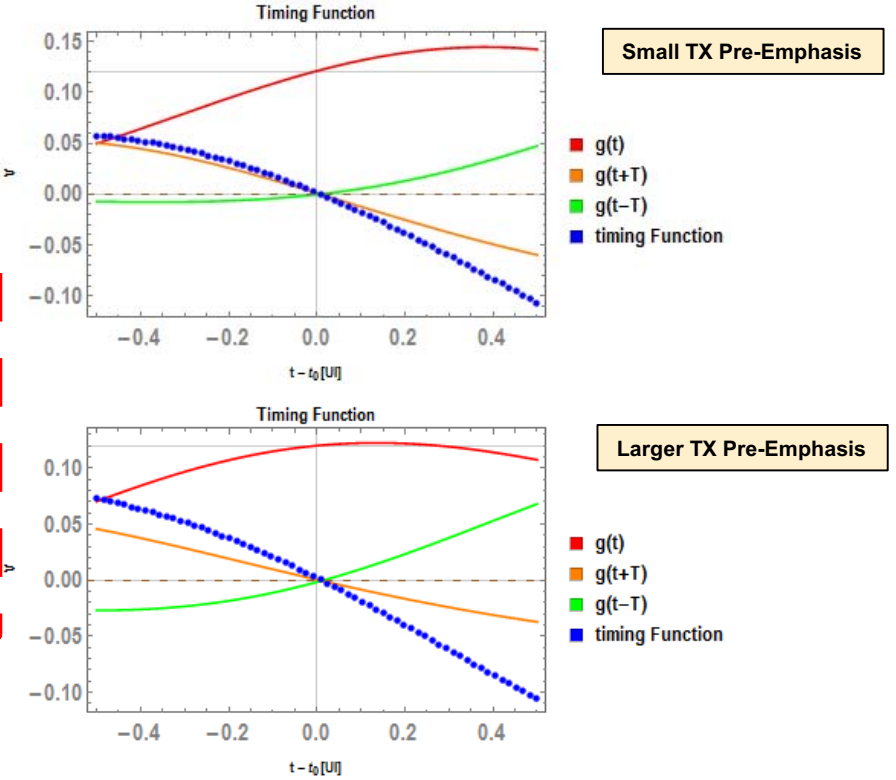
	Summers	Clock Phases	Data Slicers	Edge Slicers	Error Slicers	
Alexander PD + Ctle	0	2 N	N	N	0	
Alexander PD + Ctle + Data Dfe	N	2 N	N	N	N	
Modified Alexander PD + Ctle + Data Dfe	N	2 N	N	N	N	δ-tuning circuitry
Modified Alexander PD + Ctle + Data and Edge Dfe	2 N	2 N	N	N	N	δ-tuning circuitry
Mueller-Müller Type A + Dfe	**N**	**N**	**N**	**0**	**2 N**	

- Because the timing information is extracted from the data and error signals, the error samplers need be active at all time.
  - This **is not** a requirement for the Alexander PD.

Note: In the table above, **N** indicates the number of interleaves.

Data Filtering: $a_n \neq a_{n-1}$

a_{n+1}	a_n	a_{n-1}	$\eta_n \hat{a}_{n-1} - \eta_{n-1} \hat{a}_n$
x,+1,+1	+1,+1,+1	+1,+1,+1	0
x,+1,+1	+1,+1,+1	+1,+1,-1	0
x,-1,-1	-1,+1,+1	+1,+1,+1	0
x,-1,-1	-1,+1,+1	+1,+1,-1	0
x,+1,+1	+1,+1,-1	+1,-1,+1	-2 g_1 ⇐
x,+1,+1	+1,+1,-1	+1,-1,-1	-2 (g_1-g_{-1})
x,-1,-1	-1,+1,-1	+1,-1,+1	0
x,-1,-1	-1,+1,-1	+1,-1,-1	2 g_{-1}
x,+1,+1	+1,-1,+1	-1,+1,+1	2 g_{-1}
x,+1,+1	+1,-1,+1	-1,+1,-1	0
x,-1,-1	-1,-1,+1	-1,+1,+1	-2 (g_1-g_{-1})
x,-1,-1	-1,-1,+1	-1,+1,-1	-2 g_1 ⇐
x,+1,+1	+1,-1,-1	-1,-1,+1	0
x,+1,+1	+1,-1,-1	-1,-1,-1	0
x,-1,-1	-1,-1,-1	-1,-1,+1	0
x,-1,-1	-1,-1,-1	-1,-1,-1	0



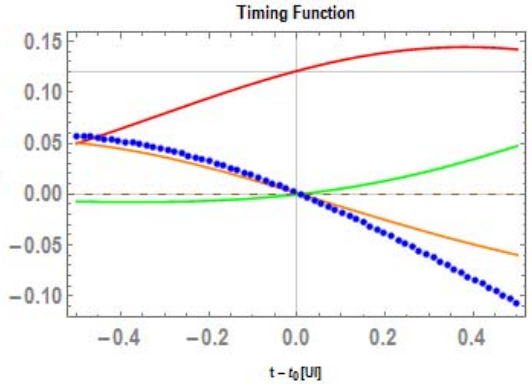
⇒ Potentially weak timing feedback cause by poor slope

Data Filtering

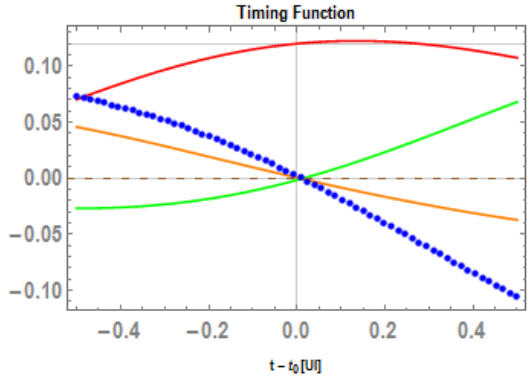
If the receiver is built by multiple interleaves (N > 1) we can use data filtering to periodically turn off the error samplers in some of the interleaves to reduce power consumption.

