ECE 621
Signaling and Synchronization
Fall 2020

Topic 7
Clock and Data Recovery (CDR) Circuits

Sameh A. Ibrahim

Ain Shams University

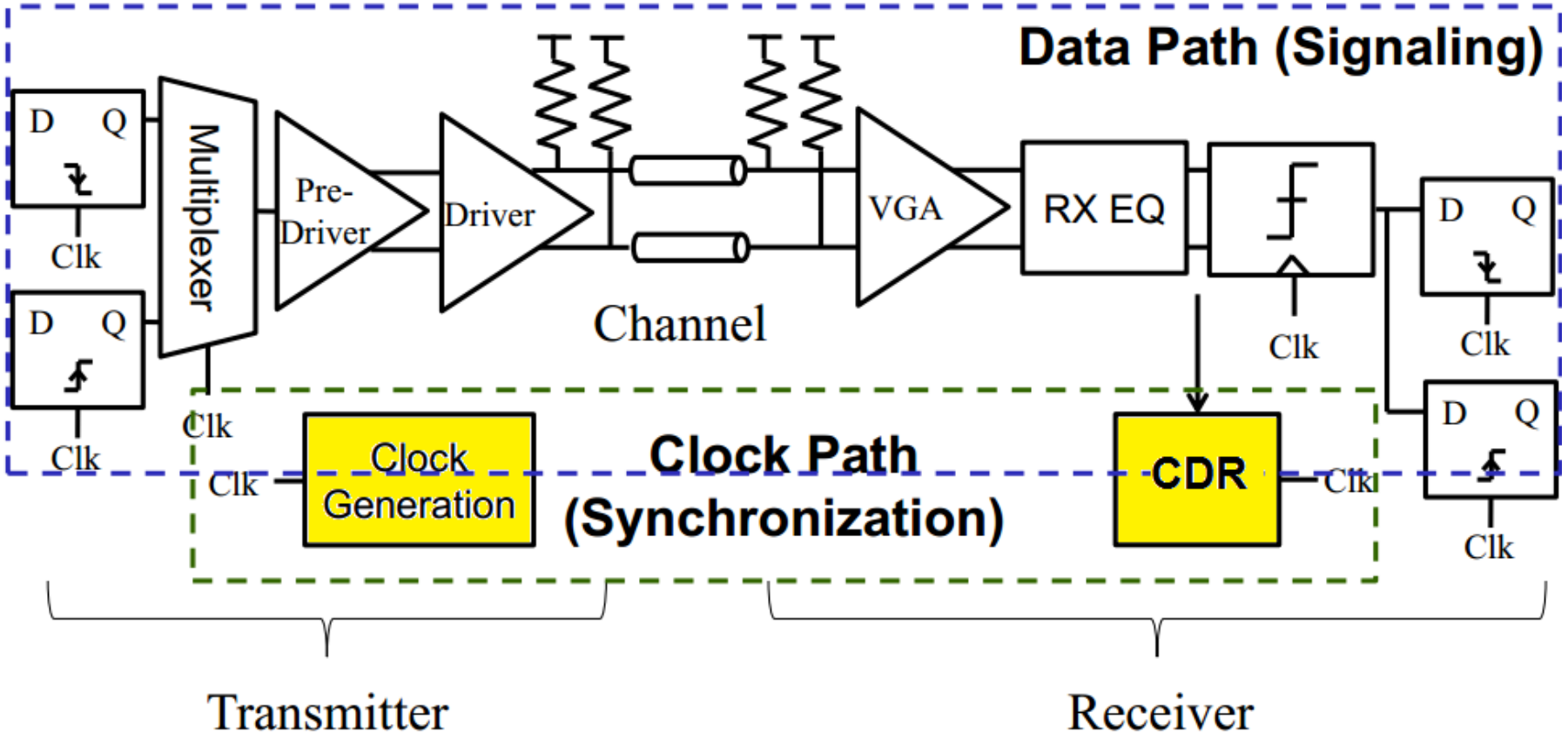
ICL

(Courtesy of B. Casper – Intel, K. Yang – UCLA, A. Elshazly – Marvell and S. Palermo – TAMU)

