ECE 546
Lecture - 25
Advanced Jitter Analysis

Spring 2024

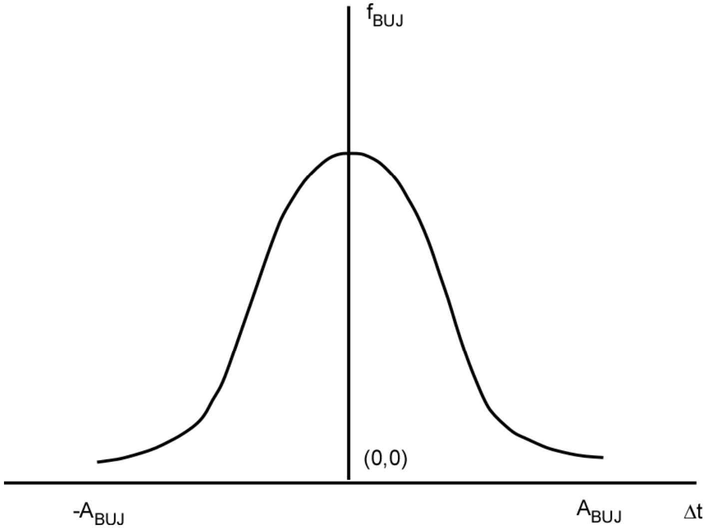
Jose E. Schutt-Aine
Electrical & Computer Engineering
University of Illinois
jesa@illinois.edu

Bounded Uncorrelated Jitter

BUJ is primarily due to crosstalk

The PDF for BUJ is given by

$$f_{PJ}(\Delta t) = \begin{cases} \dfrac{p_{BUJ}}{\sqrt{2\pi}\sigma_{BUJ}} e^{-\frac{t^2}{2\sigma_{BUJ}^2}} & for \;\; |\Delta t| \leq A_{BUJ} \\ 0 & for \;\; |\Delta t| \leq A_{BUJ} \end{cases}$$



Mix of Random and Periodic Jitters

Gaussian RJ and Rectangle PJ

➔ Obtain convolution of 2 PDFs

$$RJ * PJ_{rect} = \int_{-\infty}^{+\infty} RJ\left(t-\tau\right)\left[\delta\left(-\frac{m}{2}\right)+\delta\left(\frac{m}{2}\right)\right]d\tau$$

$$= \frac{1}{2\sigma\sqrt{2\pi}}\left[e^{-\frac{\left(t-m/2\right)^2}{2\sigma^2}}+e^{\frac{\left(t+m/2\right)^2}{2\sigma^2}}\right]$$

Result is the sum of 2 Gaussian distributions with equal RMS value offset by the PJ peak-to-peak value . It is called the DUAL DIRAC DISTRIBUTION

Jitter Mixing

- **Problem**
  - In tests, we have measured jitter histograms and need to extract the individual jitter components
  - Ideally, we could use deconvolution into components. However without prior knowledge of deterministic jitter, it is not possible
  - Use dual Dirac distribution model which would yield the worst case deterministic jitter

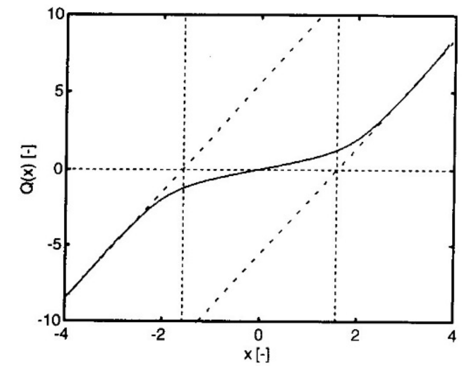
Q-Scale Transformation

Q-scale is defined such that the Gaussian distribution mapped onto the Q-scale is a straight line

Use CDF

$$CDF(x) = \frac{1}{2} + \frac{1}{2} erf\left(\frac{x}{\sigma\sqrt{2}}\right)$$

$$Q(x) = \sqrt{2} erf^{-1}\left(2CDF(x) - 1\right) = \frac{x}{\sigma}$$

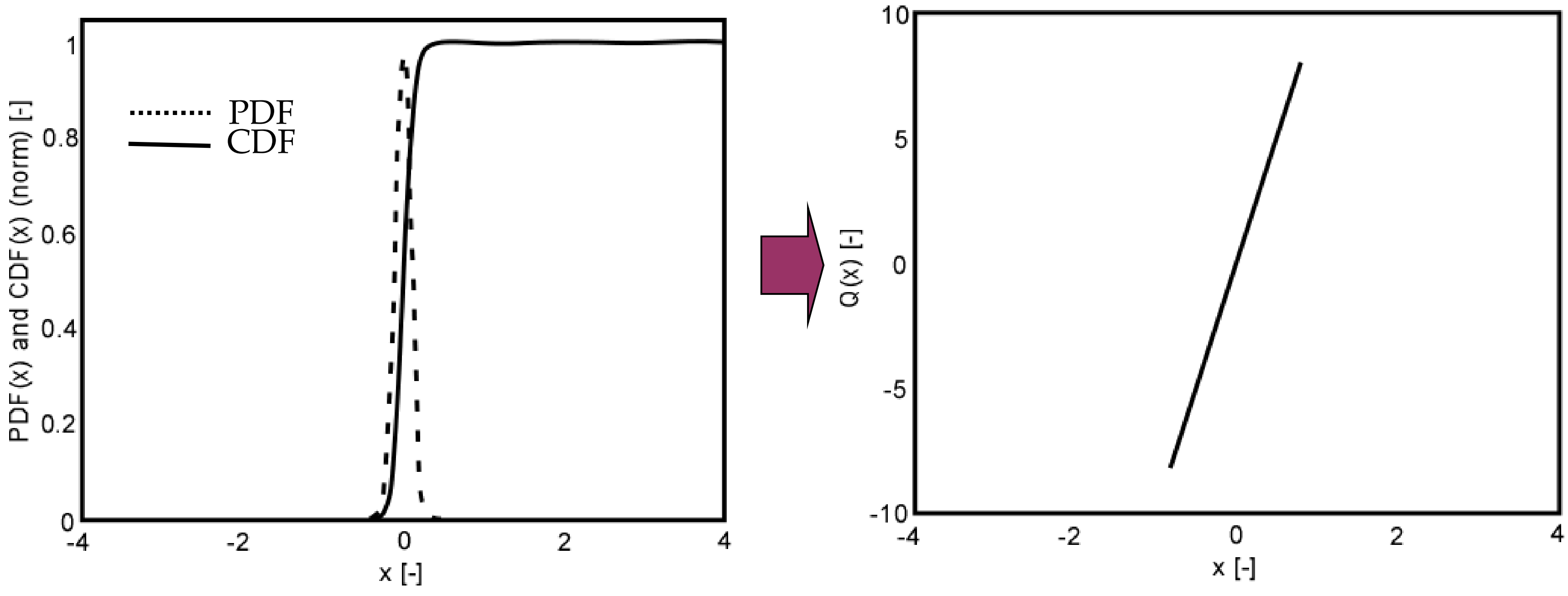


A Gaussian CDF is a straight line in the Q scale with slope 1/σ. DJ is given by distance d

