IBM Research

Topics in Design and Analysis of High Data Rate SERDES Systems

Troy Beukema
IBM T. J. Watson Research Center
Yorktown Heights, NY

IEEE SSCS, Denver Section Seminar | Fort Collins, CO September 2009

Table of Contents

Introduction : Drivers of High Data Rate Interconnect

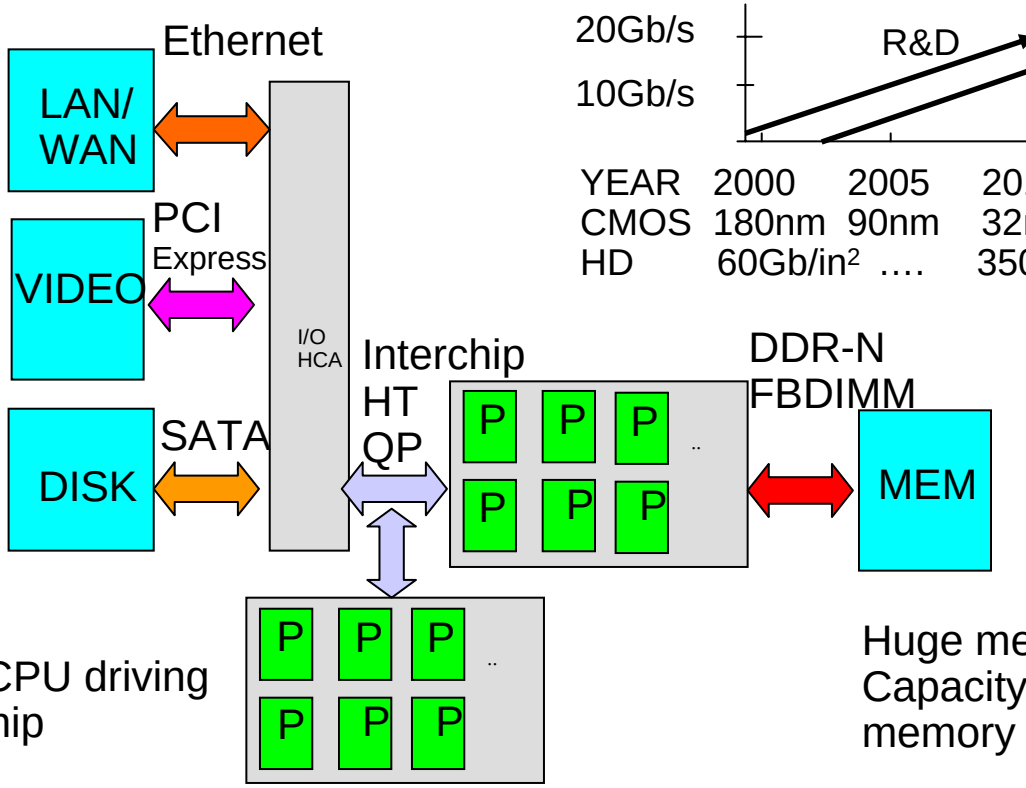
Observation : there is **no end** to the desire for higher interconnect bandwidth!

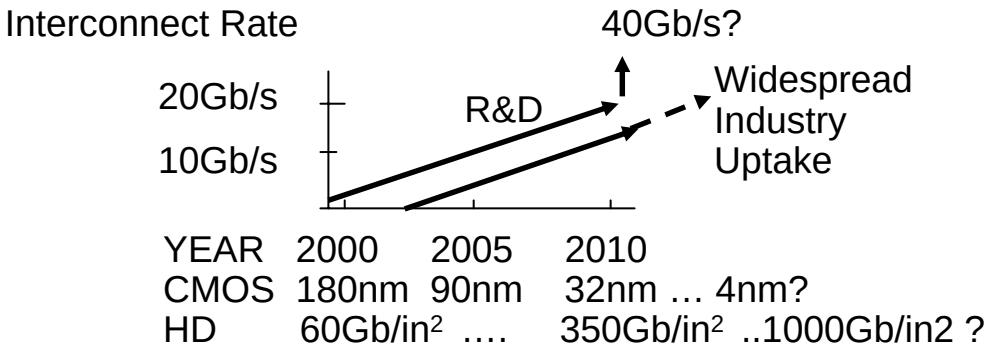
"Video" Internet
Driving exponential
bandwidth growth

3D memory >=2Gb
GP GPU >= 1Tflop
driving PCI BW and
GPU memory BW

Multiple Tb storage
and SSD driving
disk I/O BW

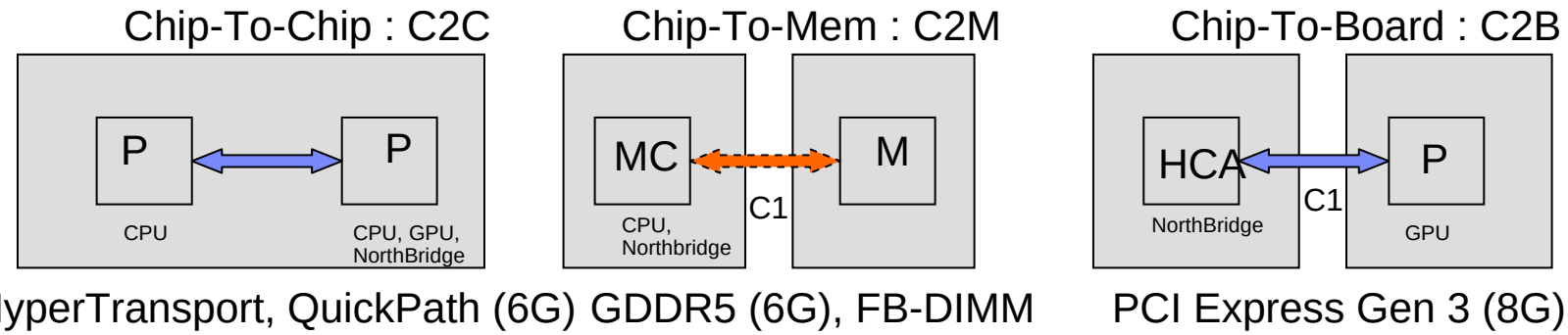
Multiple Processor CPU driving
memory and inter-chip
communication BW

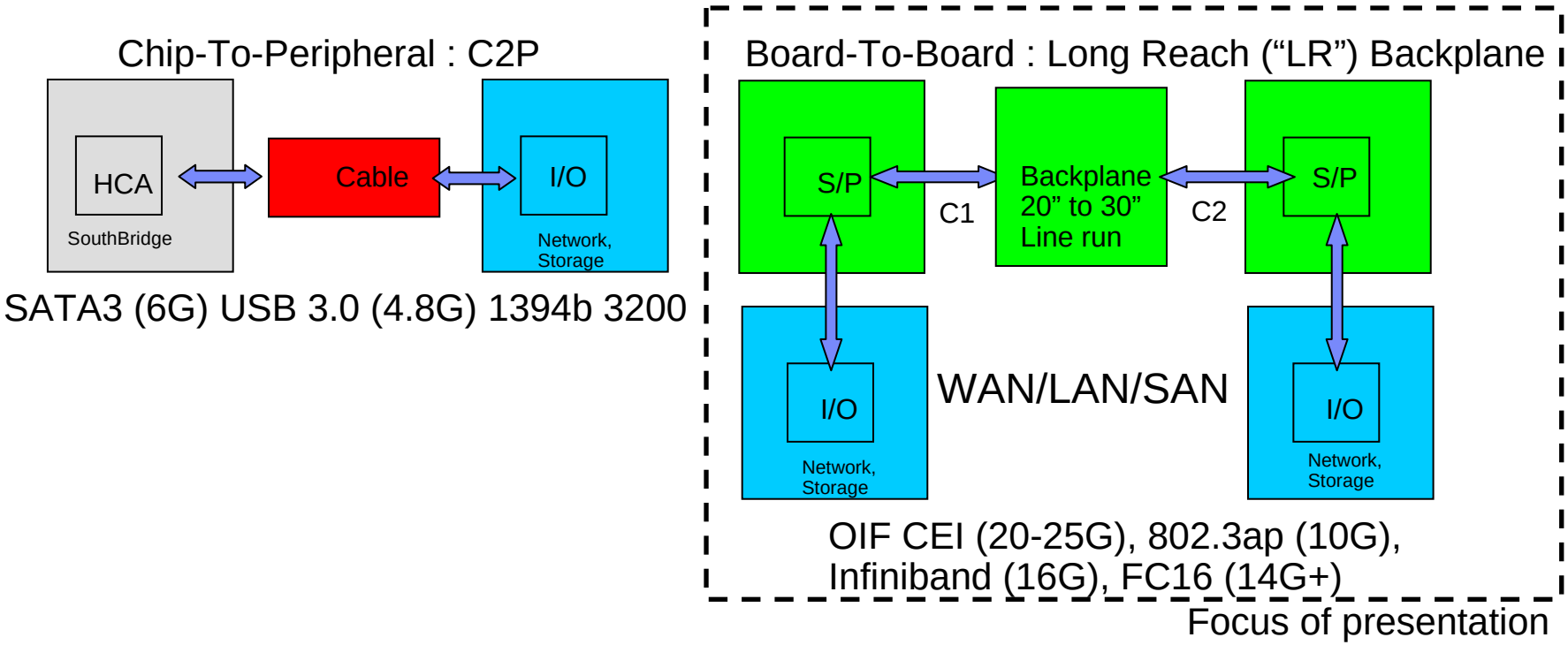




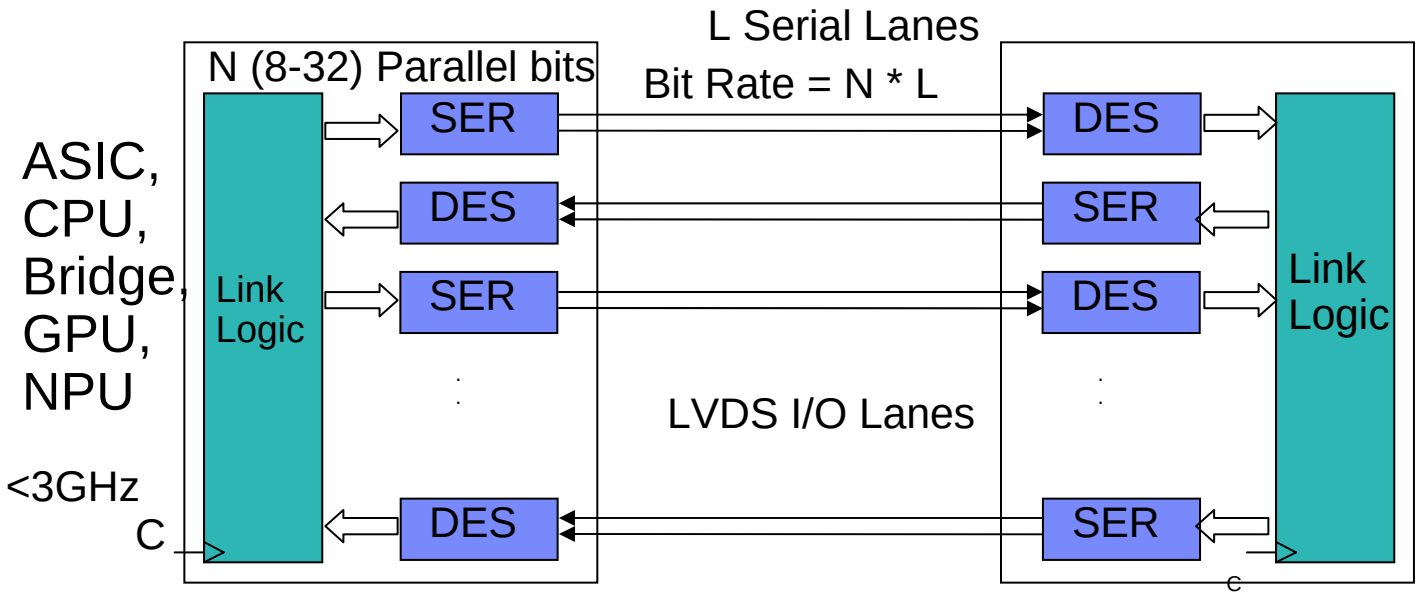
Huge memory
Capacity (>32Gb) driving
memory bandwidth

Data Interconnect Applications





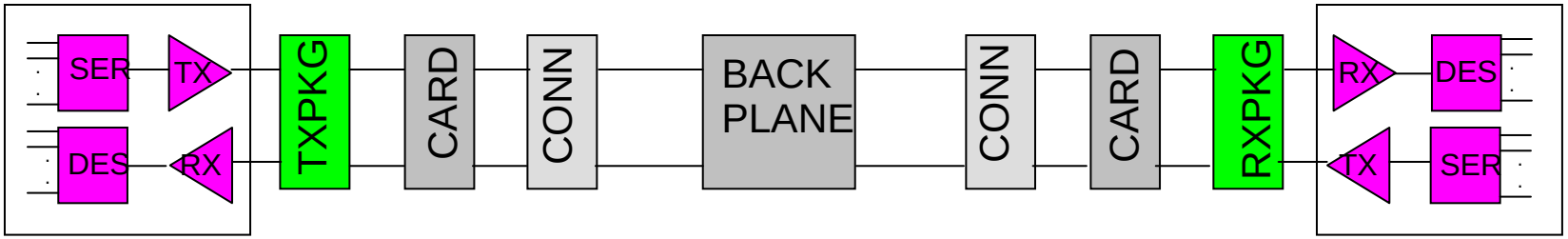
Point-to-Point SERDES I/O Architecture



Why Use SERDES I/O Architecture?

1) Impossible to escape chip with N * L parallel lines due to package pin density limits
2) Data skew for wide parallel synchronous buses becomes a problem at high data rates
3) Clock Recovery/Equalization on L lines needed instead of N * L lines : Simplifies receiver
4) Point-to-Point impedance-controlled I/O eliminates signal degradation from parallel “stub bus”

Anatomy of a Backplane Serial Interconnect System



Processor/Network Model

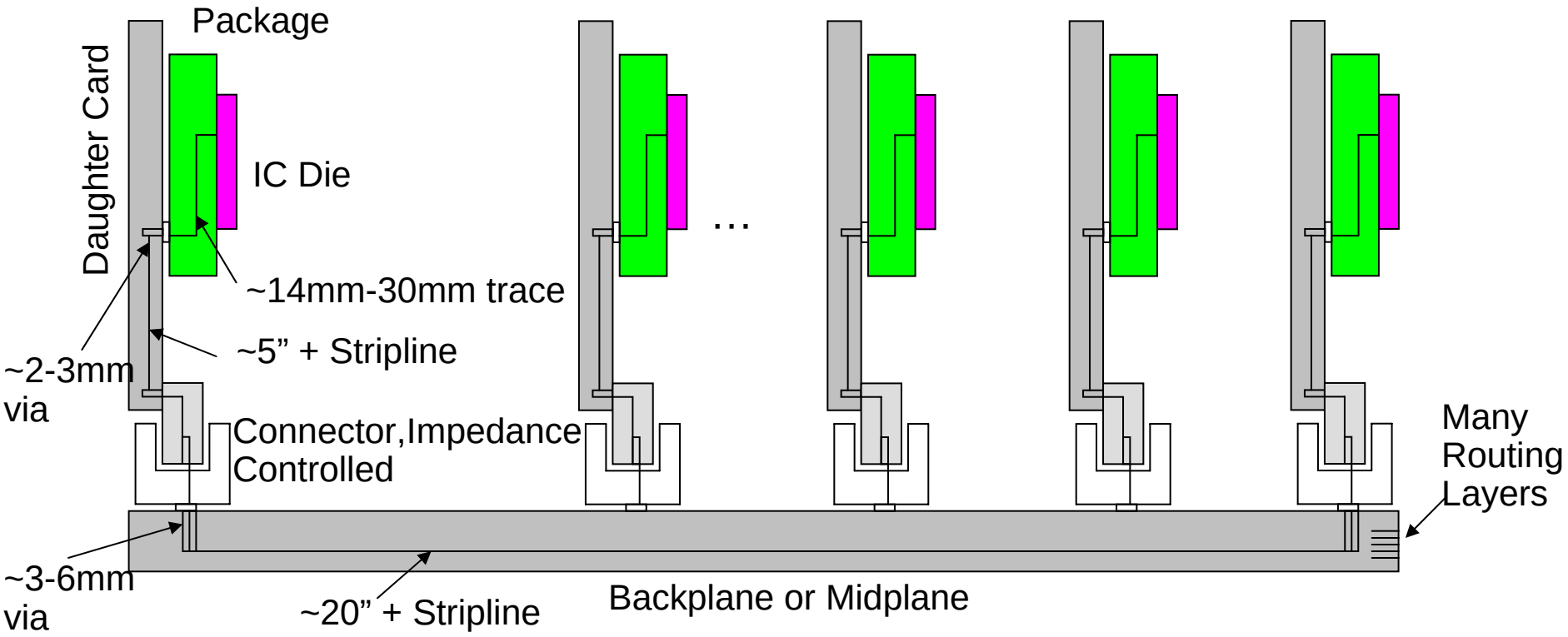
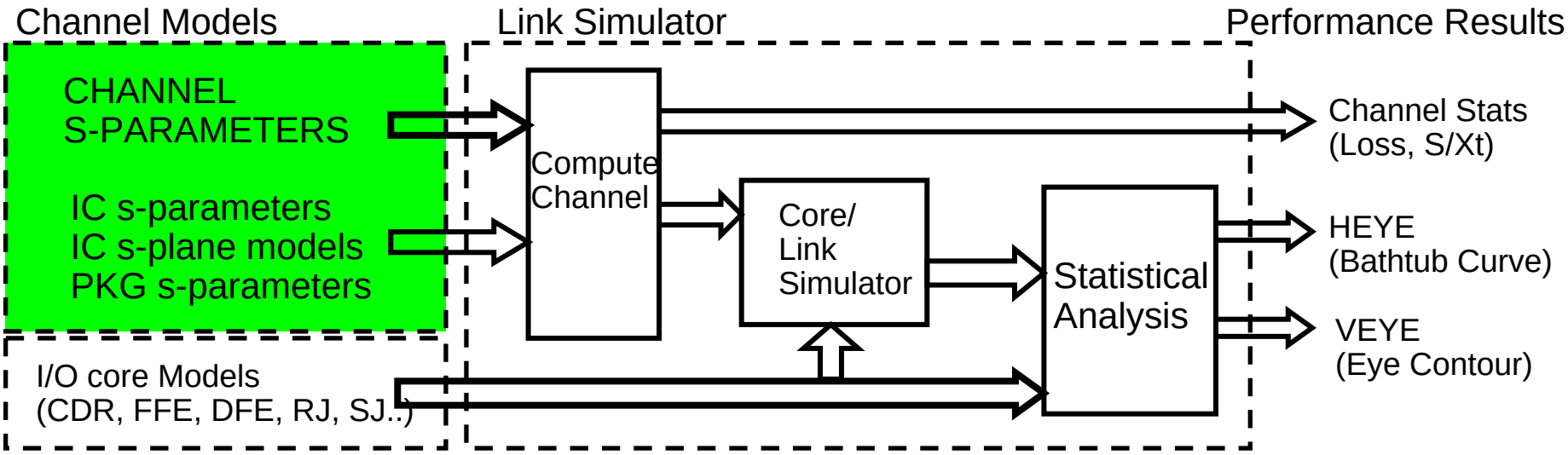
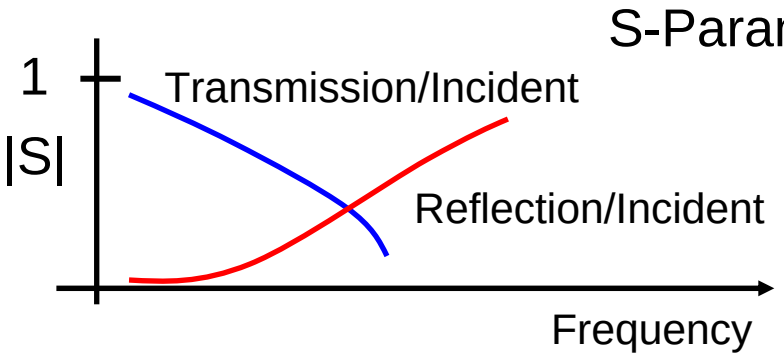


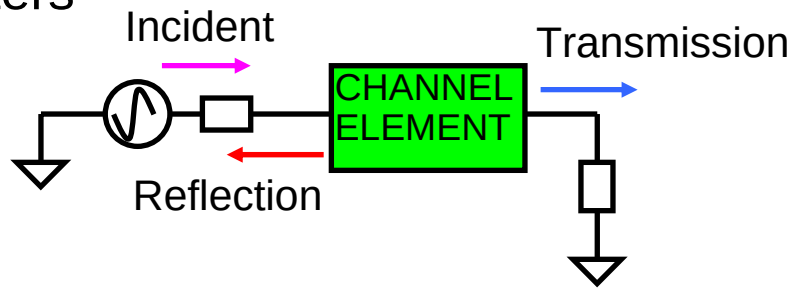
Table of Contents

Channel Models with S-Parameters

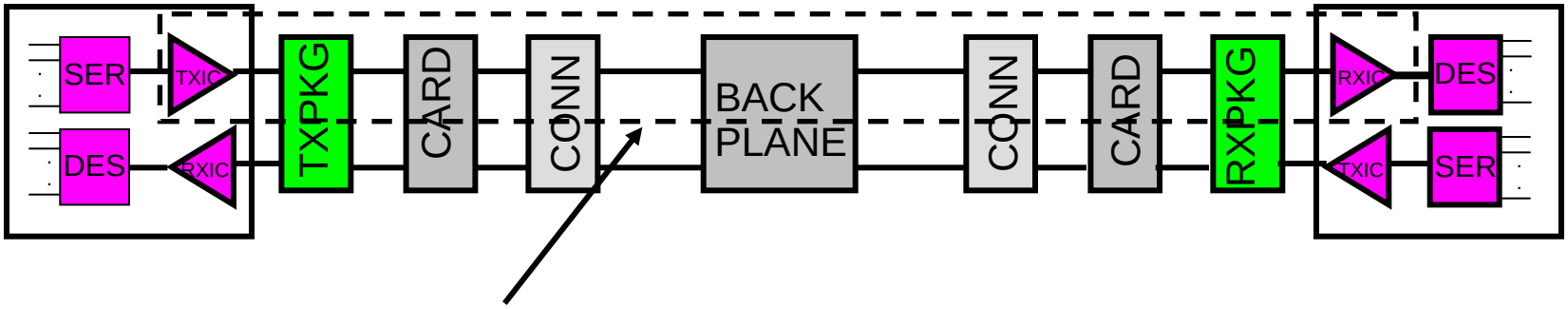
I/O system performance analysis typically starts with Scattering (S)-parameter descriptions of the frequency response of the application channel and I/O core.







S-Parameter/Frequency-Domain Based Link Modeling



Basic “End-to-End” Serial Channel

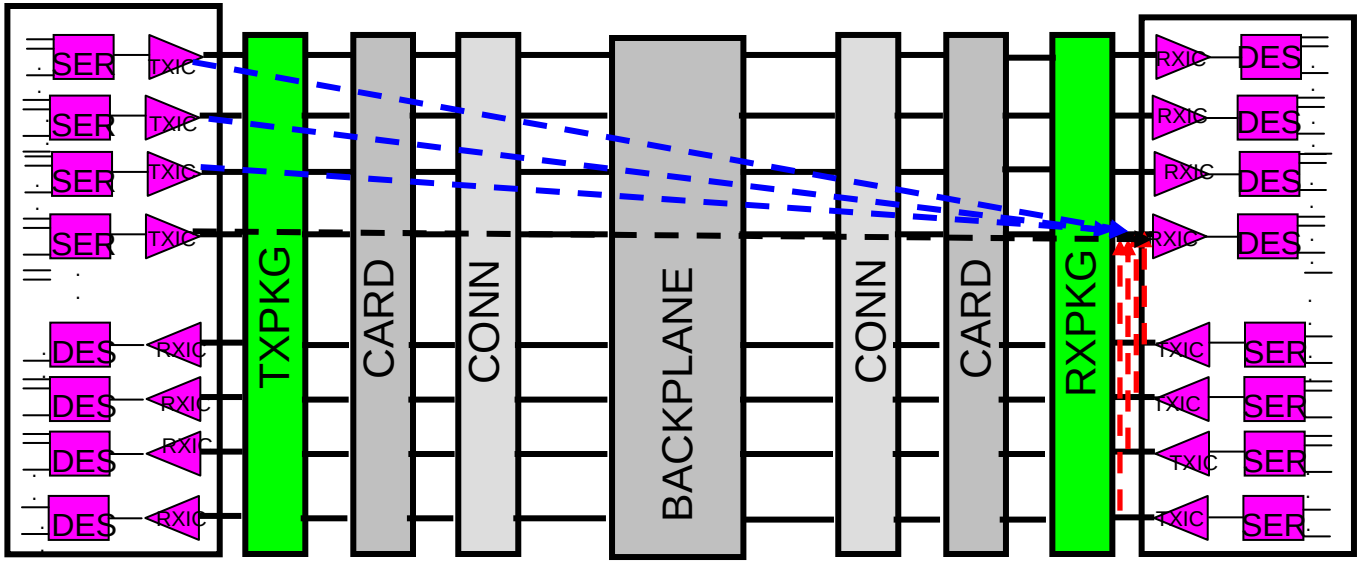
Each component of the serial link is modeled as an S-parameter, or possibly a rational s-plane pole/zero model

The “End-to-End” channel is found by cascading the S-parameters

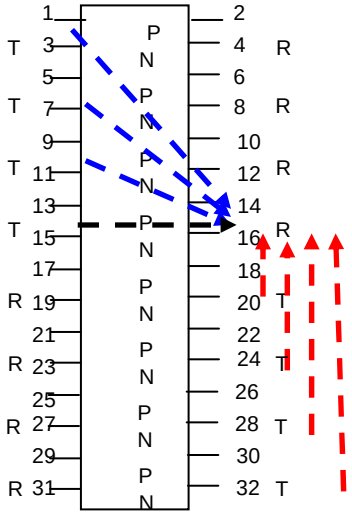
After cascading the S-parameters, the channel time-domain impulse response is found using a Frequency to Time Transform.

S-parameter models are typically created by a combination of the IC Manufacturer (IC and package models) and the system OEM (backplane+ daughter-cards)

Power of S-Parameters : Efficiently Describe Parallel Bus N-Port Models



32-Port S-Parameter L

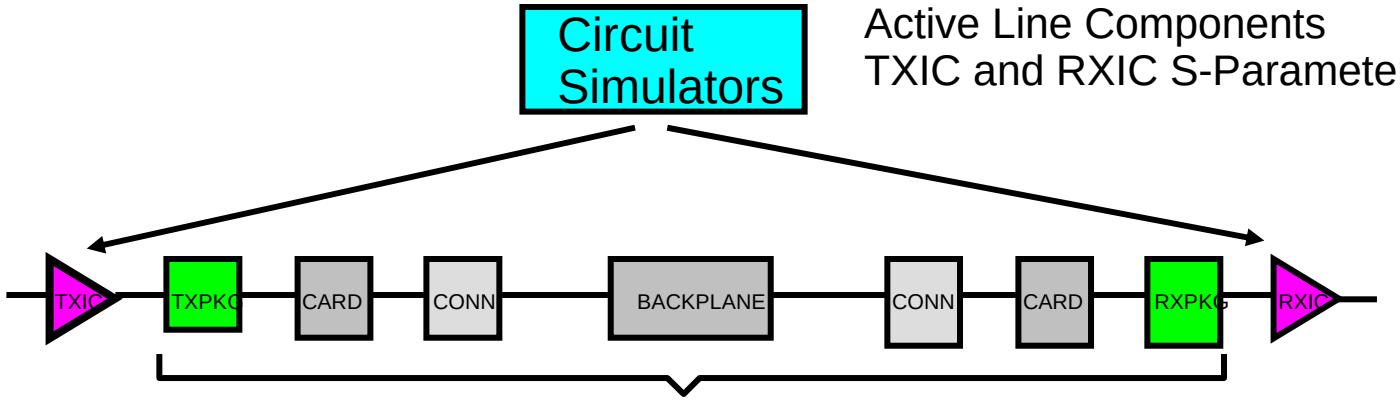


FEXT = [1,14,3,16]L,
[5,14,7,16]L,
[9,14,13,16]L = FAR END CROSSTALK

THRU = [13,14,15,16]L = “VICTIM” LINE

NEXT = [18,14,20,16]L,
[22,14,24,16]L,
[26,14,28,16]L,
[30,14,32,16]L = NEAR END CROSSTALK

Sources of S-Parameters



Active Line Components
TXIC and RXIC S-Parameter models

Vector-Network Analyzer (VNA) Measurements

2D, 3D EM Field Solvers

Passive Line Components :
- IC Package, Board Vias, Connectors
- Microstrip/Stripline transmission lines on daughter-cards and motherboards

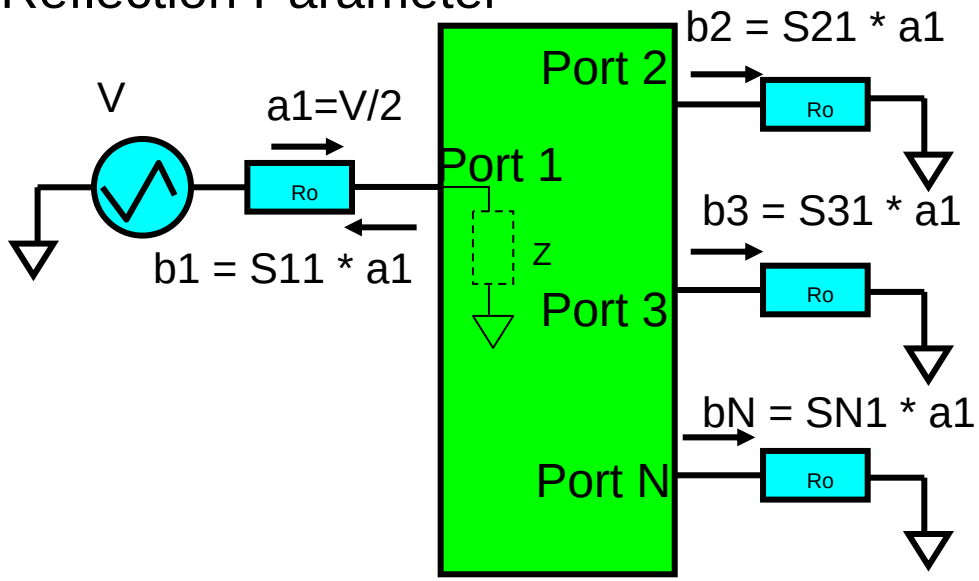
Advantages of S-parameters based channel analysis :
- "Real" hardware brought into computer simulations through VNA measurements
- Unifies disparate link building blocks to one common format

Disadvantages :
- Need linear channel assumption
- Invalid s-parameters easy to generate
  - (bad/noisy/miscalibrated measurements or invalid 2D/3D EM models)
- May require causality-passivity correction before use

Formal S-Parameter Definition and N-Port Matrix Representation

S-Parameters are defined as the ratio of a signal level transmitted out of a port to an incident signal sent into an excitation port, where all ports are terminated in a fixed impedance called the reference impedance.

Reflection Parameter



Transmission Parameters

Outputs = S-parameter matrix × Incident Voltages

$$\begin{bmatrix} b1 \\ b2 \\ . \\ . \\ bN \end{bmatrix} = \begin{bmatrix} S11 & S12 .. & S1N \\ S21 & S22 .. & S2N \\ . & & \\ . & & \\ SN1 & SN2 \ldots & SNN \end{bmatrix} \begin{bmatrix} a1 \\ a2 \\ .. \\ aN \end{bmatrix}$$

Nomenclature :

S21

Source Port

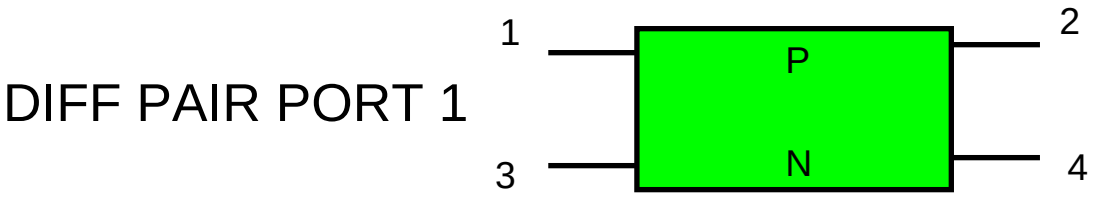
Destination Port

See Appendix A.1 for Derivation of S-Parameters from Port Node Voltages

4-port S-parameters and Mixed-Mode S-Parameter Representation

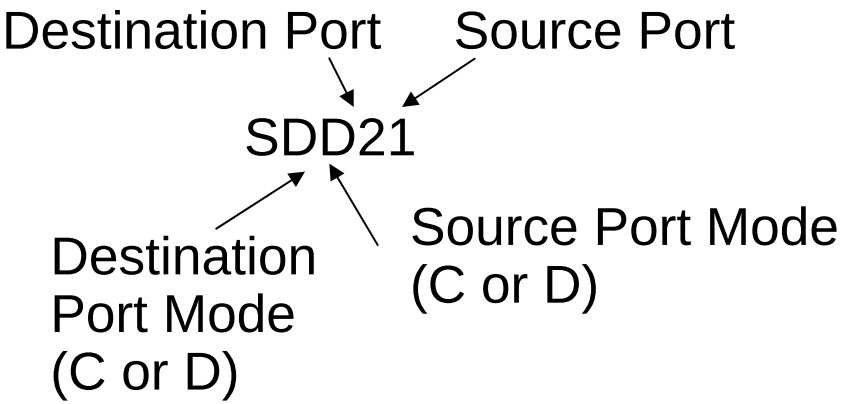
Most high rate SERDES systems use differential signaling, where a signal is Encoded as P-N where P is the voltage on a positive line and N is the voltage on a negative line. A 4-port S-parameter can describe this differential channel.