Outline

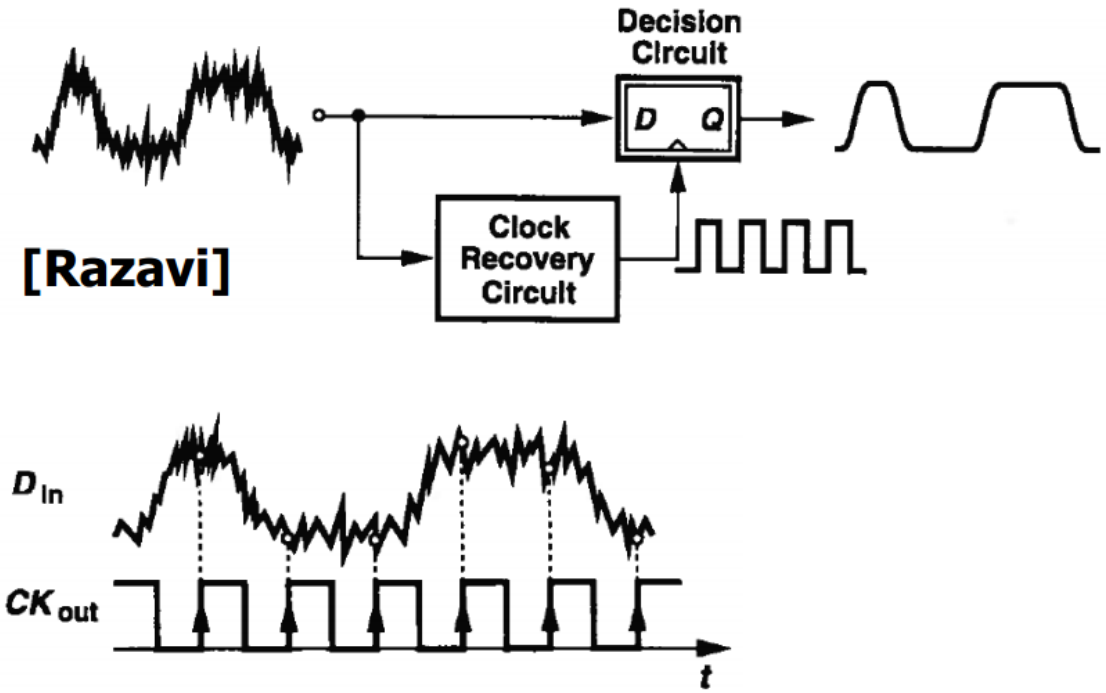
- CDR overview
- CDR Performance Metrics
- CDR Components (phase detectors)
- CDR Topologies
- CDR jitter properties calculations

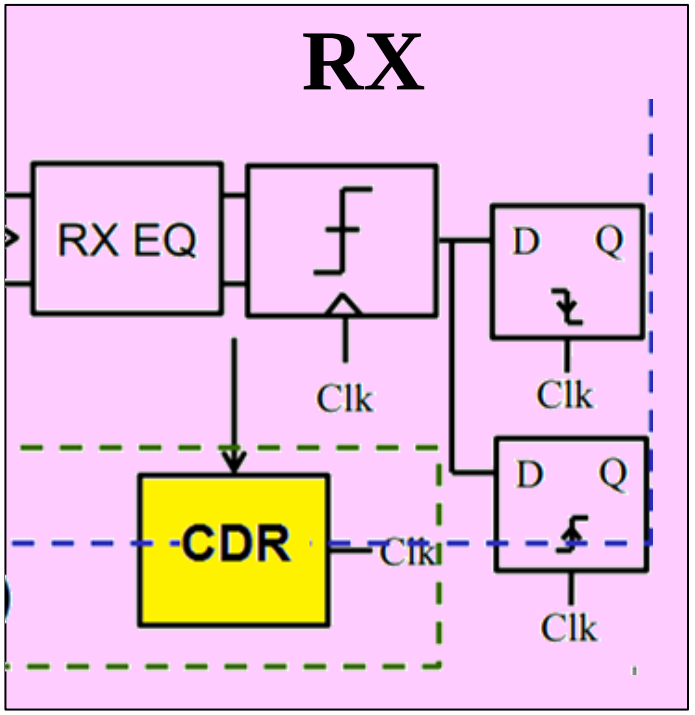
Serial-Link System Block Diagram



- TX and RX clocks needs synchronization ➔ CDRs
  - Frequency synchronization (depend on clock path)
  - Phase synchronization (data path)

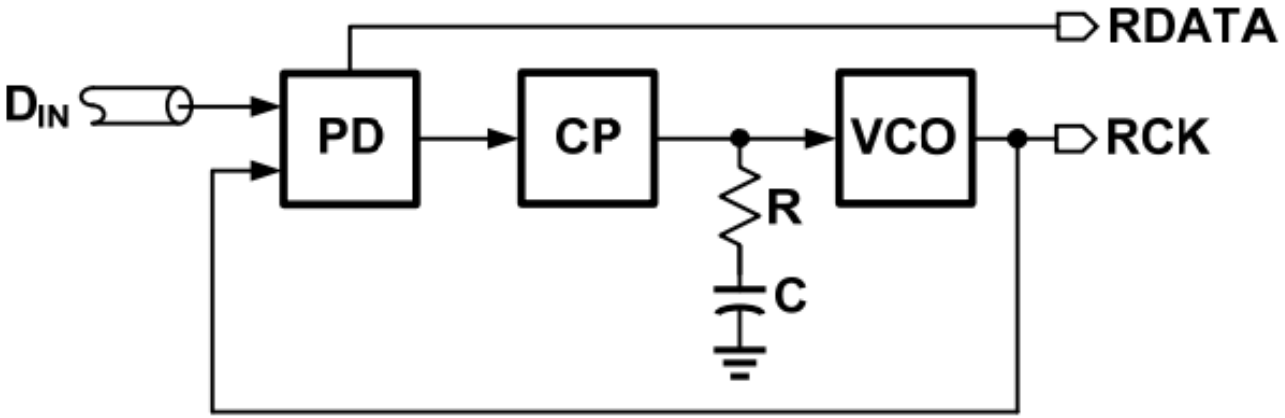
Serial-Link System Block Diagram





- A clock and data recovery system (CDR) produces the clocks to sample incoming data at optimum phase position and correct clock frequency
- Recovered Clock (CK_{OUT}) extracted from incoming data

CDR Block Diagram



- Extracts embedded clock and retimes input data
- Phase detection continuously running.