Data Filtering: $a_{n+1} \neq a_{n-1}$

a_{n+1}	a_n	a_{n-1}	$\eta_n(\hat{a}_{n-1} - \hat{a}_{n+1})$
x,**+1**,+1	+1,+1,+1	+1,**+1**,+1	**0**
x,**+1**,+1	+1,+1,+1	+1,**+1**,−1	**0**
x,**−1**,−1	−1,+1,+1	+1,**+1**,+1	**-2 (g_1-g_{-1})**
x,**−1**,−1	−1,+1,+1	+1,**+1**,−1	**-2 (g_1-g_{-1})**
x,**+1**,+1	+1,+1,−1	+1,**−1**,+1	**-2 (g_1-g_{-1})**
x,**+1**,+1	+1,+1,−1	+1,**−1**,−1	**-2 (g_1-g_{-1})**
x,**−1**,−1	−1,+1,−1	+1,**−1**,+1	**0**
x,**−1**,−1	−1,+1,−1	+1,**−1**,−1	**0**
x,**+1**,+1	+1,−1,+1	−1,**+1**,+1	**0**
x,**+1**,+1	+1,−1,+1	−1,**+1**,−1	**0**
x,**−1**,−1	−1,−1,+1	−1,**+1**,+1	**-2 (g_1-g_{-1})**
x,**−1**,−1	−1,−1,+1	−1,**+1**,−1	**-2 (g_1-g_{-1})**
x,**+1**,+1	+1,−1,−1	−1,**−1**,+1	**-2 (g_1-g_{-1})**
x,**+1**,+1	+1,−1,−1	−1,**−1**,−1	**-2 (g_1-g_{-1})**
x,**−1**,−1	−1,−1,−1	−1,**−1**,+1	**0**
x,**−1**,−1	−1,−1,−1	−1,**−1**,−1	**0**

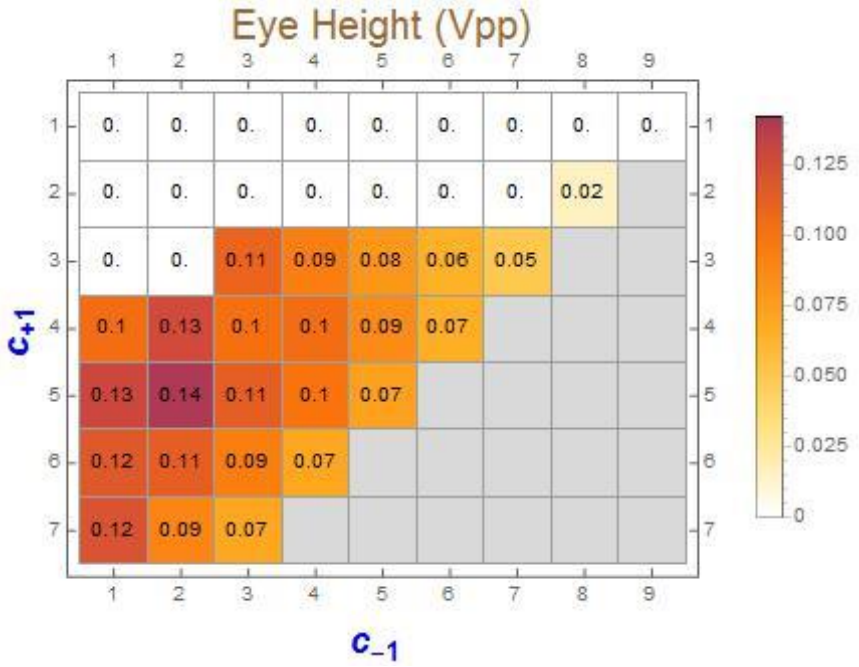


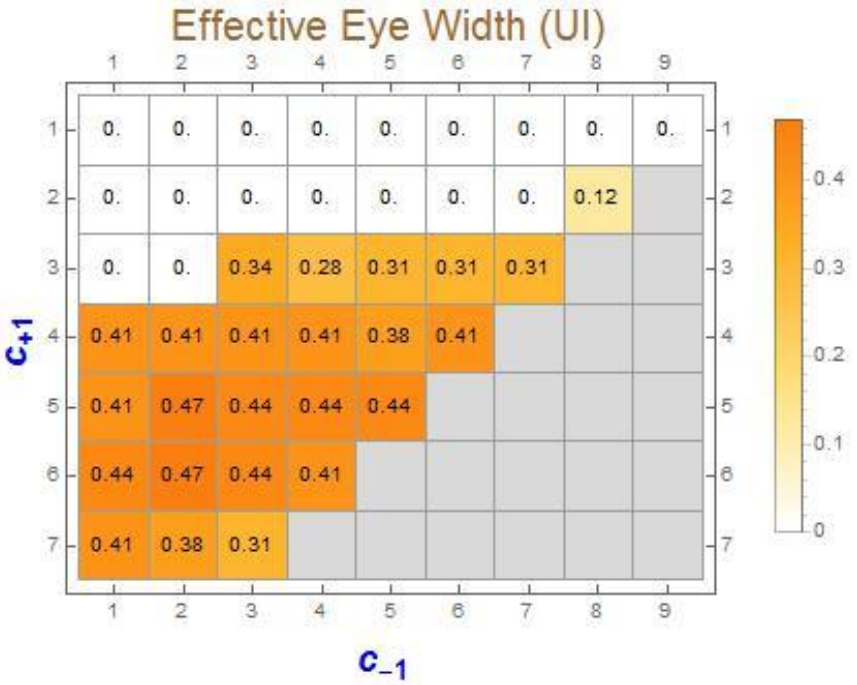
Small TX Pre-Emphasis



Larger TX Pre-Emphasis

Margin Maps
(High IL)