Q-Scale Transformation

Gaussian RJ

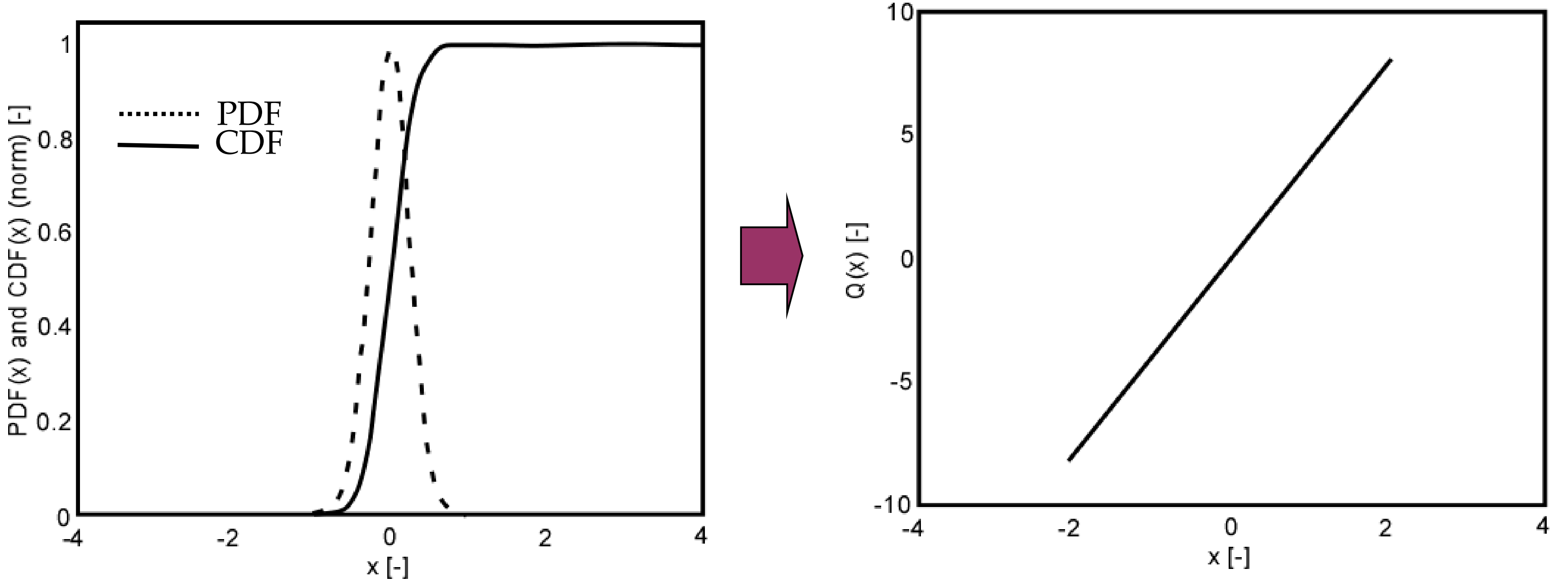
$\sigma = 0.5$



Q-Scale Transformation

Gaussian RJ

$\sigma = 0.25$

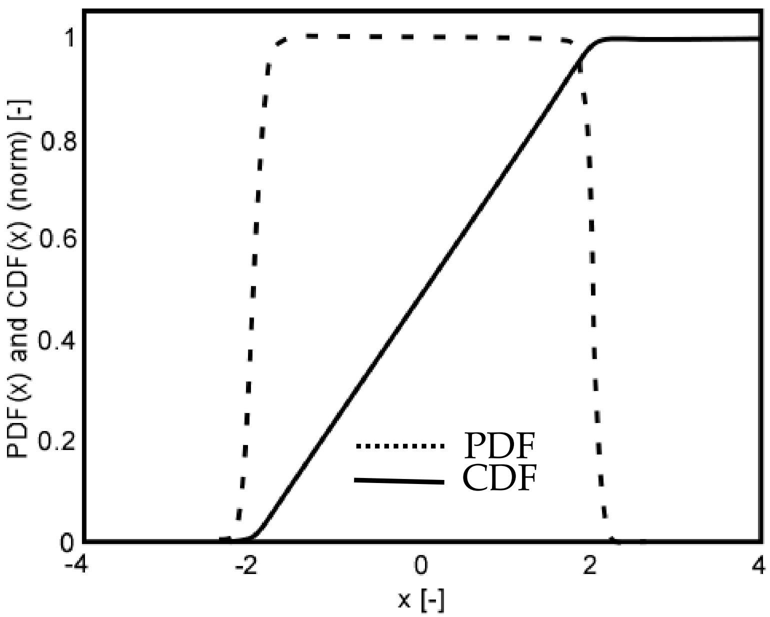


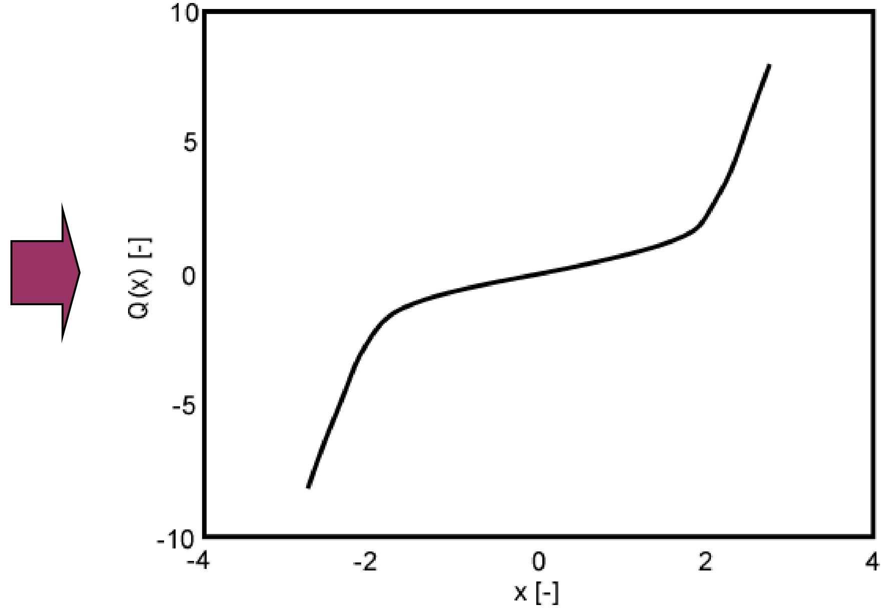
Q-Scale - Generalization

$$Q(x) = \sqrt{2}erf^{-1}\left(2CDF(x) - 1\right) = \frac{x}{\sigma}$$

Mixed Gaussian RJ and PJ

$$\sigma = 0.1$$



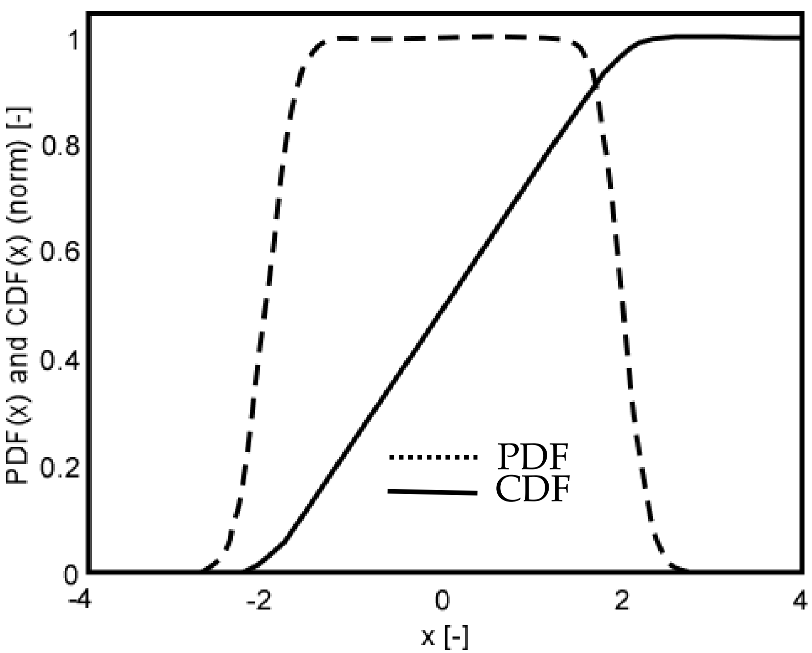


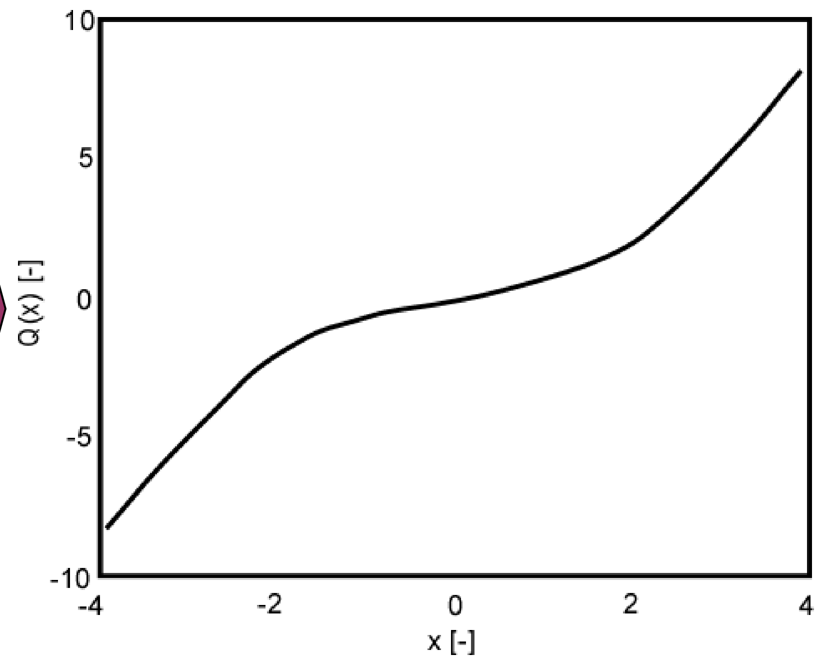
Q-Scale - Generalization

$$Q(x)=\sqrt{2}erf^{-1}\left(2CDF(x)-1\right)=\frac{x}{\sigma}$$

Mixed Gaussian RJ and PJ

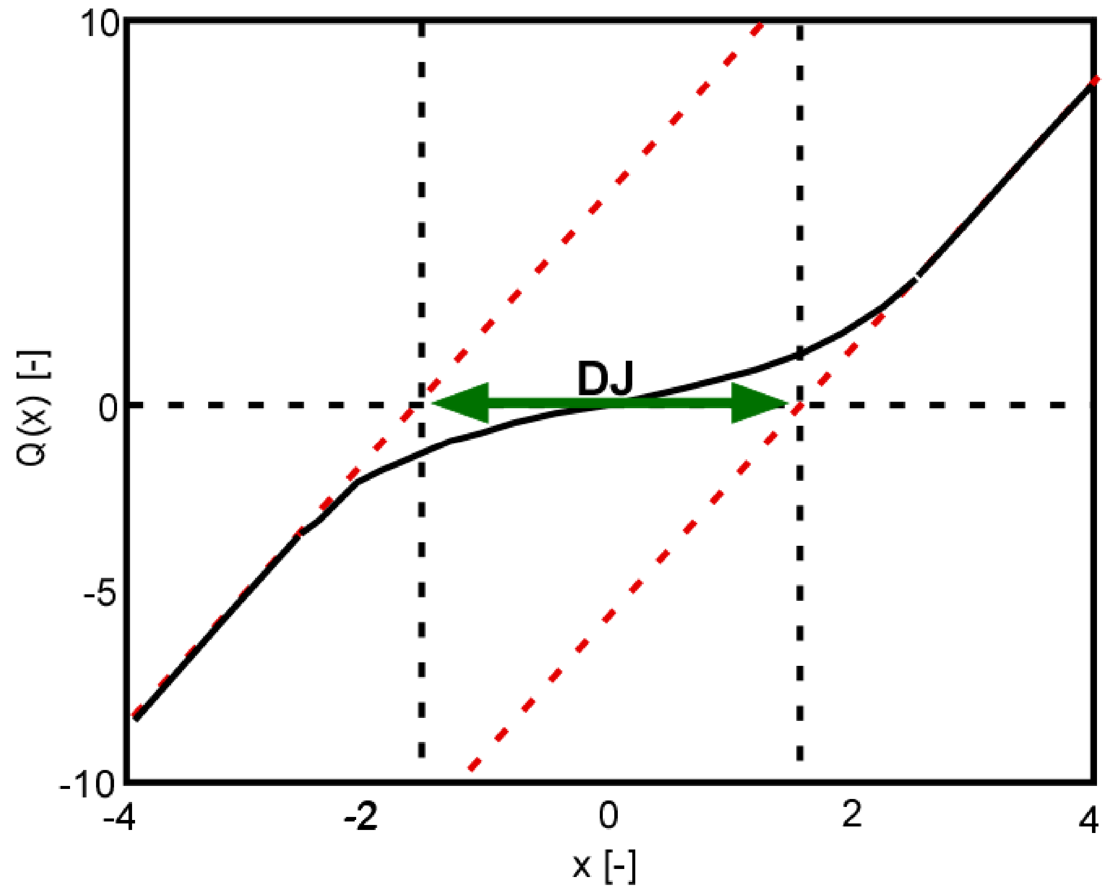
σ = 0.25





Dual Dirac Model

Mixed Gaussian RJ and Triangular PJ



Jitter Mixing

- Problem
  - In tests, we have measured jitter histograms and need to extract the individual jitter components
  - Ideally, we could use deconvolution into components. However without prior knowledge of deterministic jitter, it is not possible
  - Use dual Dirac distribution model which would yield the worst case deterministic jitter

Random Jitter Extraction

- **Spectrum Analysis**

  - Extract random jitter by using the assumption that it has a piecewise linear spectrum