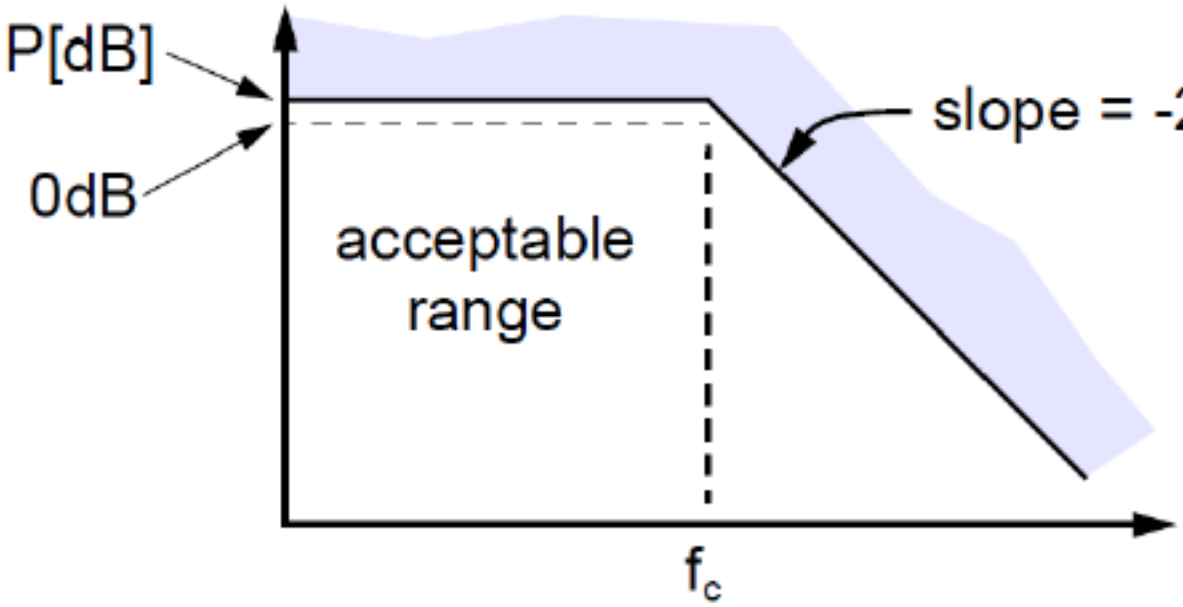
- Possible CDR implementations
  - Stand-alone PLL
  - “Dual-loop” architecture - with PLL/DLL and phase interpolators (PI)
  - Phase-rotator PLL

CDR Performance Metrics

- **Jitter transfer**
- **Jitter tolerance**
- **Jitter generation**

1- Jitter Transfer Function (JTF)

Ratio of magnitudes of output sinusoidal jitter to input sinusoidal jitter of a CDR