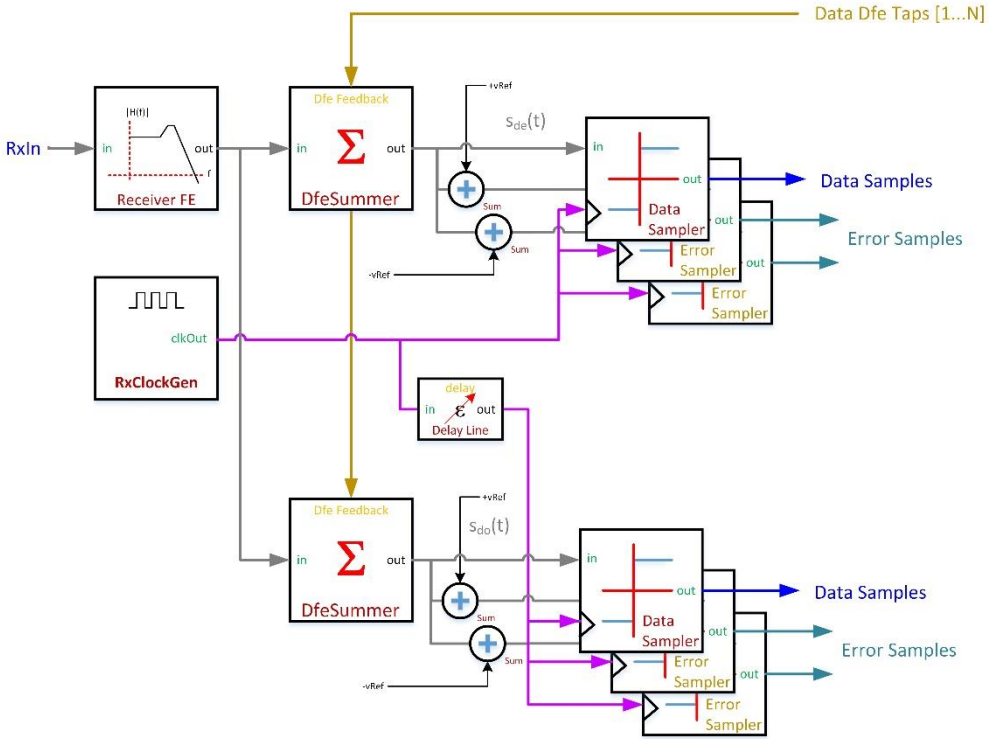
Further Improvements

As in the case of the Alexander Phase Detector performance improvements over this basic configuration can be obtained albeit a the cost of some additional circuitry.

- In particular it can shown that it possible to define a sampling point equivalent to the one obtained with the modified Alexander function described in the previous section.

While the scope of these improvements exceeds the scope of this tutorial, it is good to conclude presenting a hybrid architecture which encapsulates the benefits of the two architectures presented.