Odd In / Even Out preferred port numbering convention :



D = Differential Port Voltage $\triangleq$ (Vp – Vn)
C = Common Port Voltage $\triangleq$ (Vp + Vn)

Nomenclature :



See Appendix A.2 for conversion between single-ended and mixed-mode s-parameter formats

Mixed Mode Matrix Representation

There is no standard convention for storing mixed-mode S-parameters!

All of these matrices represent the same S-parameter information in different formats :

Single-Ended S-parameters Matrix

	1	2	3	4
1	S11	S12	S13	S14
2	S21	S22	S23	S24
3	S31	S32	S33	S34
4	S41	S42	S43	S44

Mixed-Mode S-parameters Matrix #1

	D1	D2	C1	C2
D1	SDD11	SDD12	SDC11	SDC12
D2	SDD21	SDD22	SDC21	SDC22
C1	SCD11	SCD12	SCC11	SCC12
C2	SCD21	SCD22	SCC21	SCC22

Mixed-Mode S-parameters Matrix #2

	D1	C1	D2	C2
D1	SDD11	SDC11	SDD12	SDC12
C1	SCD11	SCC11	SCC12	SCC12
D2	SDD21	SDC21	SDD22	SDC22
C2	SCD21	SCC21	SCD22	SCC22

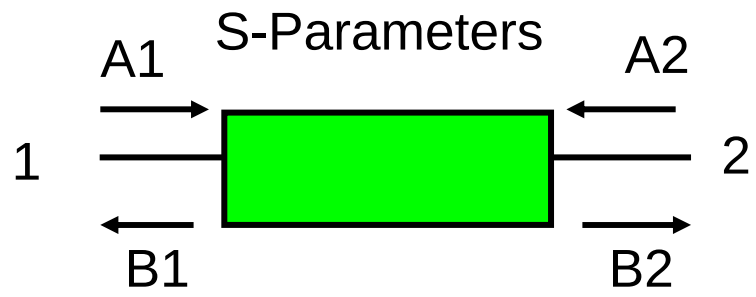
Mixed-Mode S-parameters Matrix #3

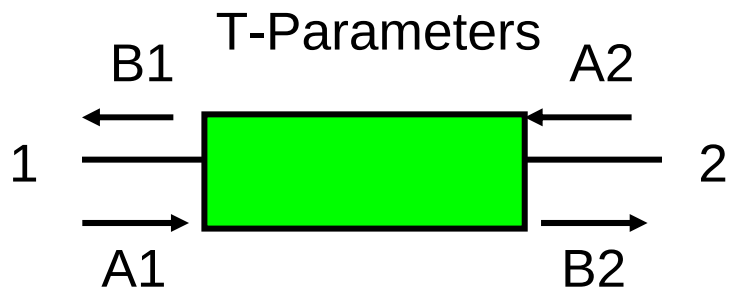
	11	12	21	22
D1D2	SDD11	SDD12	SDD21	SDD22
D1C2	SDC11	SDC12	SDC21	SDC22
C1D2	SCD11	SCD12	SCD21	SCD22
C1C2	SCC11	SCC12	SCC21	SCC22

Use of mixed-mode S-parameters in model data-bases, or as input to simulators is not recommended : **Probability (getting wrong answer) =~1**, with no warnings!

Transport Parameters

Transport Parameters re-arrange the s-parameter port variables so inputs are on LHS and outputs on RHS. This enables two series transport-parameters to be cascaded.





$$\begin{matrix} \text{INPUT} \\ \text{OUTPUT} \end{matrix} \begin{bmatrix} B1 \\ B2 \end{bmatrix} = \begin{bmatrix} S11 & S12 \\ S21 & S22 \end{bmatrix} \begin{bmatrix} A1 \\ A2 \end{bmatrix} \begin{matrix} \text{INPUT} \\ \text{OUTPUT} \end{matrix}$$

$$B1 = S11\ A1 + S12\ A2$$
$$B2 = S21\ A1 + S22\ A2$$

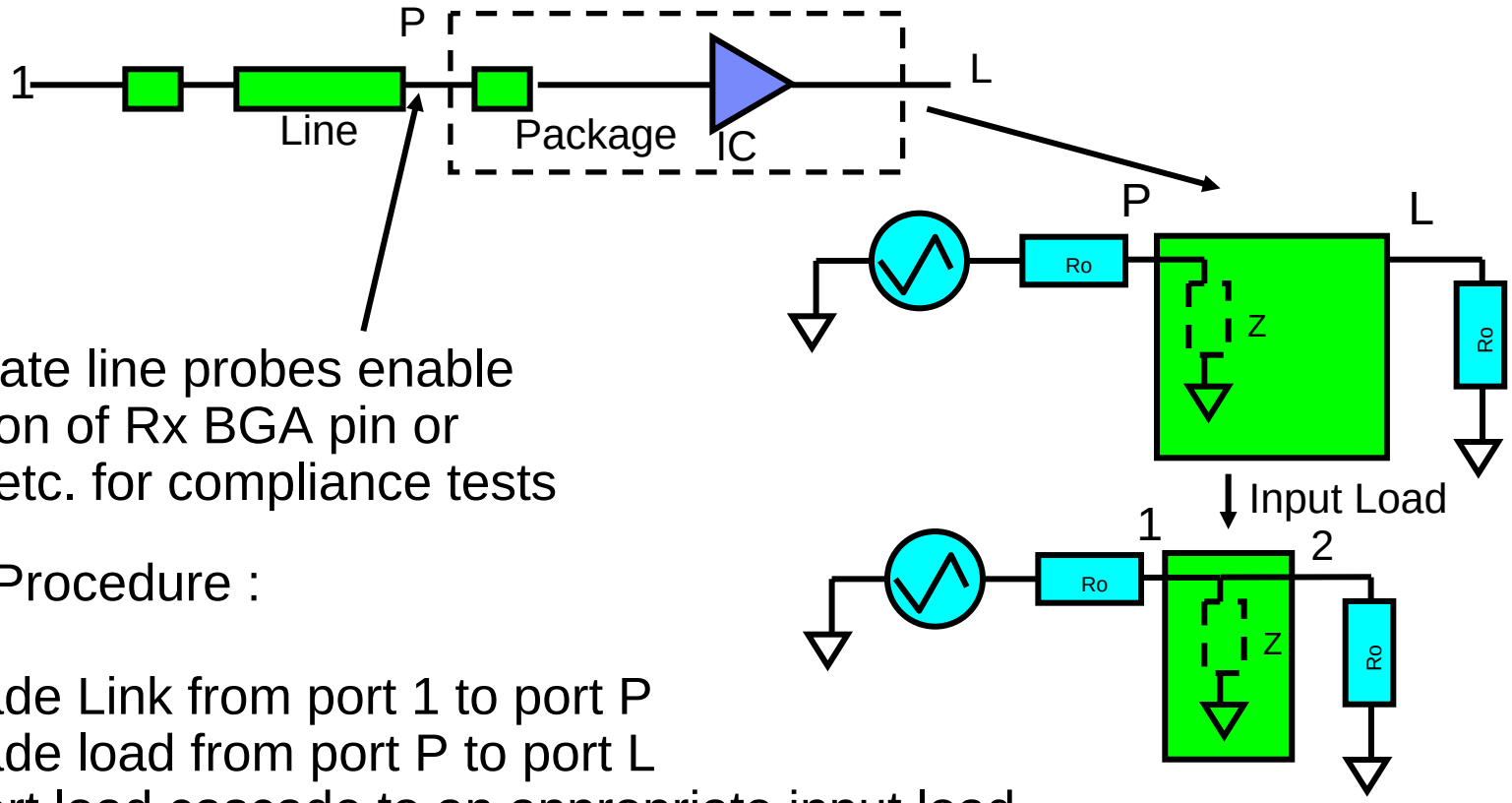
$$\begin{matrix} \text{INPUT} \\ \text{INPUT} \end{matrix} \begin{bmatrix} B1 \\ A1 \end{bmatrix} = \begin{bmatrix} T11 & T12 \\ T21 & T22 \end{bmatrix} \begin{bmatrix} A2 \\ B2 \end{bmatrix} \begin{matrix} \text{OUTPUT} \\ \text{OUTPUT} \end{matrix}$$

$$B1 = T11\ A2 + T12\ B2$$
$$A1 = T21\ A2 + T22\ B2$$

See Appendix A.3 for conversion between 2-port T and S parameters

Intermediate Line Probes with S-Parameters

Intermediate Line Probe at RxBGA



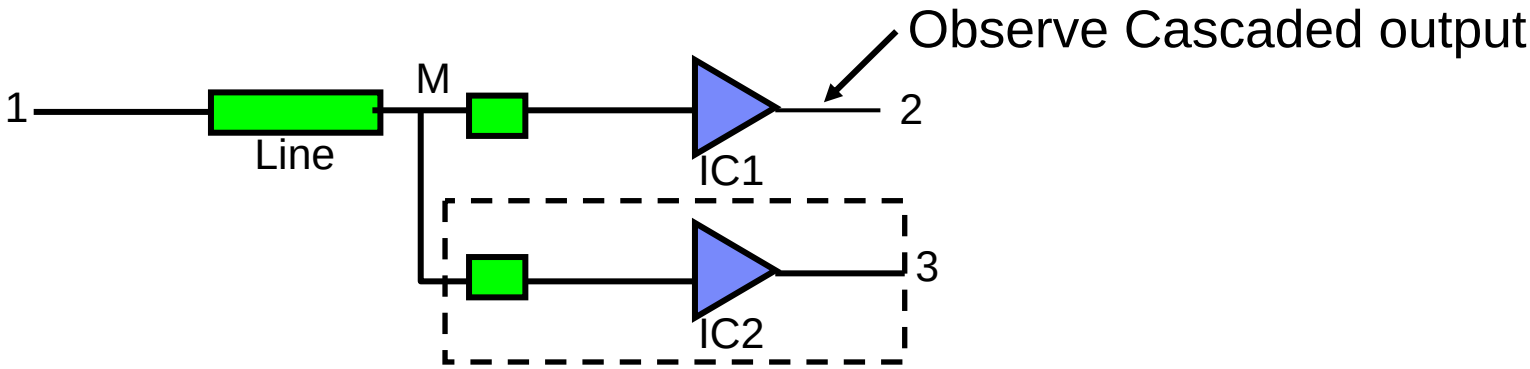
Intermediate line probes enable observation of Rx BGA pin or Tx BGA, etc. for compliance tests

Analysis Procedure :

1) Cascade Link from port 1 to port P
2) Cascade load from port P to port L
3) Convert load cascade to an appropriate input load
4) Cascade input load to end of first cascade
5) Cascade -50 ohm load to remove s-parameter probe impedance at probe point

Multidrop Cascading with S-Parameters

Multi-Drop Cascade (“Stub” Channel Analysis)

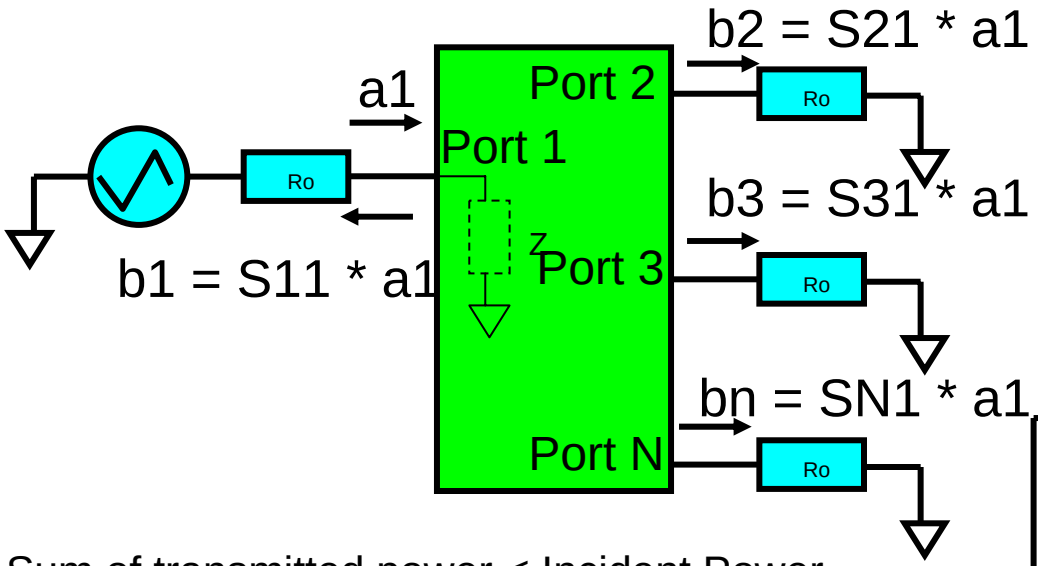


Multi-Drop Cascades enable analyzing 1-to-Many bus structures, typically found in “legacy” interconnect designs

Analysis Procedure :

1) Cascade Link from port 1 to port M
2) Cascade multidrop load from port M to port 3
3) Convert load cascade to an appropriate input load
4) Cascade input load of M->3 load
5) Finish cascading link from port M to 2

Checking Validity of S-Parameters : Passivity Requirement



$$\begin{bmatrix} b1 \\ b2 \\ .. \\ bN \end{bmatrix} = \begin{bmatrix} S11 & S12 & .. & S1N \\ S21 & S22 & .. & S2N \\ . & .. & & \\ SN1 & SN2 & \dots & SNN \end{bmatrix} \begin{bmatrix} a1 \\ a2 \\ .. \\ aN \end{bmatrix}$$

$$S^H = \begin{bmatrix} S11^* & \dots & SN1^* \\ S12^* & \dots & SN2^* \\ . & & \\ S1N^* & \dots & SNN^* \end{bmatrix} \qquad S = \begin{bmatrix} S11 & \dots & S1N \\ S21 & \dots & S2N \\ . & & \\ SN1 & \dots & SNN^* \end{bmatrix}$$

1) Sum of transmitted power < Incident Power

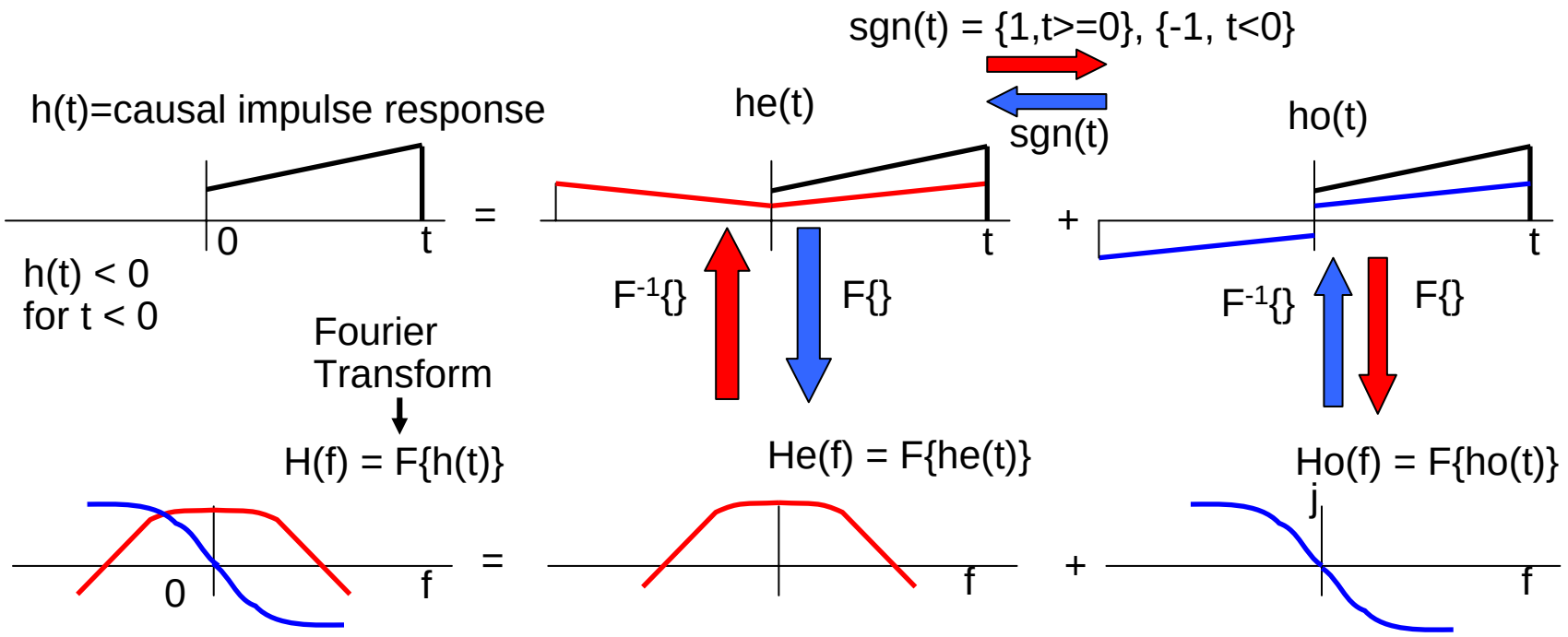
-> |b1|**2 + |b2|**2 + … |bn|**2 < |a1|**2
-> Re{S11 * conj(S11)} + Re{S21 * conj(S21)} + … Re{conj(SN1) * SN1} < 1
In the general case, $\Sigma|bi|^2 <= \Sigma|ai|^2$ -> $S^H S <= I$, where S^H = conjugate-transpose

2) all the S-parameter frequency responses must satisfy causality constraints (non-causal -> non-passive)

3) All the S-parameter frequency responses must satisfy :
Sij(jw) = conj(Sij(-jw))

Reference : “Stability, Causality, and Passivity in Electrical Interconnect Models”, Triverio et. al. , IEEE TRANSACTIONS ON ADVANCED PACKAGING, VOL. 30, NO. 4, NOVEMBER 2007, pp. 801-803

Causality of (S-Parameter) Frequency Responses



For causality to hold, ho(t) = sgn(t) he(t), -> Ho(f) = F{sgn(t) he(t)} = F{sgn(t)} conv He(f)
also, he(t) = sgn(t) ho(t), -> He(f) = F{sgn(t) ho(t)} = F{sgn(t)} conv Ho(f)

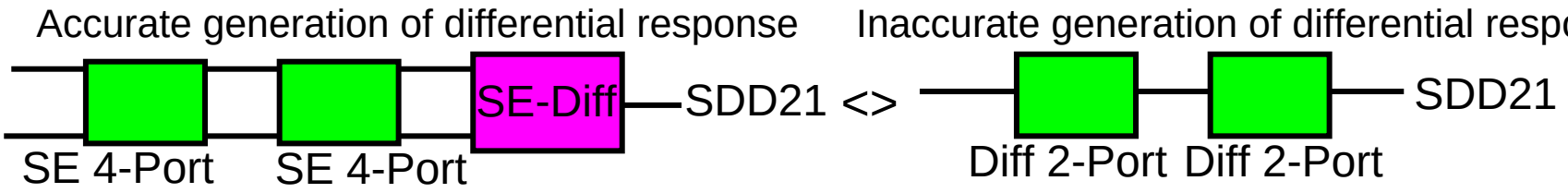
Hilbert Transforms

Hazard :

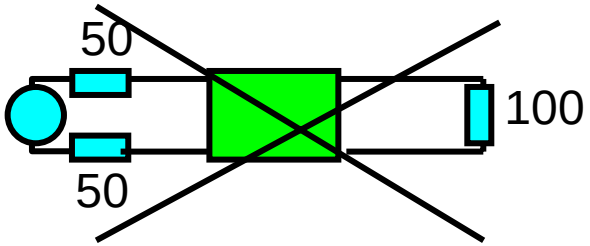
Many model-generated and measured S-parameters do not satisfy the causality constraint and require numerical correction! Non-causality can arise from “non-causal” models or measurement errors.

Some S-parameter Mistakes to Avoid

1) Cascading differential 2-ports is not the same as a 4-port differential cascade

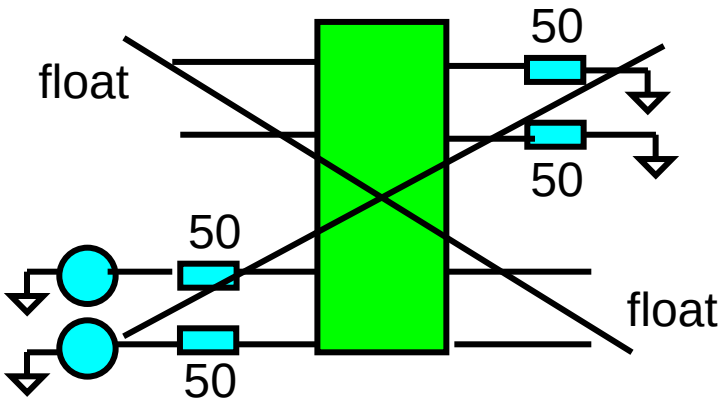


2) Do not use floating sources to measure differential s-parameters in simulators



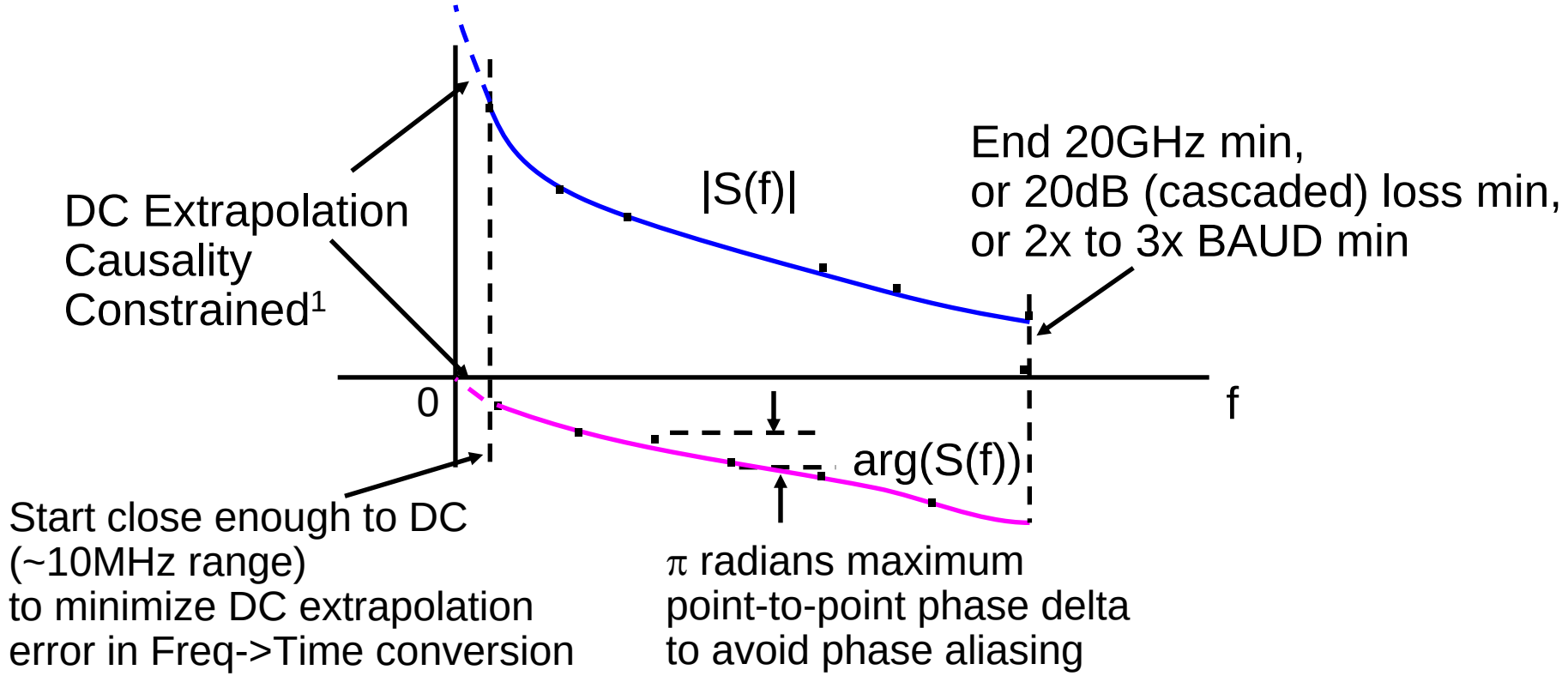
This has been seen
in commercial EDA tools!

3) Do not leave ports floating when making crosstalk measurements



Victim and aggressor ports
need termination on both input
and output for valid
crosstalk measurements

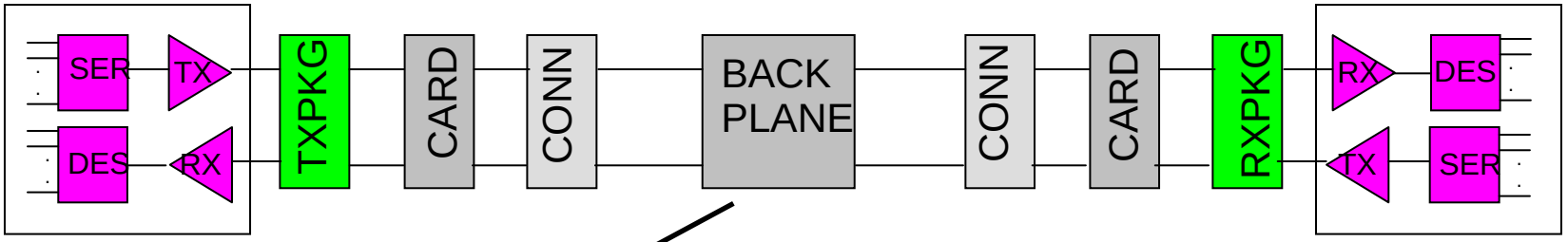
Frequency Sampling Grid and DC Extrapolation for S-Parameters



USEFUL SAMPLING GRIDS (START and STEP the same) :
5MHz to 20GHz in 5MHz STEPS (4000 POINTS)
10MHz to 20GHz in 10MHz STEPS (2000 POINTS)
20MHz to 20GHz in 20MHz STEPS (1000 POINTS)

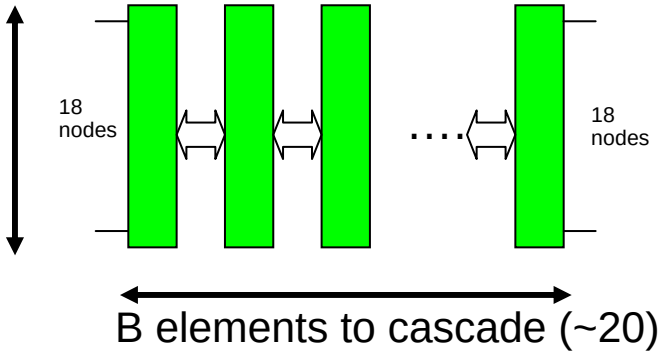
[1]P. Triverio, S. Grivet-Talocia, **"**Causality-constrained interpolation of tabulated frequency responses**"**
http://www.emc.polito.it/publications/file/cnf-2006-epep-DCreconstruction.pdf

Efficiency of S-Parameter Cascade Analysis vs. Nodal Analysis



CHANNEL CASCADE : ~17 Elements

N (36) port s-parameters to describe 9 diff pairs (4 aggressors on either side of a victim)



Cascade analysis : B * O(3) for N-ports = B * N^3

Nodal Analysis : (B+1) * (N/2) NODES =~ BN/2 NODES, O(3) <- $(BN/2)^3$

Complexity Ratio = B^2 / 8, for B = 10-20 elements, speed ratio = 10-50

S-Parameter Cascade Analysis 10X to 50X faster than Nodal Analysis
36-port cascades readily concatenated on 2G SSE Laptops in ~minutes

Summary of S-Parameters for Link Modeling

Major Advantages :

+Describes “real” hardware measurements in addition to models
+Models coupling in wide data buses
+Can describe both passive line elements and active circuits : captures the “end-to-end” link.
+Versatile analysis capability : multidrop links, intermediate line probes
+Computationally efficient in solving wide bus problems

Some Disadvantages :

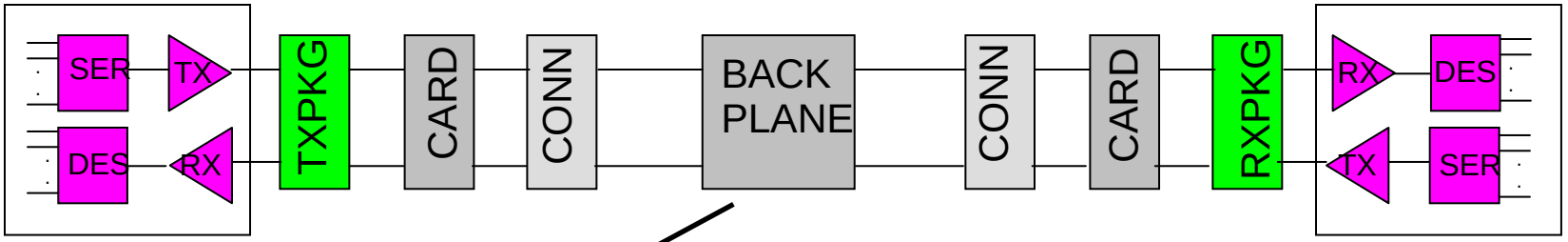
-Linear channel assumption
-Finite frequency range, accuracy of s-parameters at low and high frequency can degrade in both measurements and models

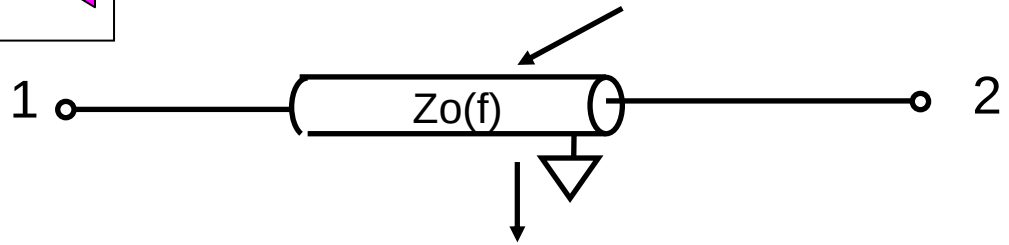
Some Caveat Emptors :

*Beware of invalid passivity and causality, undersampling, and insufficient frequency range : many s-parameters in the wild are inconsistent / invalid due to one or more of these problems.
*No standard storage convention for Differential Mode : confusing and prone to resulting in misuse/wrong answers

Table of Contents

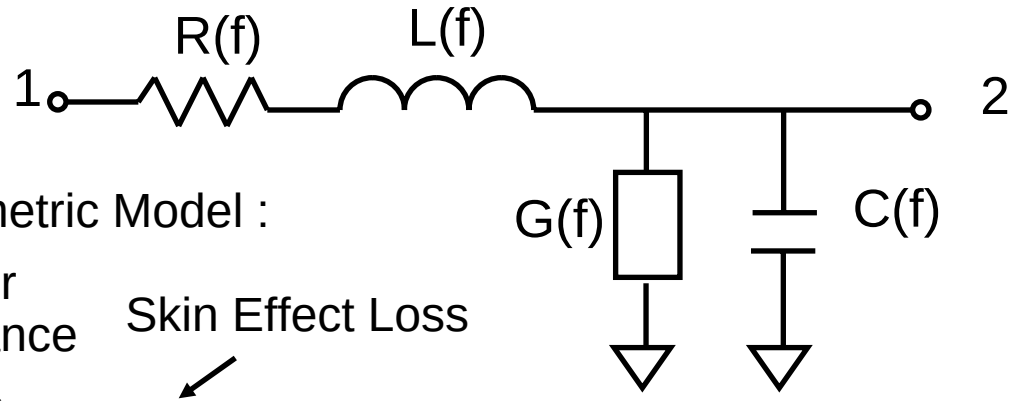
A Parametric RLGC Transmission Line Model





$$Zo(\omega) = \sqrt{\frac{(R+j\omega L)}{(G+j\omega C)}}$$

Characteristic impedance



$$\gamma(\omega) = \sqrt{(R+j\omega L)\,(G+j\omega C)}$$

Propagation Constant

Parametric Model :