  - Impulses are attributed to DJ

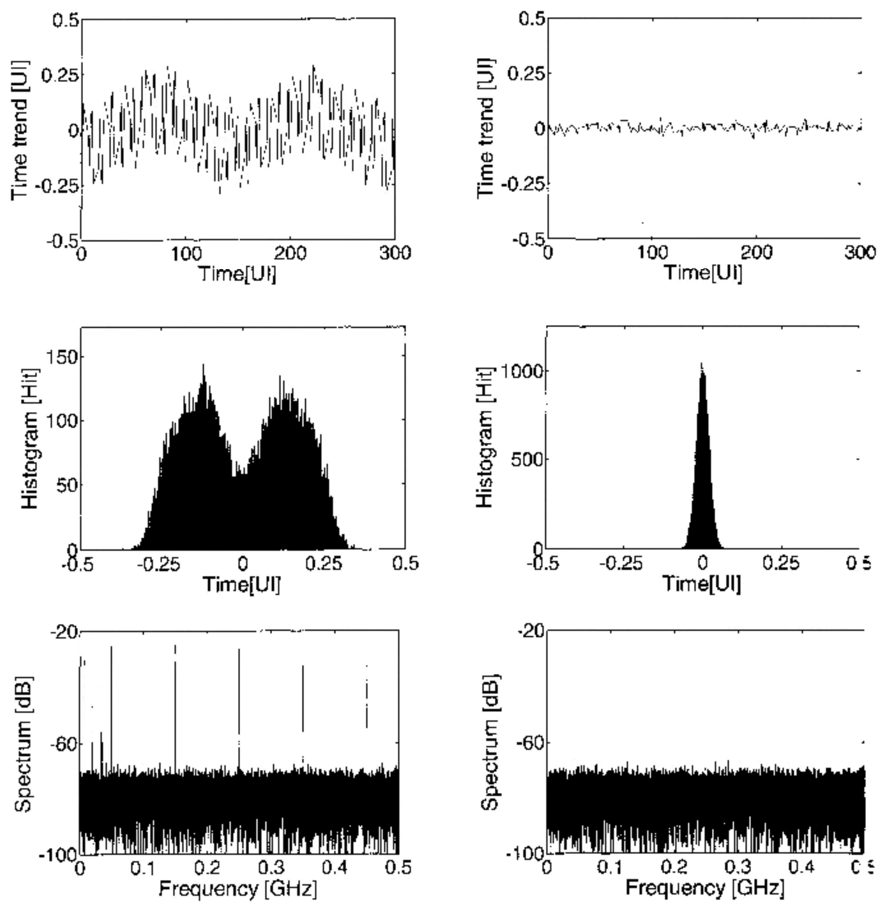
  - Noise floor is due to RJ

Extracting Random Jitter

Time domain

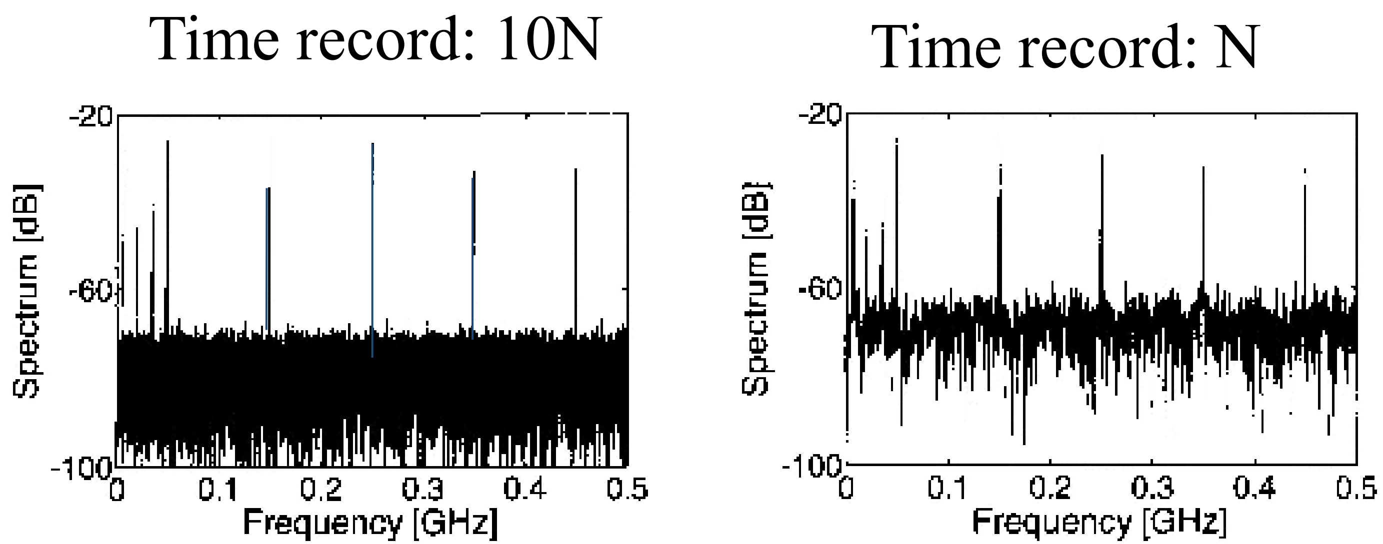
Statistical domain

Spectral domain



Total jitter **Random jitter**

Jitter Spectrum



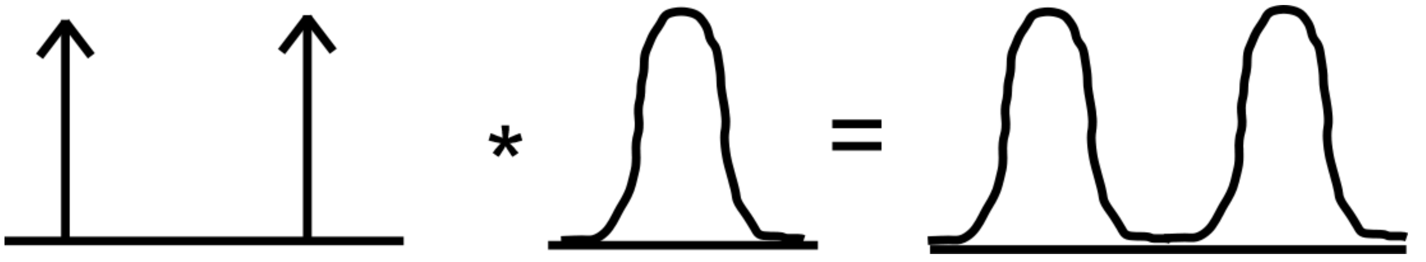
A longer FFT yields a spectrum with greater frequency resolution and lower noise floor.

Random Jitter Extraction

- **Tail-Fit**
  - Extract random jitter under the assumption that its probability density function follows a Gaussian distribution
  - Make use of the Dual-Dirac Model

Dual Dirac Model

Unknowns
- gap between 2 impulses
- σ for Gaussian distribution

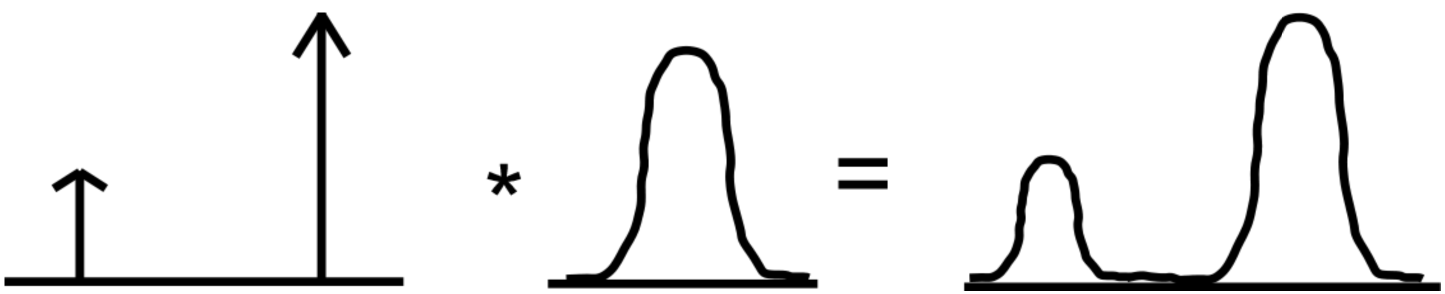


- Equal Amplitudes
  - Two unknown variables
  - Linear Problem
  - Explicit solution

Dual Dirac Model

Unknowns

- gap between 2 impulses
- σ for Gaussian distribution
- ratio of 2 impulse amplitudes



- **Unequal Amplitudes**
  - Three unknown variables
  - Nonlinear Problem
  - No explicit solution

Dual Dirac Model

Assume Gaussian RJ and Rectangle PJ

➔ Obtain convolution of 2 PDFs

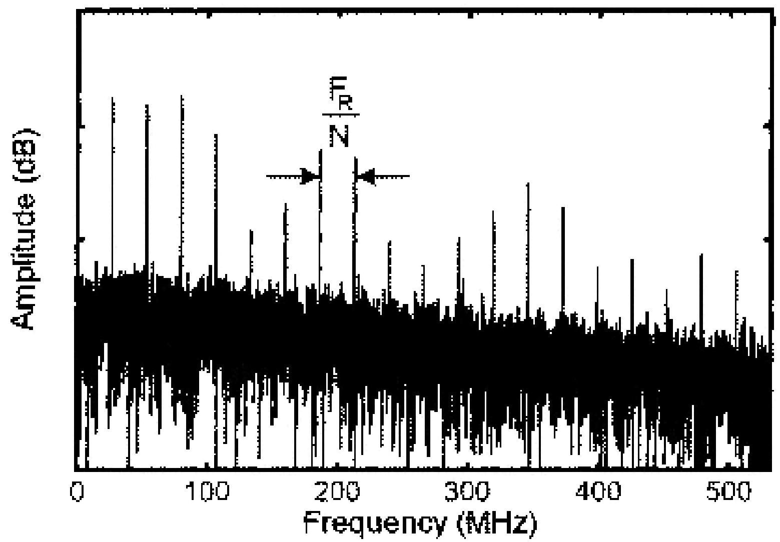
$$RJ * PJ_{rect} = \int_{-\infty}^{+\infty} RJ(t-\tau)\left[\delta\left(-\frac{m}{2}\right)+\delta\left(\frac{m}{2}\right)\right]d\tau$$

$$= \frac{1}{2\sigma\sqrt{2\pi}}\left[e^{-\frac{(t-m/2)^2}{2\sigma^2}} + e^{\frac{(t+m/2)^2}{2\sigma^2}}\right]$$

Result is the sum of 2 Gaussian distributions with equal RMS value offset by the PJ peak-to-peak value . It is called the **DUAL DIRAC DISTRIBUTION**