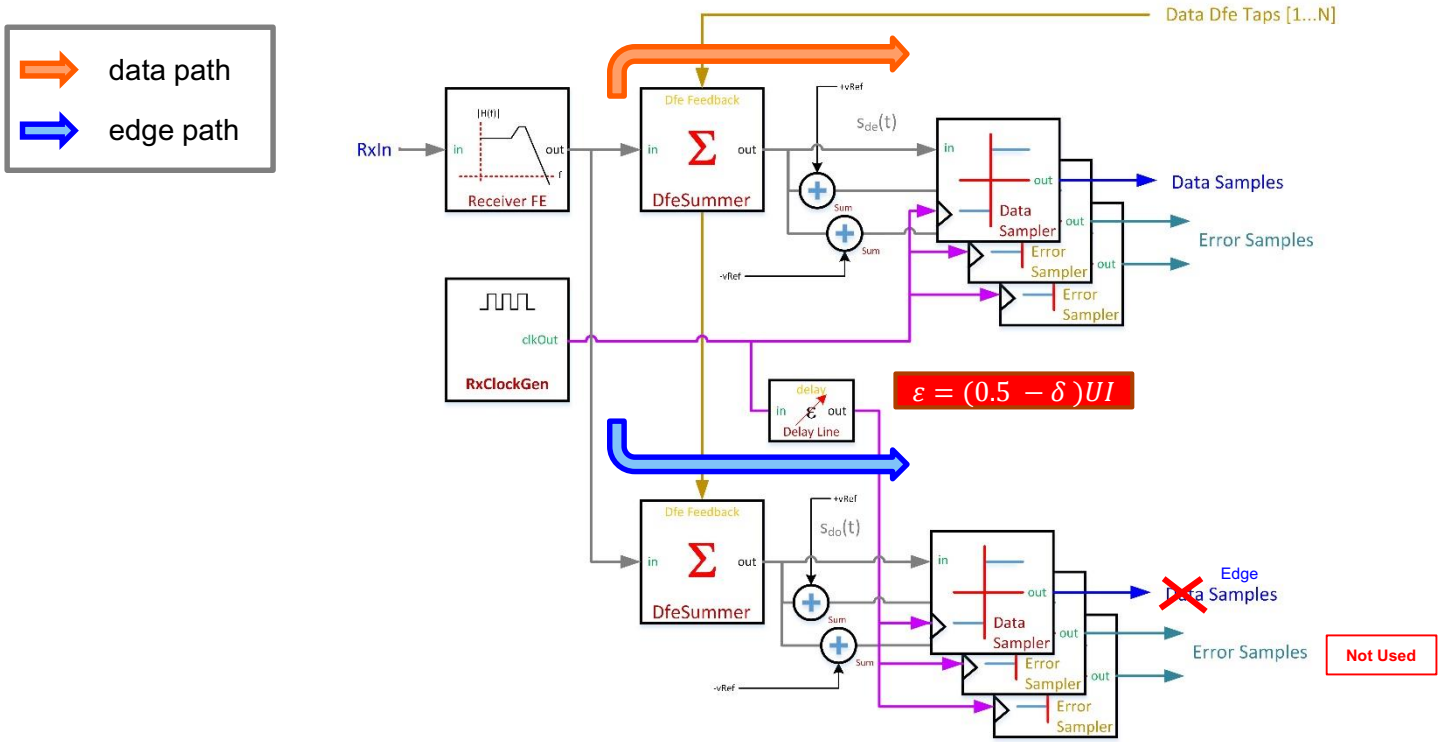
Hybrid Architecture



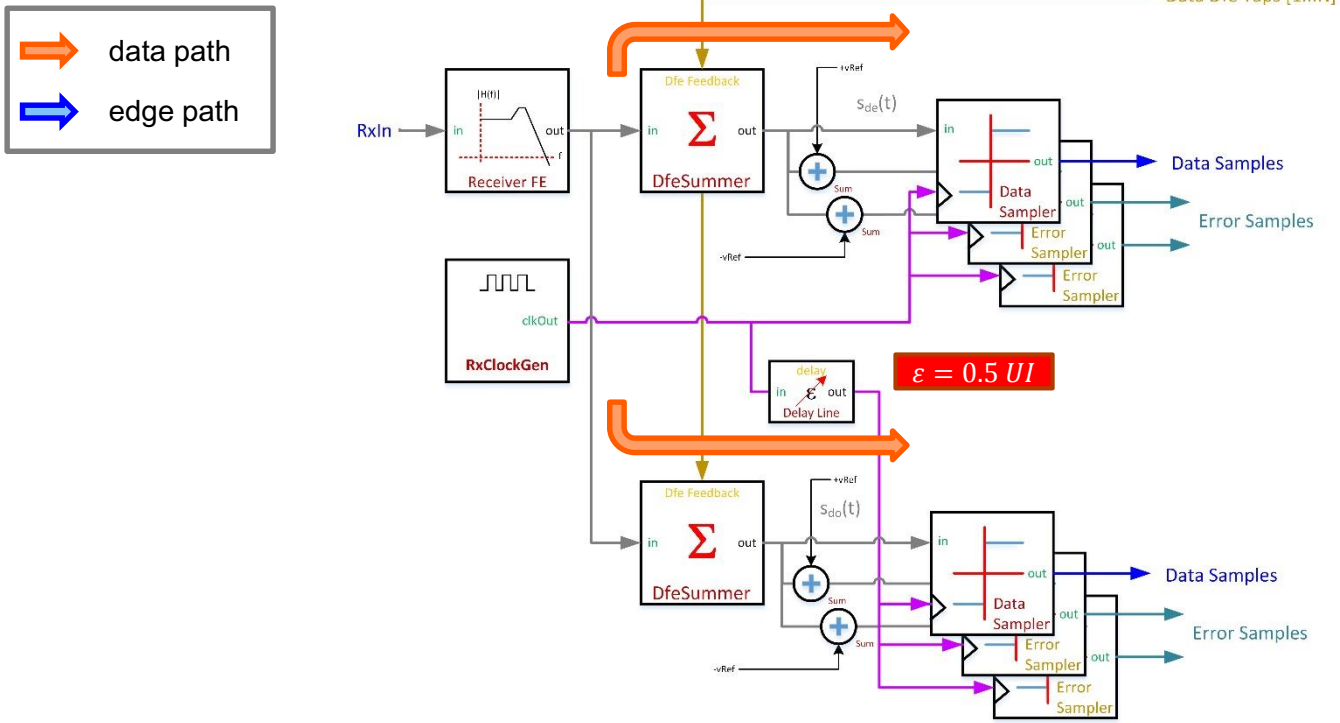
This is referred to as a ‘Hybrid Architecture’ because it operates in **full rate** Alexander PD mode at Low Data Rates and **half rate** Mueller-Müller PD mode at High Data Rates.

This arrangement obviates to the limitations of the Mueller-Müller PD at low IL (and Low Data Rate) by using the Alexander PD (albeit at the price of operating the receiver at full rate).

Hybrid Architecture
Full Rate Alexander Mode



Hybrid Architecture
Half Rate Mueller-Müller Mode



SUMMARY

Summary

With shrinking unit intervals (UI) the optimal horizontal centering of the reference sampling point becomes an important consideration for the choice of the phase detector and associated timing recovery method.

Optimization of the timing recovery and link equalization can be greatly enhanced by the proper choice of timing function which, in conjunction, with proper equalization techniques will greatly improve the system performance.

Summary

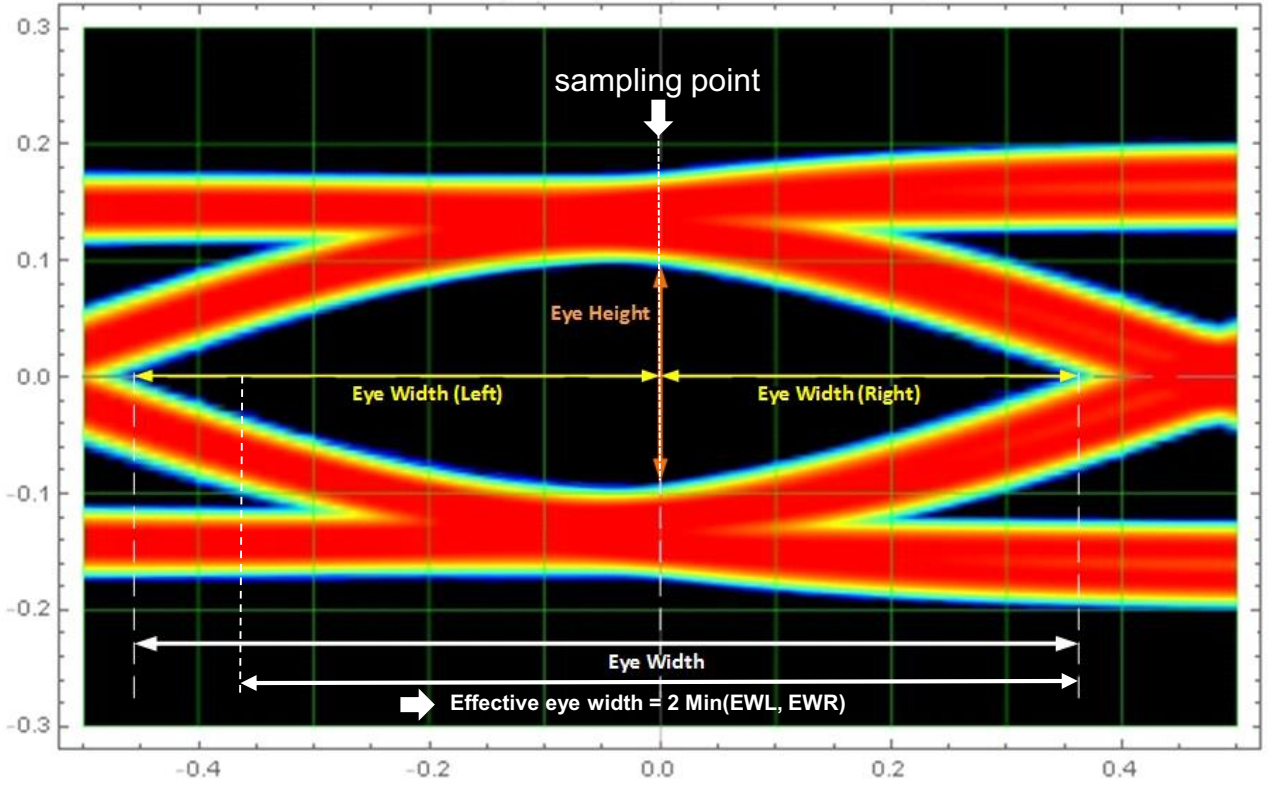
This tutorial has presented an overview of some of the considerations involved in the choice of phase detectors for good horizontal centering as well as good dynamic tracking characteristics.