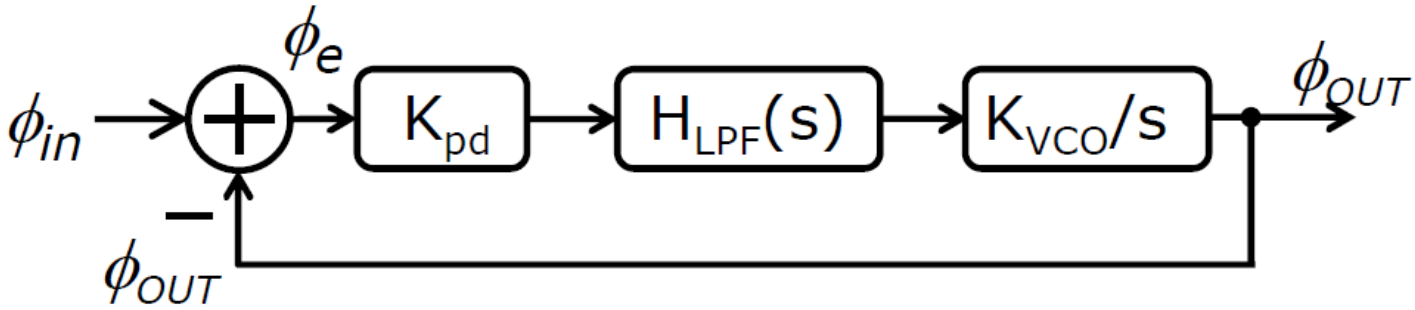


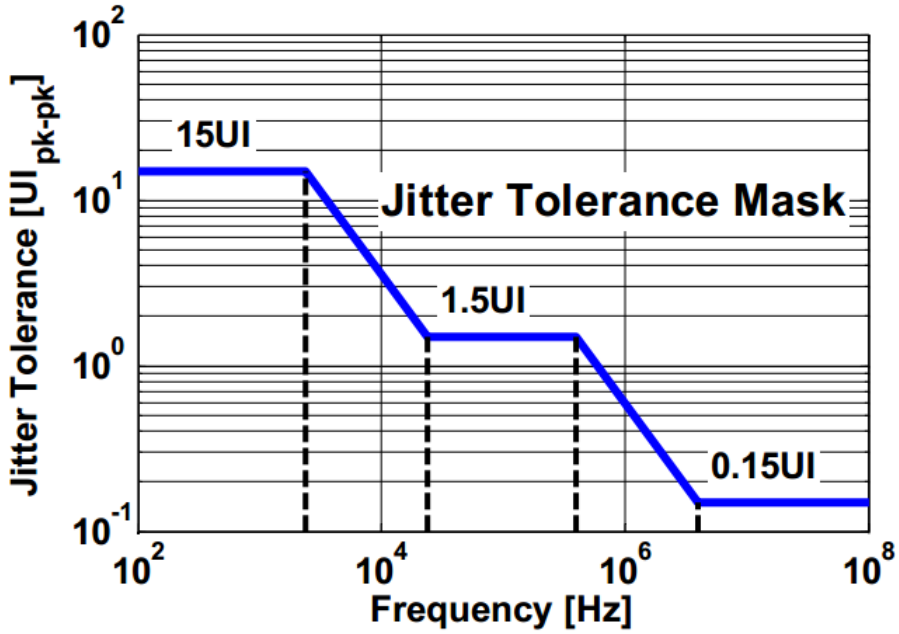
Data Rate	f_c[kHz]	P[dB]
155 Mb	130	0.1
622 Mb	500	0.1
2.488 Gb	2000	0.1

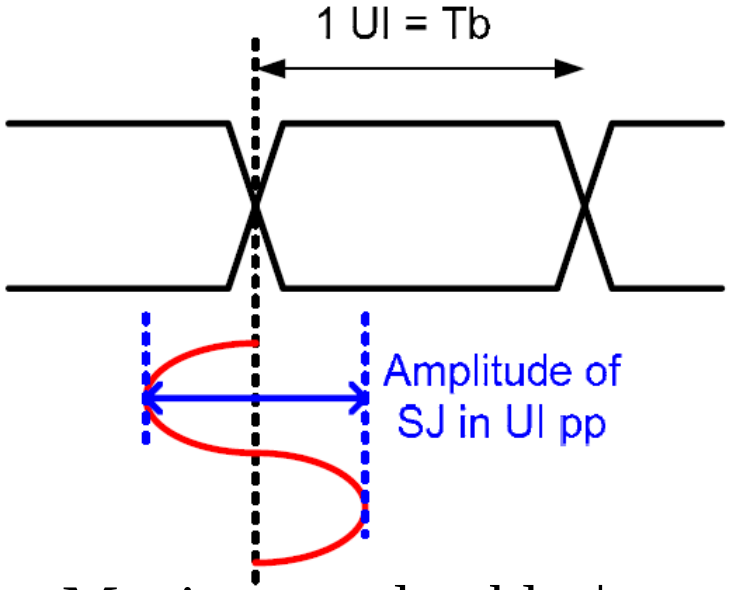
- **This specification is intended to control jitter peaking in long repeater chain**
- **For example, the SONET specification limits maximum transfer peaking of the PLL to less than 0.1 dB**

2- Jitter Tolerance (JTOL)