Copper resistance

Skin Effect Loss

$R(f) = Ro + sqrt(f) * Rf$

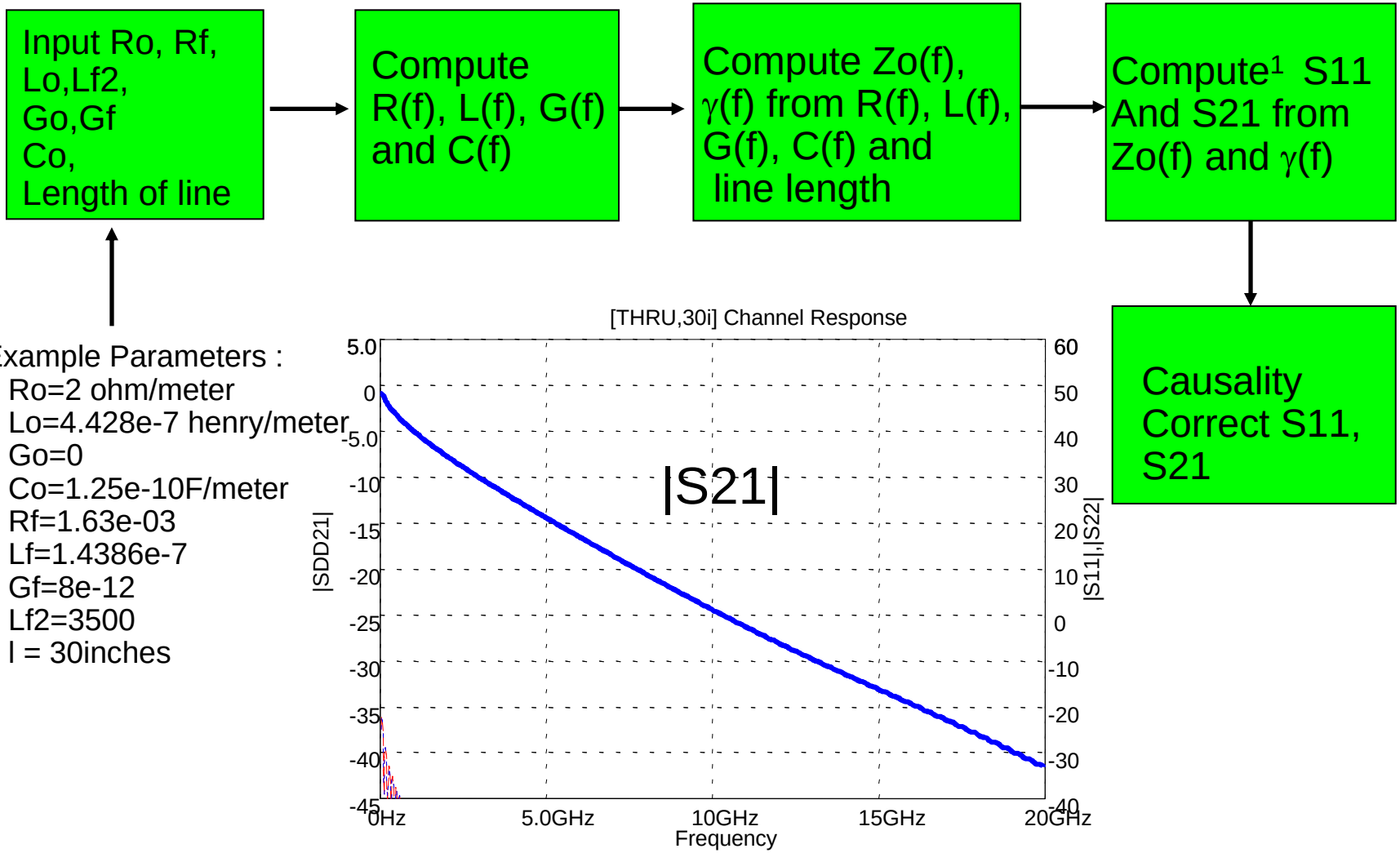
$L(f) = Lo - Lf * sqrt(f) / (Lf2 + sqrt(f))$ ← Lo at low freq, (Lo–Lf) at high freq

$G(f) = Go + f * Gf$

$C(f) = Co$

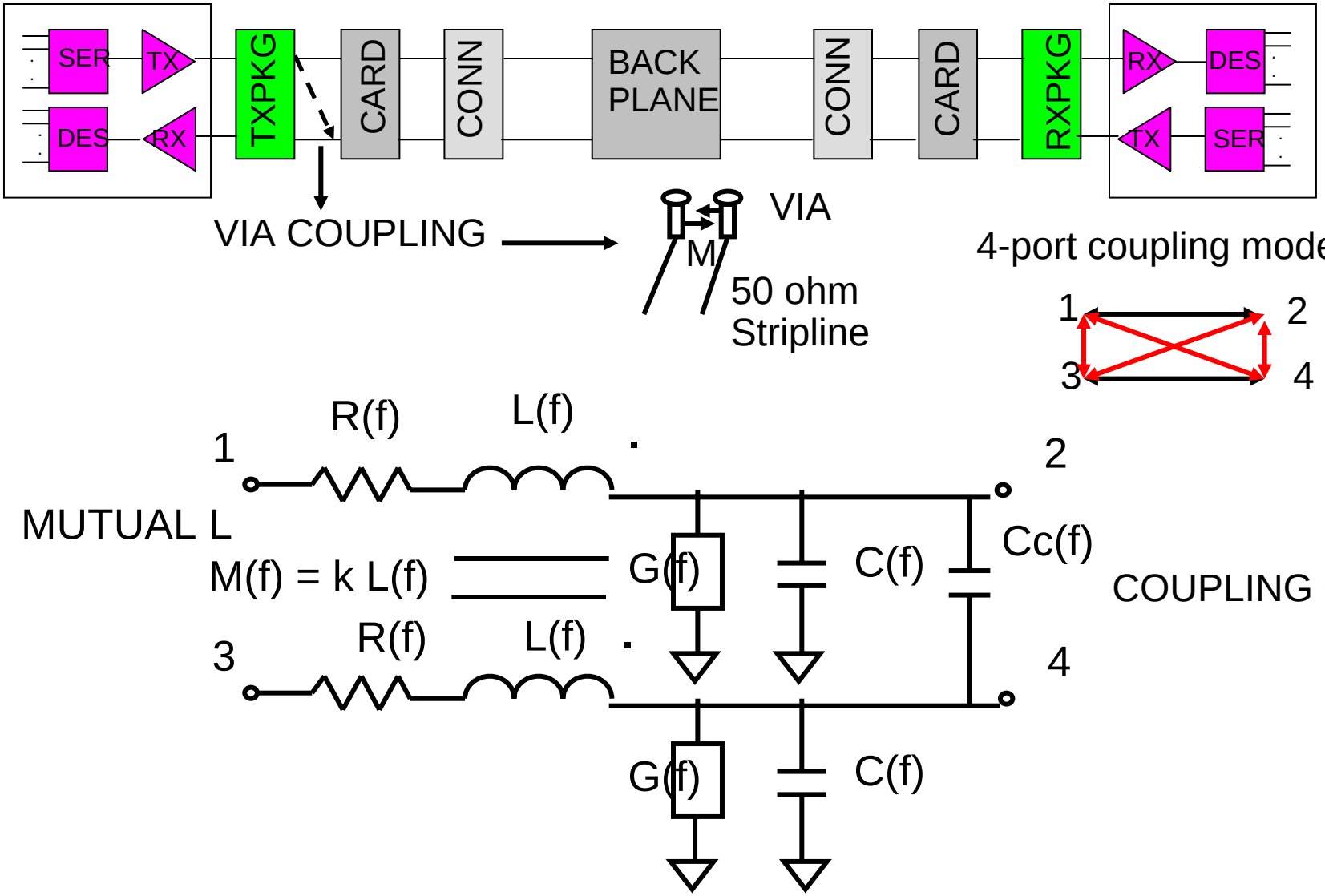
Dielectric Loss

S-Parameter Derivation from RLGC Models



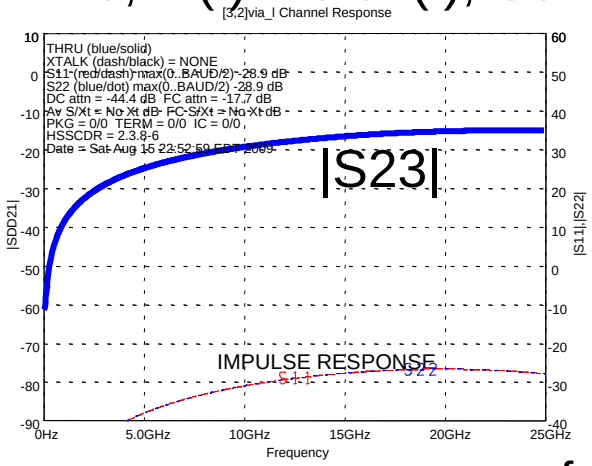
[1]See Appendix A.4 for computation of S11 and S21 from Zo and γ

A Coupled Line RLGC Transmission Line Model



Capacitive and Inductive Coupling Responses

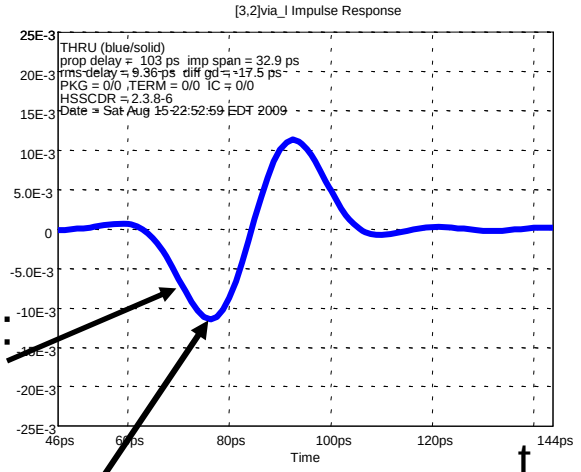
INDUCTIVE COUPLING (VIA)
3mm line, M(f) = 0.3L(f), Cc=0



f

FREQ->TIME

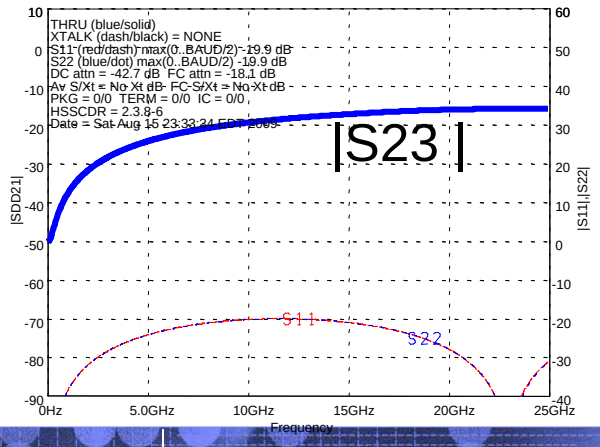
IMPULSE RESPONSE



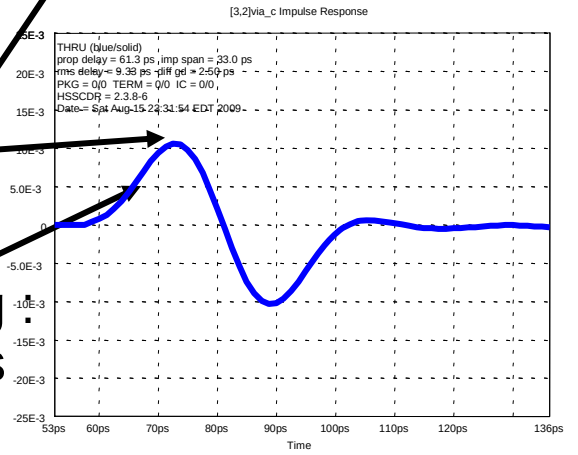
t

Inductive coupling :
Initial impulse falls

CAPACITIVE COUPLING
3mm line, M(f) = 0, Cc=26fF/mm



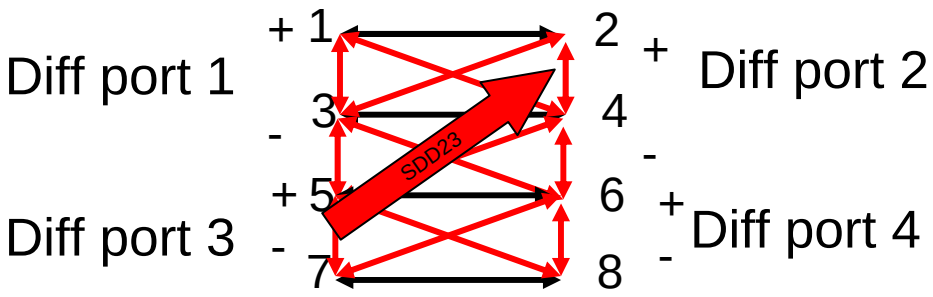
+-11mV Peak



capacitive coupling :
Initial impulse rises

A Differential Signaling Coupling Example

8-port differential coupling model built from 3 4-port coupled lines

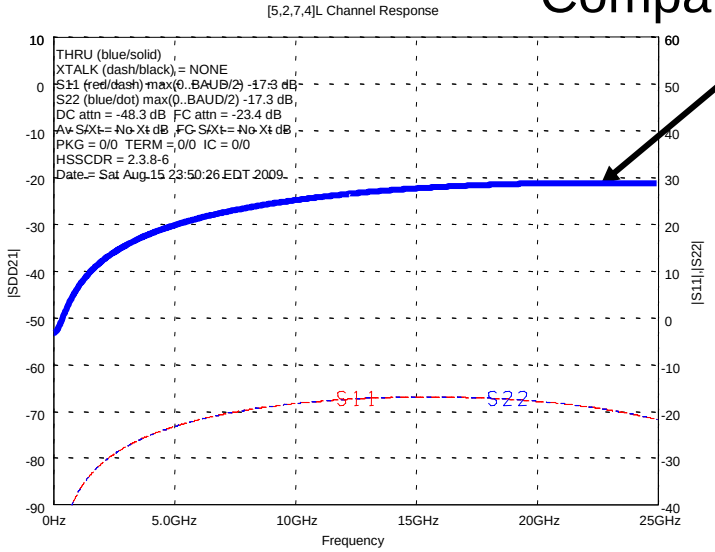


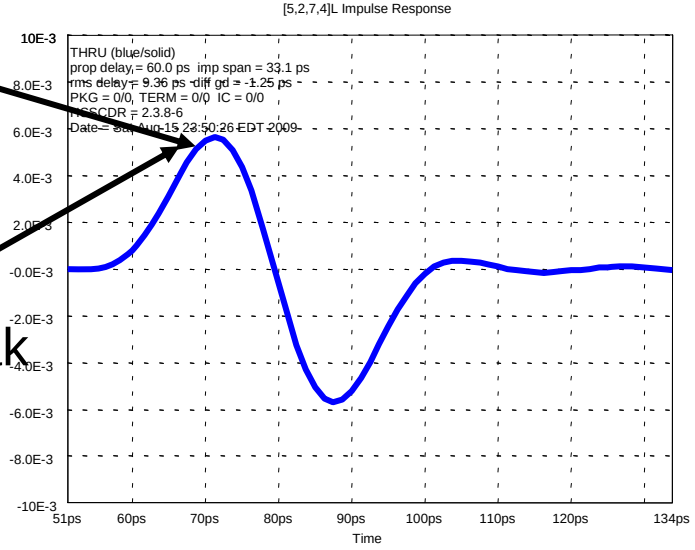
3mm line
M(f) = 0.3L(f), Cc=0

|SDD23|

Crosstalk reduced 6dB
Compared to SE

IMPULSE RESPONSE

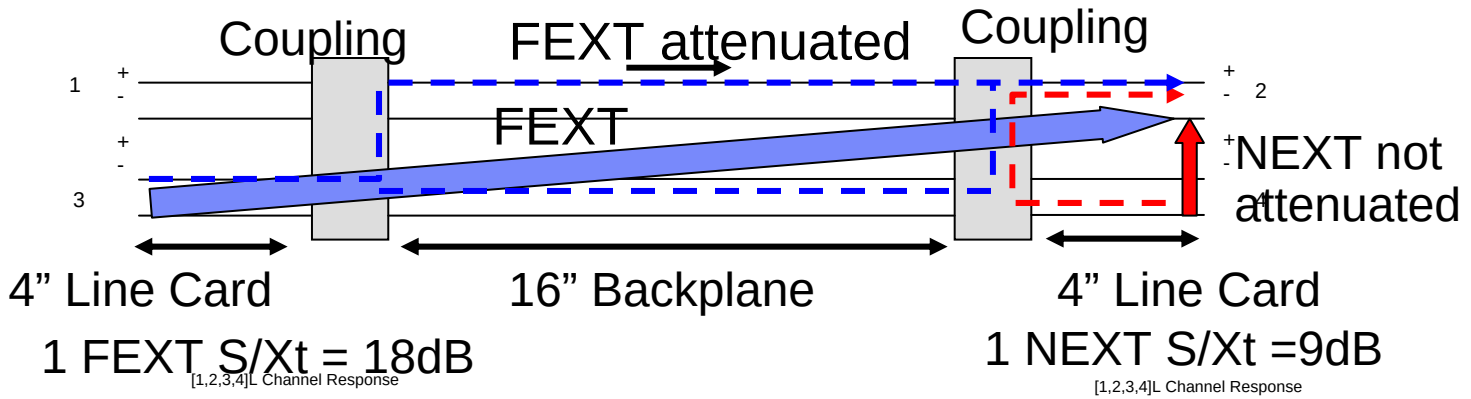


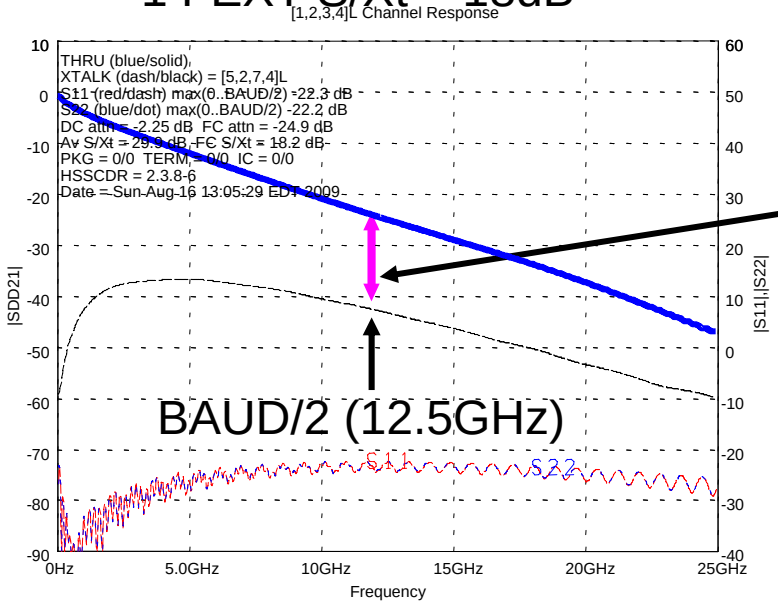


6mVpd Peak
Inverted
Polarity

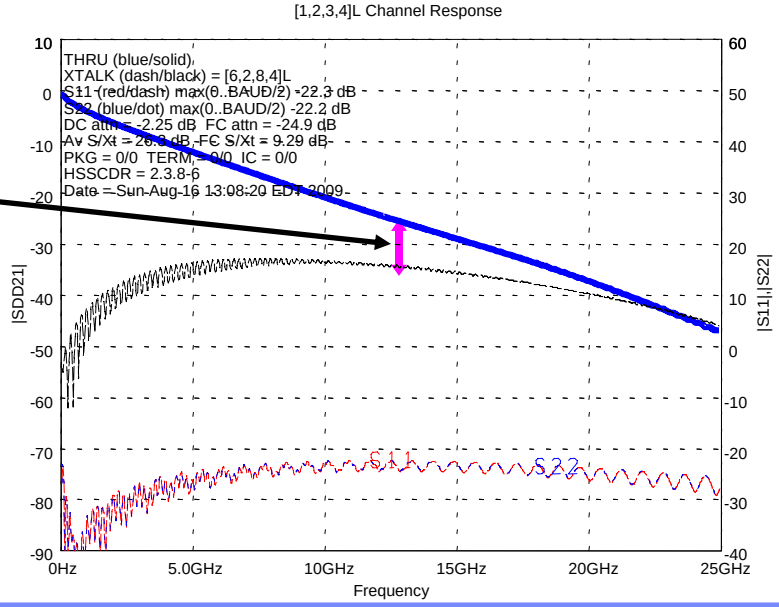
Concatenated 8-Port Net Model with Via Coupling FEXT and NEXT

FEXT is normally much smaller than NEXT since it is attenuated along with signal on line.

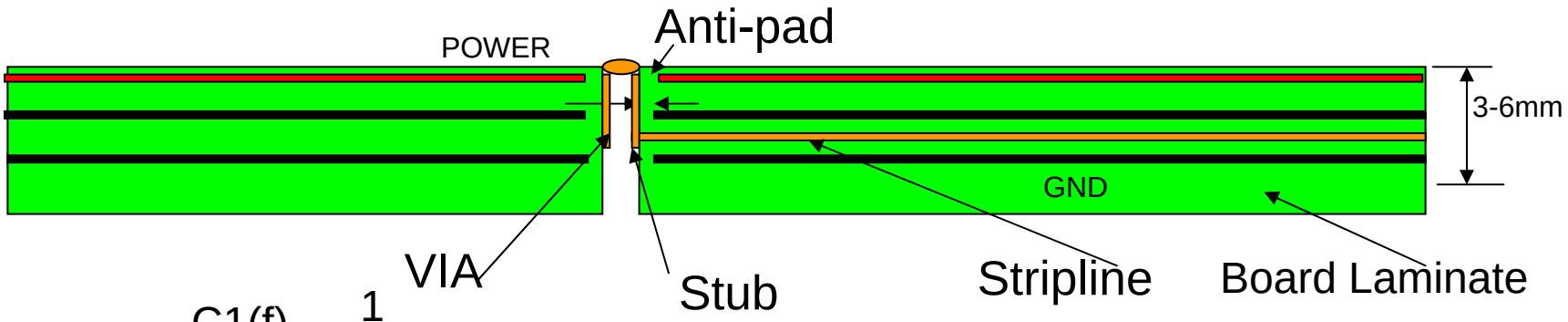


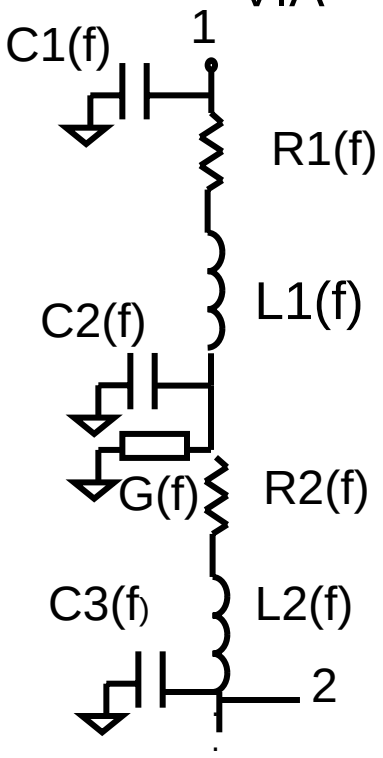


S/Xt

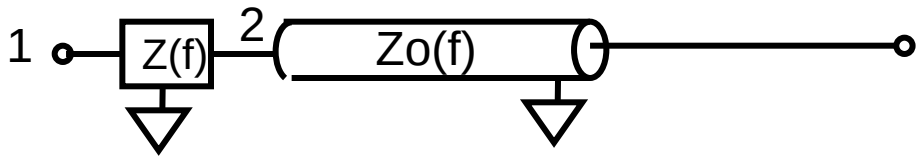


Impedance Discontinuity from Via





Via Impedance Uniform transmission line

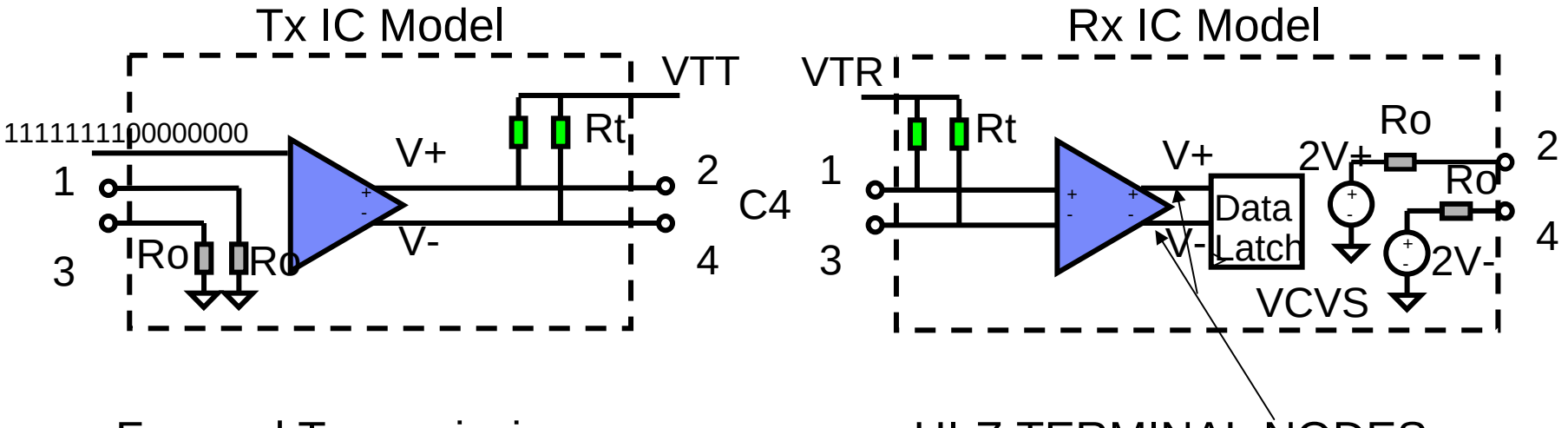


Via is not a uniform transmission line model….
can cause significant impedance discontinuity
if via geometry/stub becomes signficant with respect to
data transition frequency.

Closely spaced vias in a “via field” “breakout region”
can also introduce significant crosstalk.

Via length and number must be minimized in
high data rate systems!

IC Modeling with S-Parameters



Forward Transmission :
- Derive S21/S42 from
  - Step Response
    (111111100000000 data)

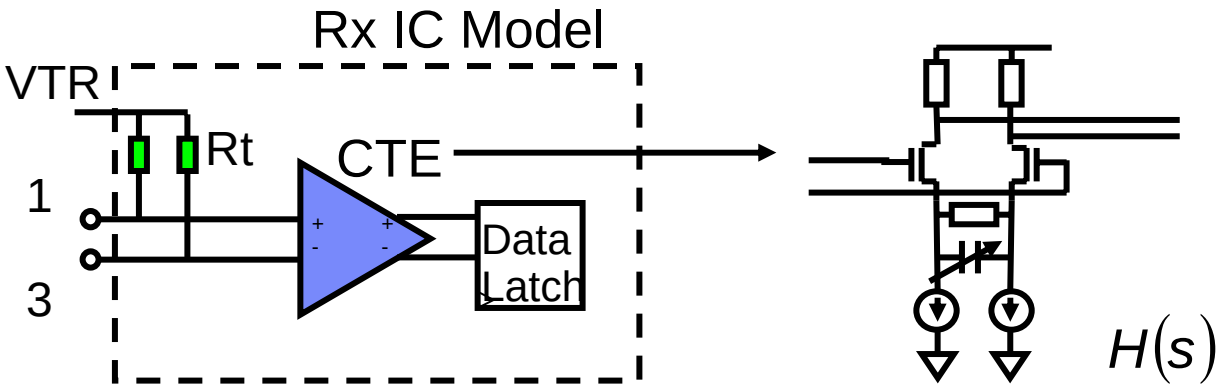
Output Reflection :
- Measure 2-port S-parameters at ports 2 and 4 to get S44, S22, S42, S24

HI-Z TERMINAL NODES

S-Parameters Computation :
- Probe latch with ideal VCVS terminated in Ro

Measure 4-port S-Parameters

Pole-Zero Behavioral Models for IC Response Modeling



3P1Z CTE Model :

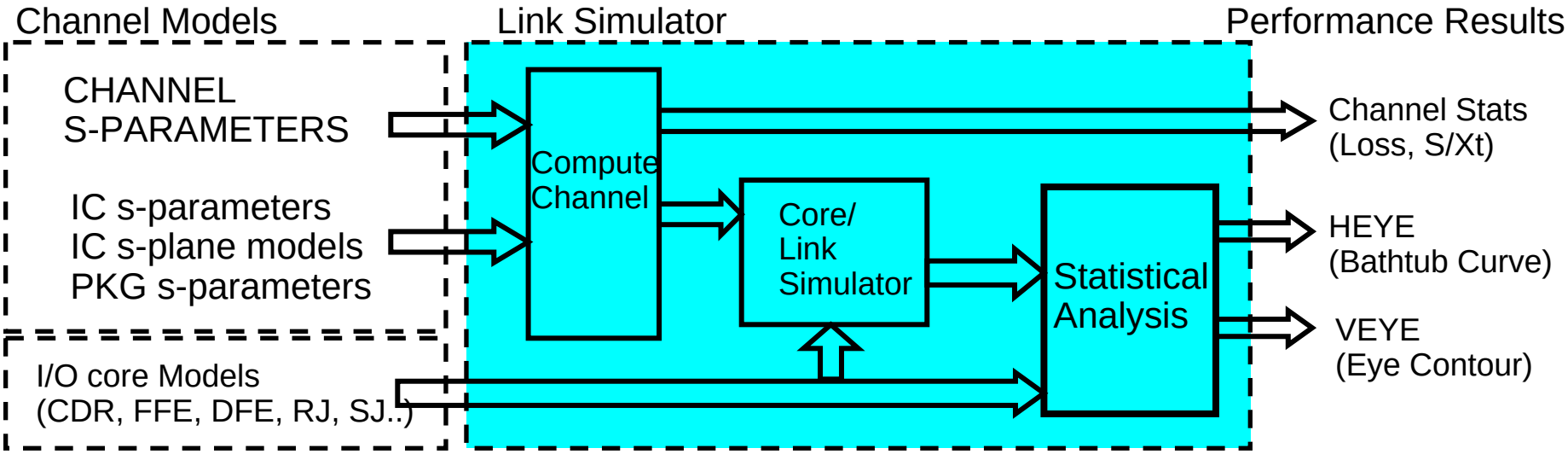
$$H(s) = \frac{(s - b_0)}{(s - a_0)(s - a_1)(s - a_2)}$$

Pole-Zero models are useful to represent programmable Continuous-Time Equalizers (CTEs) in SERDES ICs. In discrete-time simulators, they are easily converted to time domain digital filters via the (bilinear) Z-transform.

Pole-Zero models can also be fit to s-parameter responses, to enable transient simulators to incorporate s-parameters models.