This tutorial has also introduced some analysis techniques to make quantitative comparisons to enable architectural decisions.

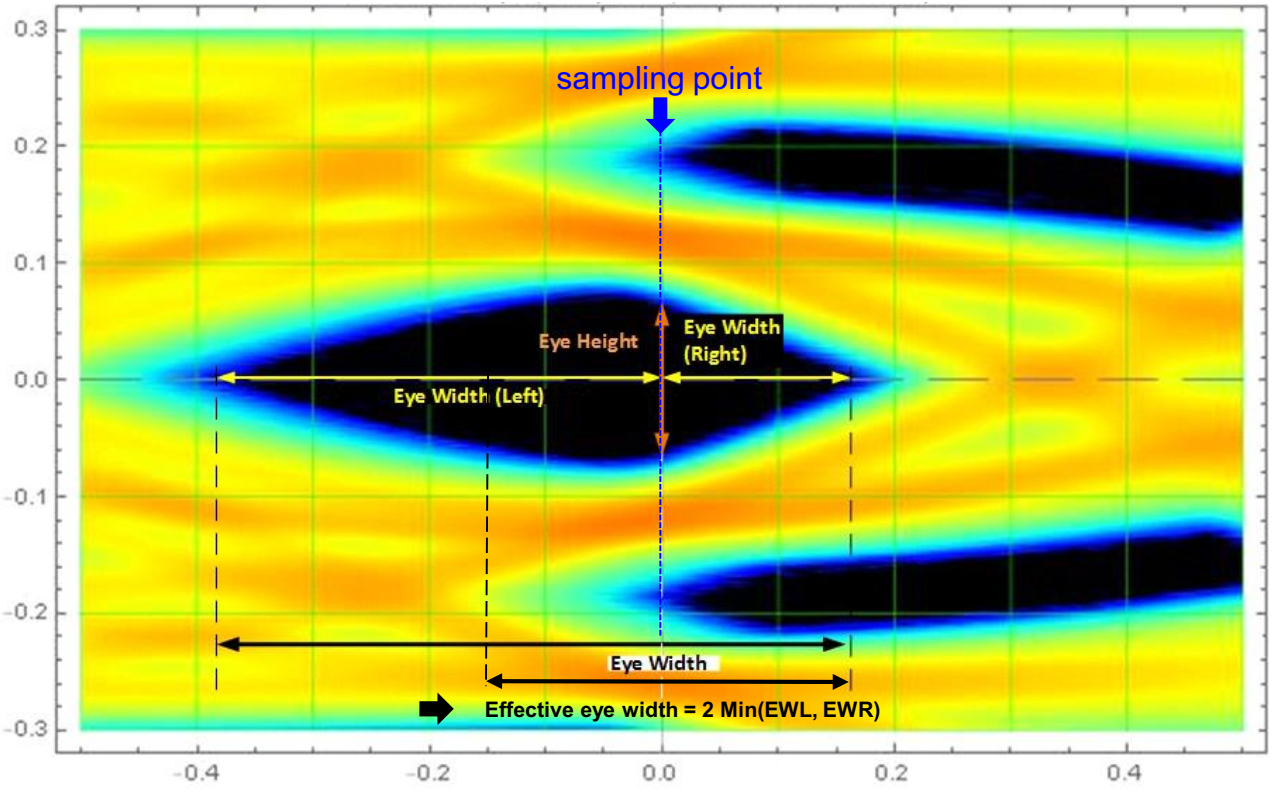
Appendix A

EYE DIAGRAMS: EXAMPLES AND DEFINITIONS

Example - 1


sampling point
Eye Height
Eye Width (Left)
Eye Width (Right)
Eye Width
Effective eye width = 2 Min(EWL, EWR)
0.3
0.2
0.1
0.0
-0.1
-0.2
-0.3
-0.4
-0.2
0.0
0.2
0.4