How much sinusoidal jitter can the CDR "tolerate" and still achieve a given BER



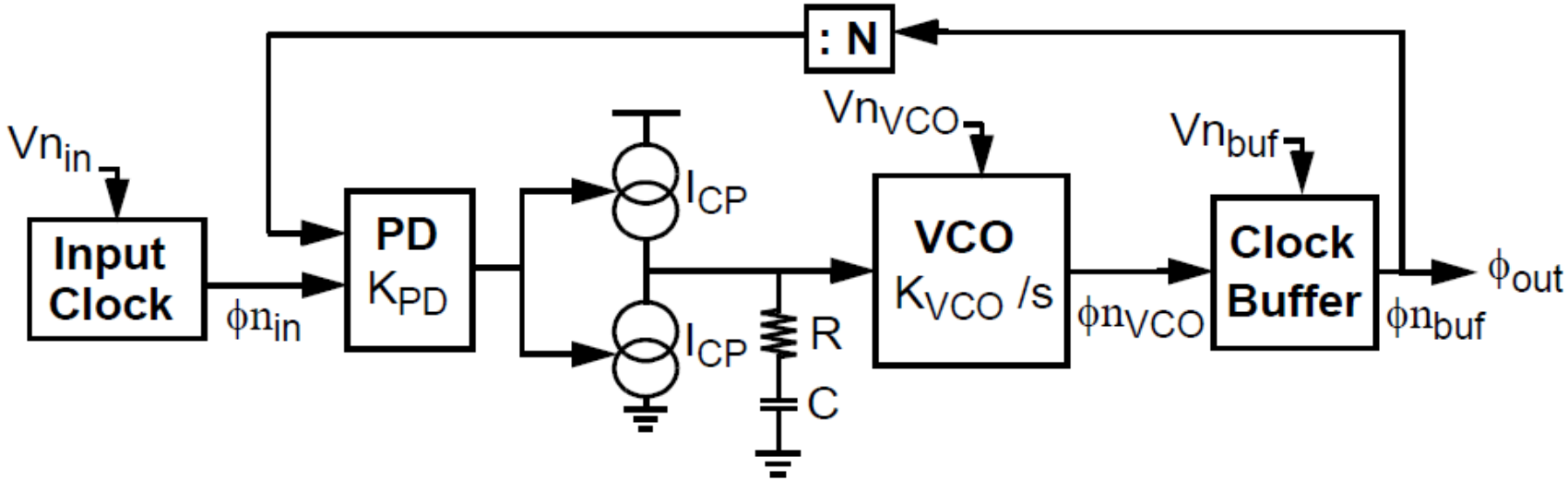




Maximum tolerable ϕ_e

3- Jitter Generation

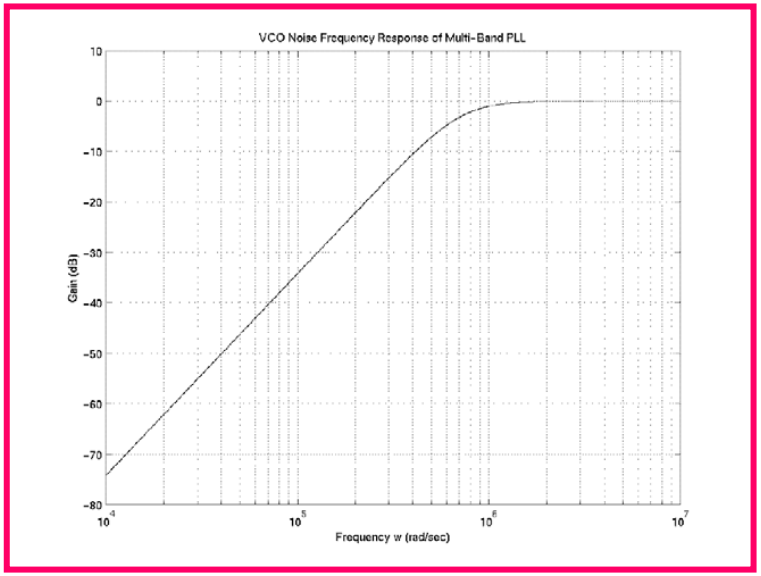
jitter produced by the CDR measured at its output

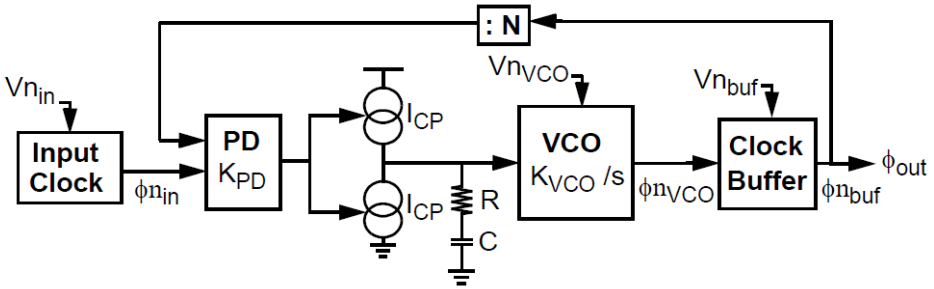


- **Jitter generation is how much jitter the CDR "generates".**
- **Assumes jitter-free serial data input.**

3- Jitter Generation (Cont.)

jitter produced by the CDR measured at its output





- **Jitter generation is how much jitter the CDR "generates".**
- **Assumes jitter-free serial data input.**
- **VCO phase noise**

  N = 1 for CDR

$$H_{n_{VCO}}(s) = \frac{\phi_{out}}{\phi_{n_{VCO}}} = \frac{s^2}{s^2 + \left(\frac{K_{Loop}}{N}\right)RCs + \frac{K_{Loop}}{N}} = \frac{s^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

- **SONET specification → rms output jitter ≤ 0.01 UI**

Outline

- CDR overview
- CDR Performance Metrics
- **CDR Components (phase detectors)**
- CDR Topologies
- CDR jitter properties calculations

CDR Phase Detectors (PDs)

- **CDR phase detectors**
  - Compares phase between input data & recovered clock
  - Provides information to adjust sampling clock phase
- **CDR Phase detectors**
  1. **Non-linear PD:** provide only sign information regarding the sampling phase error (Examples: Alexander or Bang-Bang)
  2. **Linear PD:** Provide both sign and magnitude information regarding the sampling phase error (example Hogge PD)
  3. **Oversampling (>2)**
  4. **Baud-Rate**

1- Non-Linear PD (Alexander PD)

- **Most commonly used CDR phase detector**
- **Non-linear (Binary) "Bang-Bang" PD**
  - Only provides sign information of phase error (not magnitude), 2x-Oversampled.
- **Phase detector uses 2 data samples and one "edge" sample**
- **Data transition necessary**