Table of Contents

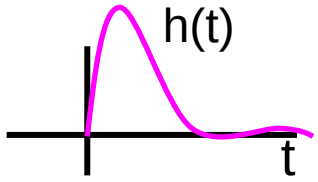
System Performance Analysis

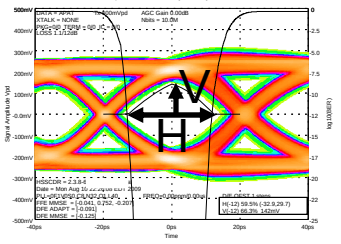


System Performance Analysis steps performed by Link Simulator :

1) Compute end-to-end channel response from channel models :
   -Cascade S-parameters
   -Combine Pole-Zero models, if any, with S-Parameters
   -Determine time-domain impulse response of channel

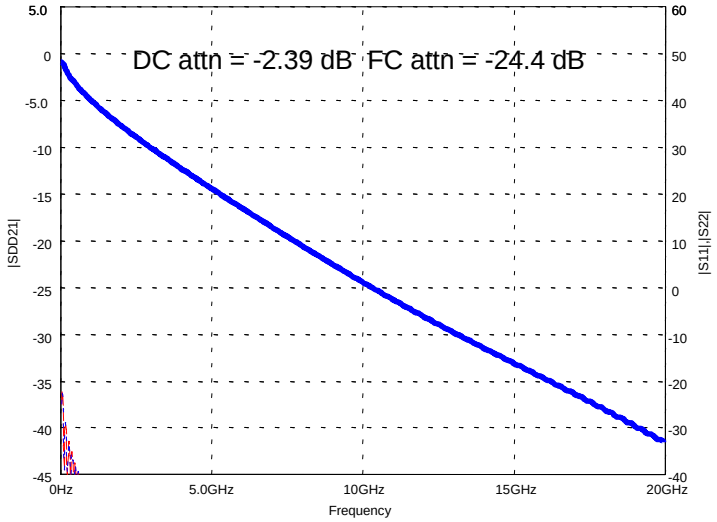
2) Determine HEYE and VEYE from channel impulse response combined with I/O core behavioral model using either pure statistical analysis, or discrete-time system simulation



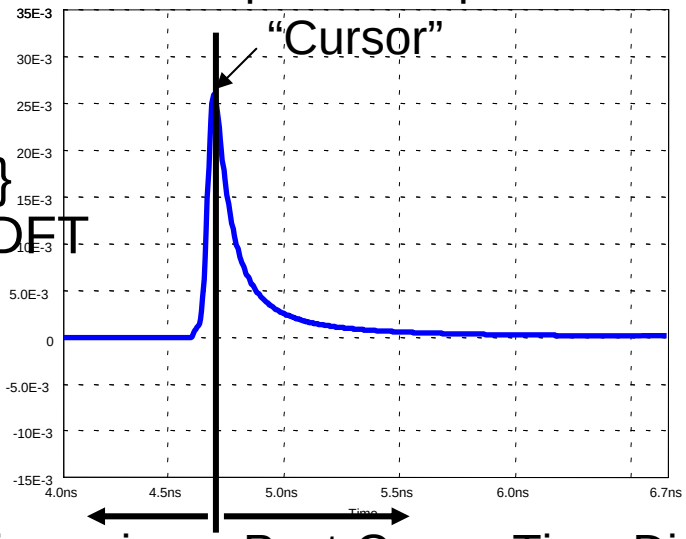


Time Conversion Example : Impulse Response of 30” RLGC T-Line

[THRU,30i] Channel Response



Impulse Response



$\rightarrow F^{-1}\{\}$

F^{-1}= iDFT

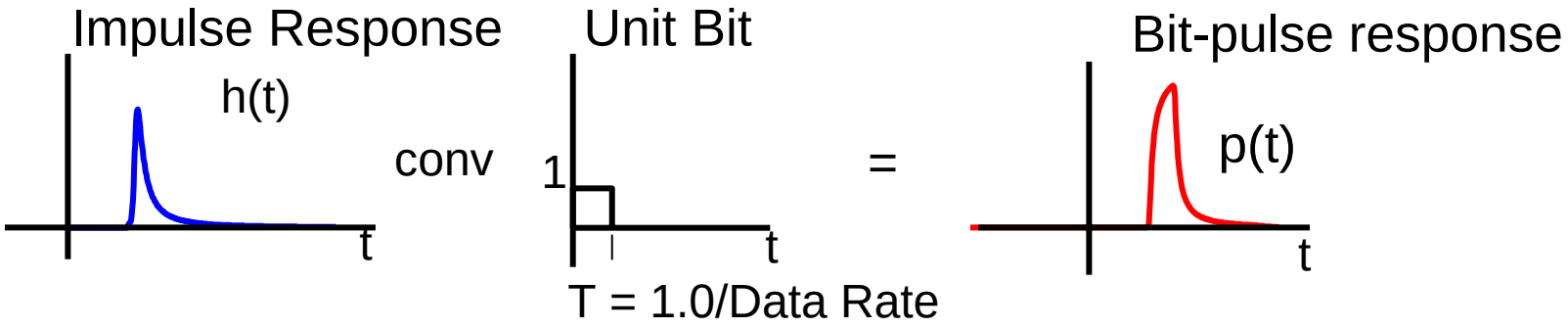
Pre-Cursor Time Dispersion Post-Cursor Time Dispersion

Sanity checks in time conversion :

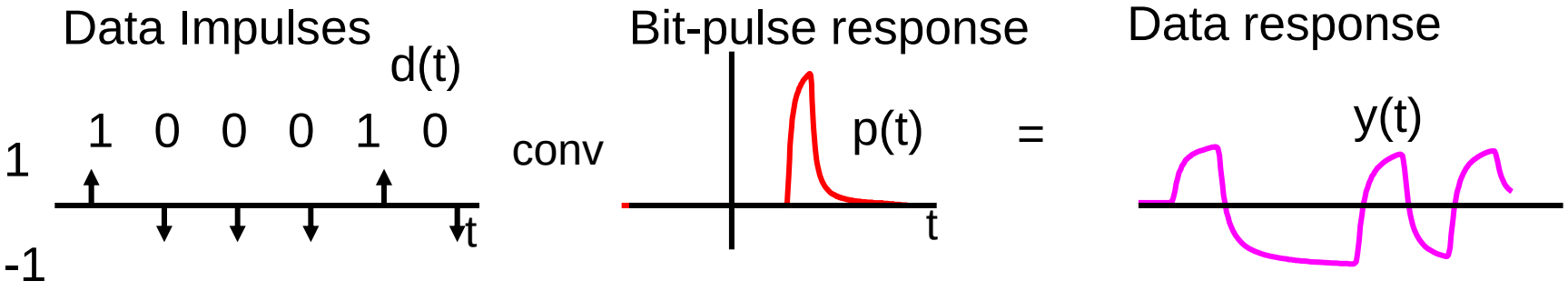
1) No significant energy in impulse for t << Tcursor
   (indicates possible causality problem in data)
2) No significant energy in impulse for t >> Tcursor
   (indicates possible aliasing into negative time/phase undersampling)
3) Rise time normally >> Fall time in a long T-line
4) Back-translate to frequency domain via FFT to compare accuracy of transform

Bit Pulse Response of Channel

Bit Pulse response = Channel Impulse Response convolved with unit bit

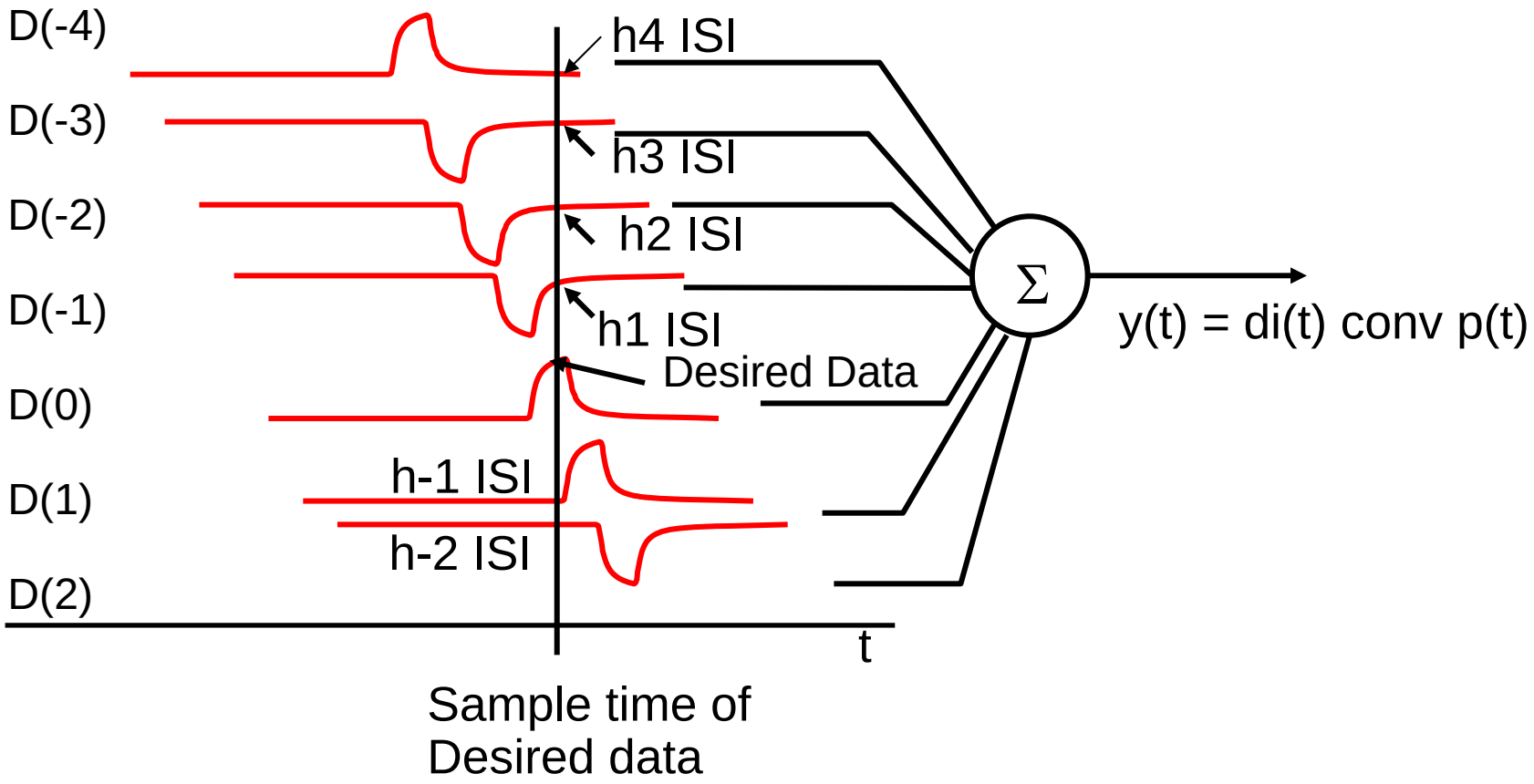


Data response is generated by convolving signed data (impulses) with bit-pulse response. This convolution is computationally efficient and forms the basis for a fast discrete time simulator.

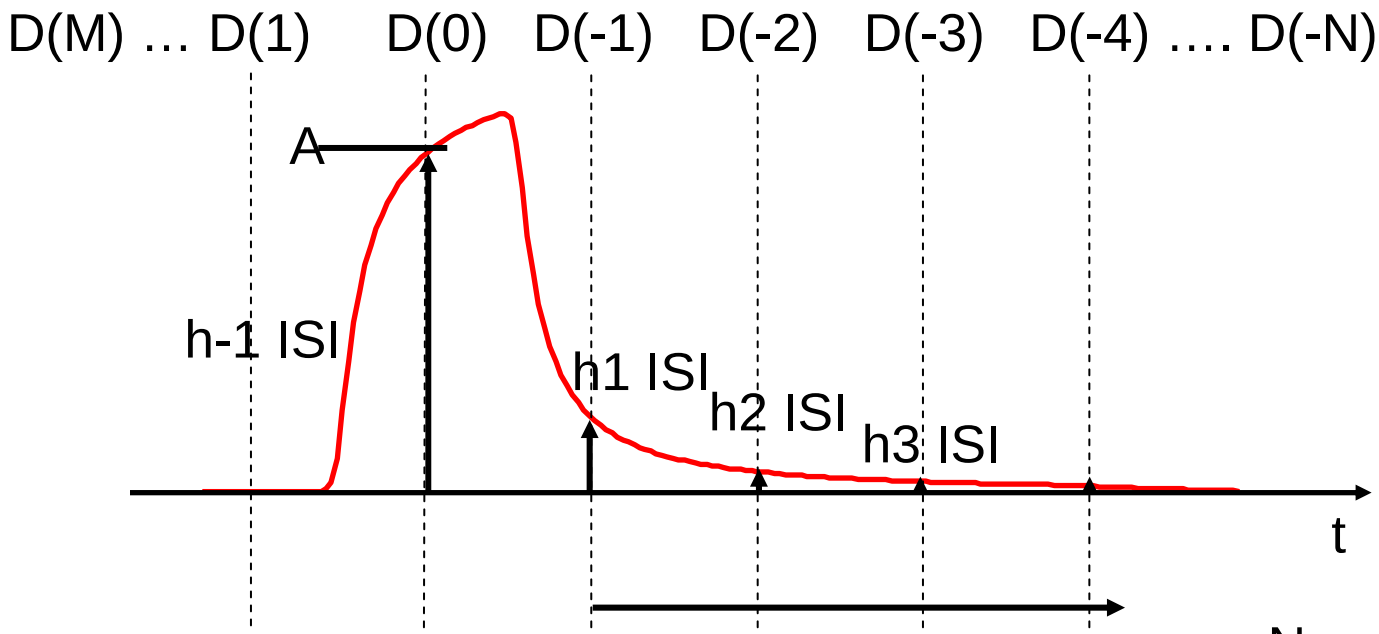


Data transmission viewed as a superposition of signed bit pulses

The transmitted data stream under a linear time-invariant (LTI) channel assumption is a superposition of time-delayed data-signed bit pulse responses.The ISI at data detection time is deterministic and can be derived directly from the bit-pulse response.



Derivation of Inter Symbol Interference (ISI) from Bit-Pulse Response



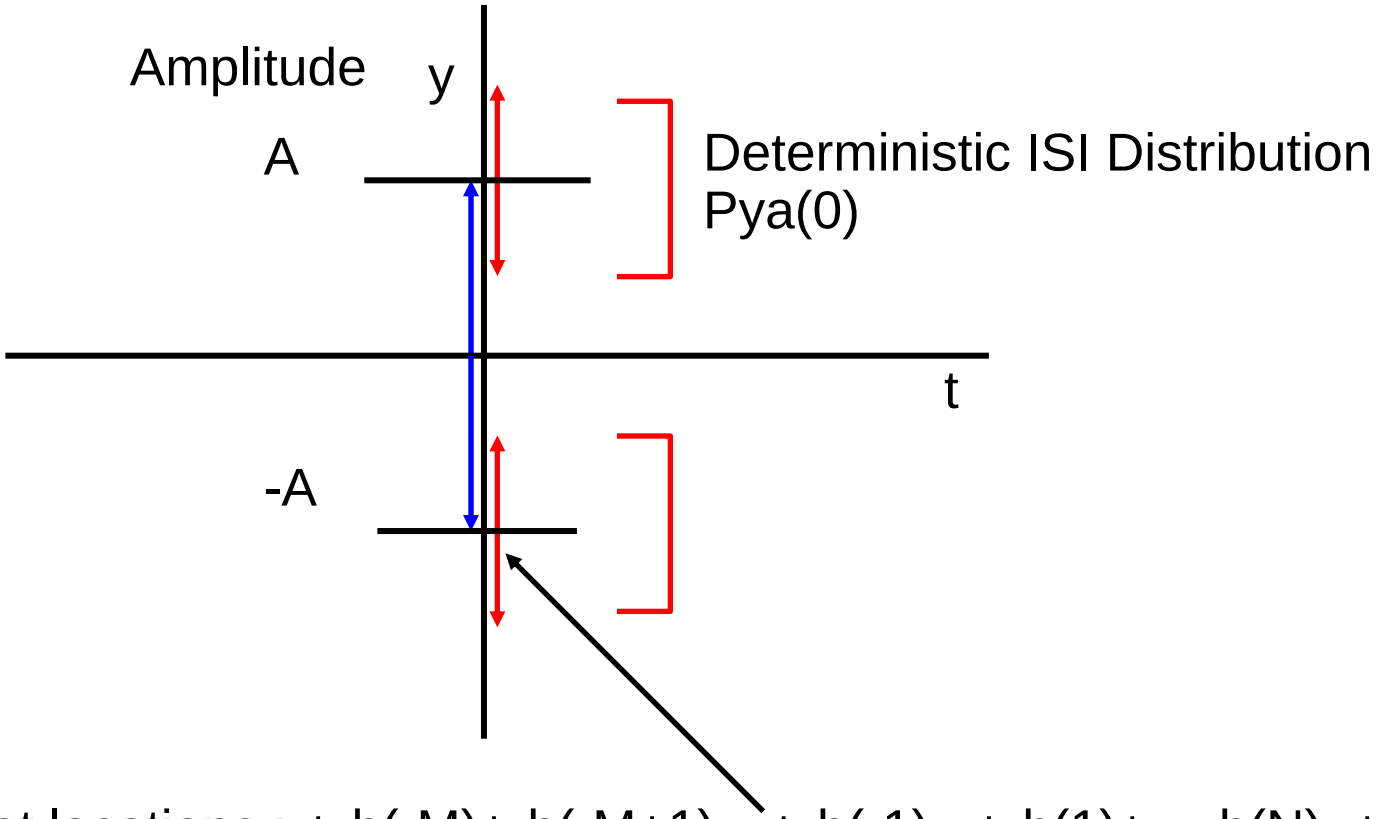
$$\text{POST-CURSOR ISI} = \sum_{j=1}^{N} h(j)\, D(-j)$$

DESIRED SIGNAL = D(0) h(0)

$$\text{PRE-CURSOR ISI} = \sum_{j=1}^{M} h(-j)\, D(j)$$

"Vertical Eye" Deterministic ISI Statistics

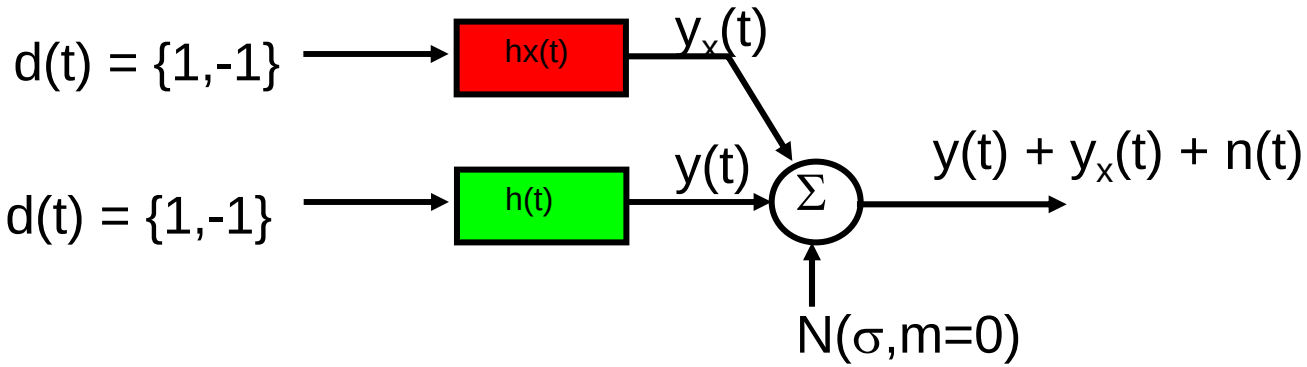
The amplitude distribution from ISI can be computed from the bit-pulse response by forming all possible combinations of ISI sums.



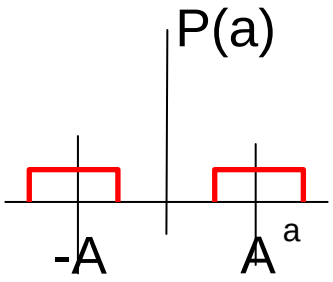
ISI Occurs at locations : +-h(-M)+-h(-M+1)…+-h(-1) +-h(1)+-…h(N) +- A
= 2**(M + N) discrete ISI points centered about A, -A

Non-Deterministic Noise and Crosstalk Noise Addition

Non-Deterministic amplitude noise and crosstalk noise can be added to the deterministic ISI distribution by convolving the deterministic distributions together with a Gaussian noise distribution

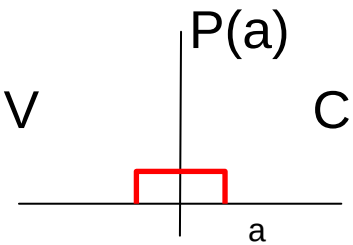


Deterministic Data ISI Distribution



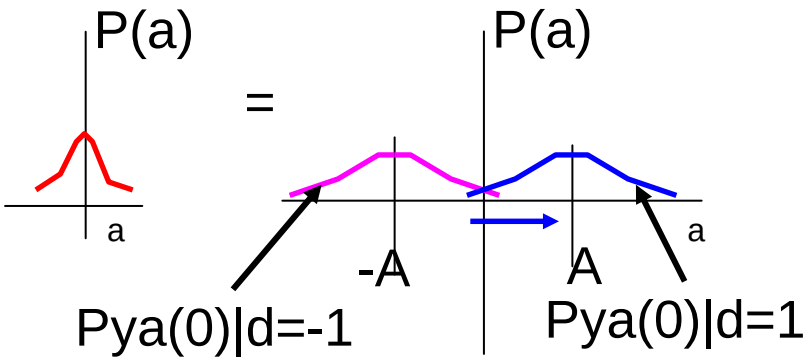
CONV

Deterministic Crosstalk Noise Distribution



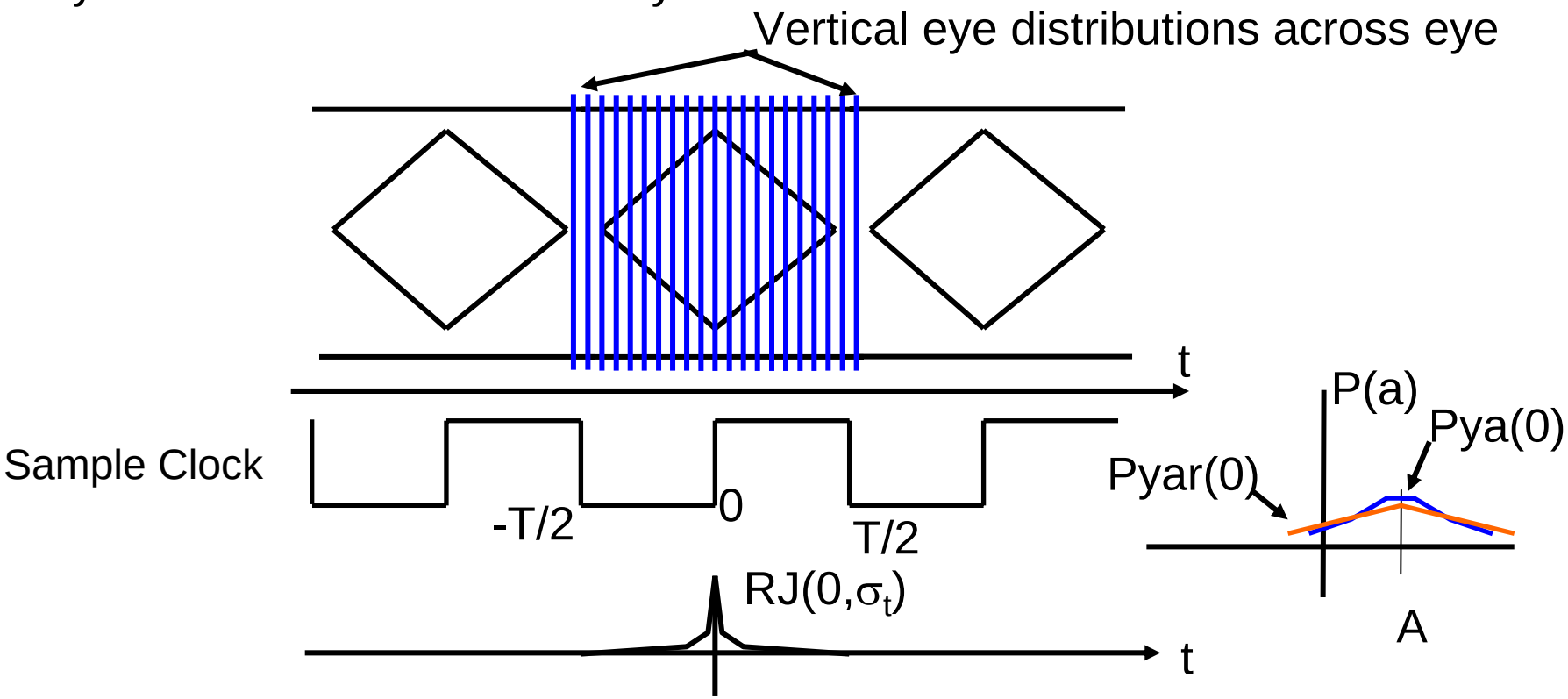
CONV

Non-Deterministic Noise Distribution



Sample Time Jitter Addition to Vertical Eye Statistics

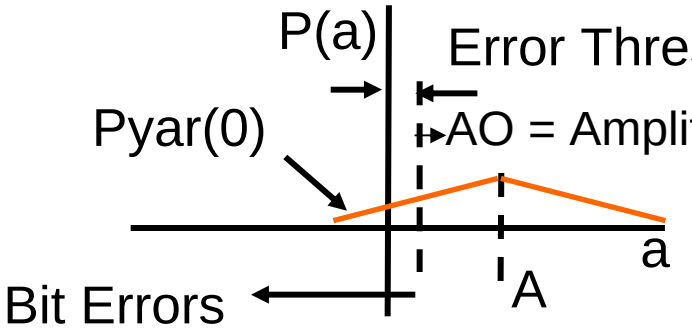
The effect of clock jitter is incorporated into the HEYE and VEYE analysis by computing time-jitter weighted averages of the vertical eye distributions across the eye



For center vertical eye distribution : $Pyar(0) = \sum_{t=-T/2 \text{ to } T/2} Pya(t)\, RJ(t)$

HEYE and VEYE Computation

The final vertical amplitude distribution at eye center (t=0) is Pyar(0)



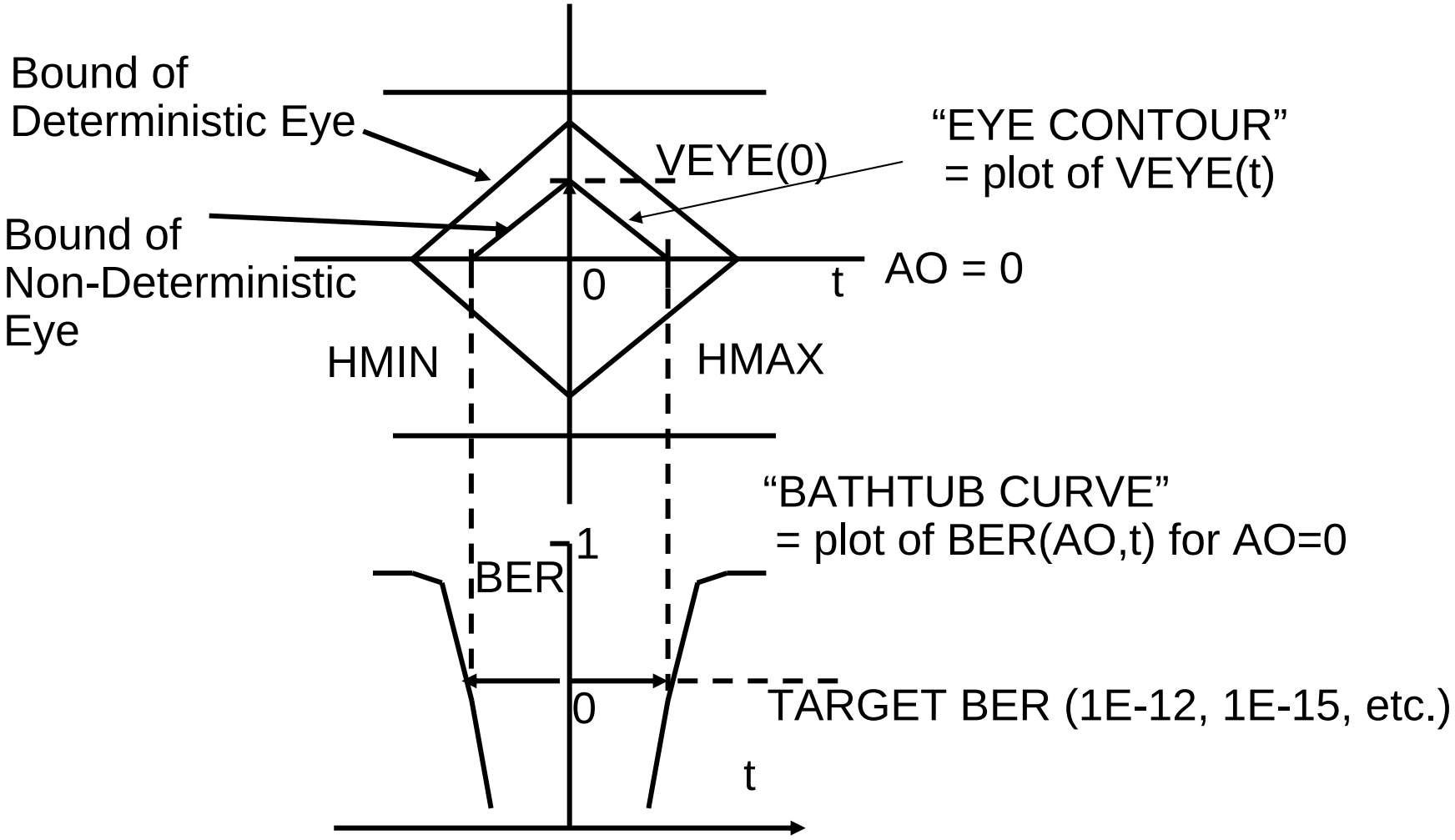
$$BER(AO,t) = \int_{-\infty}^{AM+AO} Pyar_t(a)da$$

VEYE(t) = AO which results in BER(AO,t) = target BER
VEYE = VEYE(0)

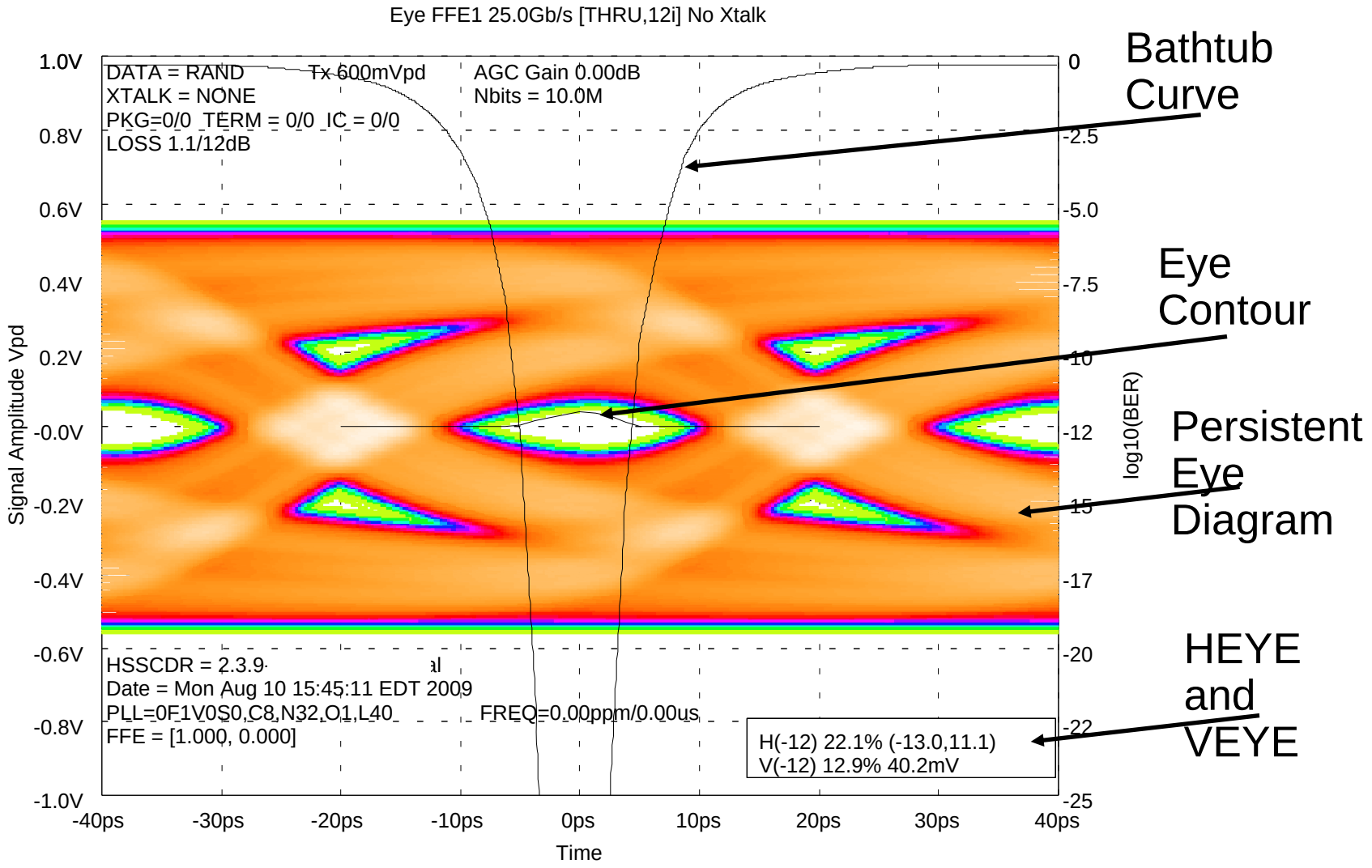
HMIN = t which results in BER(0,t) = target BER for t < 0
HMAX = t which results in BER(0,t) = target BER for t > 0
HEYE = 2 * min(|HMIN|,HMAX) (Minimum P-P Jitter Tolerance)
HEYEPP = HMAX-HMIN (Maximum P-P Jitter Tolerance)

Eye Contour and Bathtub Curve

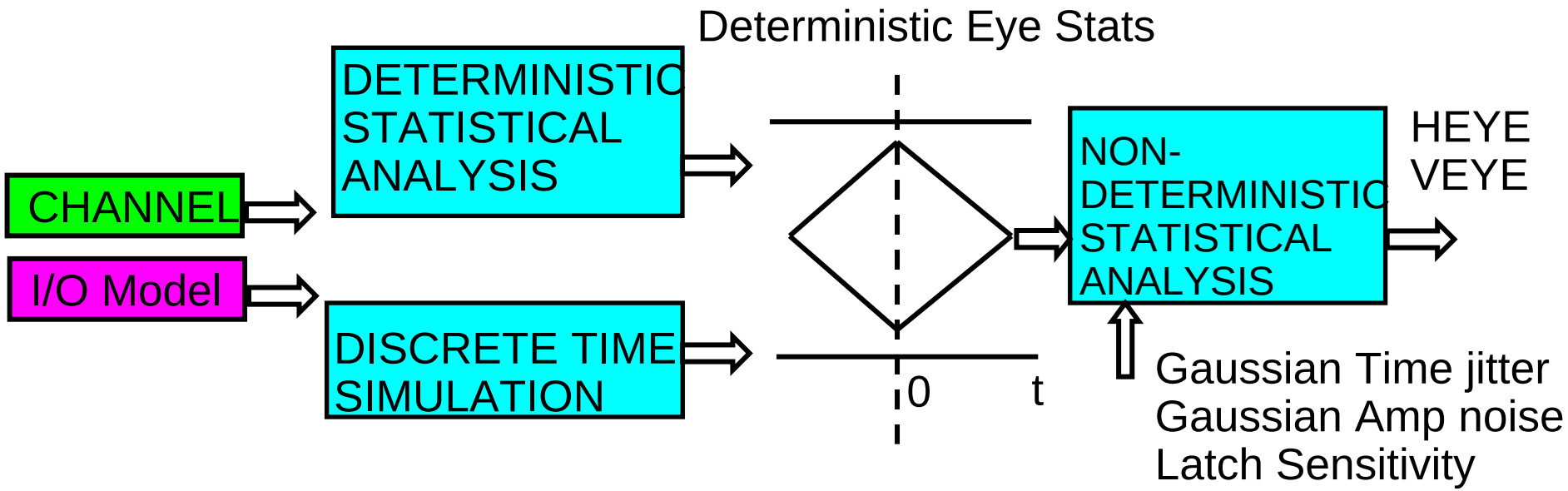
The eye contour directly shows vertical eye margin at a desired BER confidence
The bathtub curve directly shows horizontal eye margin at a desired BER confidence



Final Goal! Eye Diagram + Bathtub + Eye Contour + HEYE + VEYE



Statistical Analysis vs. Discrete-Time Simulation Analysis



Advantage of Discrete-Time Simulation analysis :

+Automatically incorporates “algorithmic jitter” of receiver clock-recovery function

+Can add non-linear compression easily if desired

+Can easily find effect of special data patterns (8/10 code, etc.)
no assumption of uniform random data needed

Disadvantage of Discrete-Time Simulation analysis :

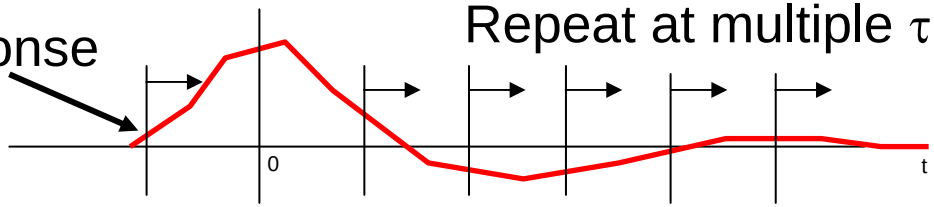
-Slower : must simulate 10M bits or more before statistics start to converge

-Does not guarantee coverage of all data combinations, although “worst case” data combinations can be “forced” into the discrete-time simulation

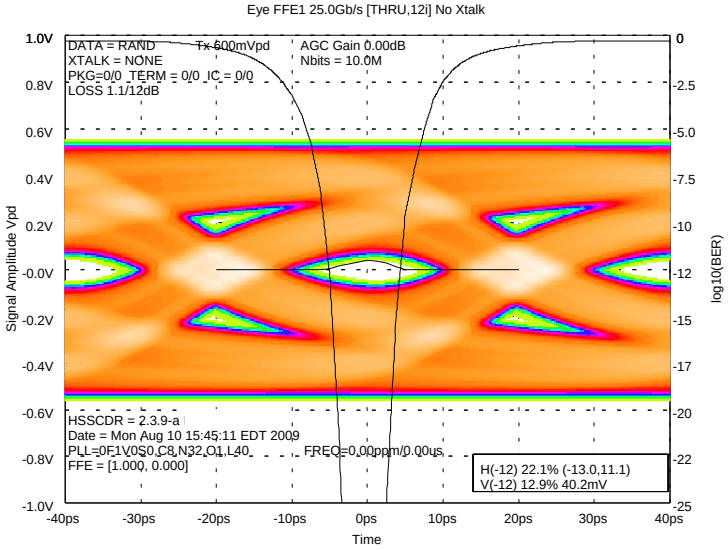
Worst-Case Data Pattern Analysis

"Worst Case" data patterns exist for both in-channel ISI and crosstalk excitation. They are found by lining up data bit polarities with the sign of the bit-pulse responses.