DDJ and DC D

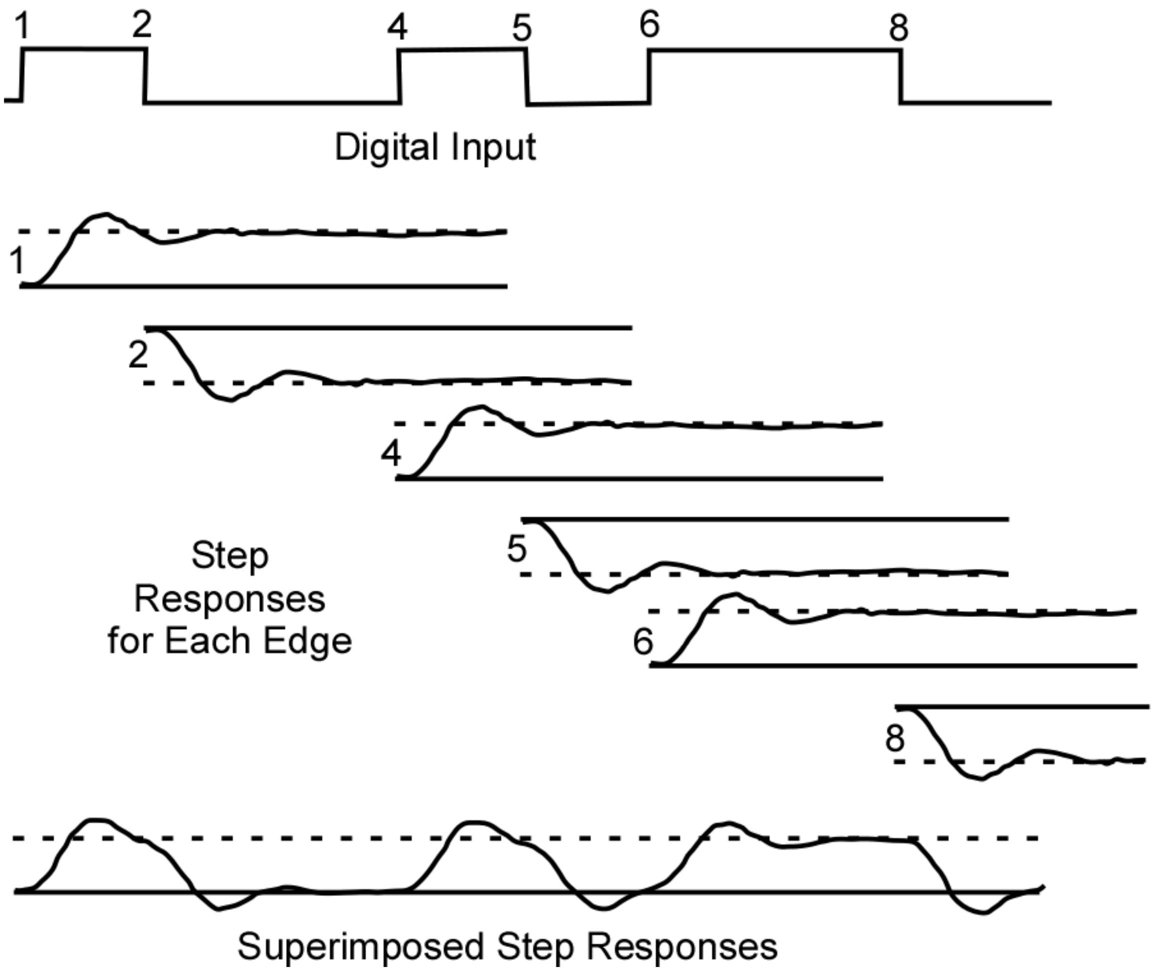
- DDJ and DCD are correlated to the data pattern



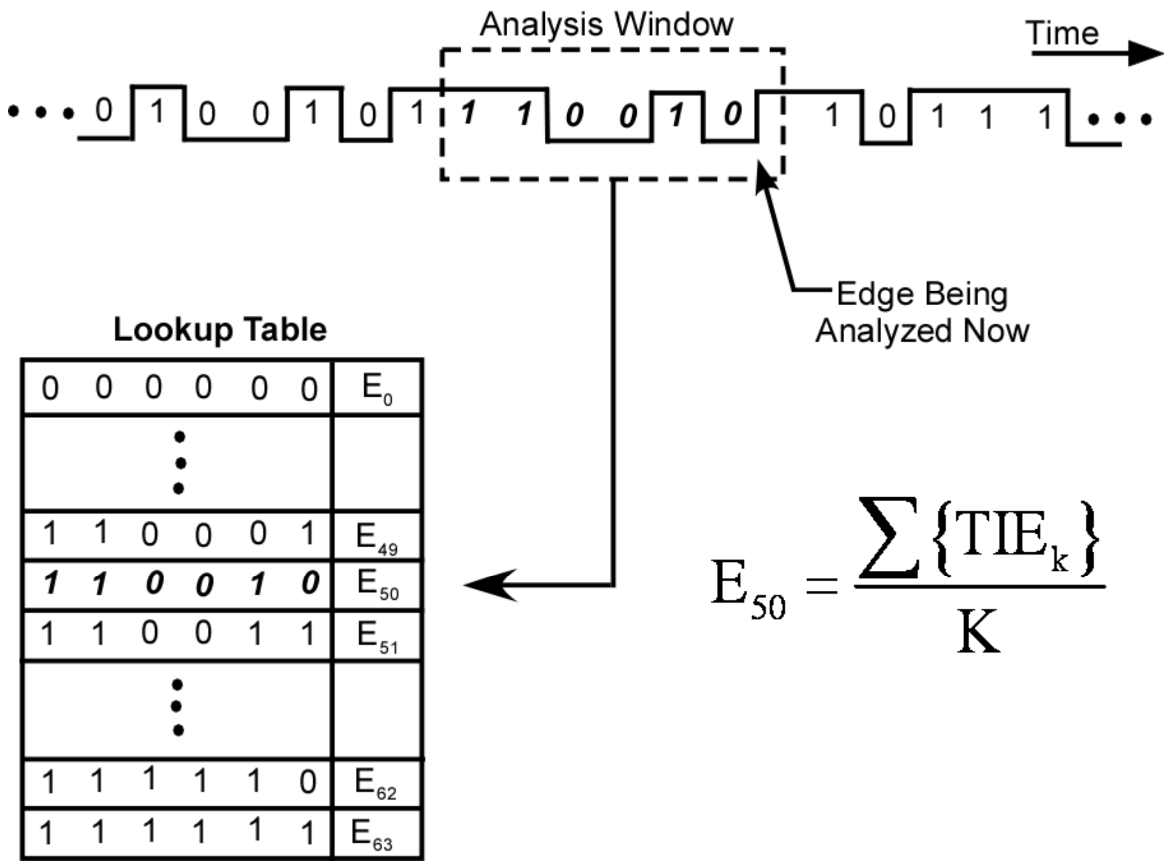
F_R=1.0625 Gbits/s
N=40 bits

For N bits, transmitted at rate F_R, the jitter components due to DDJ and DCD will appear in the spectrum at multiple of F_R/N

Pattern Correlation



Pattern Correlation



$$E_{50} = \frac{\sum \{\mathrm{TIE}_k\}}{K}$$

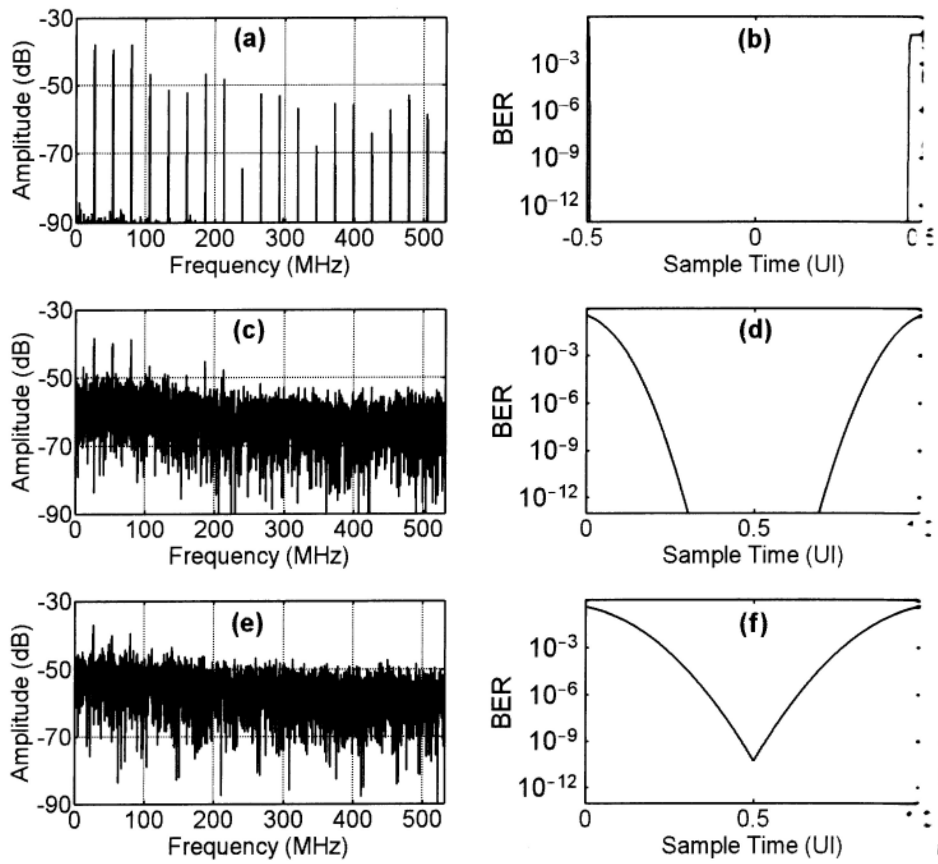
The phase errors from all occurrences of each M-bit patterns are averaged together to estimate the phase error due to that M-bit pattern

Extracting DDJ

DDJ Dominant

DDJ & RJ

RJ Dominant



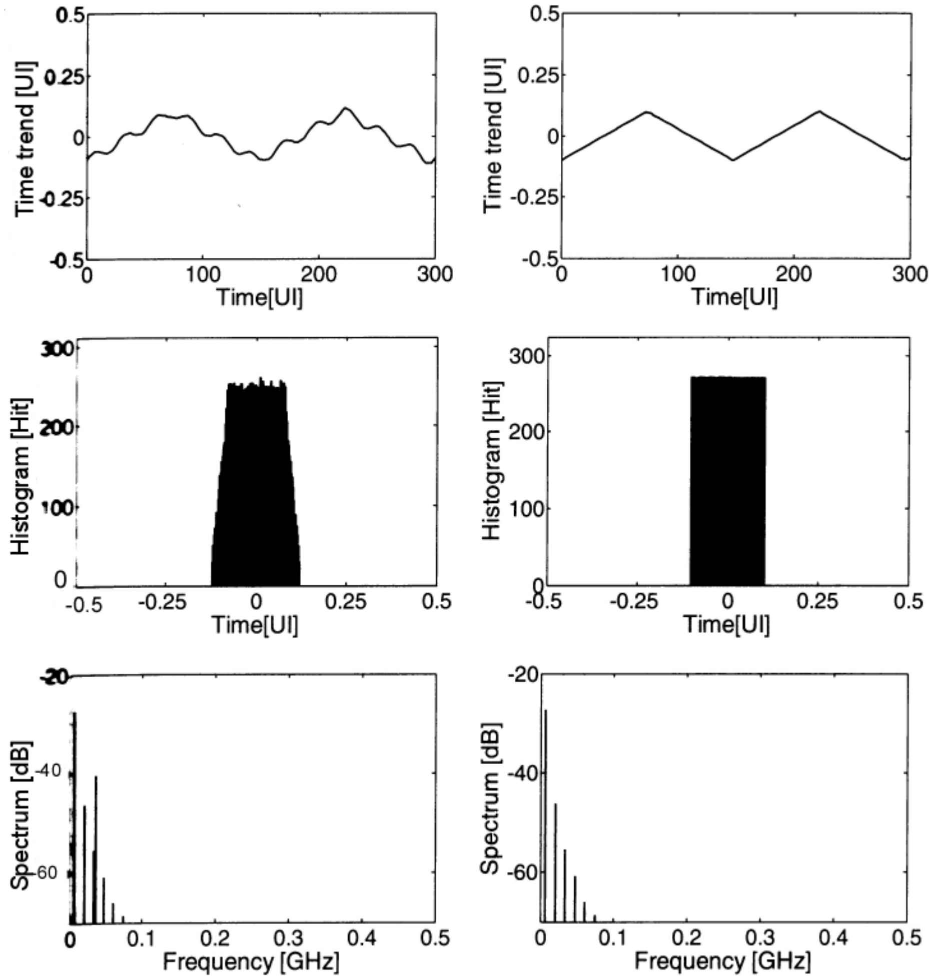
Spectral domain **Eye**

Periodic Jitter

Time domain

Statistical domain

Spectral domain



PJ | **PJ subcomponent**

Clock Jitter

In a computer system, the clock is used to provide timing or synchronization for the system.