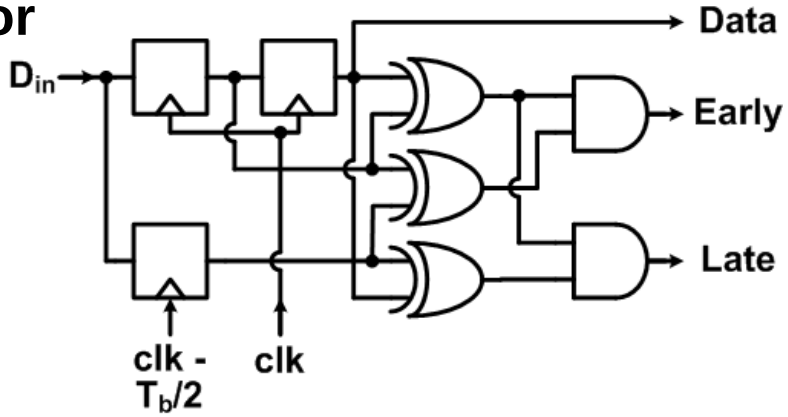
$$D_n \oplus D_{n+1}$$

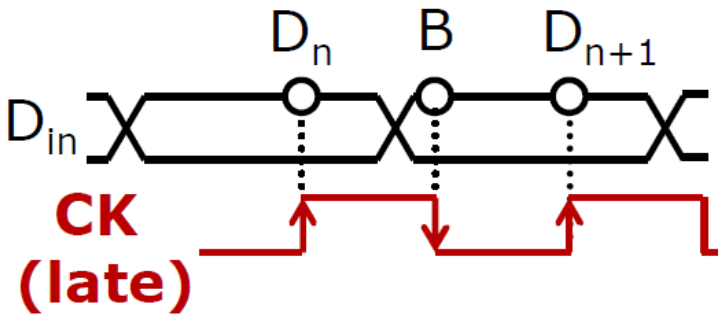
  - If "edge" sample is same as second bit (or different from first), then the clock is sampling "late"

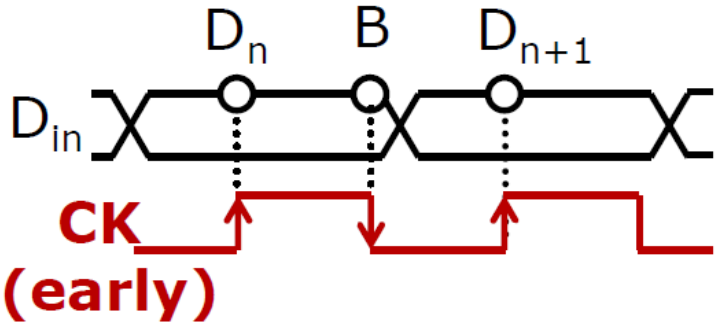
$$E_n \oplus D_n$$

  - If "edge" sample is same as first bit (or different from second), then the clock is sampling "early"

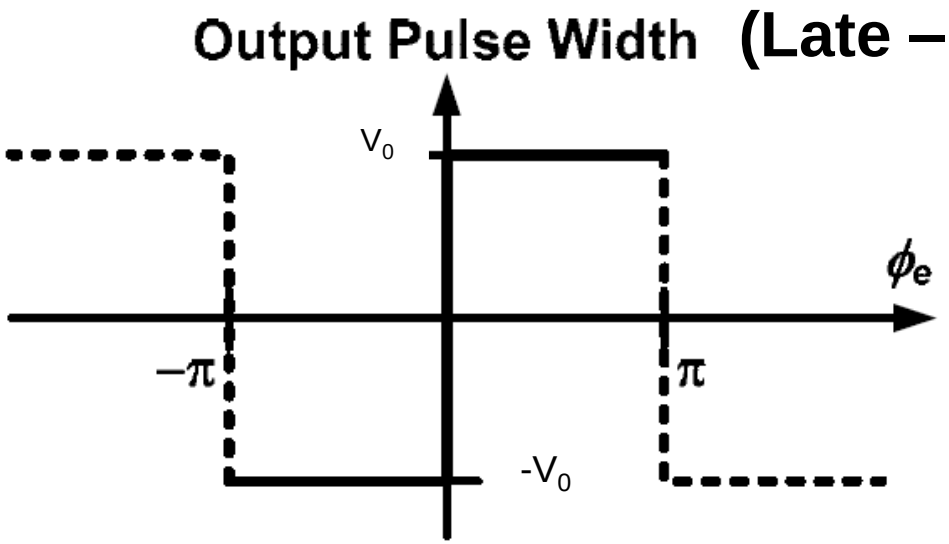
$$E_n \oplus D_{n+1}$$





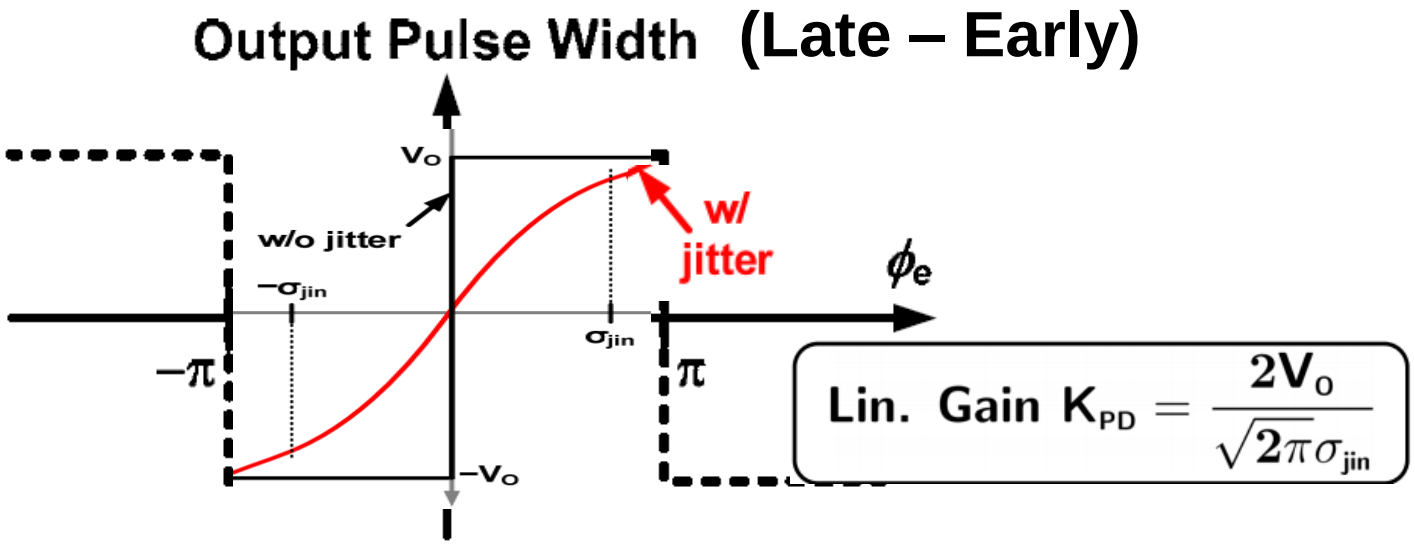


Alexander PD Characteristic (Ideal)