Bit Pulse Response (Thru or Xtalk)



"Random" Data



"Worst-Case" ISI

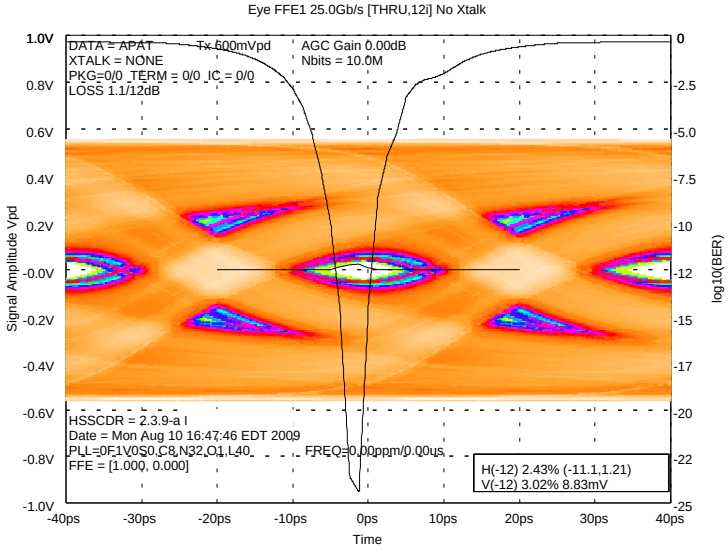


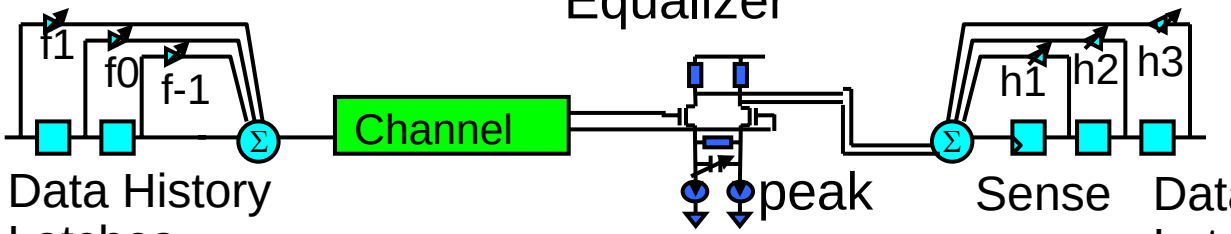
Table of Contents

Line Equalization Building Blocks

Feed-Forward Equalizer

Continuous Time Linear Equalizer

Non-Linear Decision-Feedback Equalizer



Data History Latches

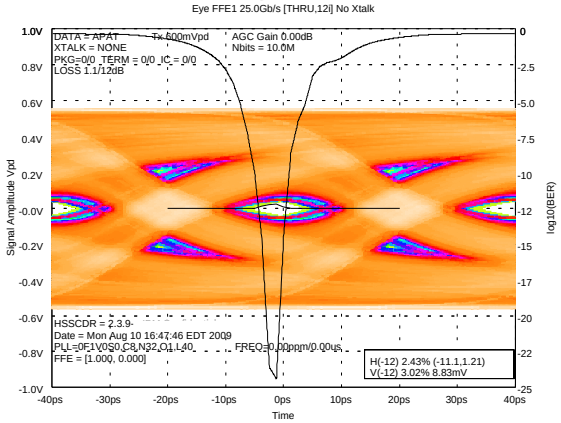
Sense amp

Data History Latches

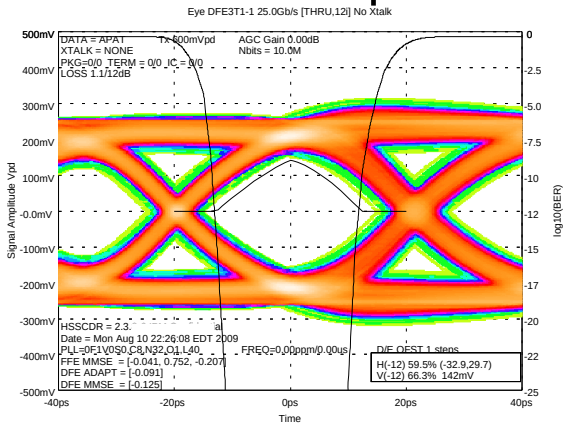
Tx Pre-Distortion

Rx Post Equalization

Closed eye at Rx input
HEYE = 0%



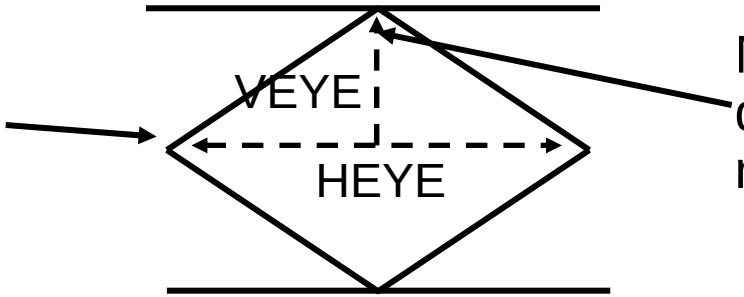
Open eye at Rx Sense amp
HEYE = 60% Open



General Equalization Error Criteria for NRZ Signaling

Equalization Goal

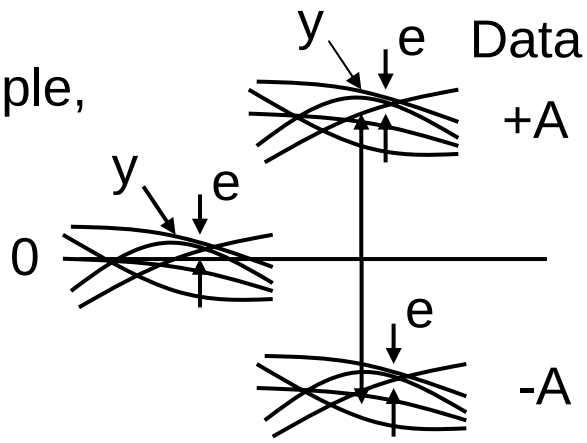
Minimize error at Edge crossover to Maximize HEYE



Minimize error at data sample time to maximize VEYE

HEYE and VEYE Error Criteria:

Edge Sample, Drive to 0



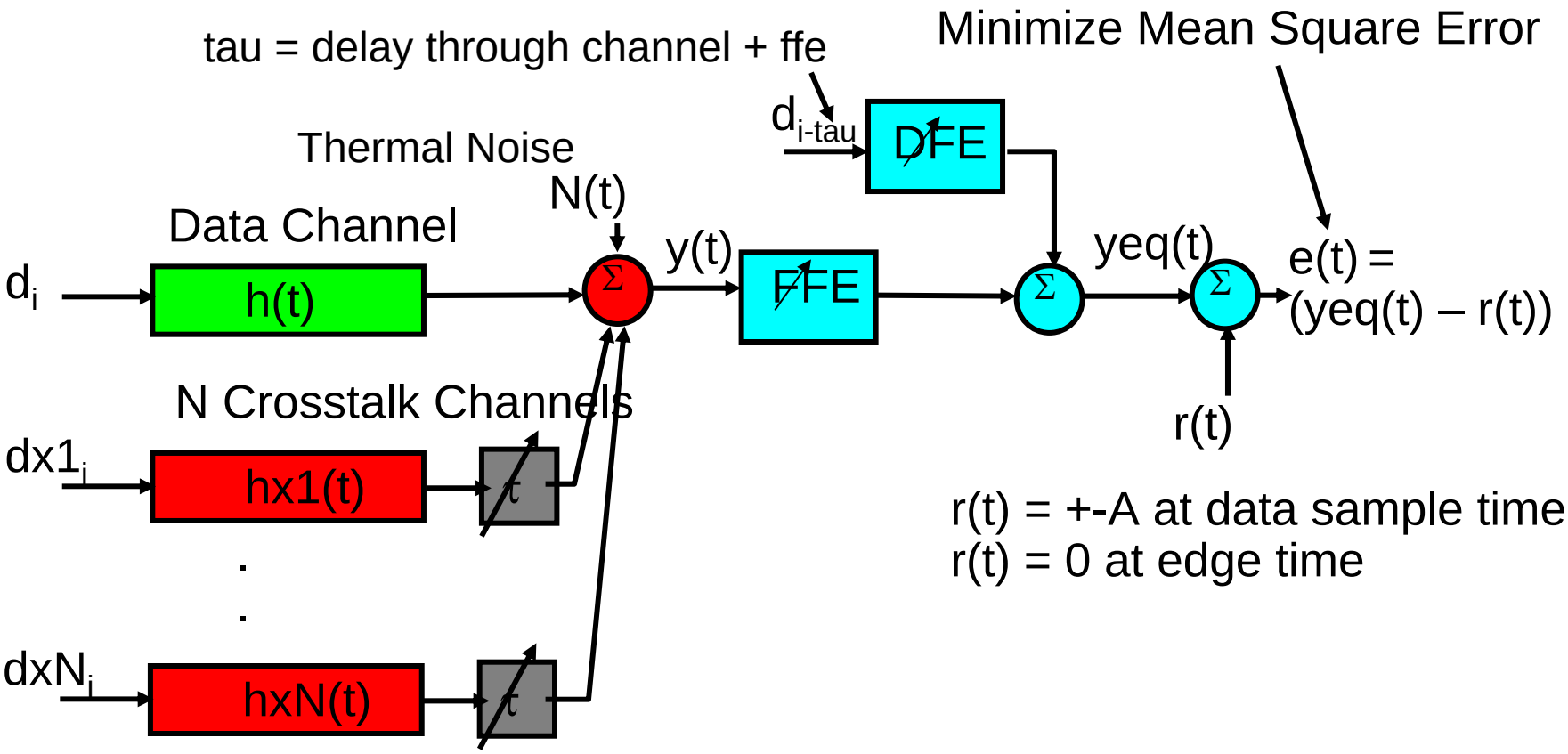
Data Sample, Drive to +-A

+A

-A

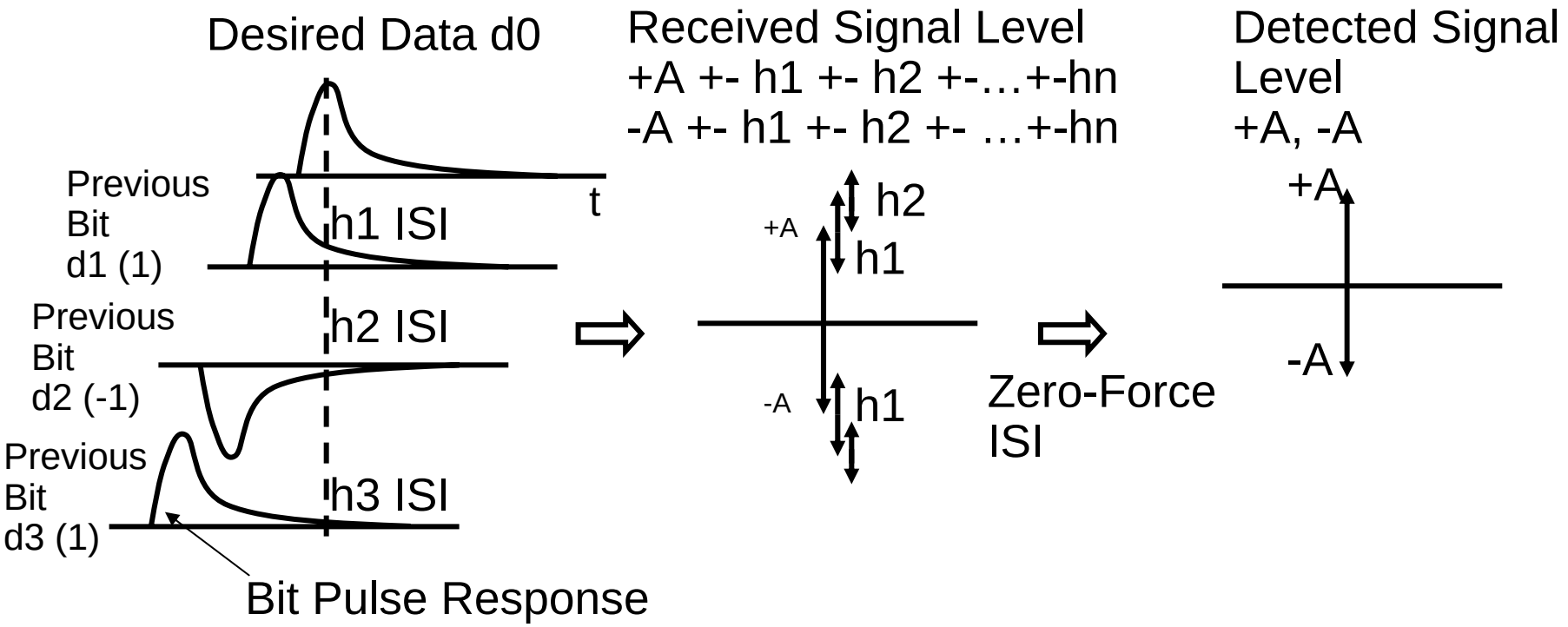
Minimum-Mean-Square-Error (MMSE) Equalization

MMSE Equalization finds FFE/DFE taps to minimize a mean-square error metric at the output of the equalizer



Zero-Force ISI Equalization

"Zero-Force" ISI minimizes error at data and/or edge crossing[1] through ISI cancellation



[1]T. Toifl, et.al "Low-Complexity Adaptive Equalization for High-Speed Chip-to-Chip Communication Paths by Zero-Forcing of Jitter Components", IEEE TRANSACTIONS ON COMMUNICATIONS, VOL. 54, NO. 9, SEPTEMBER 2006

Zero-Force Equalization Adaptation

"Zero-Force" ISI Error Criteria drives correlation of averaged ISI sign error with data to 0:

$E\{e * d_i\} = 0$ (data history does not correlate with error)

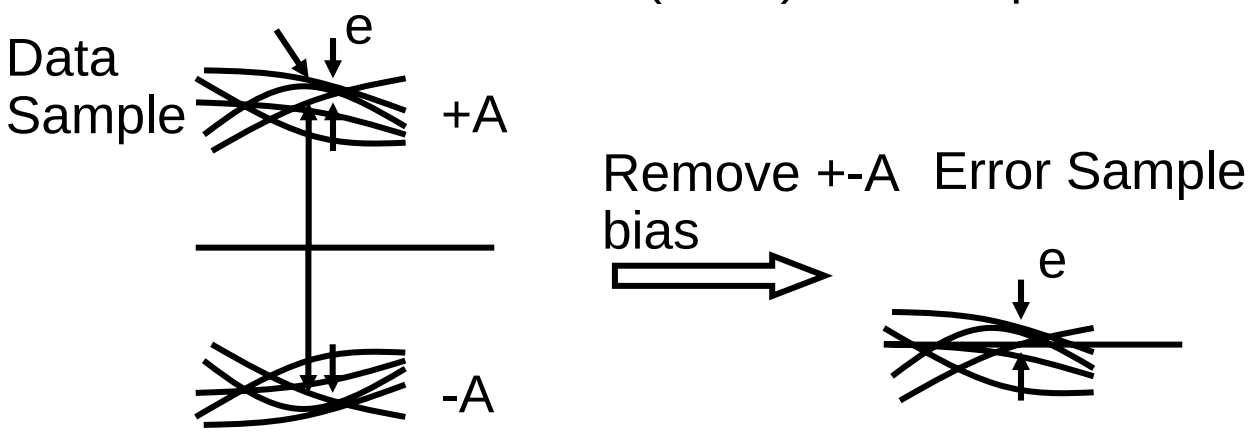
Solution :

$$h_i = \int e * d_i = k \int \mathrm{sgn}(e) * d_i$$

Data History

Small Integration Constant (<< hi)

Limited Level Sample Error

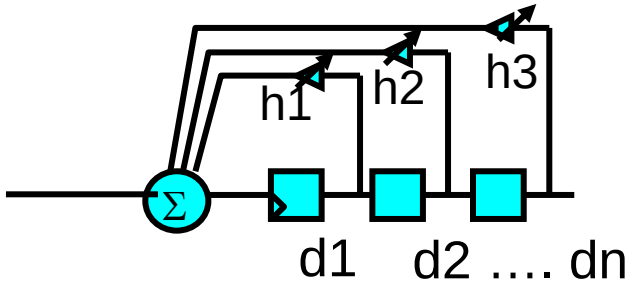


Data Correlation Problem for Zero-Force Adaptation

If two (or more) data bits which are driving a zero-force ISI adaptation are fully correlated, the ISI error is in turn fully correlated between these bits.

As a result :

1) There is no unique solution for single-tap ISI convergence
2) Data correlated tap values can wander around almost anywhere
3) Taps can bias from ISI far removed from actual tap



Convergence to unique solution at tap “i” requires that

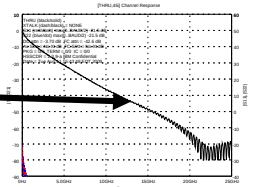
$$E\{di * dj\} < 1 \text{ for all } j <> i$$

-> Do not run adaptive equalization if 100% (or extremely high) data correlation found among data bits!

Equalization of Lossy Channels

Use of receiver CTE provides a bigger received signal, since less de-emphasis needed @ Tx, and also directly helps keep a Tx FFE from running out of post-cursor equalization range.

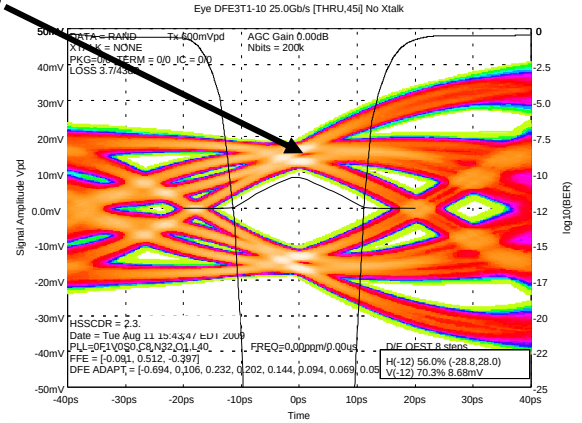
43dB loss at 12.5GHz



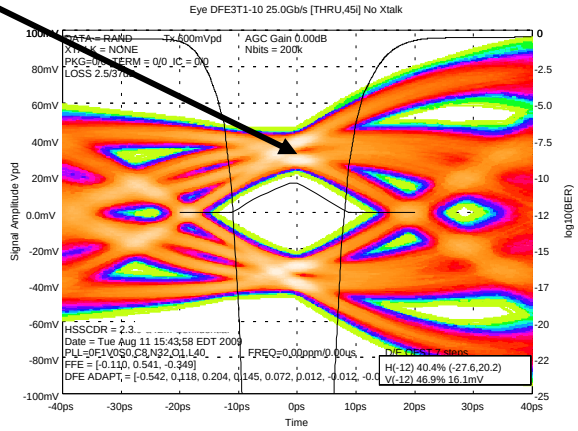
FFE → 43dB Loss → DFE

Large De-Emphasis Needed At Transmitter

15mV signal level with no Rx CTE



30mV signal level with Rx CTE



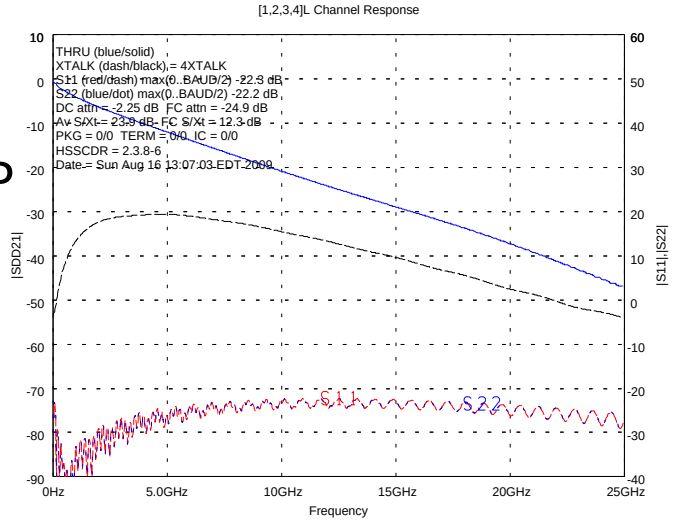
>=6dB Peak Rx CTE

FFE → 43dB Loss → CTE → DFE

6dB less de-emphasis needed at Transmitter

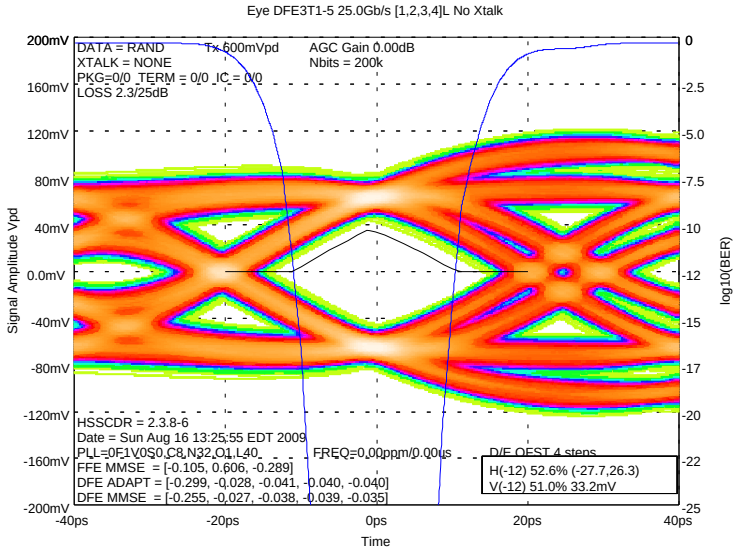
Equalization of High Crosstalk Channels

CHANNEL :
24” 2Connector LR BP
4FEXT model
BAUD/2
LOSS : 25dB
S/Xt : 12dB



I/O :
FFE 3 DFE 5
ASYNC CDR
1% UI RMS RJ
2% Mean VEye AN
25Gb/s

NO CROSSTALK



S/Xt = 12dB : CLOSE TO LIMIT OF OPERATION

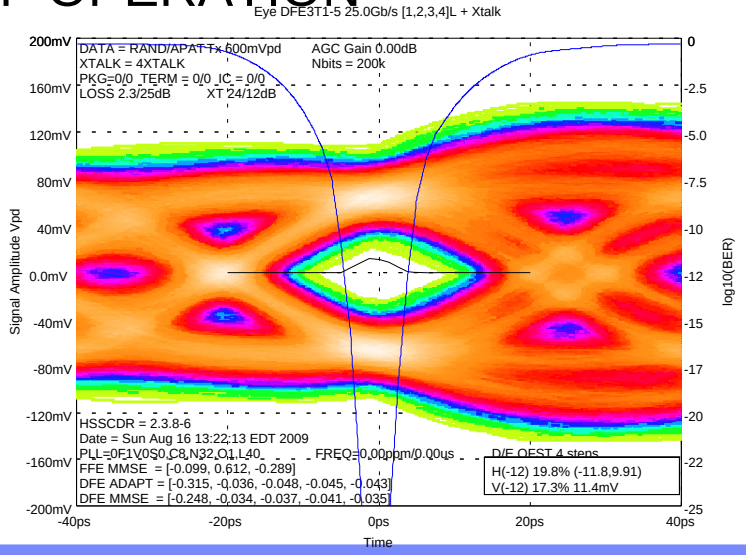
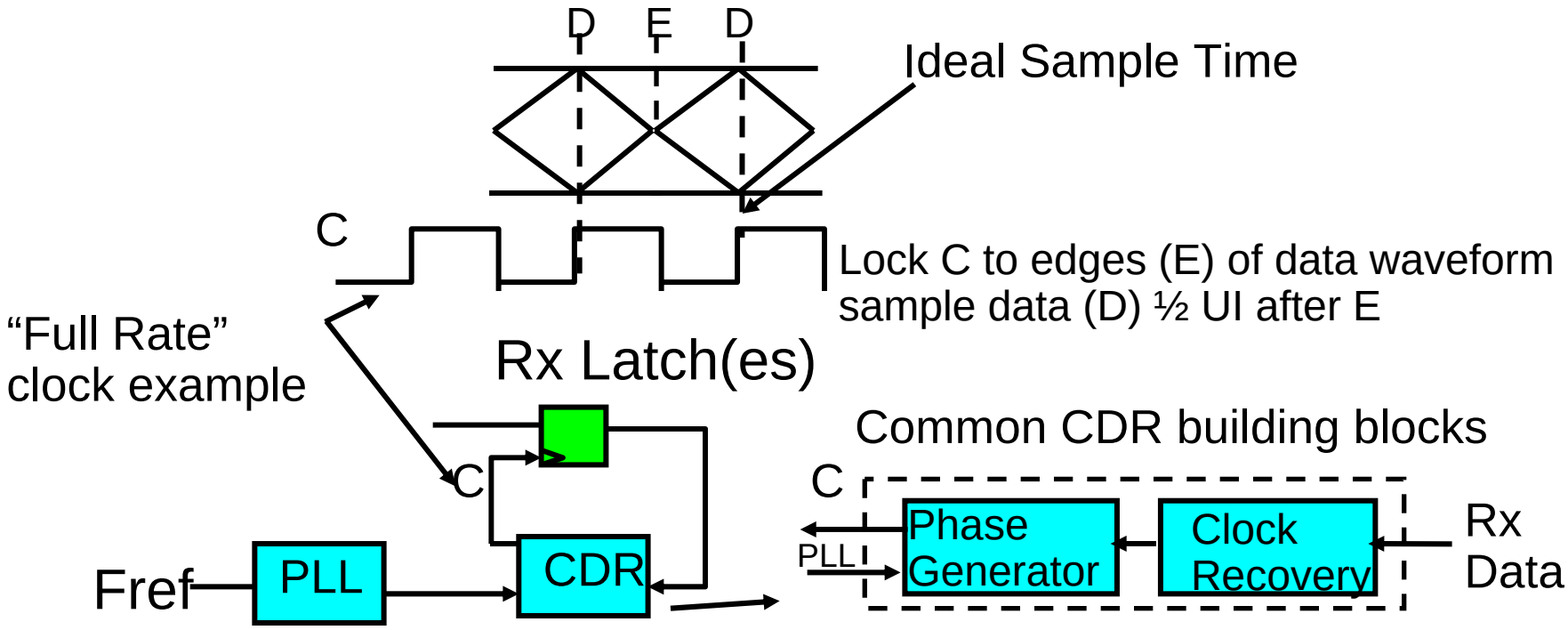


Table of Contents

Clock and Data Recovery

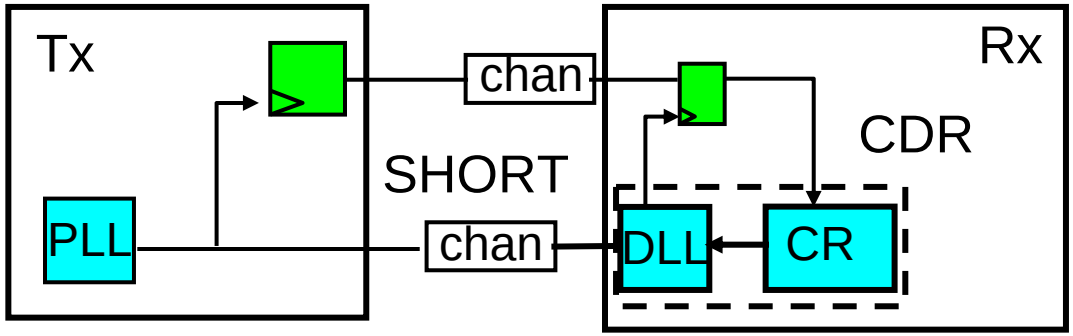
Clock and Data recovery in the receiver determines a sample time for the received signal which provides optimum data detection.