In a communication system, the clock is used to specify when a data switch or bit transaction should be transmitted and received

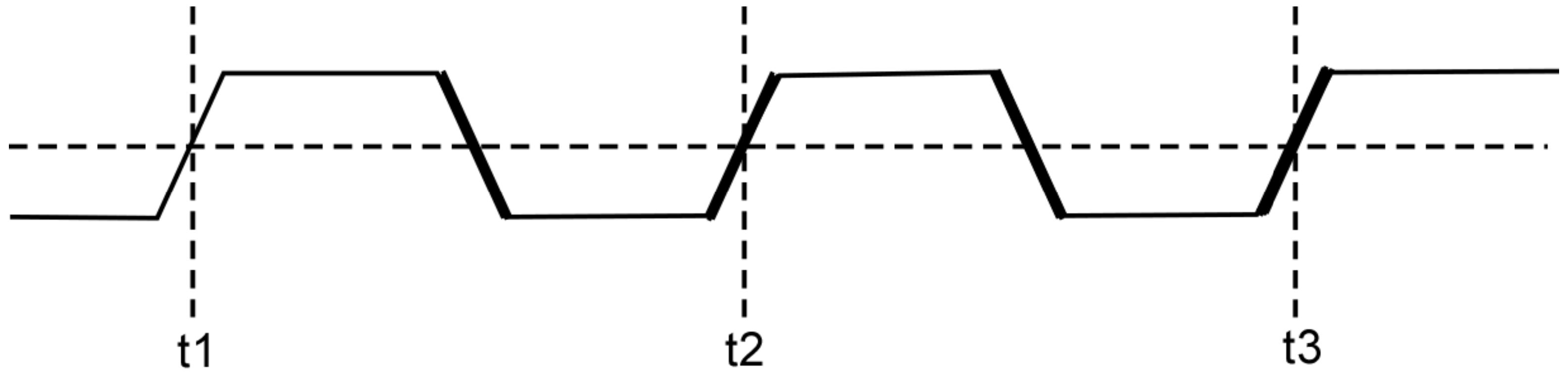
In a synchronized system, a central global clock is distributed to its subsystem

Clock jitter is the single most important degrader of clock performance

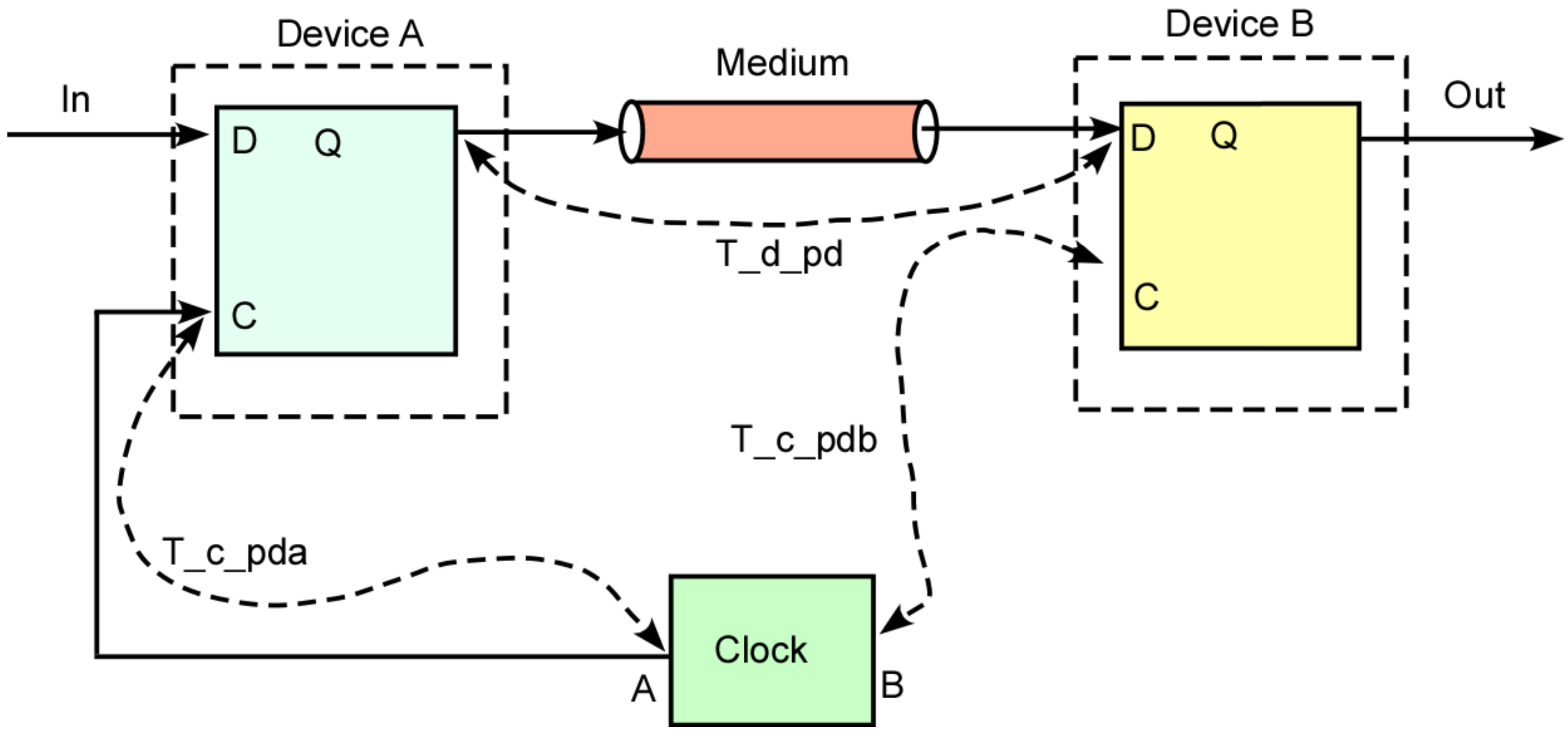
Definition

- Most of the definitions of data jitter (DJ, Rj,…) apply to clock jitter

- ISI does not apply to clock jitter

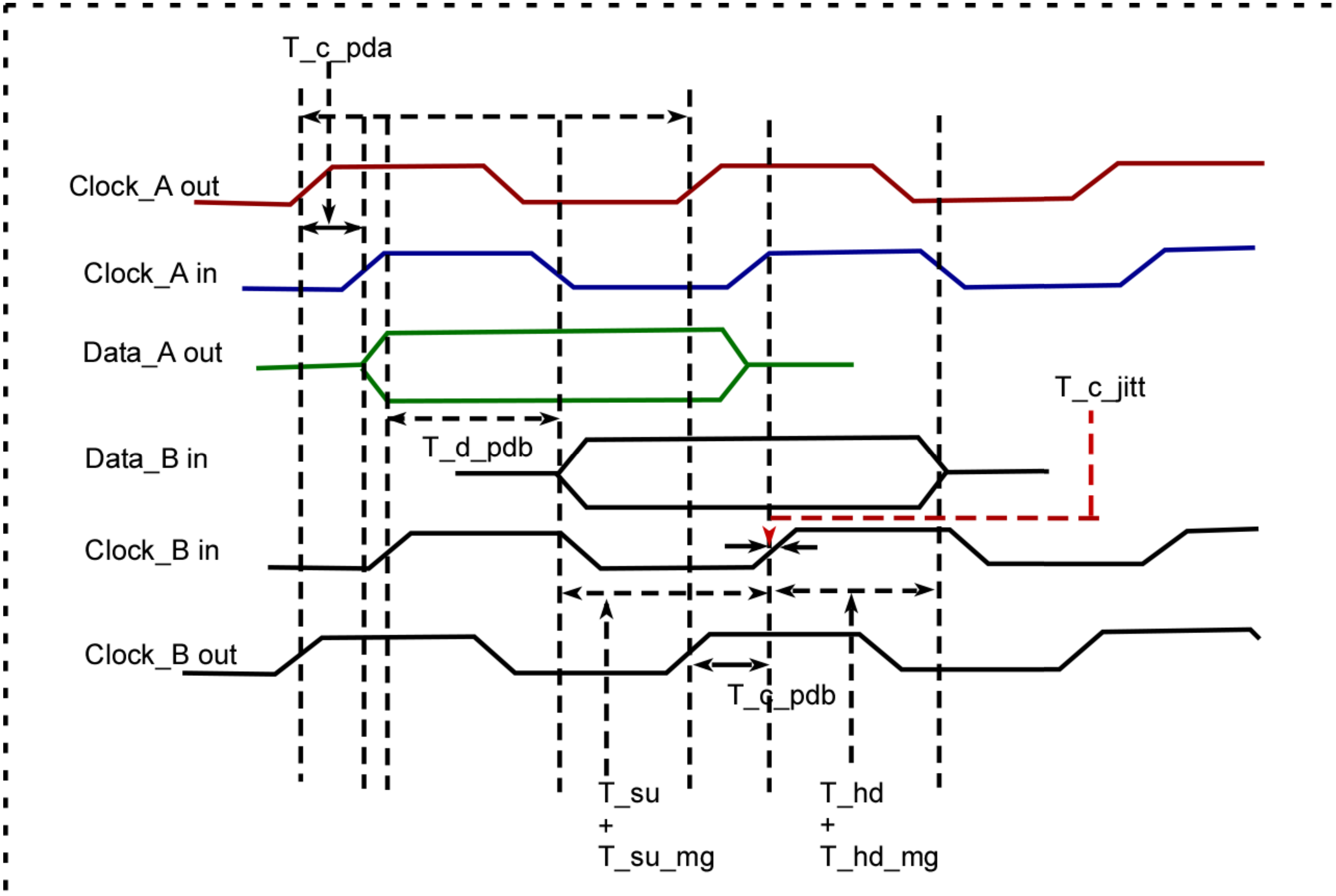


Synchronized System



- Initial clock pulse causes A to latch data from input and launch it into channel
- Second clock causes B to latch the incoming data

Timing Parameters



Timing Conditions

The minimum conditions are that both setup time and hold time margin should be larger than 0

$$T_0 \geq -T_{c\_jitt} + T_{c\_skew} + T_{d\_pd} + T_{su}$$

$$T_{hd} \leq T_{d\_pd} + T_{c\_skew} - T_{c\_jitt}$$

These give a quantitative description of how clock jitter and clock skew affect the performance of the synchronized system in which a common or global clock for both driver and receiver is used

Skew Impact

- $T_{c\_jitter}=0$, $T_{c\_skew}>0$
  - The minimum clock period increases. The maximum hold time increases ➔hold time condition easier to meet

- $T_{c\_jitter}=0$, $T_{c\_skew}<0$
  - The minimum clock period decreases. The maximum hold time decreases ➔hold time condition harder to meet (race condition)