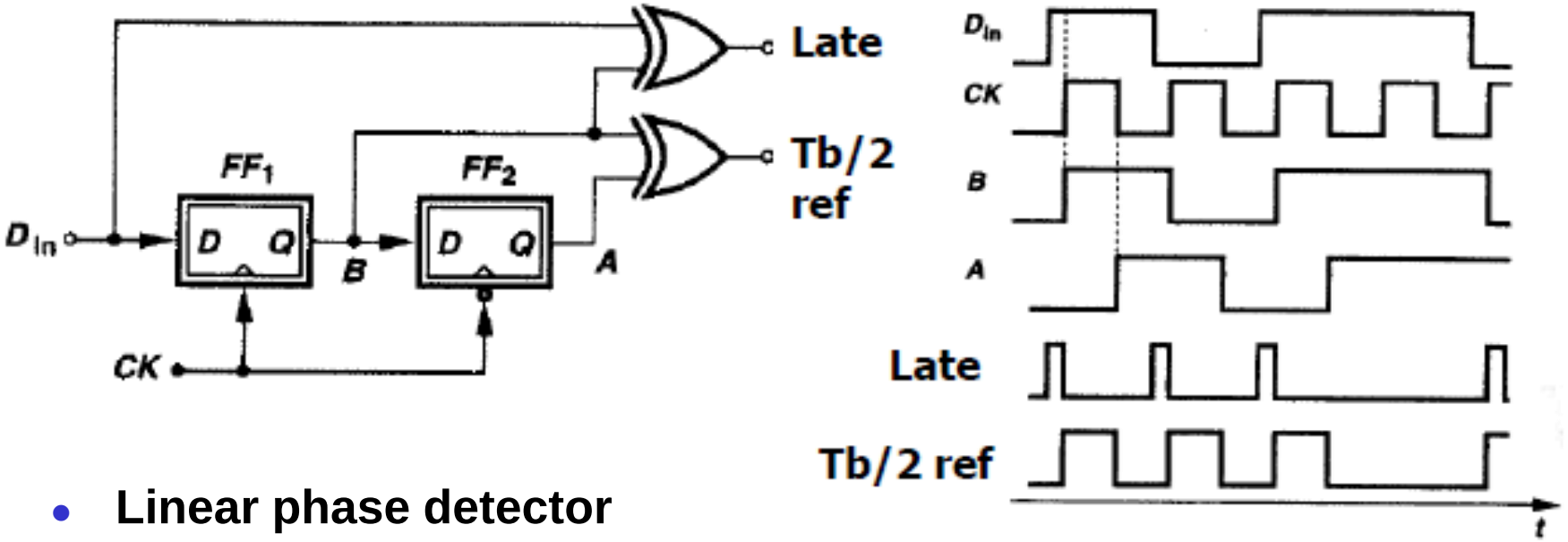
- **Phase detector only outputs phase error sign information in the form of a late OR early pulse whose width doesn't vary,**
- **Phase detector gain is ideally infinite at zero phase error.**

Alexander PD Characteristic (with Noise)



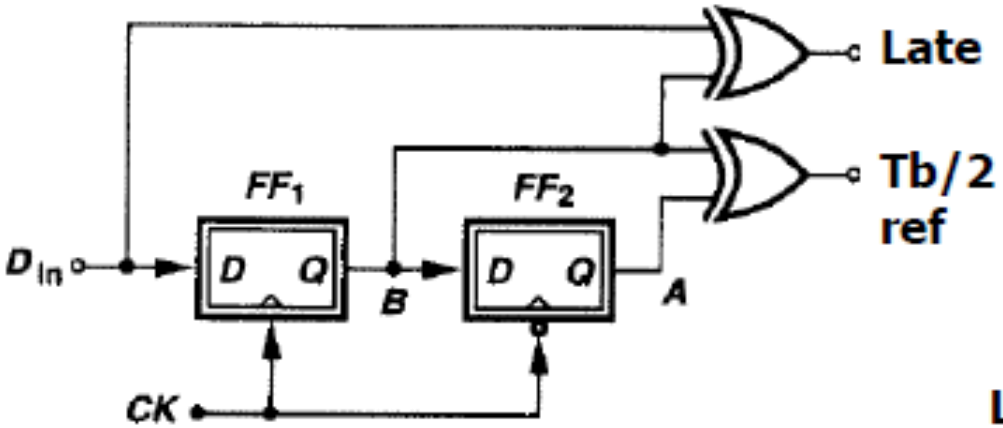
- Phase detector only outputs phase error sign information in the form of a late OR early pulse whose width doesn't vary,
- Phase detector gain is ideally infinite at zero phase error.
- Finite gain will be present with noise, clock jitter, sampler metastability, ISI.
- Unpredictable loop dynamics