Example - 2

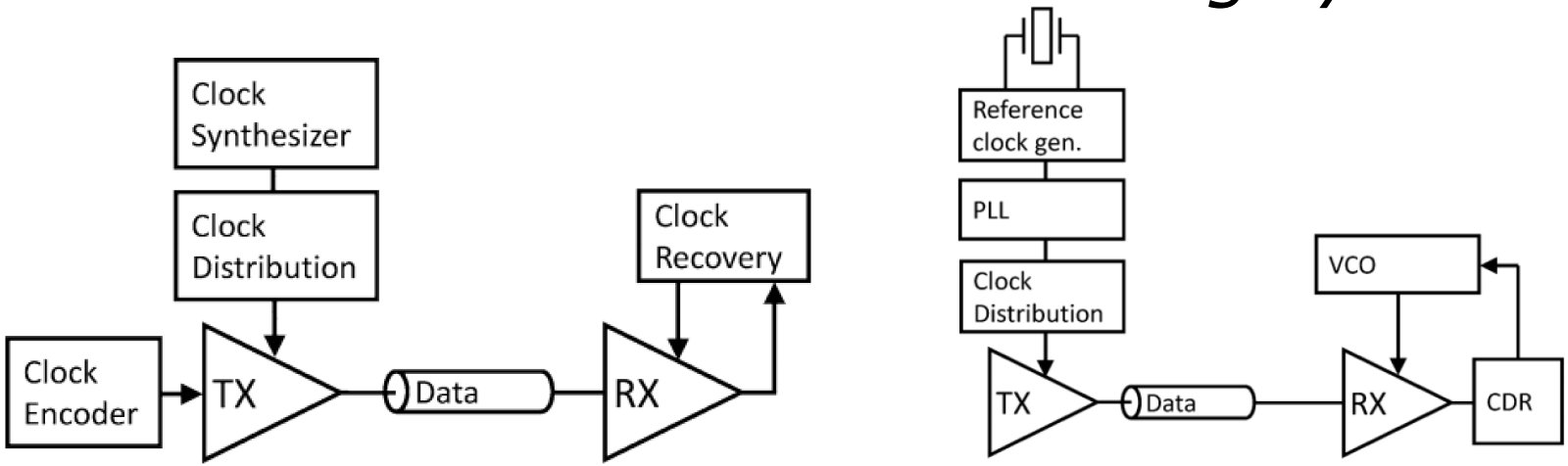

sampling point
Eye Height
Eye Width (Right)
Eye Width (Left)
Eye Width
Effective eye width = 2 Min(EWL, EWR)
0.3
0.2
0.1
0.0
-0.1
-0.2
-0.3
-0.4
-0.2
0.0
0.2
0.4

Appendix B

TIMING RECOVERY TOPOLOGIES

Timing Recovery Topologies - 1

A system in which the timing information is extracted solely from the received signal is referred to as an *embedded clocking system.*

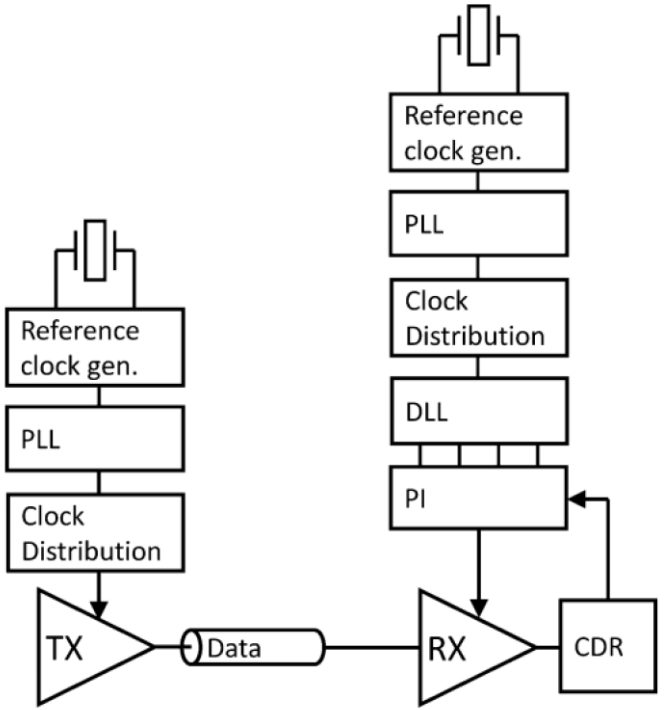


Casper et al., "Clocking Analysis, Implementation and Measurement Techniques for High-Speed Data Links - A Tutorial ", IEEE Transaction on Circuits and Systems, VOL. 56, NO. 1, January 2009

Timing Recovery Topologies - 2

A variant of the embedded clock recovery topology commonly used in HSIO is the Phase Interpolator (PI) architecture.

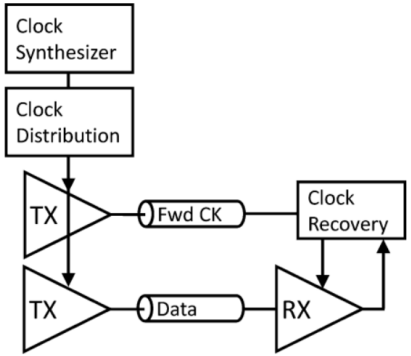
This is a configuration which is well suited for Common Clock links, where the TX and RX PLLs share a common reference clock.

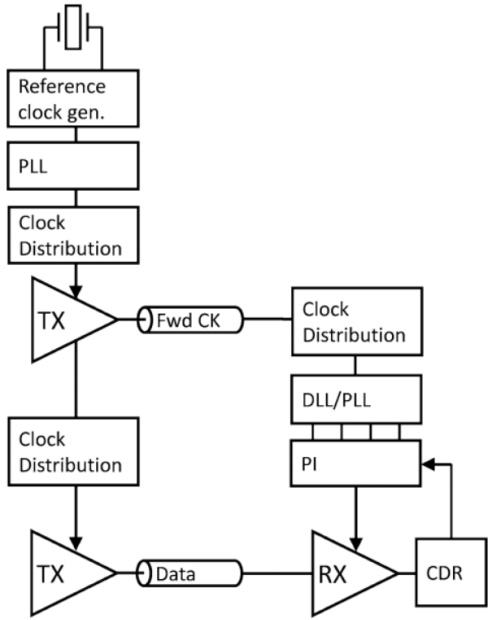


Casper et al., "Clocking Analysis, Implementation and Measurement Techniques for High-Speed Data Links - A Tutorial ", IEEE Transaction on Circuits and Systems, VOL. 56, NO. 1, January 2009

Timing Recovery Topologies - 3

When the timing information is extracted by a clock signal which is sent to the receiver together with the data signal we refer to it as a *forwarded clock system*.