Jitter Impact

- $T_{c\_skew}=0$, $T_{c\_jitter}>0$ (longer cycle)
  - The minimum clock period increases. The maximum hold time decreases ➔hold time condition harder to meet

- $T_{c\_skew}=0$, $T_{c\_jitter}<0$ (shorter cycle)
  - The minimum clock period decreases. The maximum hold time increases ➔hold time condition easier to meet

System Performance

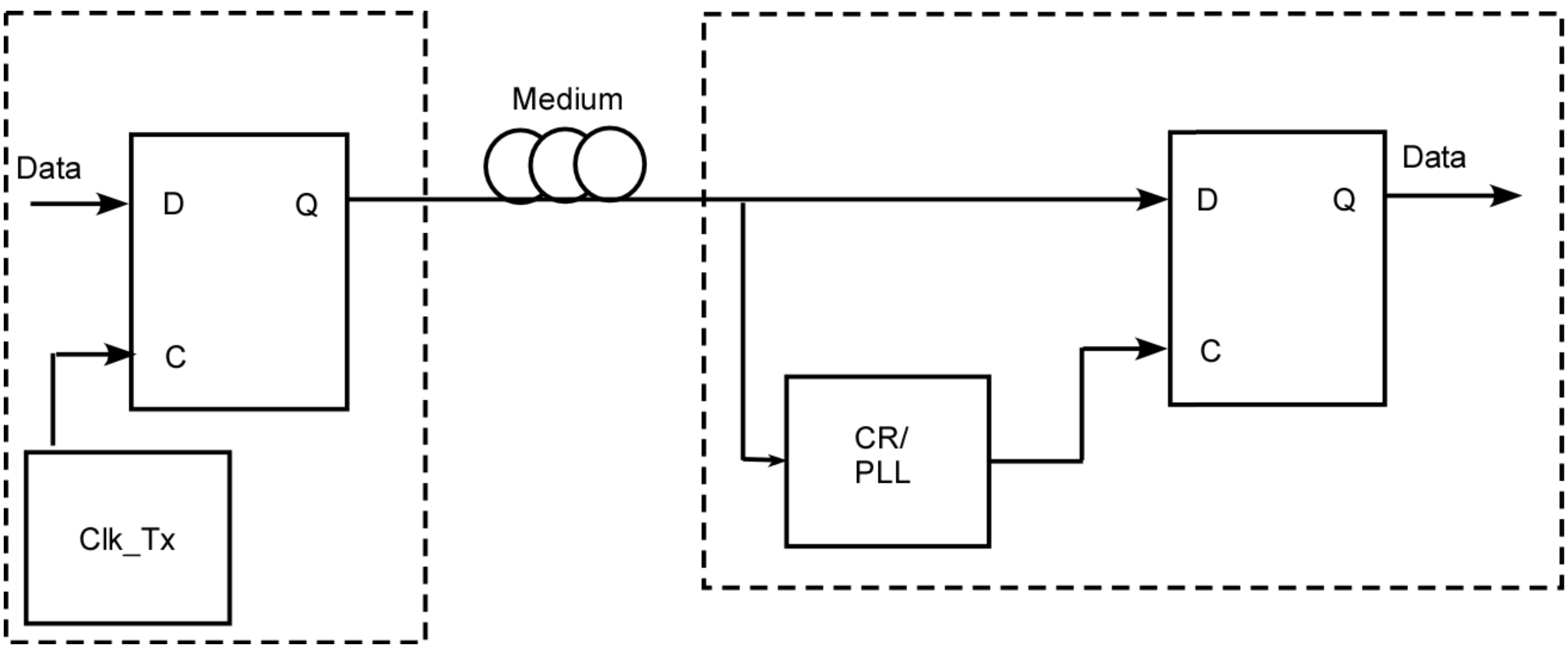
1. Positive jitter over one clock period makes both clock period and hold time hard to meet

2. A longer cycle does more harm to system performance

3. When both skew and jitter are present, system performance can be any of the four scenarios just discussed

Asynchronized System

The skew of a synchronized system becomes hard to manage when the data rate increases(~1 Gb/s). At multiple Gb/s data rates, an asynchronized system is commonly used.



Clock Types

- Synchronized System
  - Global clock is used to update and determine bits
- Asynchronized System
  - Only data is sent
  - Clock is embedded in data
  - Clock recovery unit (CRU) recovers clock at receiver

Asysnchronized Link

$$DJ_{clk\_tot} = DJ_{clk\_tx} + DJ_{clk\_rx}$$

$$\sigma^2_{clk\_tot} = \sigma^2_{clk\_tx} + \sigma^2_{clk\_rx}$$

Low-frequency jitter from the transmitter clock can be tracked or attenuated by the clock recovery function if it has a high enough corner frequency. A low phase noise oscillator within a PLL clock recovery also provides smaller random jitter generations.

Phase Jitter

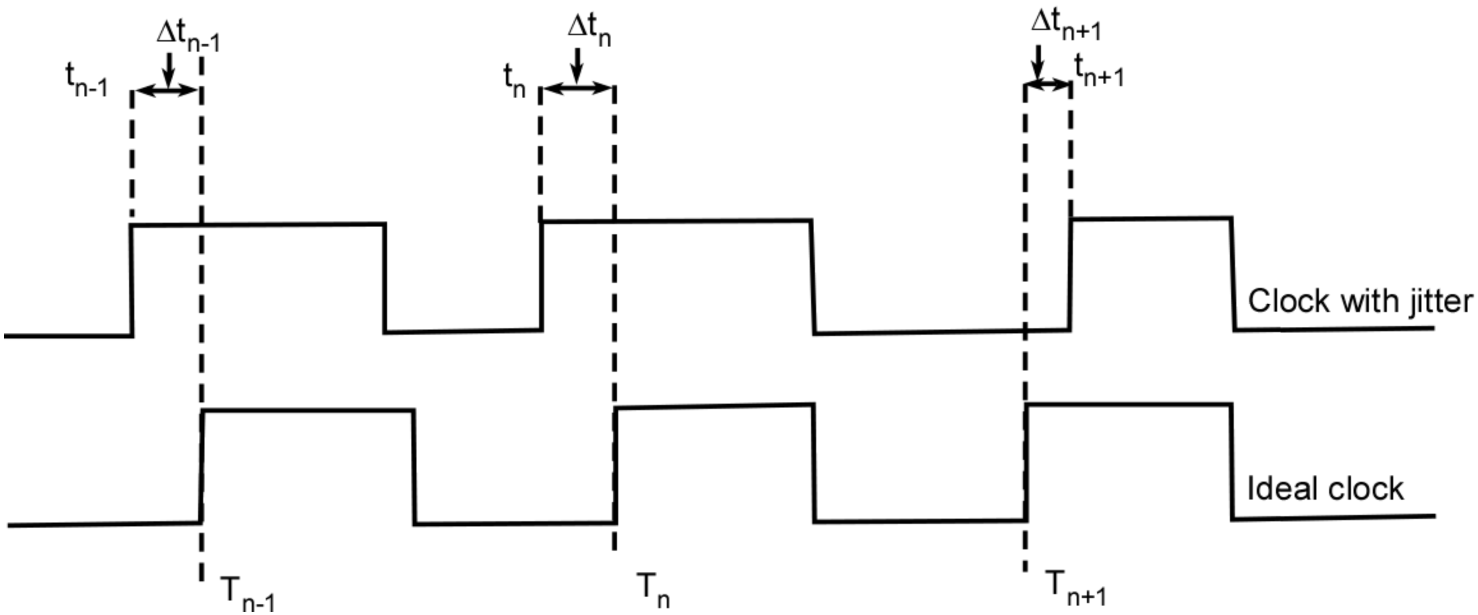
t_n : timing for nth edge for **jittery** clock

T_n : timing for nth edge for **ideal** clock

T_o : ideal clock period

$$\Delta t_n = t_n - T_n$$

$$T_n = nT_o$$



Phase Jitter

Phase jitter captures the instance timing deviation from the ideal for each transition. Jitter measured with phase jitter is absolute and accumulates over time.

In frequency domain

$$\phi_n = \frac{t_n}{T_o} 2\pi$$

Period Jitter

Period jitter is defined as the period deviation from the ideal period.

$$\Delta t_{pn} = \left(t_n - t_{n-1}\right) - T_o$$

using previous relations

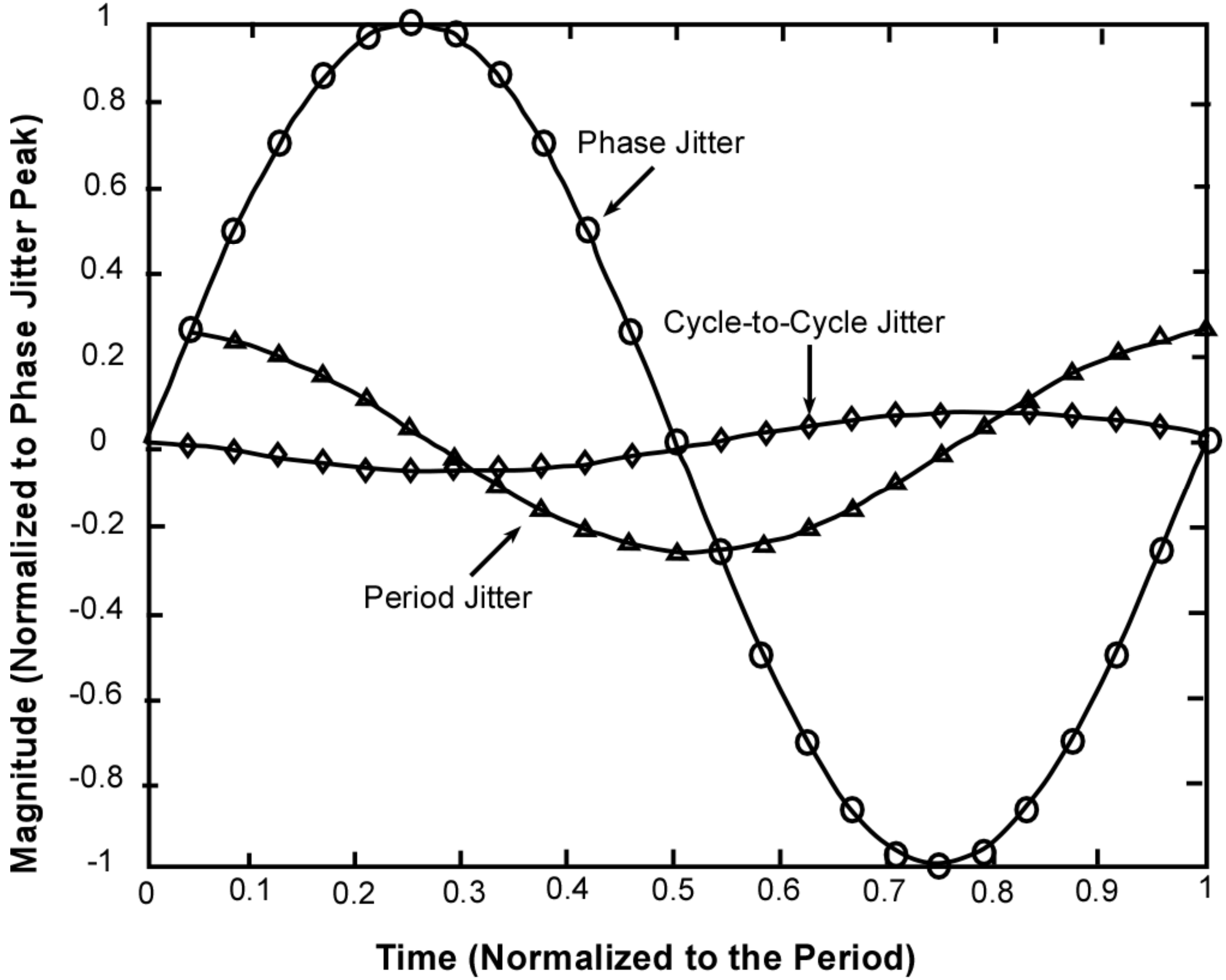
$$\Delta t_{pn} = \Delta t_n - \Delta t_{n-1}$$