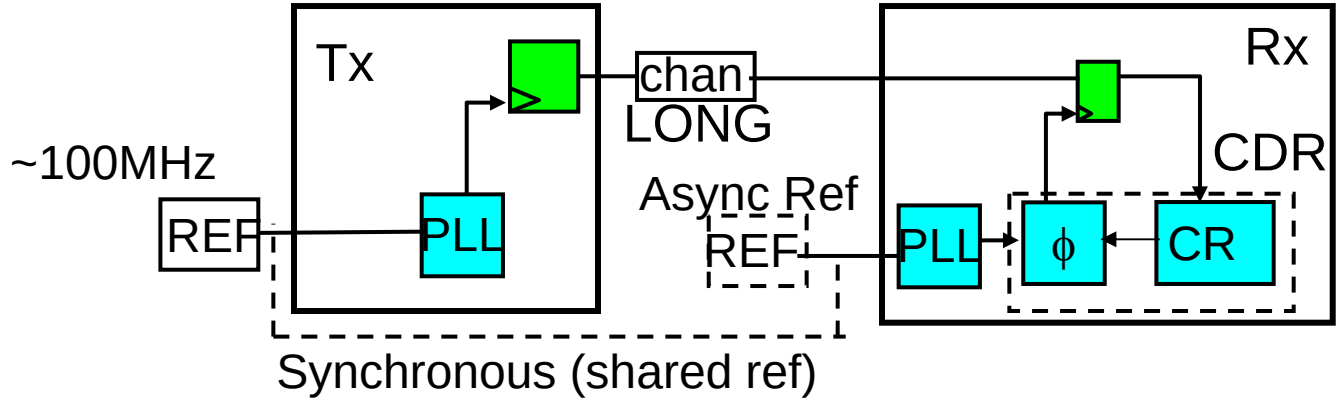
Synchronous and Asynchronous CDR Systems

Transmit and Receive clocks may be synchronous via forwarded clock, synchronous via PLL locked to a common reference, or fully asynchronous

Forwarded Clock (source synchronous) (Memory systems or C2C)

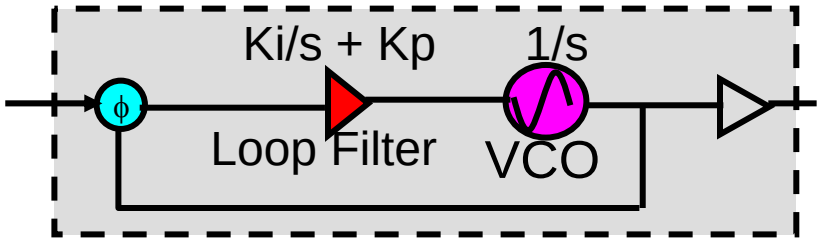


Local PLL Clock (asynchronous or shared ref) (B2B/ Long Reach systems)



Early/Late Second-Order CDR for Asynchronous I/O

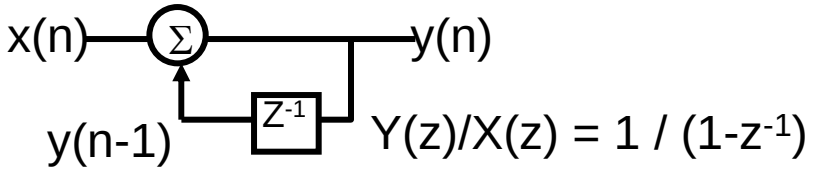
A second-order digital CDR is derived from a second-order analog PLL :

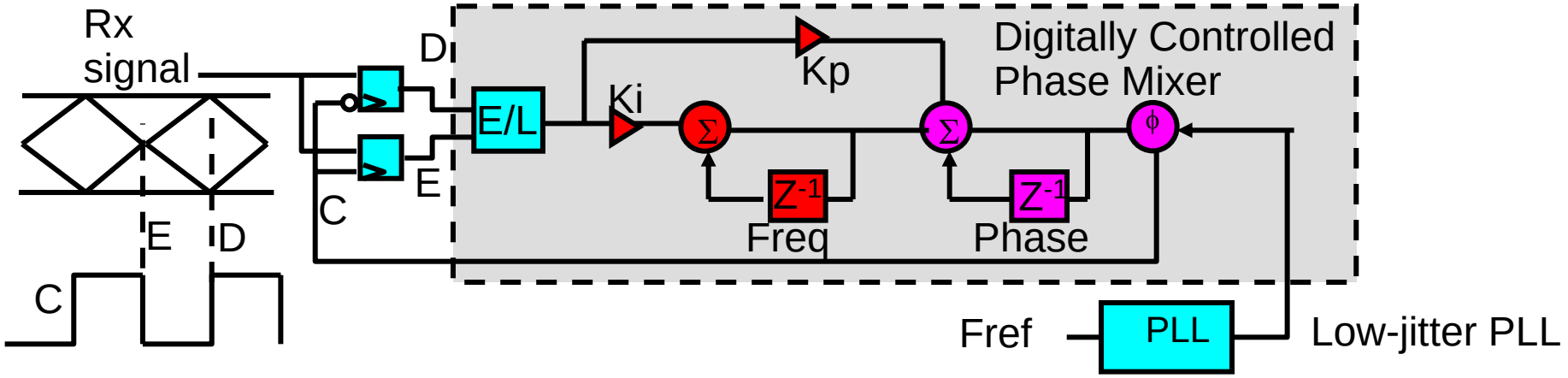


Translate block diagram from analog to digital domain :

1/s = Analog integrator

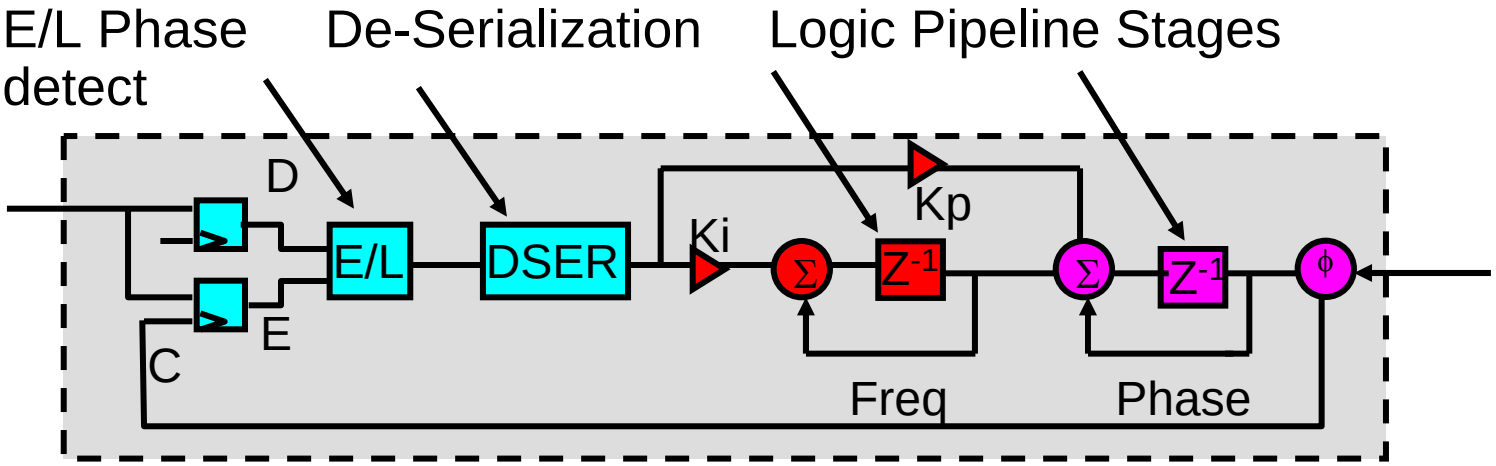
y(n) = y(n-1) + x(n) = digital integrator





Source of Latency in Digital CDR Loops

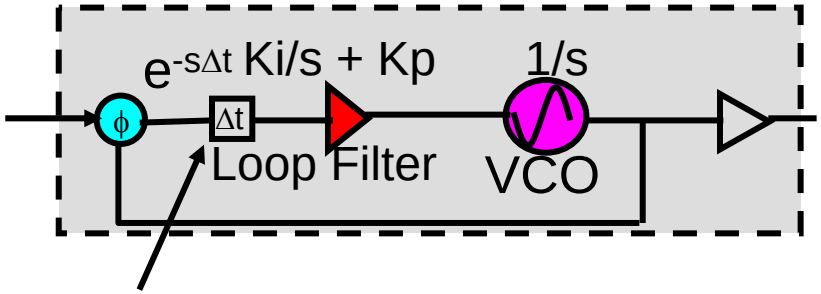
Practical digital CDR adds latency to CDR loop through E/L phase detection, deserialization and logic pipelining.



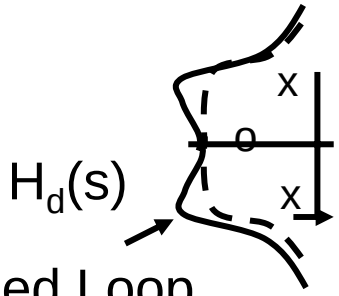
DSER must slow clock down enough that the digital CDR logic/adders can time.

Effect of Latency in PLL/CDR Loops

Excess loop latency de-stabilizes the PLL or CDR loop and results in jitter transfer peaking.



Time delay in analog elements

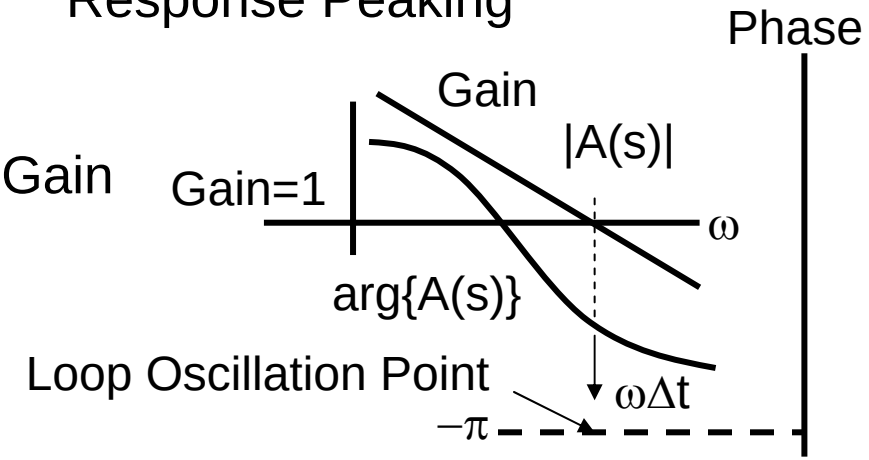


Δt increases Poles head to Jw axis

Closed Loop Response Peaking

$A(s) = e^{-s\Delta t}(Ki/s + Kp)/s$ = Open-Loop Gain

No gain loss + added phase delay = direct hit on phase margin

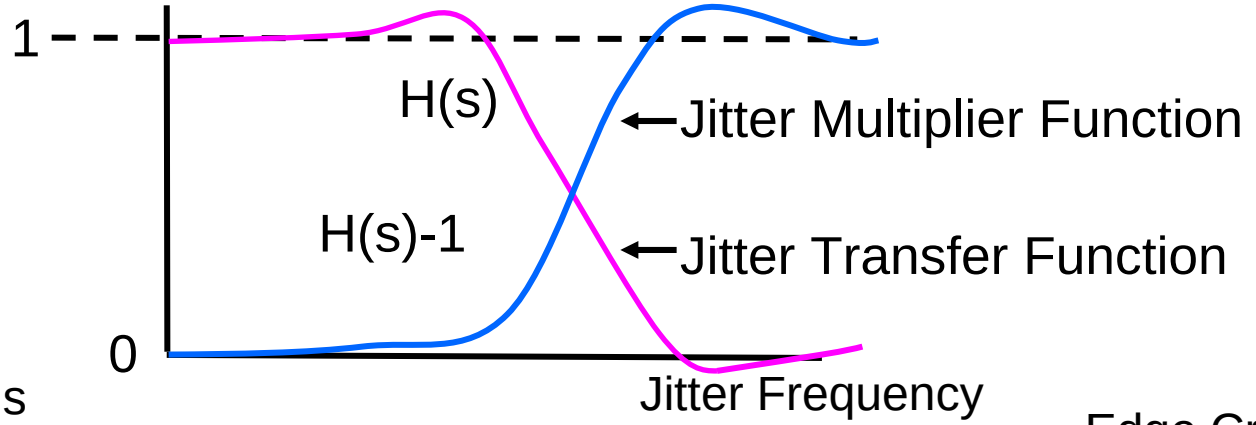


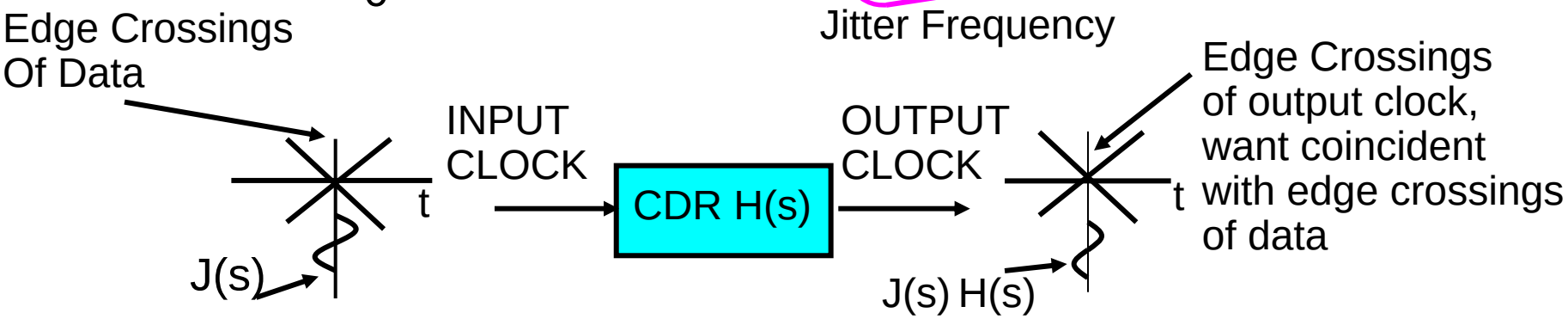
$\omega_o \Delta t$ = Loss in phase margin, ω_o = frequency where the open loop and closed loop gain meet

Jitter Transfer Function and Jitter Multiplier Function of a CDR Loop

The CDR can eliminate or attenuate the effect of jitter within its tracking bandwidth. The amount of jitter attenuation is quantified by the jitter multiplier function.

Jitter Transfer Function = CDR output clock jitter / Input clock jitter = H(s)
Jitter Multiplier Function $\triangleq$ (CDR output clock jitter – Input clock jitter) / Input clock jitter
= H(s)-1

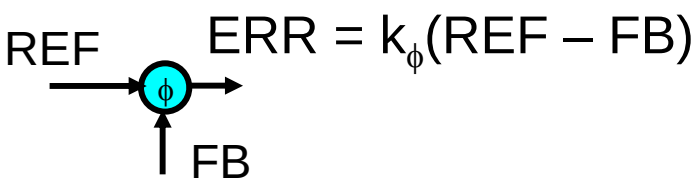




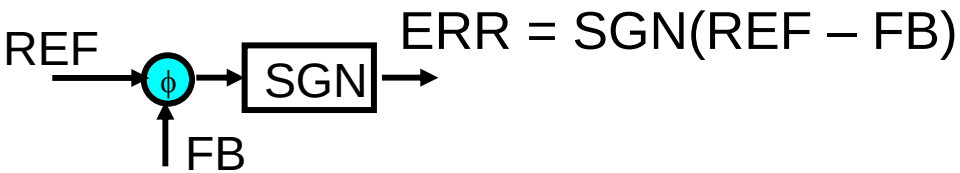
Jitter Magnitude Dependent Transfer Function of Early/Late CDR

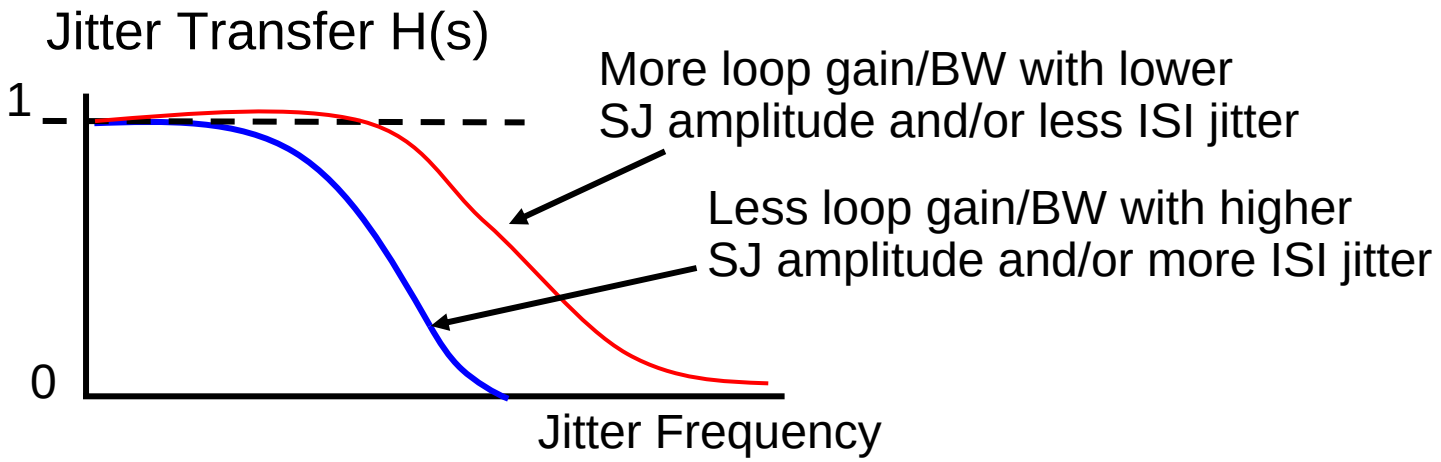
The Early/Late CDR jitter transfer function depends on the magnitude of the input jitter due to non-linear phase detector.

LINEAR ϕ DETECT :
LOOP GAIN ~ ERR



LIMITING ϕ DETECT :
LOOP GAIN ~ SGN(ERR)

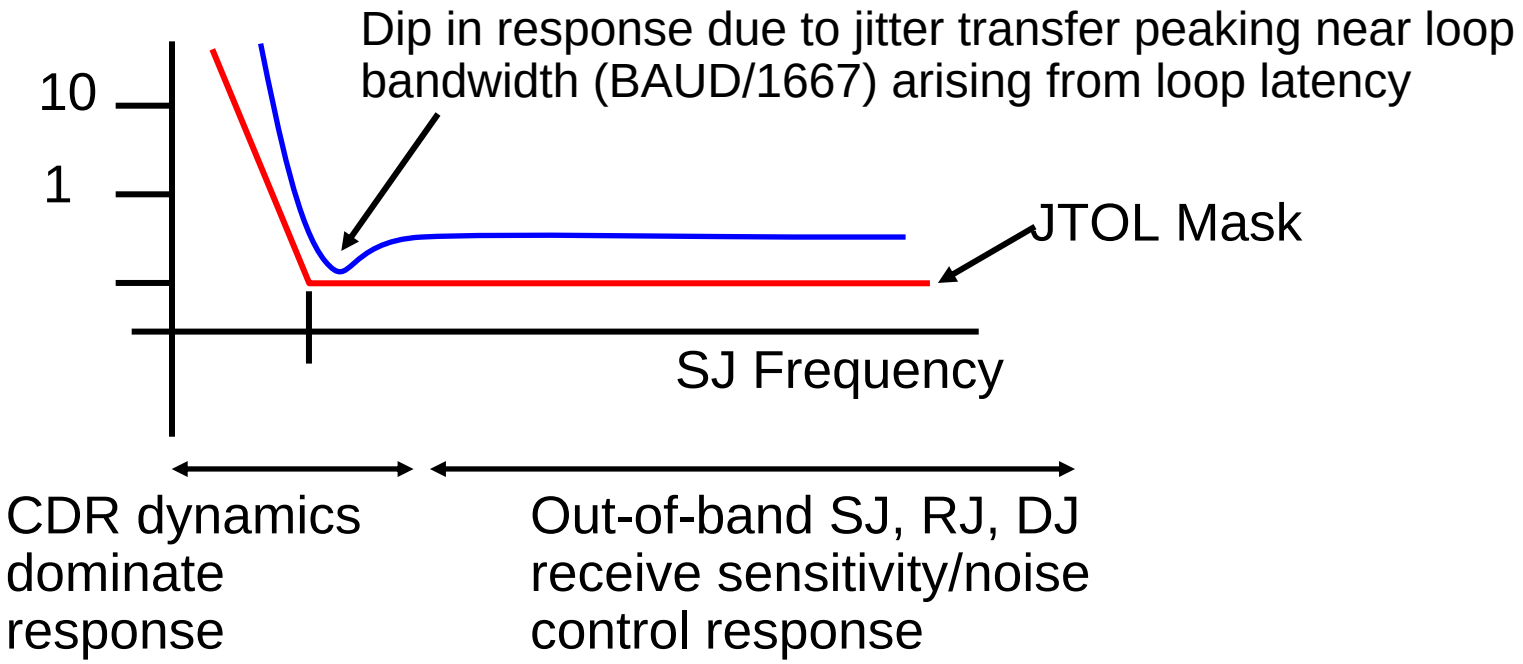




Jitter Tolerance Optimization for an Asynchronous CDR

CDR JTOL Optimization for Async CDR is accomplished by tuning the design latency and loop parameters (proportional and integral path gain) to meet the required JTOL mask.

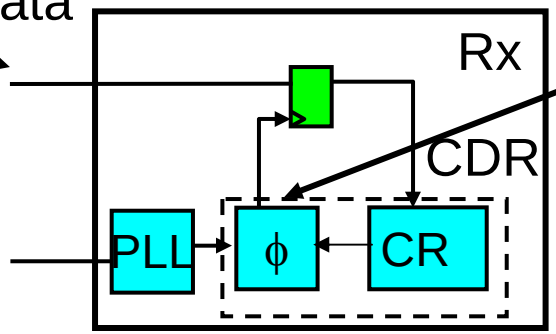
Beyond CDR tracking bandwidth, JTOL is dominated by system high-frequency SJ, DJ, RJ, and receiver sensitivity.

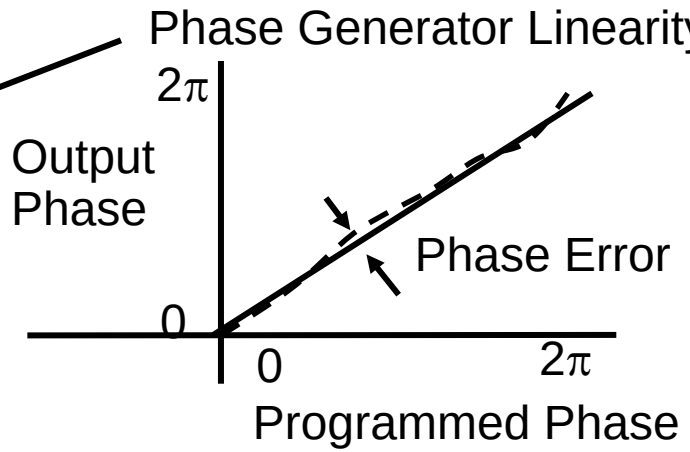


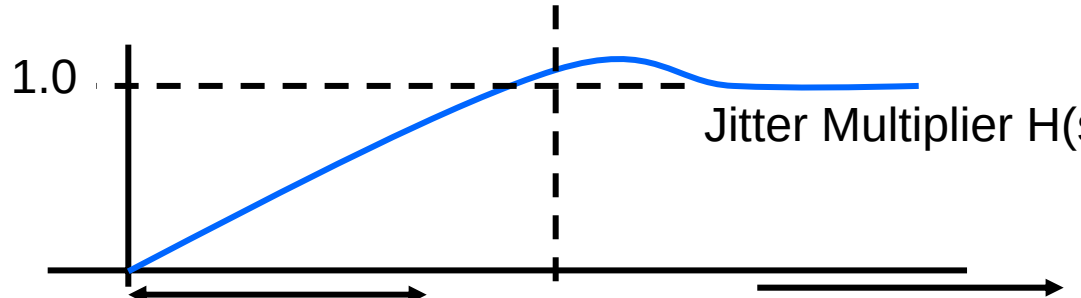
Non-Linearity in CDR Phase Generation Subsystem

Non-Linearity in a local CDR Clock Phase Generator can generate un-trackable excess jitter if the frequency offset between source and receiver is large.

Spread-Spectrum Clocking (SSC)
Input data







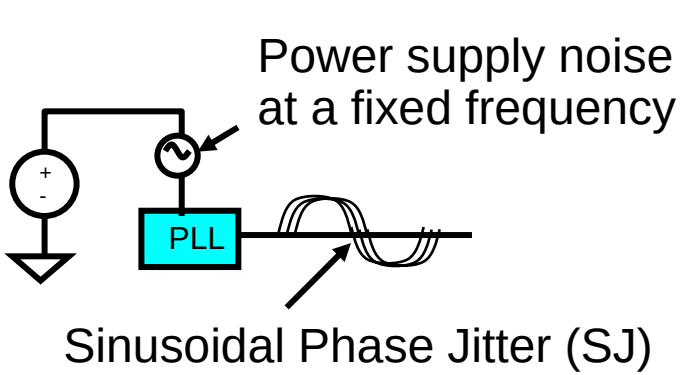
Low PPM Offset, ϕ non-linearity
Is tracked out by CDR

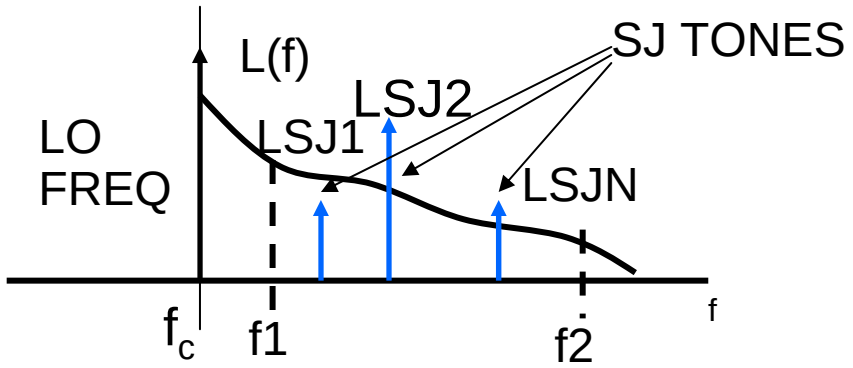
BAUD/1667=600ppm

High PPM Offset, ϕ non-linearity
comes straight through

Table of Contents

Time Jitter Computation From Phase Noise + SJ Tones





$$\text{Time Jitter (RMS)} = \frac{1}{2\pi f_c}\sqrt{2\int_{f_1}^{f_2} L(f)df + 2\sum LSJi}$$

Non-Deterministic Phase Noise Power arising from noise

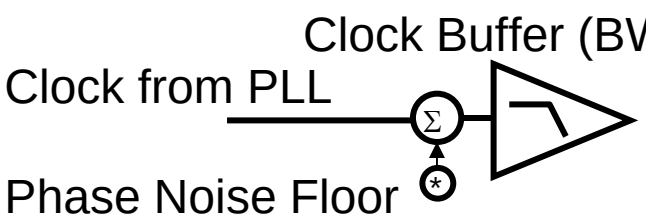
Bounded Peak-Peak Phase Noise Power arising from spurs
(power sum valid if all are de-correlated)

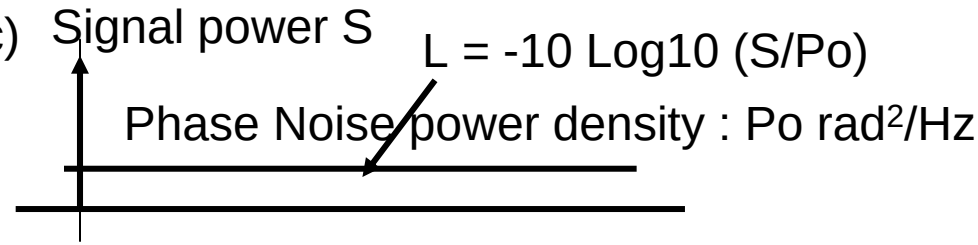
If one (or more) SJ terms greatly dominates L(f) over its integration range, SJ tones can be separated from clock RMS RJ and added separately to the system jitter analysis as bounded SJ.

See Appendix B for PLL phase noise model and jitter formula derivations

Time Jitter Contribution from Phase Noise Floor

Clock buffer noise floor can contribute significant integrated jitter since it is present over wide bandwidth. Lowering the clock signal level increases the dBc noise in direct proportion to loss in level.

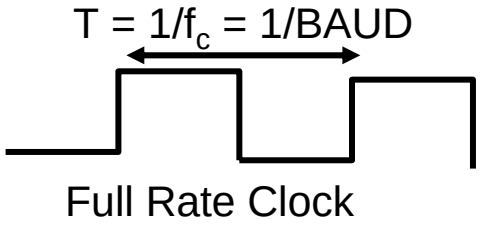




Time Jitter =~ $1 / (2\pi f_c) \sqrt{2(Po/S)}$

Jitter goes up with lower clock level

Jitter goes up with larger Po



Maximum TOTAL jitter wanted (not just noise floor) is ~0.7% UI

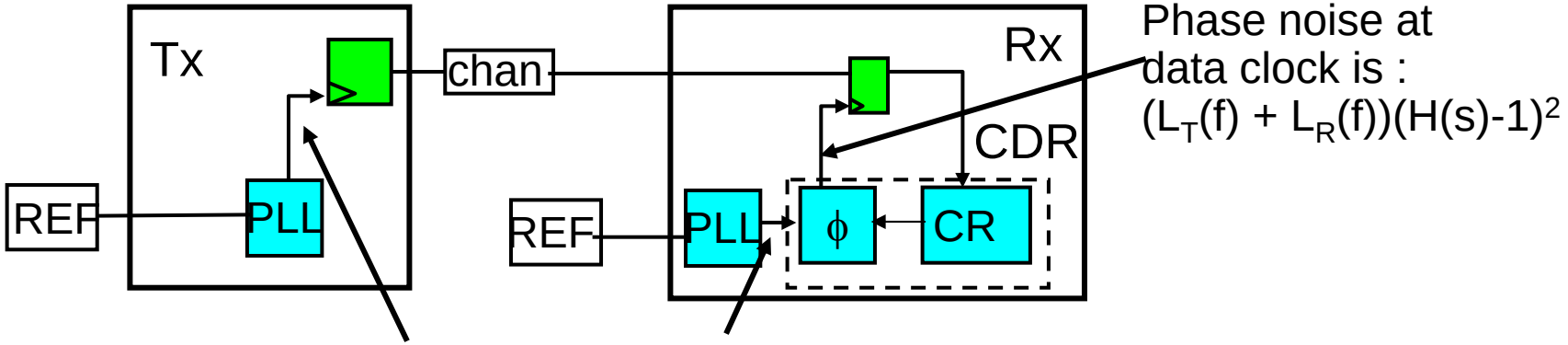
Jitter vs. Noise

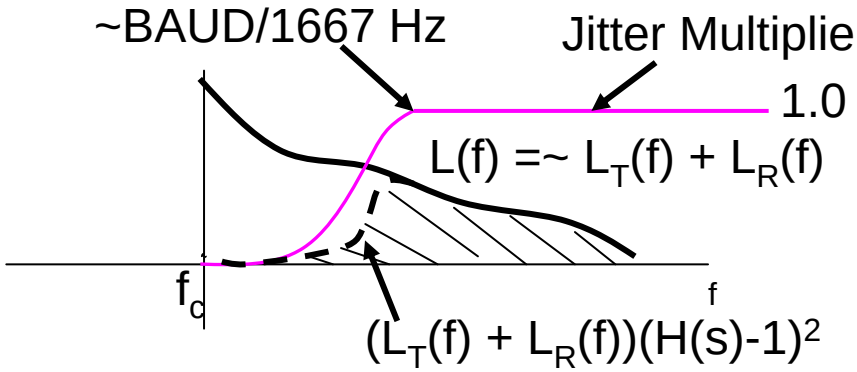
Phase noise (dBc/Hz)	fc	Jitter (ps RMS)	BAUD (Gb/s)	Jitter (% UI RMS)
-140/-134/-128	6GHz	0.3/0.6/1.2	6	0.17/0.35/0.7
-140/-134/-128	12GHz	0.2/0.4/0.8	12	0.25/0.5/1
-140/-134/-128	24GHz	0.15/0.3/0.6	24	0.35/0.7/1.4

See appendix B.5 for derivation of phase noise power density from broadband additive thermal noise

Random Jitter Analysis for an Asynchronous PLL + CDR System

Combined T-R Random Jitter Analysis Approximation : In dominant Noise part of phase noise power spectrum (<1GHz), channel is flat… -> power sum T and R phase noise @ Rx.