- In this tutorial we will focus on *embedded clocking systems*.

Casper et al., "Clocking Analysis, Implementation and Measurement Techniques for High-Speed Data Links - A Tutorial ", IEEE Transaction on Circuits and Systems, VOL. 56, NO. 1, January 2009

Appendix C

EARLY LATE PROBABILITY COMPUTATION

Timing Functions

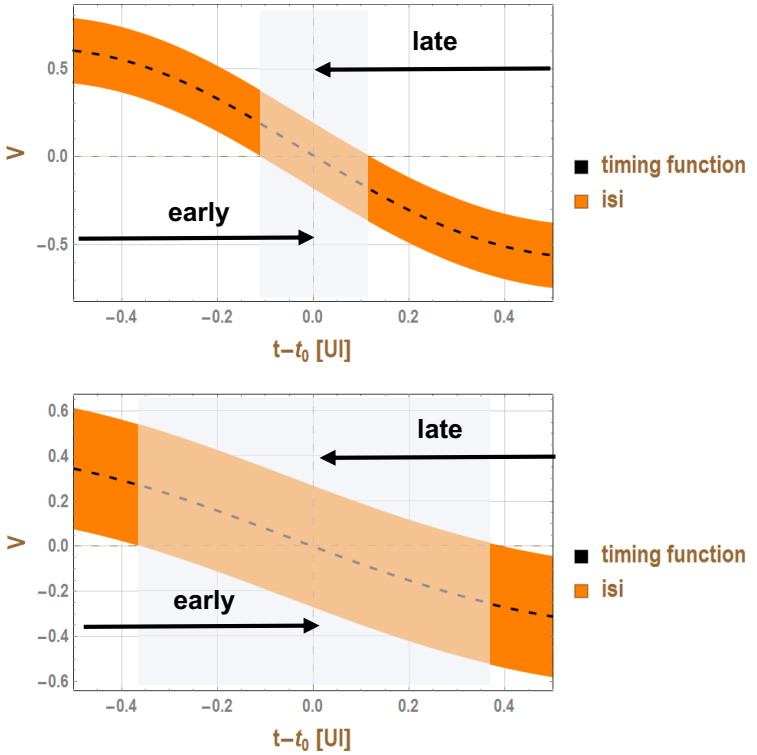
In the linear case the generation of the Phase Detector Output transfer function follows directly from the timing function itself.

In the case of a bang-bang phase detector the process of generating the output transfer function is a bit more complex.

Timing Functions

Let's assume that the system timing function is known.

- The first step is to quantify the error caused by residual ISI.
- In the examples on the right the two plots show two cases with different amounts of residual ISI.

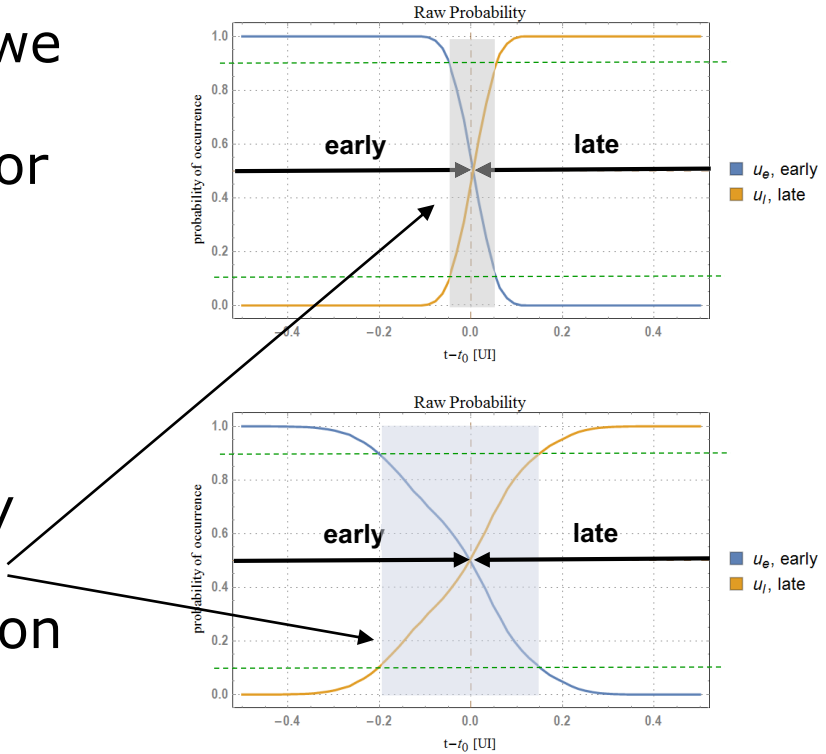


Region where **both** Late and Early results are possible.

Timing Functions

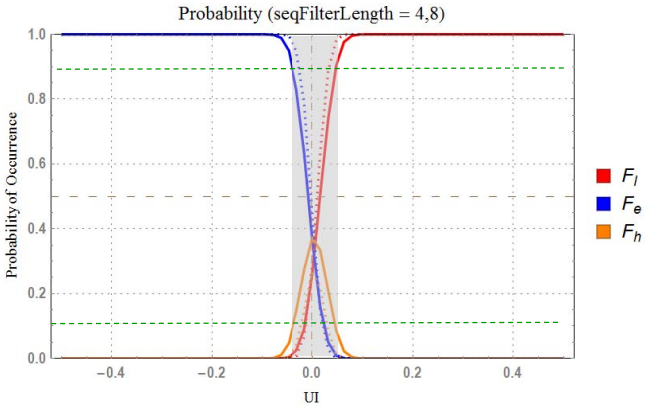
For each point on the x-axis, we can compute the (raw) probability that the timing error detected be positive or negative.