in terms of phase units

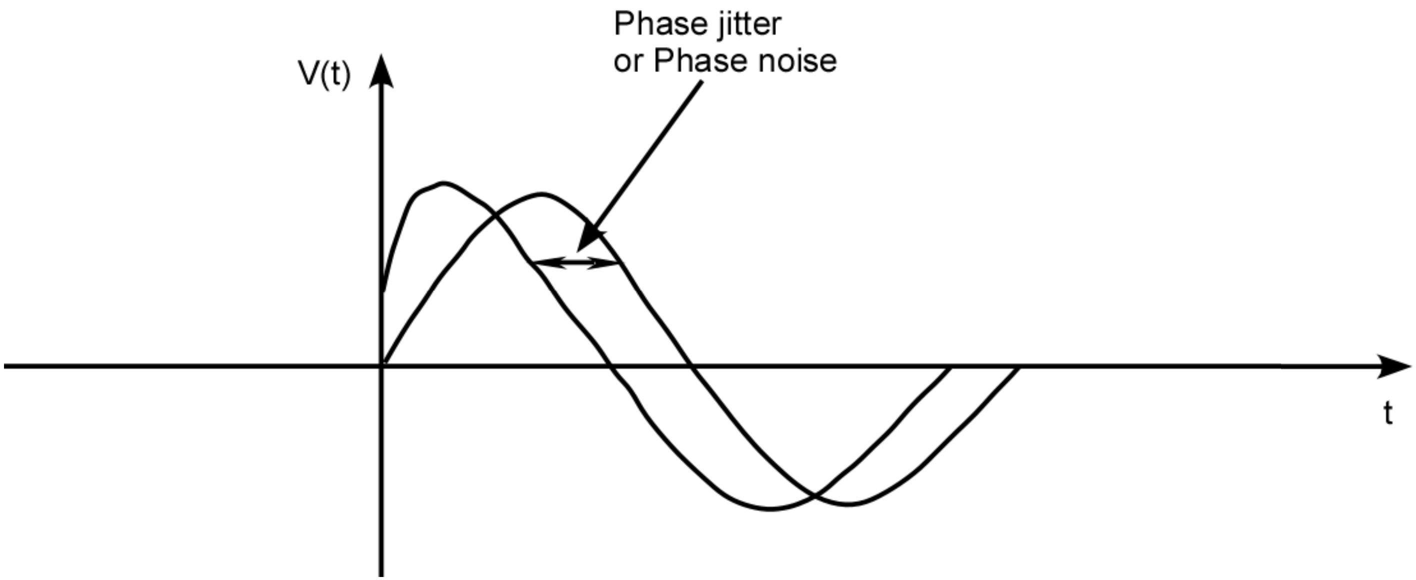
$$\phi_n^{'} = \Phi_n - \Phi_{n-1}$$

Period jitter and phase jitter are not independent → we can derive one from the other.

Phase, Period and CTC Jitter

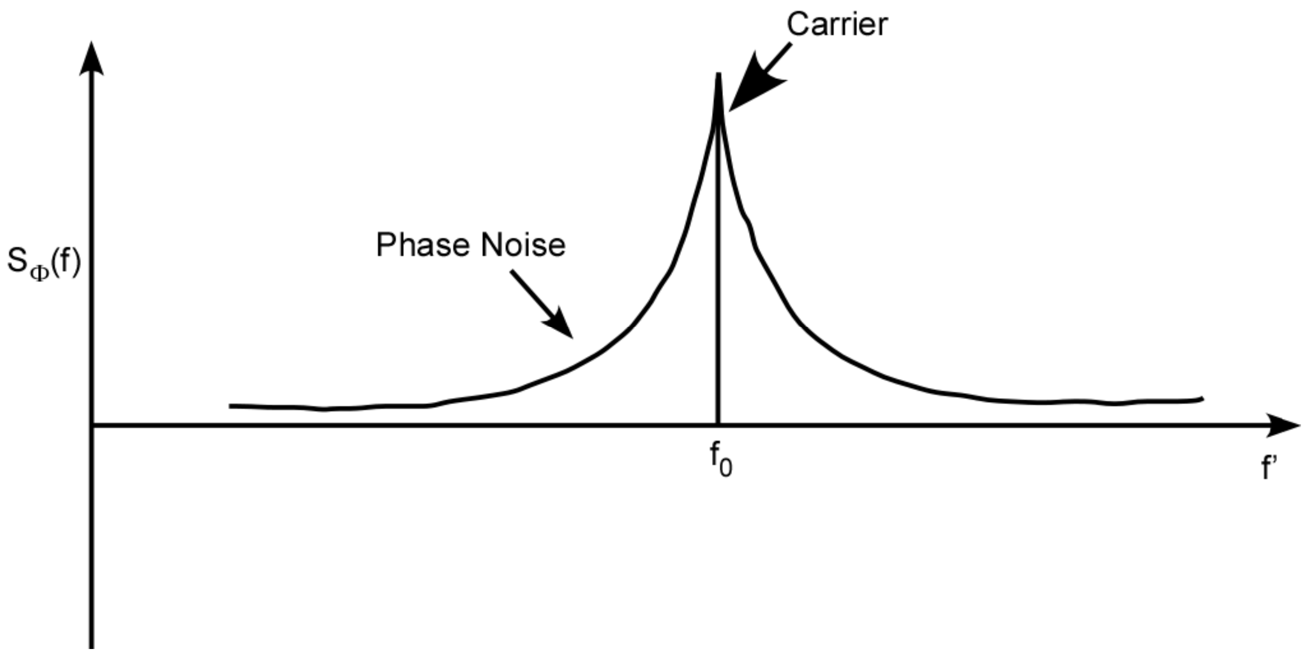


Phase Jitter in Time Domain



If the phase varies, the waveform V(t) shifts back and forth along the time axis and this creates phase jitter

Phase Jitter in Spectral Domain



Phase noise appears as sidebands centered around the carrier frequency

Phase Jitter

Phase noise magnitude is specified relative to the carrier's power on a per-hertz basis

$$L(f) = \frac{P_n(f)}{P_o \Delta f}$$

$P_n(f)$: phase noise power (in watts)

P_o : carrier's power (in watts)

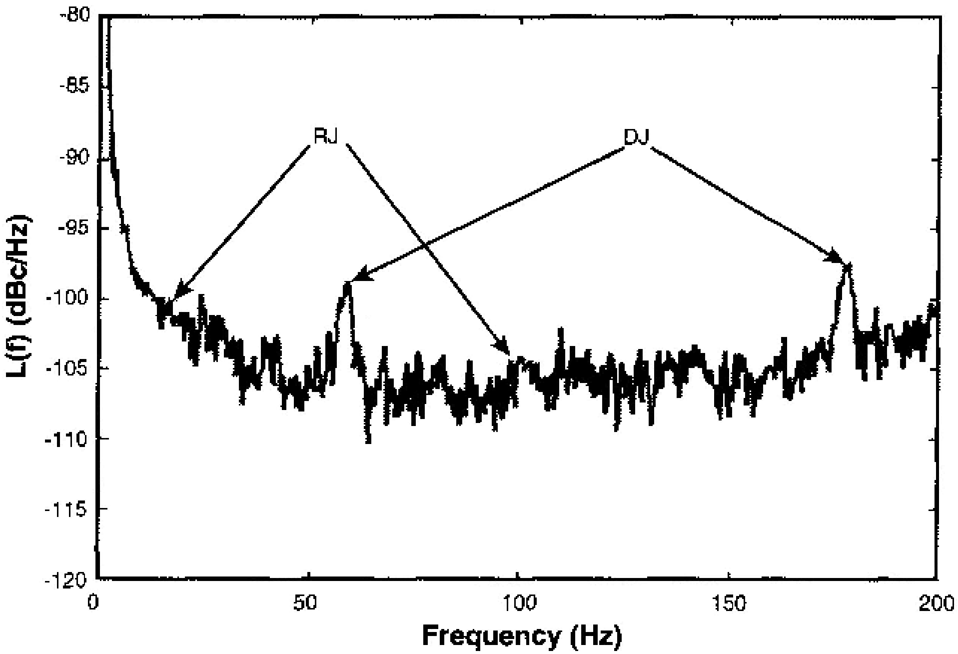
Δf : phase noise bandwidth (in hertz)

$$L(f) = \frac{1}{2} S_\Phi(f) \qquad \text{or} \qquad L(f) = 10\log_{10}\left(\frac{S_\Phi(f)}{2}\right)$$