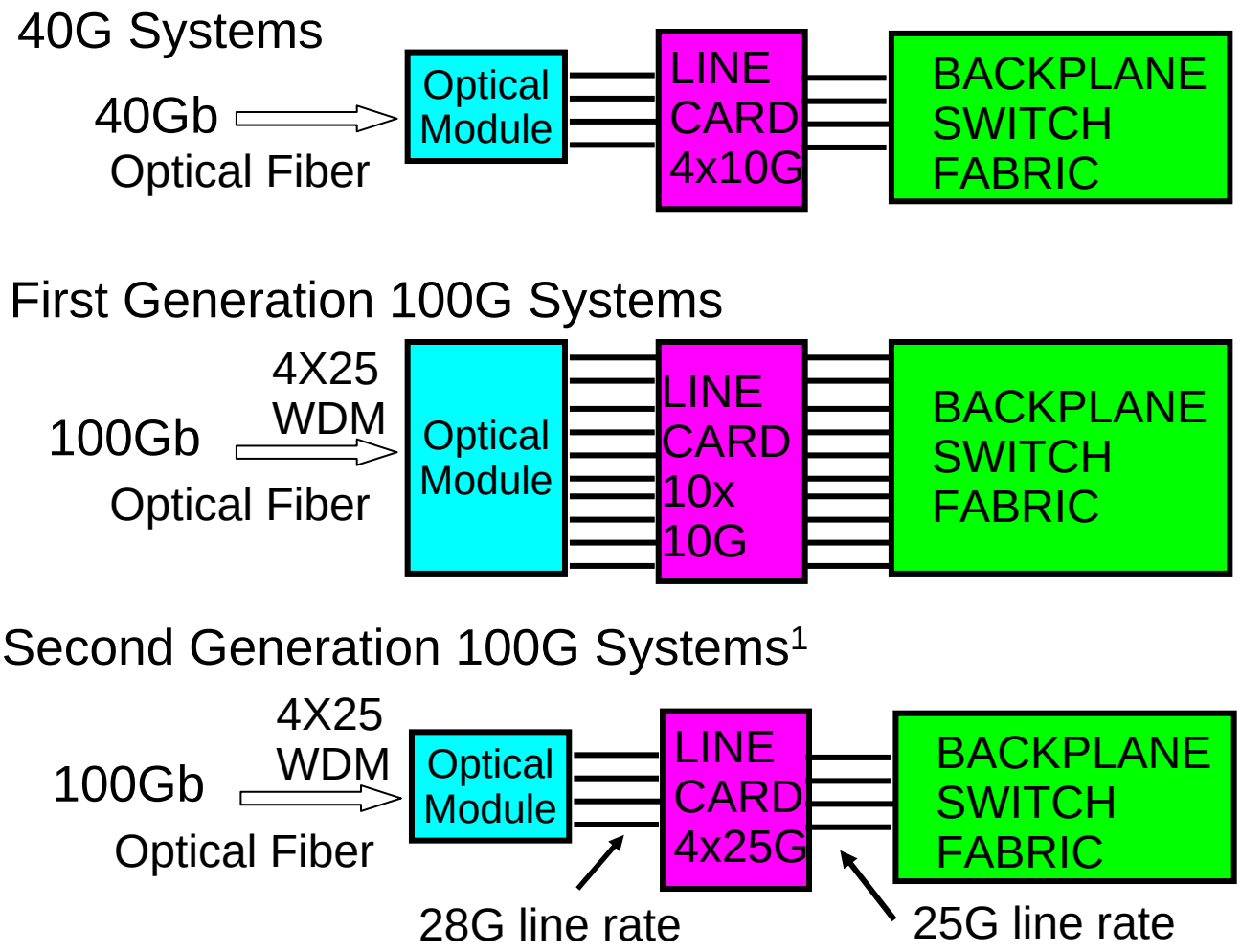


$$RJ = \frac{1}{2\pi f_c}\sqrt{2\int_0^{\infty}\left[H(s)-1\right]^2 L(f)\,df}$$

Table of Contents

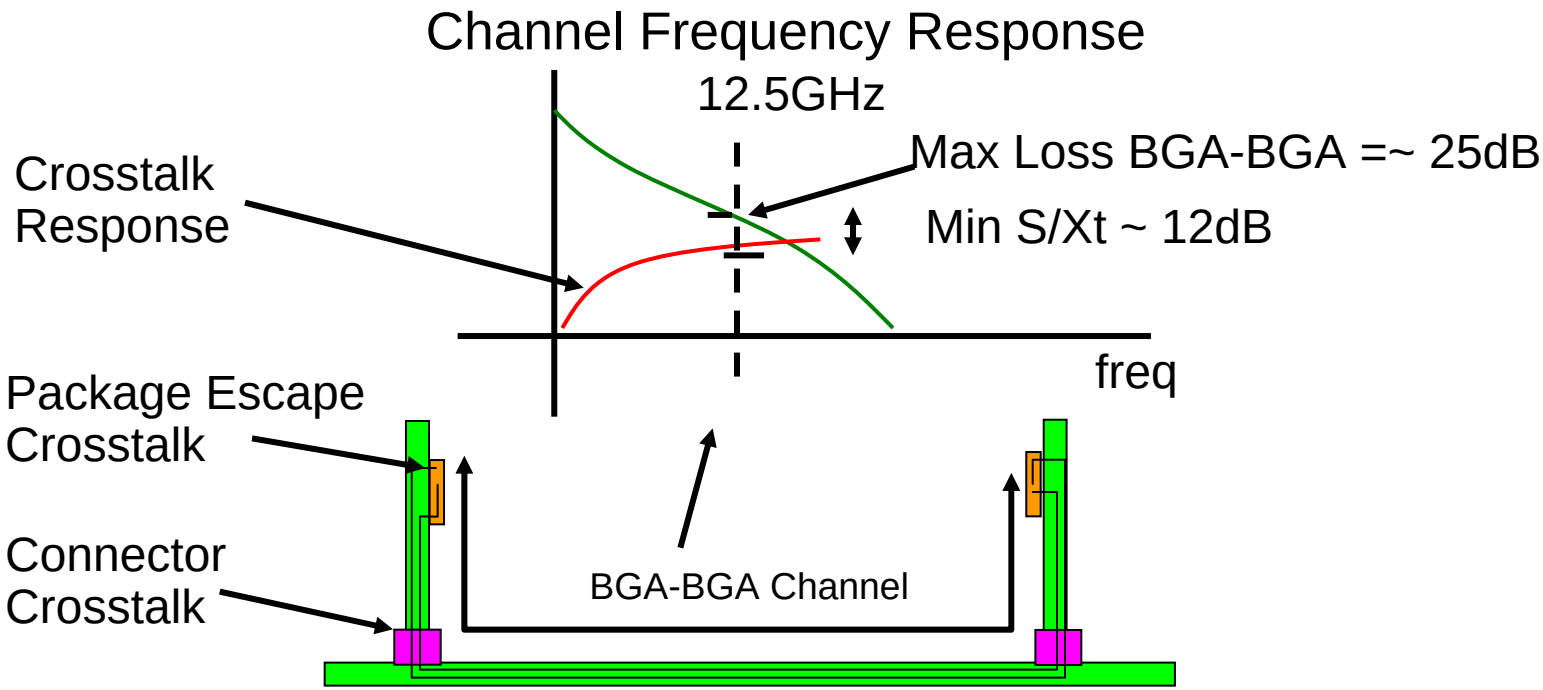
Evolution to 25Gb/s Applications

Industry standards/work groups driving rate evolutions : OIF-CEI, 802.3ba, ITU-T



[1]http://www.oiforum.com/public/documents/OIF_CEI-28G_WP_Final.pdf

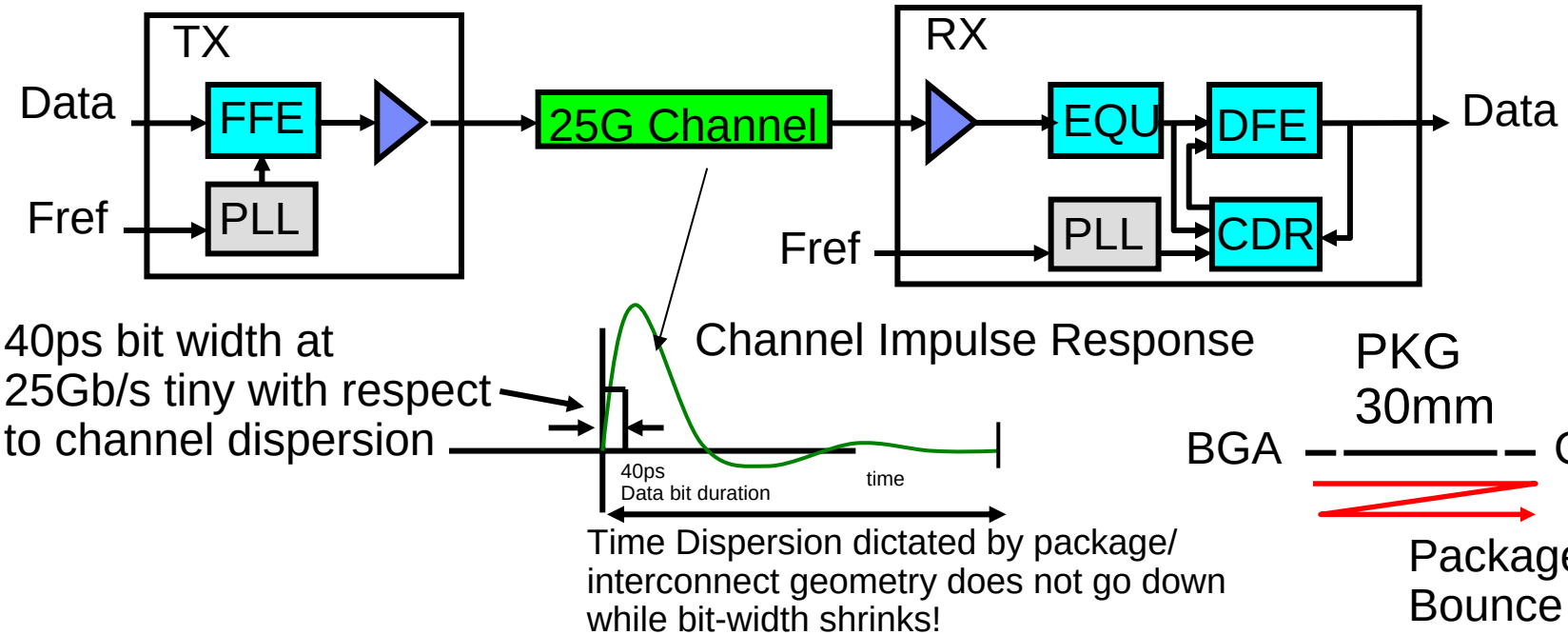
Channel Optimization Requirements for 20-25G Systems



Key Optimizations required for 25G physical channels :

1) Use low-loss dielectric material for Daughter Card and Backplane
2) Minimize crosstalk in connectors
3) Eliminate all stubs in channel… no extra resonance loss acceptable
4) Minimize number and length of via drops and associated impedance discontinuities
5) Minimize impedance discontinuity between package/board/connector/backplane
6) Maximize T-R isolation by physical design : minimize NEXT by design!
   -optimize keep T and R buses isolated/separated in connectors

SERDES Optimization Requirements for 20-25G Systems



Key Optimizations required for 25G SERDES :

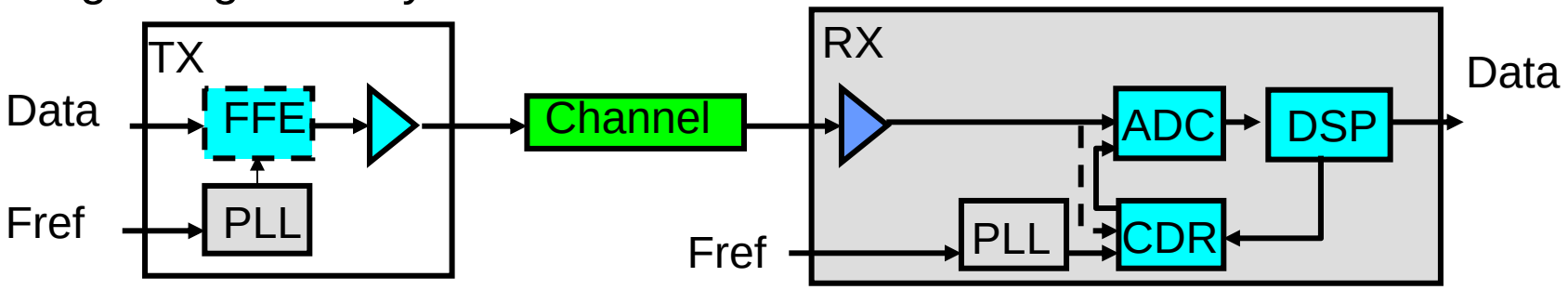
1) Increase equalization capability : Enhance line equalizers to handle channel dispersion
2) Lower jitter on launch and sample clocks to as close to 0 jitter as possible (<<40ps bit!)
   -Optimize CDR design to meet JTOL with BAUD/1667 (15MHz) jitter tracking bandwidth
3) Provide needed bandwidth for transmitter and receiver to minimize end-to-end loss
   -> Equalization system must handle combined loss of channel + package + I/O
4) Minimize noise in wide receiver bandwidth
5) Maximize T-R isolation on-die and in package escape region by physical design.
   -optimize ground/signal ratio in off-die pin escape[1]

[1] *D. Kam et. Al , "Is 25Gb/s on-board signaling viable?" IEEE Transactions on Advanced Packaging, vol 32, no. 2, May 2009*

Table of Contents

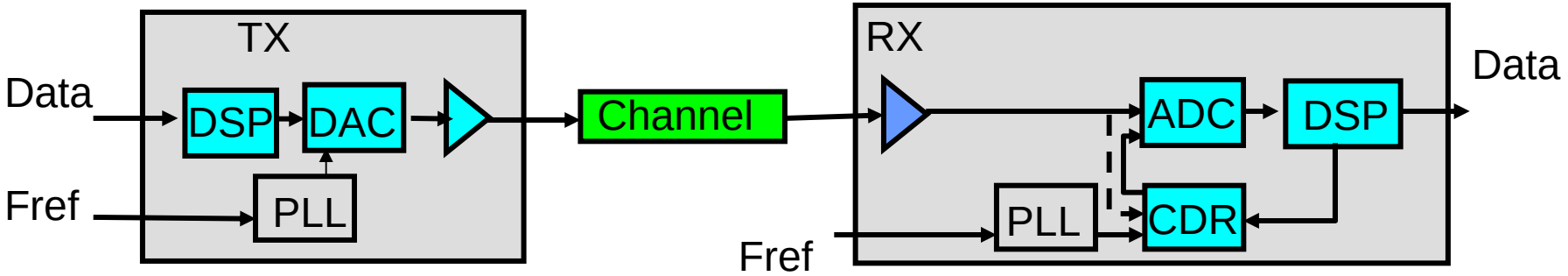
Backplane I/O ADC-Based System Designs

Analog T/Digital R Hybrid



Optimized for NRZ signaling line standards
Benefit limited to any extra performance the ADC+DSP decode can provide, and potential elimination of need for use of Tx FFE

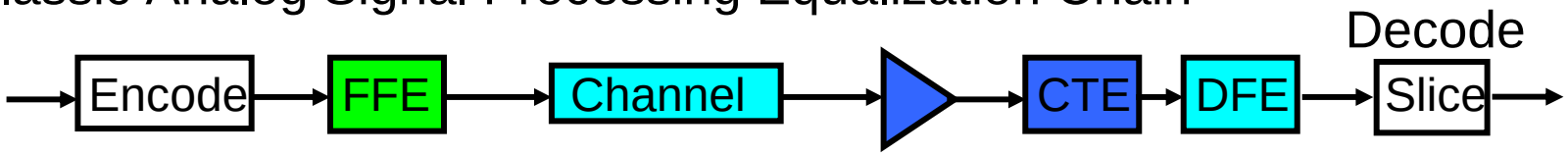
Digital T/Digital R : “Wildcard” for 25Gb/s and higher Backplane I/O



Unlimited line signaling variations possible
Supports complex Tx modulations such as TCM/OFDM, other dense symbol constellations
Use of complex modulation/coding requires industry standardization for widespread use!

Analog vs. ADC/Digital I/O Architecture Tradeoffs

Classic Analog Signal Processing Equalization Chain



NRZ Partial Response Line Signaling

+Low complexity data encode/decode

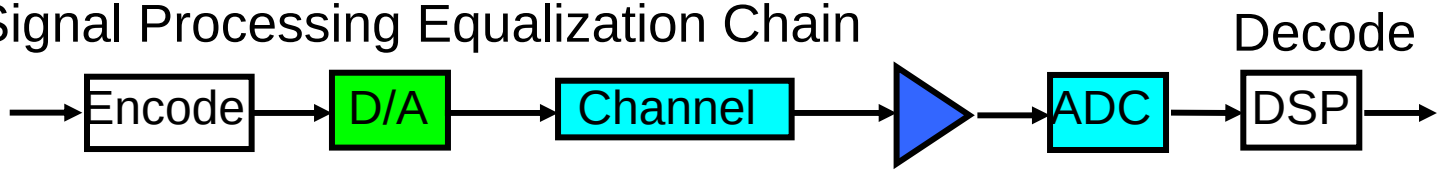
+Very good power efficiency

+No inherent quantization in binary slicer

-Non-optimal data detection with simple data slicer

-Most efficient with NRZ data transmission, bounds spectral efficiency

Digital Signal Processing Equalization Chain



Trellis-Coded Modulation, OFDM, Convolutionally Encoded data

+Closer to optimum data detection possible (using maximum likelihood sequence estimation, etc.)

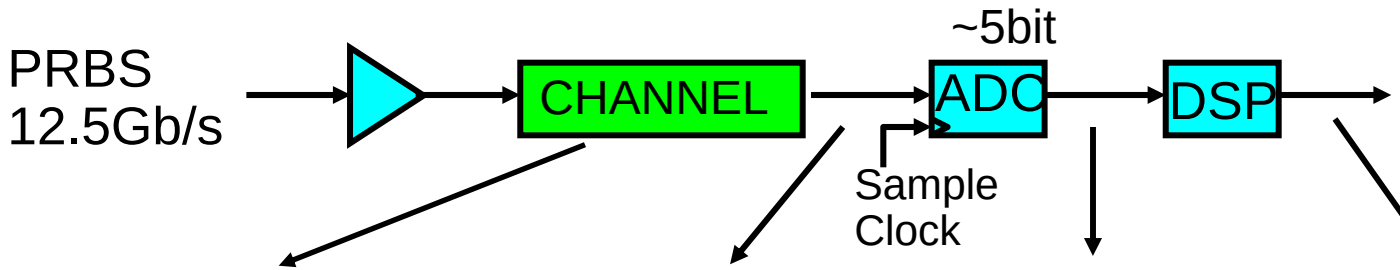
+Spectral efficiency improvement potential

-Complexity and decode latency go (way) up with modulation/coding complexity.

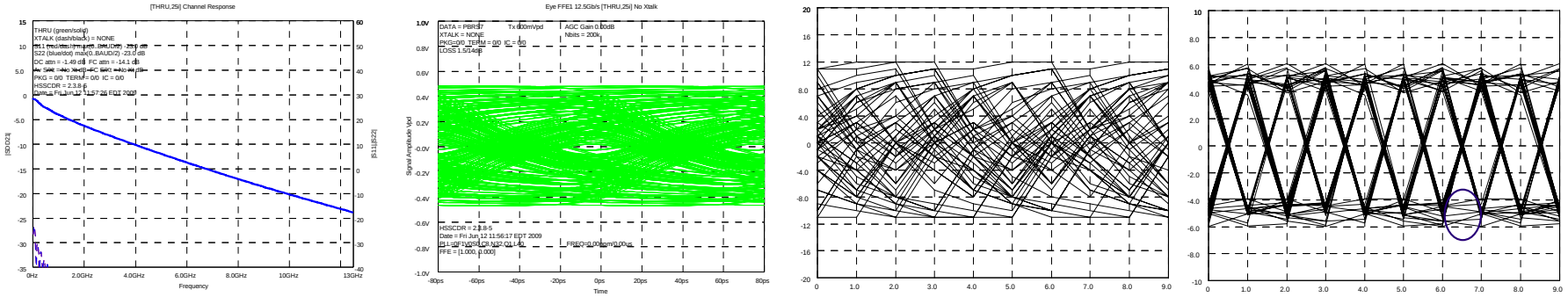
-Higher power and area than an optimized analog core (uncompetitive in NRZ I/O applications at 20% more power)

-Quantization noise/sample error of ADC

Example ADC-based NRZ Line Equalizaton



Frequency Response | Received Signal | Quantized Signal | Equalized Signal

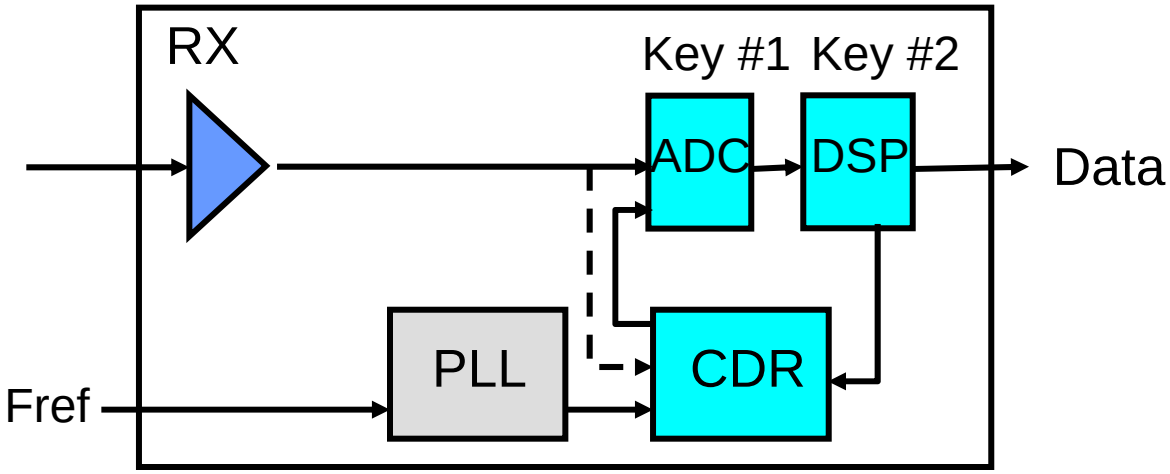


Nominal ADC Characteristics for Backplane I/O :
~4.5 bit Enob
Effective sample rate = BAUD rate (1 sample per UI)
2X to 4X interleaved converter

Nominal DSP for moderately dispersive backplane I/O :

FFE3 + DFE5

Advances Needed to unlock potential of ADC-based High-Density SERDES



Key #1 ADC :

converter power must be lowered through a combination of efficient architecture and advanced technology (45nm, 32nm…)

sample error due to non-idealities and fundamental quantization noise must be minimized to avoid degrading NRZ detection sensitivity

Key #2 DSP:

Post-Processing power must be lowered through a combination of efficient processing algorithms and technology improvements, again leaning heavily on advanced CMOS technology for low digital power

Summary and Conclusions

Electrical interconnect data rates are heading to 20Gb/s and higher as advances in IC technology, storage technology and internet growth continue to drive demand for higher information bandwidth.

S-parameters are an extremely useful behavioral modeling tool which can capture both IC and passive line model characteristics. Many pitfalls await in generating and applying them accurately.

End-to-end system analysis techniques which incorporate accurate behavioral models of the channel and I/O core are needed to design and optimize electrical links.

Key components of the Electrical I/O are :

PLL : to generate low jitter clocks
CDR : to provide optimum jitter-suppressed data sampling
Equalization : to compensate ISI from line dispersion

Comprehensive System Jitter Analysis is necessary to optimize performance of high data rate serial interconnect systems.

Co-optimization of both electrical I/O core and channel is needed at higher and higher data rates… it is not possible to move to extremely high data rates with only one of these two pieces!

Digital / ADC I/O architecture can become interesting for backplane I/O as ADC power and digital logic power go down with advanced architectures and CMOS technology, opening up the possibility of more spectrally efficient line coding and optimal decoding algorithms to increase link operating margins.

Appendix/Backup

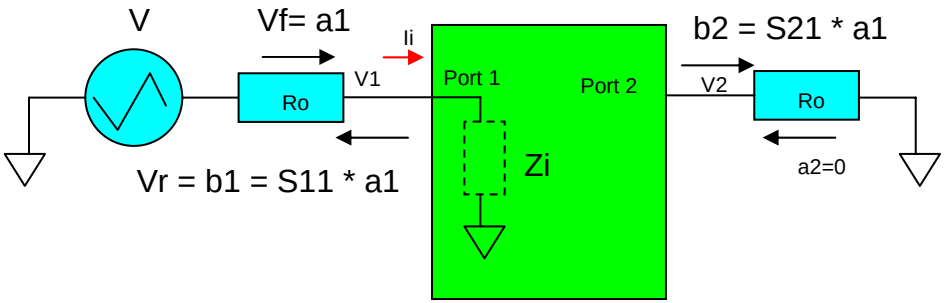
A. S-Parameter Topics :
   1. Derivation of S-Parameters from Port Node Voltages
   2. Mixed-Mode to Single-Ended S-Parameter conversion
   3. 2-port S-parameter to T parameter converson
   4. Determination of S21 and S11 from a T-Line Zo and γ
   5. Other useful line element S-Parameters
      - Open-circuit stub
      - Series Z (Series R, series L)
      - Shunt Z (Shunt G, shunt C)

B. PLL/Jitter Topics
   1. 2nd Order PLL Response
   2. Simplified PLL Phase Noise Model
   3. Time Jitter Derivation for an SJ Tone
   4. Time Jitter Computation From SSB Phase Noise
   5. Time Jitter Derivation for additive noise

C. Digital Modulation Examples
   1. Digital Line-Coding/Modulation : TCM
   2. Subchannel Modulation : OFDM

A.1 Derivation of S-parameters from Port Node Voltages

S-Parameters can be derived in circuit simulators from node voltages as follows :



V1 = Vf + Vr, Vr/Vf = S11 -> V1 = Vf(1 + S11) -> V1/Vf = 1 + S11 -> **S11 = V1/Vf – 1**

Zi = V1 / Ii = Ro * V1 / (V-V1) -> Ro/Zi = (V-V1)/V1 = V/V1 - 1

S11 = Γ = (Zi – Ro) / (Zi + Ro) = (1-Ro/Zi) / (1+Ro/Zi) = (1-(V/V1-1)) / (1+(V/V1-1)) = (2-V/V1) / (V/V1) = 2V1/V -1 = V1/(V/2)-1

-> **Vf = V/2 = a1** = incident voltage a1 for a source excitation of V, by definition of s-parameters

-> **S11 = V1 / (V/2) – 1 -> V1 = (1+S11) * (V/2) -> If V is set to 2, S11 = port 1 node voltage - 1**

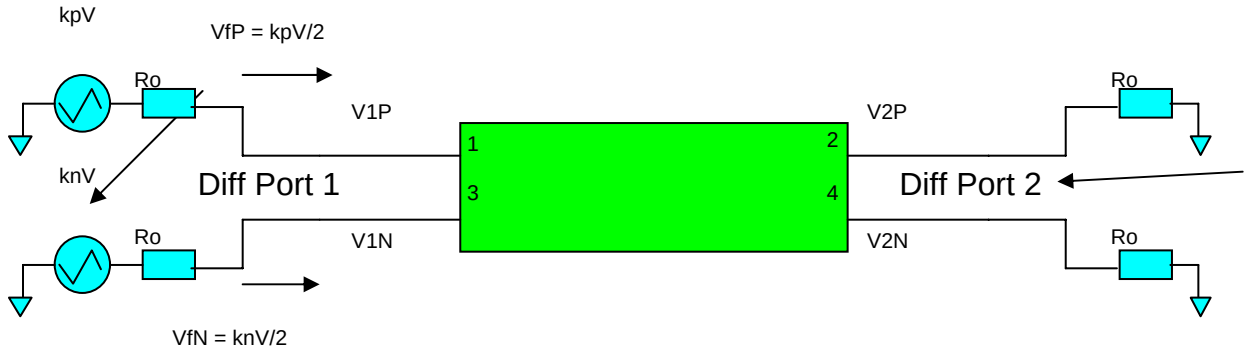
S21 = b2 / a1 = V2 / (V/2) -> V2 = S21 * (V/2) -> If V is set to 2, S21 = port 2 node voltage

Note also that :

S21 = V2 / Vf V1 = Vf + Vr = Vf(1+S11) -> Vf = V1 / (1+S11) -> **S21 = (1+S11) * V2 / V1**

-> S21 is a voltage transfer from output to input only if input is perfectly matched
-> **S21 = (1+S11)** if Node 1 and Node 2 are the same, so V2 = V1 (load impedance only)

A.2 Mixed-Mode to Single-Ended S-Parameter conversion



Interested in Differential Voltage at end of link, since this is what the data latch detects.

For Differential excitation:
kp = 1, kn = -1, VfD = V/2-(-V/2) = V

For Common excitation :
kp = 1, kn = 1, VfC = V/2 + V/2 = V

By definition these are normalized to the same incident voltage by defining the input common-mode voltage as ViP + ViN

Differential Input Reflection Coefficient :

SDD11 = Differential Input reflection / Differential excitation = (b1 – b3) / (a1-a3)
= ((S11 * a1 + S13 * a3) – (S33 * a3 + S31 * a1)) / (a1-a3)
a1 = V/2 a3 = -V/2
SDD11 = 0.5 * (S11 – S13 + S33 – S31)

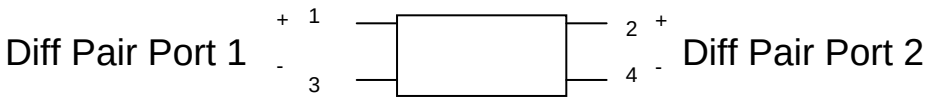
Differential Transmission Parameters :

SDD21 = Differential Transmission / Differential excitation = (b2 – b4) / (a1 – a3)
= ((S21 a1 + S23 a3) – (S43 a3 + S41 a1)) / (a1-a3)
a1 = V/2 a3 = -V/2
SDD21 = 0.5 * (S21 - S23 + S43 - S41)

SDC21 = Differential Transmission / Common excitation = (b2 – b4) / (a1 + a3)
= ((S21 a1 + S23 a3) – (S43 a3 + S41 a1)) / (a1 + a3)
a1 = V/2 a3 = V/2
SDC21 = 0.5 * (S21 + S23 – S43 – S41)

Remainder of Mode Conversion and conversion from Diff to SE formulas presented on next page.

A.2 Mixed-Mode to Single-Ended S-Parameter conversion (cont.)