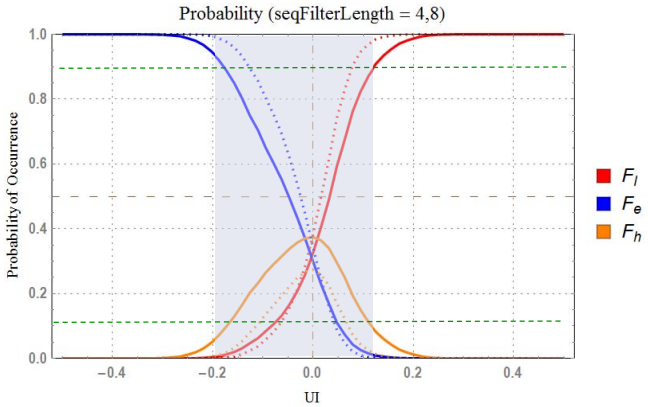
The effect of a larger residual ISI is clearly visible in the way it affects the slope and the width of the early/late transition region.



u_e, probability of a late event
u_l, probability of an early event

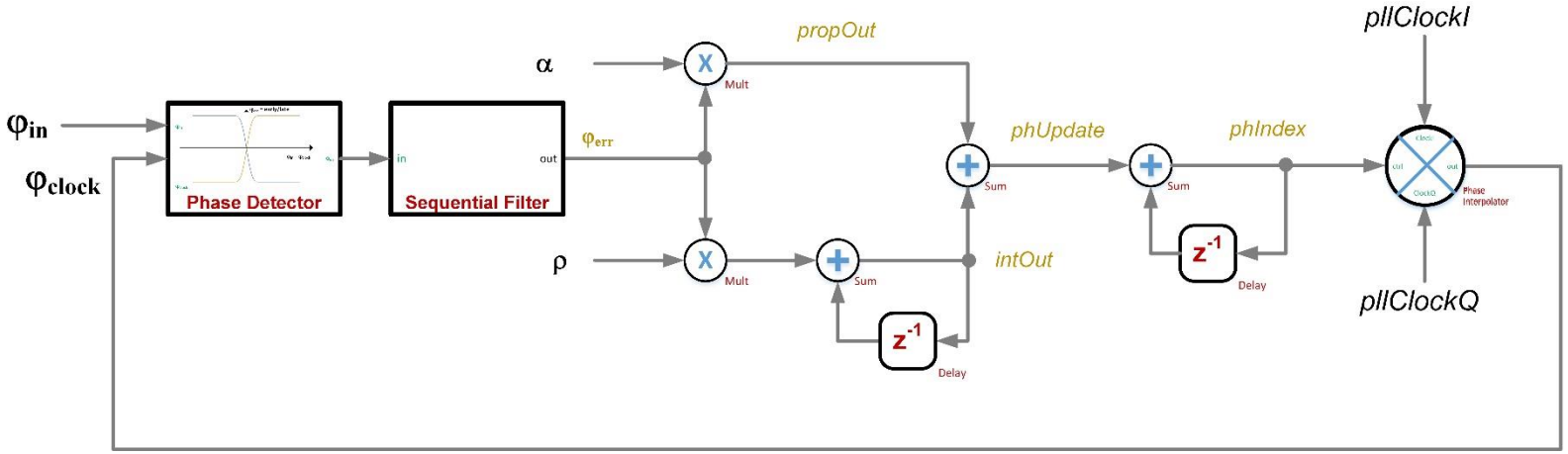
Timing Functions





- A majority vote filter at the output of the phase detector removes some of the *'softening'* introduced by ISI.
- The effect becomes more pronounced as the filter length, λ, is increased but that comes at the price of higher latency and slower update rate which may ultimately translate in reduced tracking bandwidth.

Timing Functions



For implementation reasons, the raw PD output is not directly consumed by the timing recovery loop which, instead, acts on a phase error estimate, φ_{err}, which results from the combination (filtering) of a number, λ, of raw PD outputs.

Timing Functions

One filter, very attractive because it lends itself to a simple hardware implementation, is that which implements a 'majority rule'.

- In this scheme the final PD output depends whether, in a window of constant length λ, the number **m** of *early* raw outputs exceeds the number λ-**m** of *late* raw outputs.
- The likelihood that **m** out of λ samples are *late* is given by:

$$f_l(m) = \frac{\lambda!}{m!\,(\lambda - m)!} {u_l}^m (1 - u_l)^{\lambda - m} = \frac{\lambda!}{m!\,(\lambda - m)!} {u_l}^m {u_e}^{\lambda - m} = \binom{\lambda}{m} {u_l}^m {u_e}^{\lambda - m}$$

and a corresponding expression can be written for $f_e(m)$.

Timing Functions

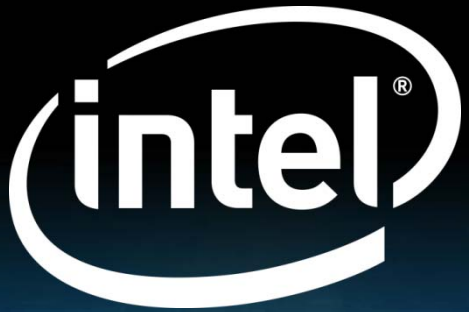
Because of the majority rule, the probabilities for the filtered outputs are:

$$F_l = \sum_{m=\frac{\lambda}{2}+1}^{\lambda} \frac{\lambda!}{m!\,(\lambda-m)!} {u_l}^m (1-u_l)^{\lambda-m} = \sum_{m=\frac{\lambda}{2}+1}^{\lambda} \binom{\lambda}{m} {u_l}^m {u_e}^{\lambda-m}$$

$$F_e = \sum_{m=\frac{\lambda}{2}+1}^{\lambda} \frac{\lambda!}{m!\,(\lambda-m)!} {u_e}^m (1-u_e)^{\lambda-m} = \sum_{m=\frac{\lambda}{2}+1}^{\lambda} \binom{\lambda}{m} {u_e}^m {u_l}^{\lambda-m}$$

To which we need to add, if λ is even, the probability that the output is neither *early* or *late:*

$$F_h = 1 - F_e - F_l$$


intel