$S_\Phi(f)$: PSD of phase noise

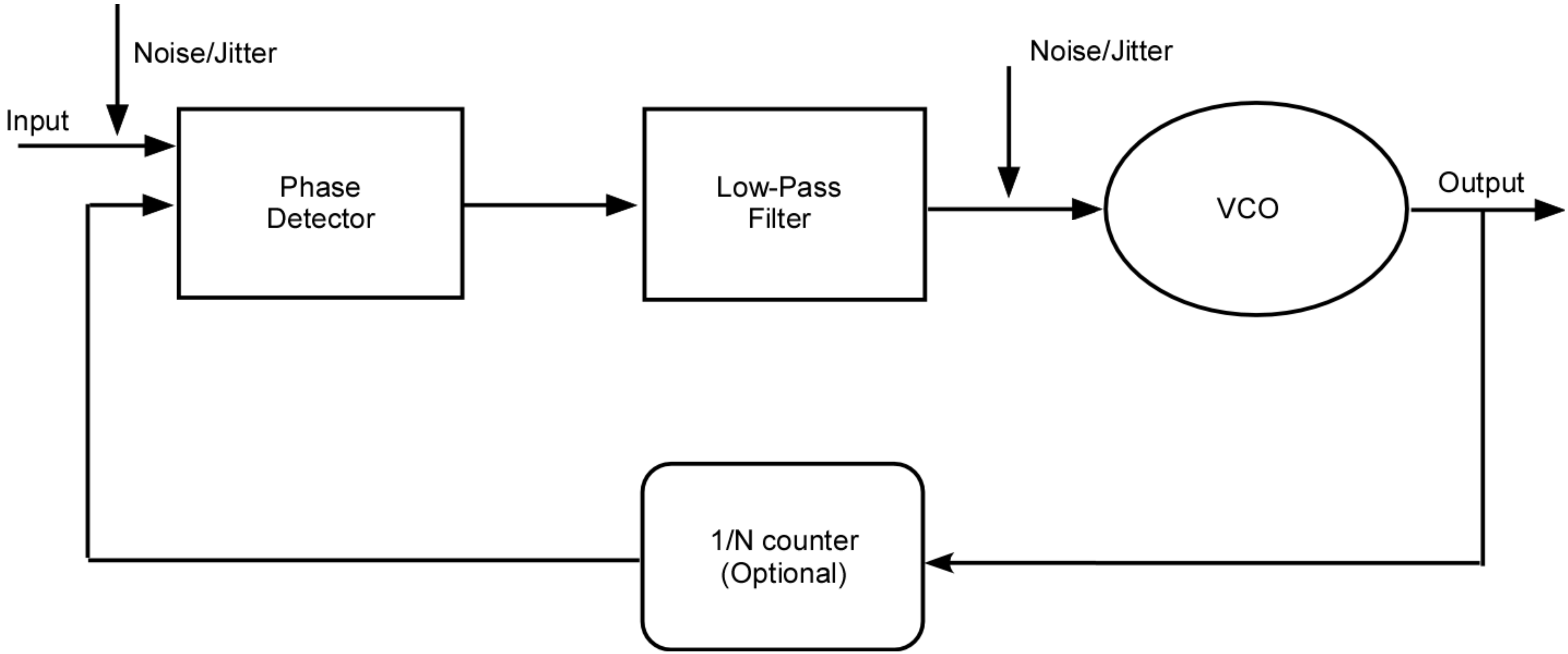
Phase Noise to Phase Jitter

Need: convert phase noise measured in the frequency domain to phase jitter for PLLs, clocks and oscillators



From the phase noise PSD, random jitter and deterministic jitter can be identified

Phase Lock Loop



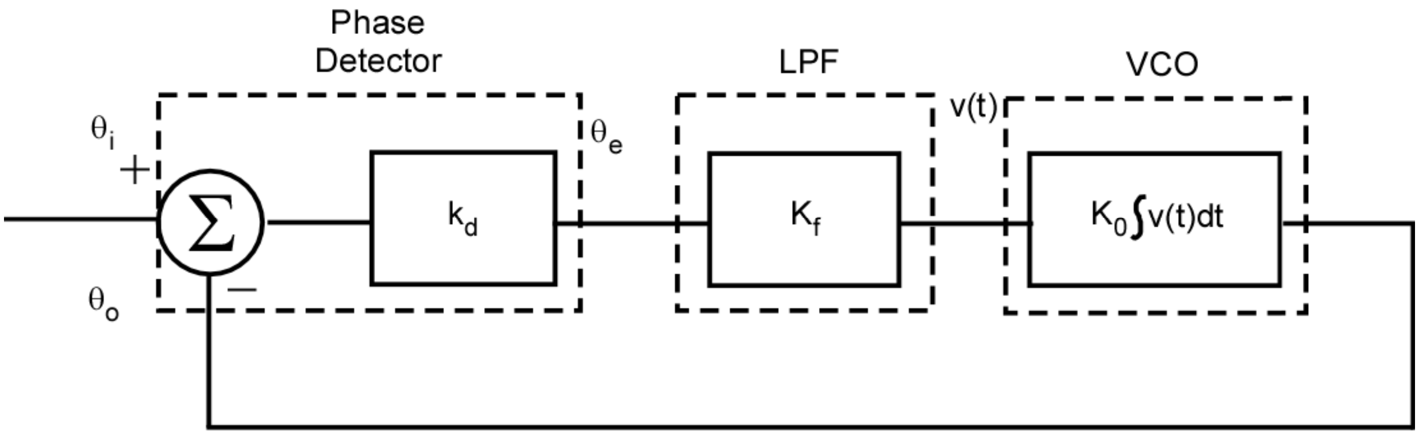
Phase noise or jitter is the key metric for evaluating the performance of a PLL system

Jitter in PLLs

- External Source
  - Reference clock input

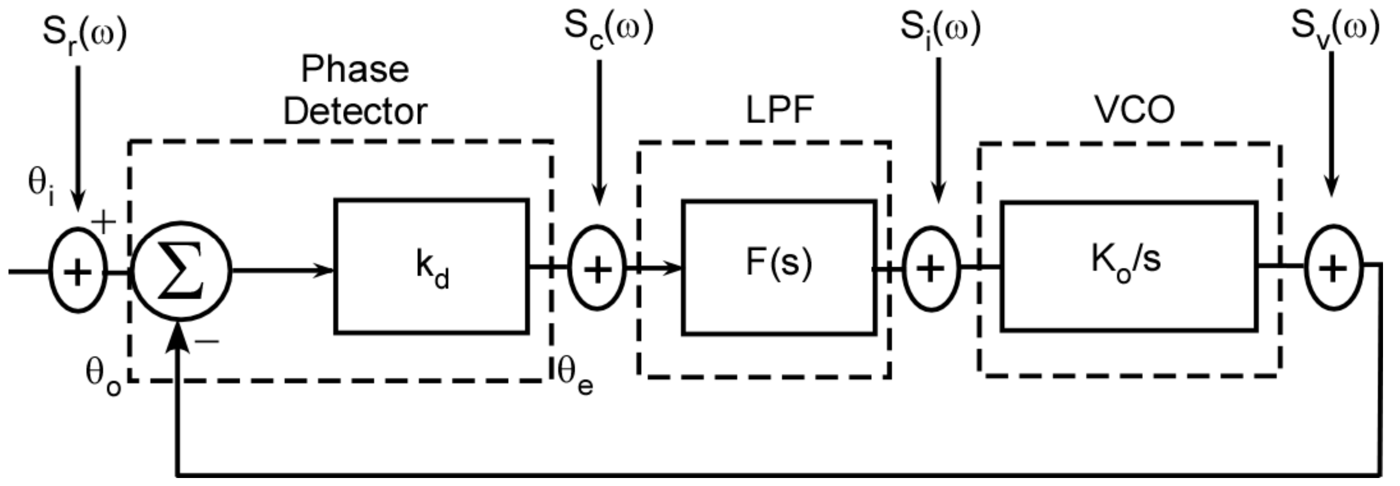
- Internal Source
  - Voltage controlled oscillator (VCO)

Time Domain PLL Analysis



- When PLL is a first-order system, it can be modeled by a closed-form solution
- It is not straightforward to model jitter/noise process with loop components in the time domain

Frequency- Domain PLL Analysis

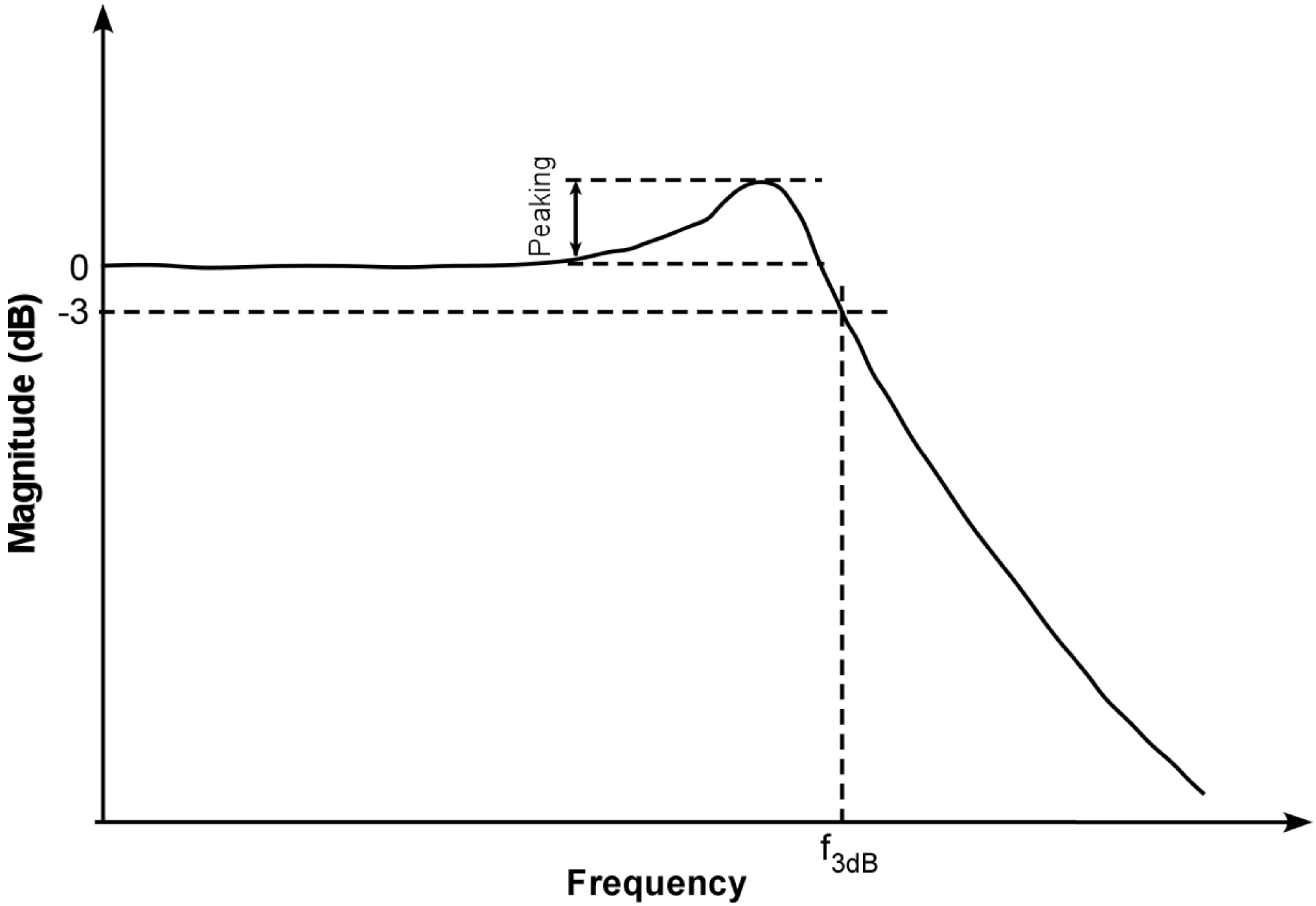


$$H_o(s) = \frac{\theta_o(s)}{\theta_i(s)} = \frac{K_d K_o F(s)}{s + K_d K_o F(s)}$$

The error transfer function is:

$$H_e(s) = \frac{\theta_e(s)}{\theta_i(s)} = 1 - H_o(s)$$

PLL Transfer Function



PLL Frequency Response

- Large peaking causes PLL to be unstable
- Larger 3dB frequency ➔ faster PLL tracking
- Larger peaking ➔ jitter amplification➔ bit error

For PLL stability, Barkhausen condition must be satisfied

$$\left| K_d K_o \frac{F(s)}{s} \right| = 1$$

$$Arg\left[K_d K_o \frac{F(s)}{s} \right] = 180^\circ$$

PLL Frequency Response