Differential Voltage $\hat{=} V^+ - V^-$
Common Voltage $\hat{=} V^+ + V^-$

SDD11 = 0.5 * (S11 - S13 + S33 - S31)
SDD12 = 0.5 * (S12 - S32 + S34 - S14)
SDD21 = 0.5 * (S21 - S23 + S43 - S41)
SDD22 = 0.5 * (S22 - S24 + S44 - S42)

SDC11 = 0.5 * (S11 + S13 - S33 - S31)
SDC12 = 0.5 * (S12 + S32 - S34 - S14)
SDC21 = 0.5 * (S21 + S23 - S43 - S41)
SDC22 = 0.5 * (S22 + S24 - S44 - S42)

SCD11 = 0.5 * (S11 - S13 - S33 + S31)
SCD12 = 0.5 * (S12 - S32 - S34 + S14)
SCD21 = 0.5 * (S21 - S23 - S43 + S41)
SCD22 = 0.5 * (S22 - S24 - S44 + S42)

SCC11 = 0.5 * (S11 + S13 + S33 + S31)
SCC12 = 0.5 * (S12 + S32 + S34 + S14)
SCC21 = 0.5 * (S21 + S23 + S43 + S41)
SCC22 = 0.5 * (S22 + S24 + s44 + S42)

S11 = 0.5 * (SDD11 + SDC11 + SCD11 + SCC11)
S12 = 0.5 * (SDD12 + SDC12 + SCD12 + SCC12)
S21 = 0.5 * (SDD21 + SDC21 + SCD21 + SCC21)
S22 = 0.5 * (SDD22 + SDC22 + SCD22 + SCC22)

S13 = 0.5 * (-SDD11 + SDC11 - SCD11 + SCC11)
S32 = 0.5 * (-SDD12 + SDC12 - SCD12 + SCC12)
S23 = 0.5 * (-SDD21 + SDC21 - SCD21 + SCC21)
S24 = 0.5 * (-SDD22 + SDC22 - SCD22 + SCC22)

S33 = 0.5 * (SDD11 - SDC11 - SCD11 + SCC11)
S34 = 0.5 * (SDD12 - SDC12 - SCD12 + SCC12)
S43 = 0.5 * (SDD21 - SDC21 - SCD21 + SCC12)
S44 = 0.5 * (SDD22 - SDC22 - SCD22 + SCC22)

S31 = 0.5 * (-SDD11 - SDC11 + SCD11 + SCC11)
S14 = 0.5 * (-SDD12 - SDC12 + SCD12 + SCC12)
S41 = 0.5 * (-SDD21 - SDC21 + SCD21 + SCC21)
S42 = 0.5 * (-SDD22 - SDC22 + SCD22 + SCC22)

A.3 Translating Between 2-Port T-Parameters and S-Parameters

$$\begin{matrix} \text{INPUT} \\ \text{OUTPUT} \end{matrix} \begin{bmatrix} B1 \\ B2 \end{bmatrix} = \begin{bmatrix} S11 & S12 \\ S21 & S22 \end{bmatrix} \begin{bmatrix} A1 \\ A2 \end{bmatrix} \begin{matrix} \text{INPUT} \\ \text{OUTPUT} \end{matrix}$$

$$\begin{matrix} \text{INPUT} \\ \text{INPUT} \end{matrix} \begin{bmatrix} B1 \\ A1 \end{bmatrix} = \begin{bmatrix} T11 & T12 \\ T21 & T22 \end{bmatrix} \begin{bmatrix} A2 \\ B2 \end{bmatrix} \begin{matrix} \text{OUTPUT} \\ \text{OUTPUT} \end{matrix}$$

B1 = S11 A1 + S12 A2
B2 = S21 A1 + S22 A2 -> A1 = (B2 – S22 A2) / S21

B1 = S11 (B2-S22A2)/S21 + S12A2
A1 = (B2-S22A2)/S21

B1 = (S12 – S11S22/S21) A2 + S11/S21 B2
A1 = -S22/S21 A2 + 1.0/S21 A1

B1 = T11 A2 + T12 B2
A1 = T21 A2 + T22 B2 -> B2 = (A1 – T21 A2) / T22

B1 = T11 A2 + T12 / T22 (A1 – T21 A2)
B2 = (A1 – T21 A2) / T22

B1 = T12/T22 A1 + (T11 – T12T21/T22) A2
B2 = 1.0/T22 A1 -T21/T22 A2

$$\begin{bmatrix} B1 \\ A1 \end{bmatrix} = \begin{bmatrix} S12\text{-}S11S22/S21 & S11/S21 \\ -S22/S21 & 1.0/S21 \end{bmatrix} \begin{bmatrix} A2 \\ B2 \end{bmatrix}$$

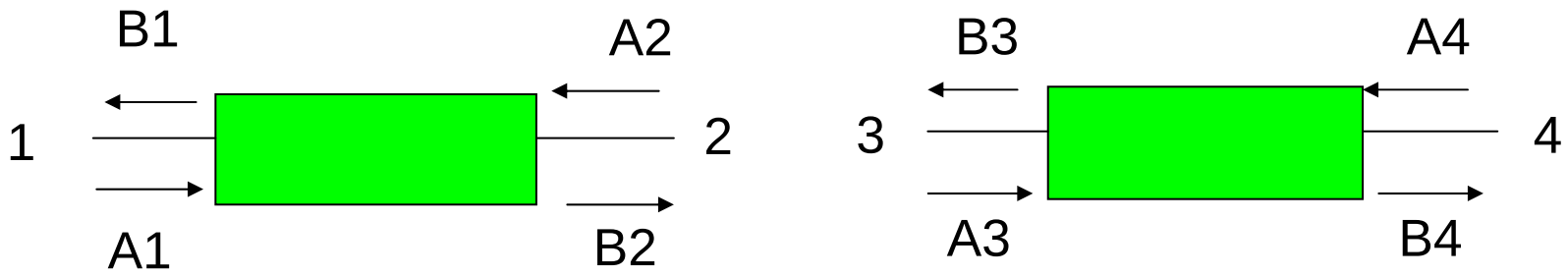
$$\begin{bmatrix} B1 \\ B2 \end{bmatrix} = \begin{bmatrix} T12/T22 & T11\text{-}T12T21/T22 \\ 1.0/T22 & -T21/T22 \end{bmatrix} \begin{bmatrix} A1 \\ A2 \end{bmatrix}$$

T11 = S12-(S11*S22/S21)
T12 = S11/S21
T21 = -S22/S21
T22 = 1.0/S21

S11 = T12 / T22
S12 = T11-(T12*T21/T22)
S21 = 1.0/T22
S22 = -T21/T22

This translation can be generalized to N-port T and S parameters, for any N even (i.e. # port outputs = # port inputs)

Cascading S-Parameter Elements using T-Parameters



$$\begin{bmatrix} B1 \\ A1 \end{bmatrix} = \begin{bmatrix} T11 & T12 \\ T21 & T22 \end{bmatrix} \begin{bmatrix} A2 \\ B2 \end{bmatrix} \qquad \begin{bmatrix} B3 \\ A3 \end{bmatrix} = \begin{bmatrix} T33 & T34 \\ T43 & T44 \end{bmatrix} \begin{bmatrix} A4 \\ B4 \end{bmatrix}$$

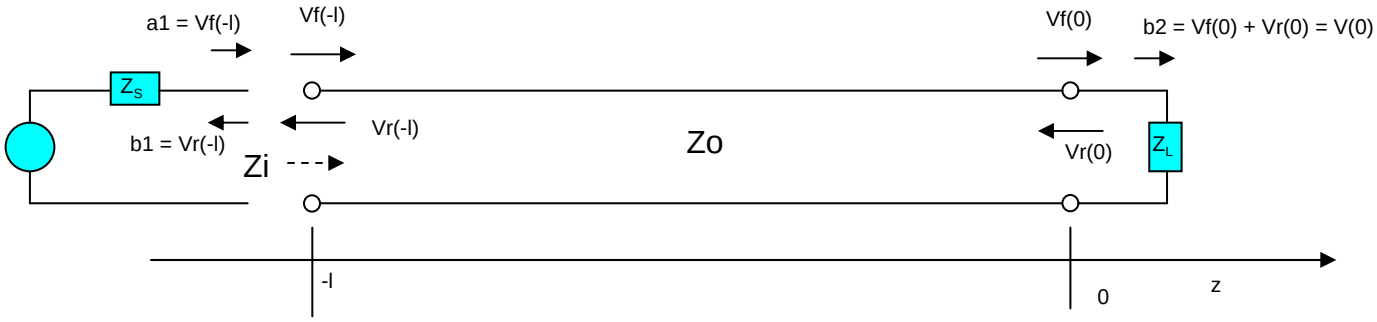
$$\begin{bmatrix} B1 \\ A1 \end{bmatrix} = \begin{bmatrix} T11 & T12 \\ T21 & T22 \end{bmatrix} \begin{bmatrix} T33 & T34 \\ T43 & T44 \end{bmatrix} \begin{bmatrix} A4 \\ B4 \end{bmatrix}$$

$$\begin{bmatrix} B1 \\ A1 \end{bmatrix} = \begin{bmatrix} TC11 & TC12 \\ TC21 & TC22 \end{bmatrix} \begin{bmatrix} A4 \\ B4 \end{bmatrix}$$ T-Parameters of Cascaded Link

$$\begin{bmatrix} B1 \\ B4 \end{bmatrix} = \begin{bmatrix} SC11 & SC12 \\ SC21 & SC22 \end{bmatrix} \begin{bmatrix} A1 \\ A4 \end{bmatrix}$$ S-Parameters of Cascaded Link

This procedure is generalized to N-port dimension for wide bus models.

A.4 Determination of S21 and S11 from a T-Line Zo and γ



$V(z) = Vf(0)\, e^{-\gamma z} + Vr(0)\, e^{\gamma z}$

$\Gamma_L = (Z_L - Zo) / (Z_L + Zo) = Vr(0) / Vf(0)$ (note $Vr(0) = Vf(0)$ for Z_L = infinity)

$V(z) = Vf(0)\,(e^{-\gamma z} + \Gamma_L\, e^{\gamma z})$

$Zi = Zo * (Z_L + Zo \tanh(\gamma l)) / (Zo + Z_L \tanh(\gamma l))$

$\Gamma_S = (Zi - Z_S) / (Zi + Z_S) = Vr(-l) / Vf(-l)$ (note $Vr(-l) = Vf(-l)$ for Zi = infinity)

$S11 = b1/a1 = Vr(-l) / Vf(-l) = \Gamma_s$

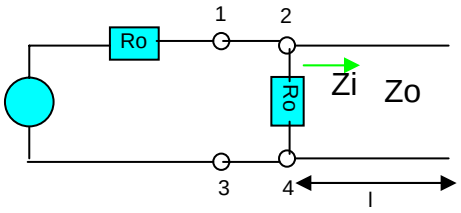
$S11 = \Gamma_S \,|_{ZS=ZL=Ro}$ where Ro = S-parameter reference impedance (normally 50 ohm)

$S21 = b2 / a1 = V(0) / Vf(-l)$
$V(0) = Vf(0)(1 + \Gamma_L)$
$V(-l) = Vf(0)\,(e^{-\gamma l} + \Gamma_L\, e^{\gamma l}) = Vf(-l)+Vr(-l) = Vf(-l)\,(1+\Gamma_S)$, -> $Vf(-l) = Vf(0)\,(e^{-\gamma l} + \Gamma_L\, e^{\gamma l}) / (1+\Gamma_S)$

$S21 = (1+\Gamma_L)\,(1+\Gamma_S) / (e^{\gamma l} + \Gamma_L\, e^{-\gamma l}) \,|_{ZS=ZL=Ro}$

A.5 Other useful line element S-Parameters

Open Stub



S11 = (Zi||Ro)-Ro / ((Zi||Ro)+Ro)

= (ZiRo – Ro(ZiRo))/

(ZiRo + Ro(Zi+Ro))

= -Ro / (Ro + 2Zi)

= -1 / (1 + 2Zi/Ro)

Zi = Zo * (Z_L + Zo tanh(γl))/

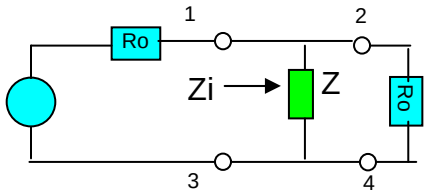
(Zo + Z_L tanh(γl)) | Zl = inf

= Zo / tanh(γl))

-> **S11 = -1 / (1 + 2Zo/(Ro*tanh(γl)))**

S21 = (1+S11)

Shunt Z



S11 = (Z||Ro)-Ro / ((Z||Ro)+Ro)

= (ZRo - Ro(Z+Ro))/

(ZRo + Ro(Z+Ro))

= -Ro / (Ro + 2Z)

= -1 / (1 + 2Z/Ro)

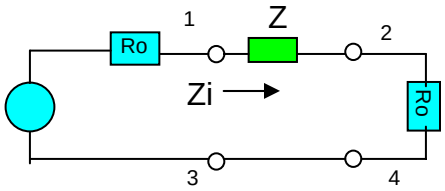
If Z = 1/sC (shunt cap)

S11 = -1 / (1 + 2/(sCRo))
S21 = (1+S11)

If Z = 1/G (shunt conductance)

S11 = -1 / (1 + 2/(G*Ro))
S21 = (1+S11)

Series Z



S11 = ((Z+Ro)-Ro) / ((Z+Ro)+Ro)

= Z / (Z+2Ro)

= 1 / (1 + 2Ro/Z)

S21 = 2Ro / (2Ro + Z)

= 1 / (1 + Z/(2Ro))

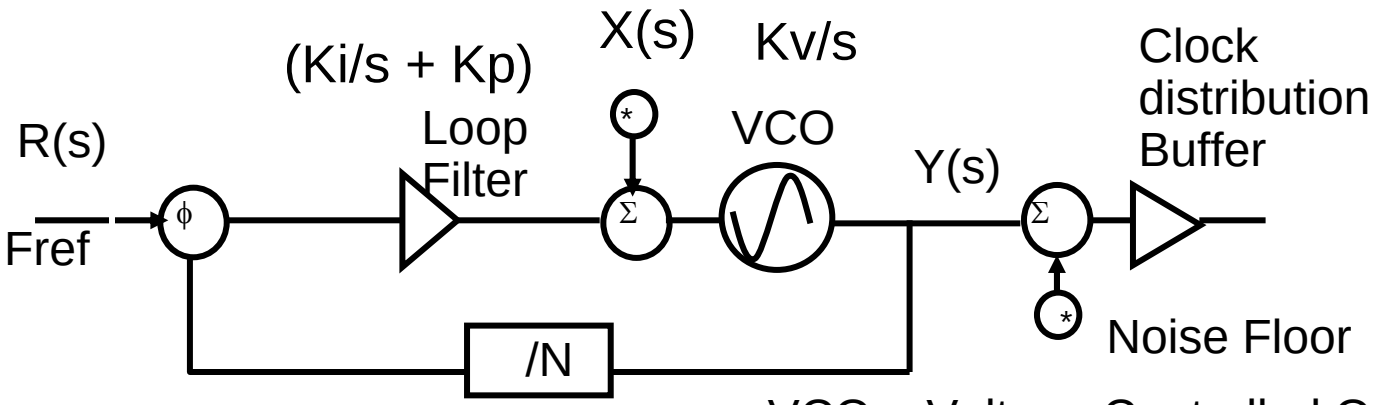
If Z = sL (series L)

S11 = 1 / (1 + 2Ro/(sL))
S21 = 1 / (1 + sL / (2Ro))

If Z = R (series R)

S11 = 1 / (1 + 2Ro/R))
S21 = 1 / (1 + R/(2Ro))

B.1 2nd Order PLL Response



Response from input — Response from VCO noise

$$Y(s) = R(s)\,A(s) / (1 + A(s)\beta(s)) + X(s)\,(Kv/s) / (1 + A(s)\beta(s))$$

$A(s) = (Ki/s + Kp)(Kv/s)$

$\beta(s) = 1.0/N$

$\rightarrow H(s) \triangleq Y(s) / R(s) \mid X(s)=0$ = Transfer Function

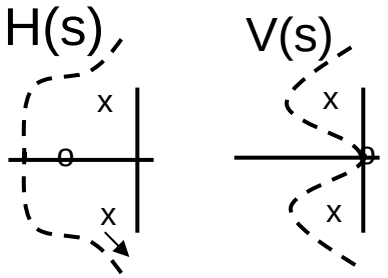
$= Kv(Kp\,s + Ki) / (s^2 + KvKp/N\,s + KvKi/N)$

$V(s) \triangleq Y(s) / X(s) \mid R(s) = 0$

$= Kv\,s / (s^2 + KvKp/N\,s + KvKi/N)$

$= Kv / s$ (s <- large)

$= 0$ (s <- 0)

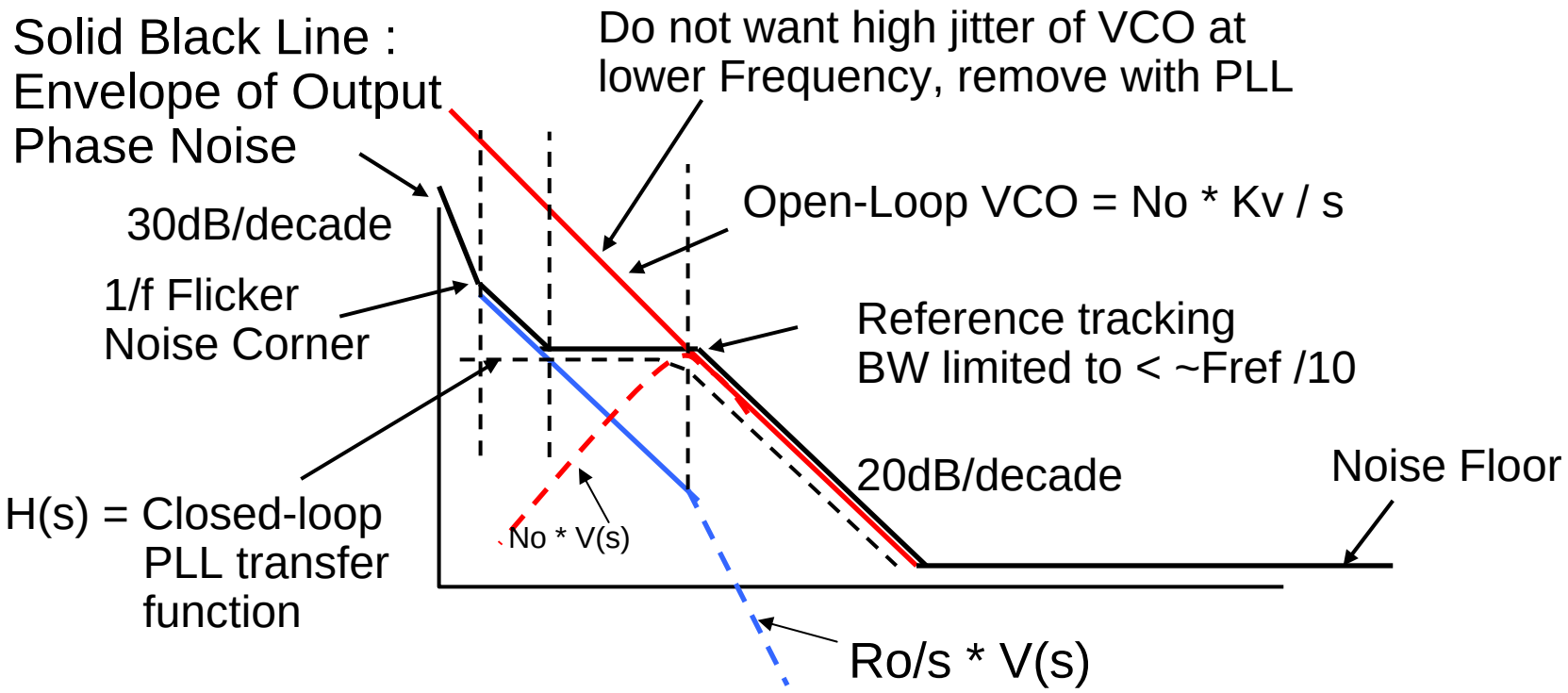


As Kp gets smaller, poles migrate
To jw axis and de-stabilize loop

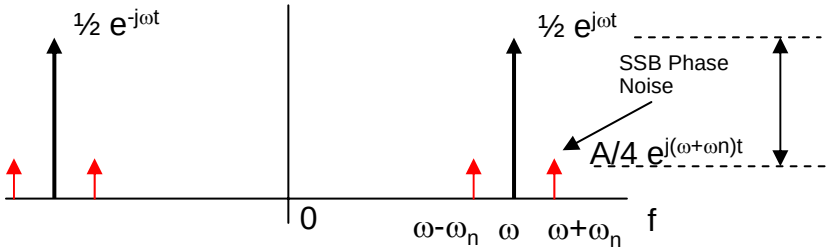
B.2 Simplified PLL Phase Noise Model

For R(s) = Ro / s (integrated noise) and X(s) = No (white noise)

$Y(s) = Kv\,(Ro\,*(Kp + Ki/s) + sNo) / (s^2 + KvKp/N\,s + KvKi/N)$



B.3 Time Jitter Derivation for an SJ Phase Tone



Power ratio = L = $(A/4)^2 / (1/2)^2 = (A/2)^2$

L (dBc) = 10 * log10(Power ratio) = 10 * log10($(A/2)^2$) dBc

L (numeric) = $(A/2)^2$

Phase jitter

$y(t) = \sin(\omega t + \phi_n(t))$

$= \sin(\omega t)\cos(\phi_n(t)) + \cos(\omega t)\sin(\phi_n(t))$

Let $\phi_n(t)$ = SJ tone = $A*\cos(\omega_n t)$, where $A << 1$

-> $y(t) =\sim \sin(\omega t) + \cos(\omega t)A\cos(\omega_n(t))$, using $\sin(\phi) =\sim \phi$ and $\cos(\phi) =\sim 1$ for $\phi << 1$

Jitter (in radians) = $A\cos(\omega t)\cos(\omega_n(t)) = (A/2)\cos(\omega+\omega_n(t)) + (A/2)\cos(\omega-\omega_n(t))$

RMS value of total jitter = 2 * (A/2) / sqrt(2) = A / sqrt(2)

Power of total jitter (in rad^2) = $A^2 / 2$

$L = (A/2)^2$, -> $A^2/2 = 2L$

Power /1Hz of total jitter (in rad^2 / 1Hz) = 2 * L

Integrated Jitter (in rad^2) = 2 * L

RMS phase jitter (in rad) = sqrt(2 * L)

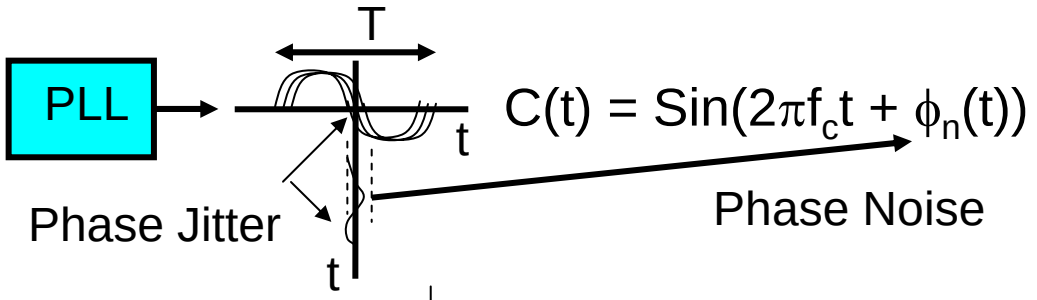
Generalization :

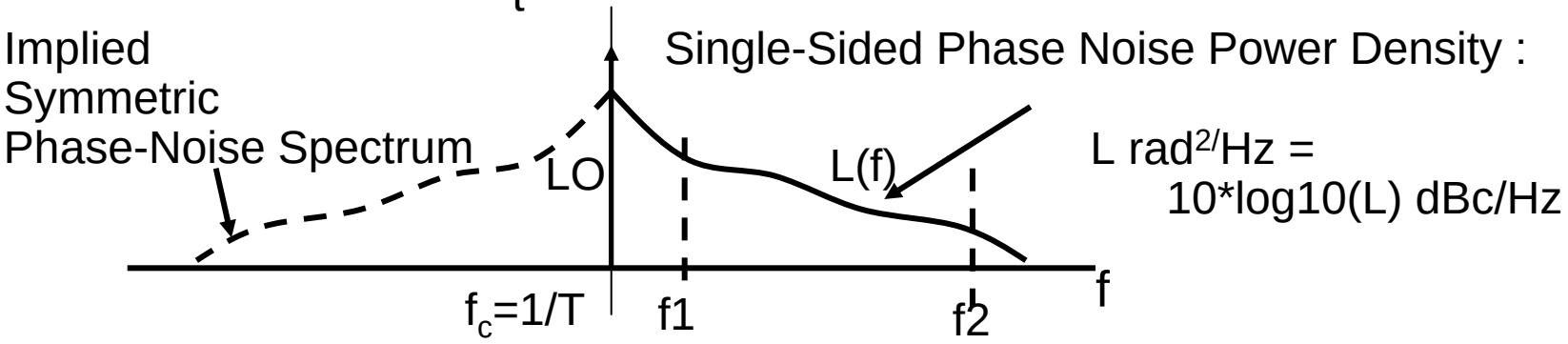
Jitter = 1 / ω * sqrt(2 * integrated SSB phase noise)

With ω = 2*pi*f, T = 1/f, there are (T seconds of time) / (2*pi) radians

-> Total Time Jitter RMS (in seconds) = T / (2*pi) * sqrt(2*L) = 1 / (2*pi*f) * sqrt(2*L) = 1/ω * sqrt(2*L)

B.4 Time Jitter Integration From SSB Phase Noise





Single-Sided Phase Noise Power Density :

L rad²/Hz =
10*log10(L) dBc/Hz

$$\sigma_N^2 = 2\int_{f_1}^{f_2} L(f)df \quad \text{rad}^2$$

rad

T seconds of time = 2π rad
1 rad = T / (2π) seconds of time

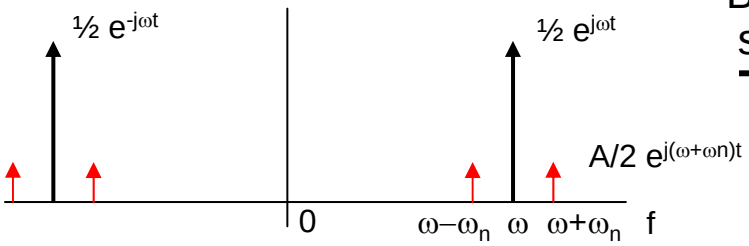
$$\text{-> Time Jitter (RMS)} = \frac{T}{2\pi}\sqrt{2\int_{f_1}^{f_2} L(f)df} = \frac{T}{2\pi\, f_c}\sqrt{2\int_{f_1}^{f_2} L(f)df}$$

B.5 Time Jitter Derivation for additive noise

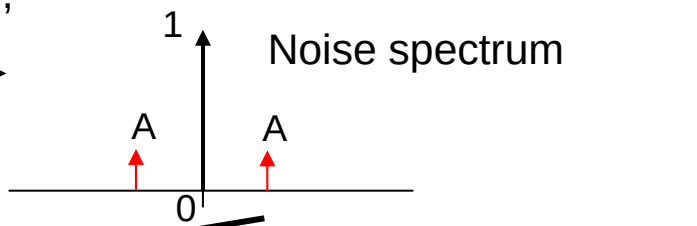
$\sin(\omega t) \rightarrow (\Sigma) \rightarrow y(t) = \sin(\omega t) + A\sin((\omega+\omega_n)t+\phi_1(t)) + A\sin((\omega-\omega_n)t+\phi_2(t)),\ A<<1$

N(t)

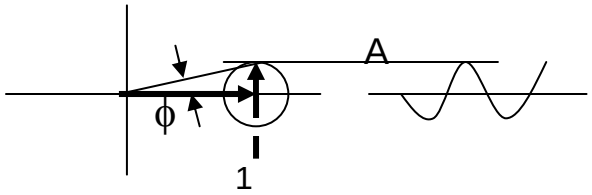
Randomized phases

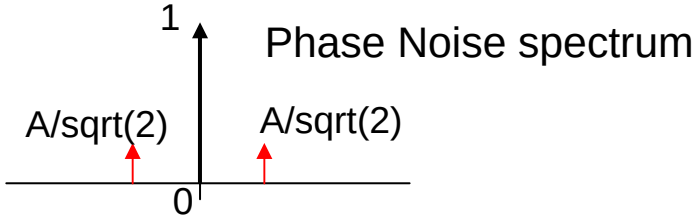


Baseband signal, scale x2



Phase noise from one ssb tone :





$\sin(\phi) =\sim A / 1,\ \phi =\sim A$

$L = (A/\mathrm{sqrt}(2))^2\ \mathrm{rad}^2/\mathrm{Hz} = A^2/2$ (see note1)

Peak phase noise = A rad
RMS phase noise = A / sqrt(2) rad
RMS phase power = A^2 / 2 rad^2
Total phase power from 2 ssb tones : $A^2\ \mathrm{rad}^2$

Total Phase noise power = $A^2\ \mathrm{rad}^2$ / Hz
= 2L
-> Jitter = $1.0/(2\pi f) * \mathrm{sqrt}(2L)$

->Jitter = $1.0/(2\pi f)$ *A seconds

note1 ½ of noise power goes onto phase noise,
½ of noise power goes onto amplitude noise

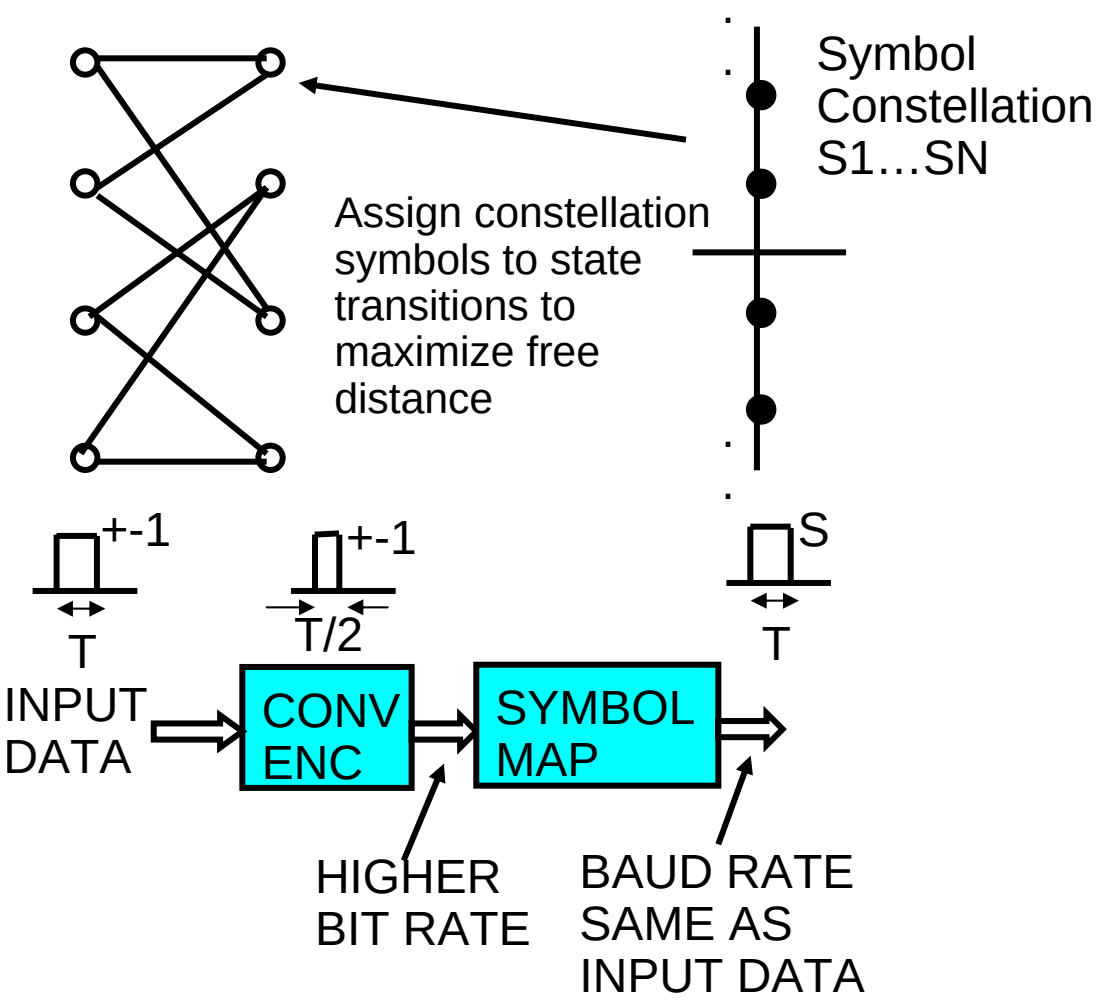
C.1 Digital Line-Coding/Modulation : TCM

Trellis-Coded Modulation (TCM)

Advantages :
- Add coding gain with no bandwidth extension

Disadvantages :
- Complexity
- Constellation moves away from NRZ, increases Peak/Average power

Used in :
- Phone modems
- Cellular standards



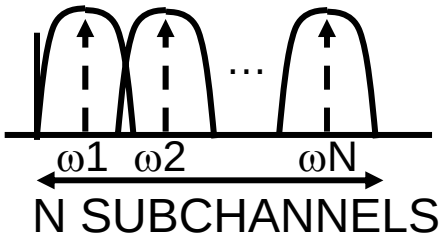
C.2 Subchannel Modulation : OFDM

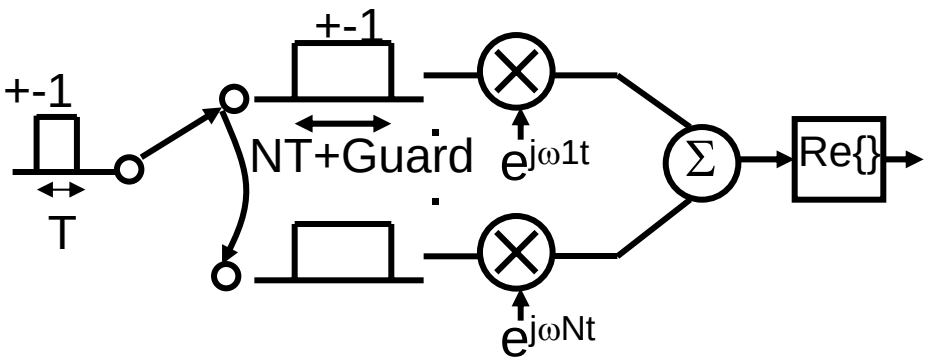
Orthogonal Frequency Divison Multiplexing (OFDM)

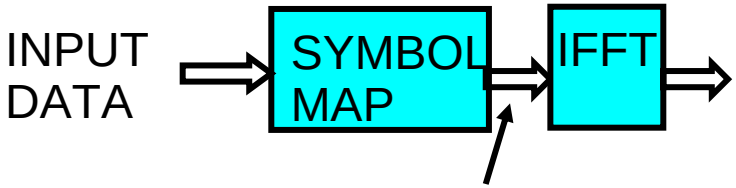
Advantages :
- Handles dispersive channel well

Disadvantages :
- -Complexity
- -High Peak/Average Power
- -"Guard time" needed to maintain subchannel orthogonality

Used in :
- DSL Broadband
- 802.11a/g Wireless







SUBCHANNEL BAUD RATE
~1/N AS FAST AS INPUT DATA