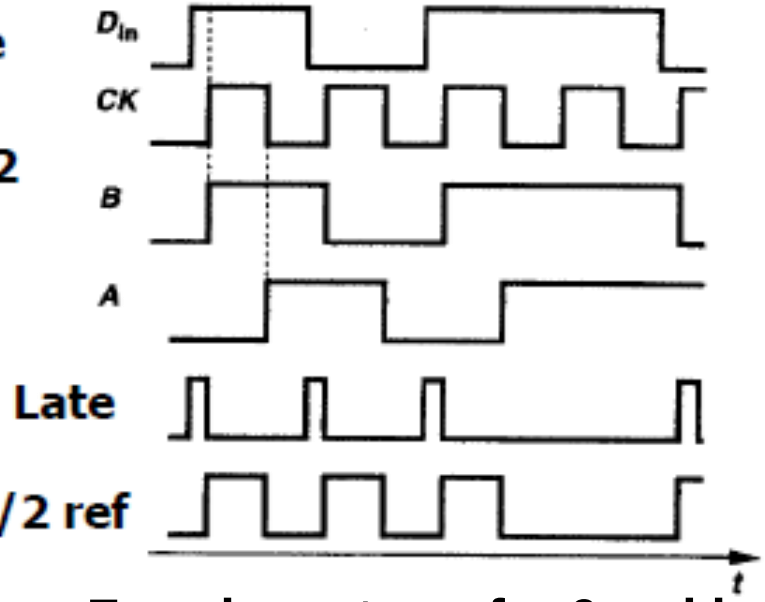
2- Linear PD: Hogge PD (1)

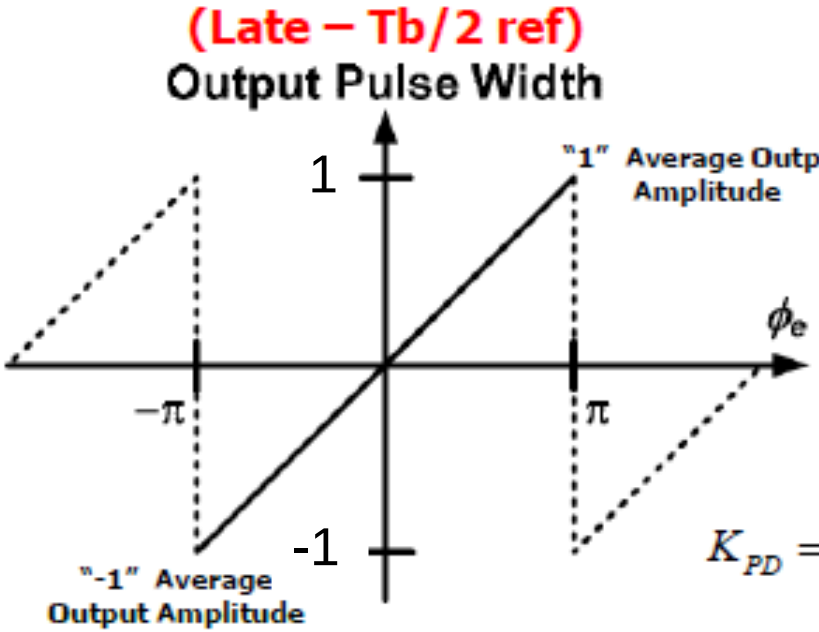


- **Linear phase detector**
- **With a data transition and assuming a full-rate clock**
  - The late signal produces a signal whose pulse width is proportional to the phase difference between the incoming data and the sampling clock.
  - A Tb/2 reference signal is produced with a Tb/2 delay.
- **If the clock is sampling early, the late signal will be shorter than Tb/2 and vice-versa.**

2- Linear PD: Hogge PD (2)



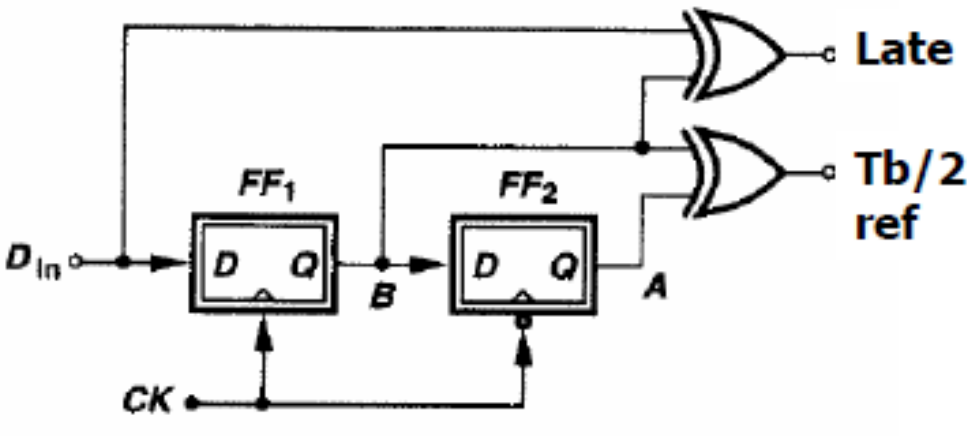


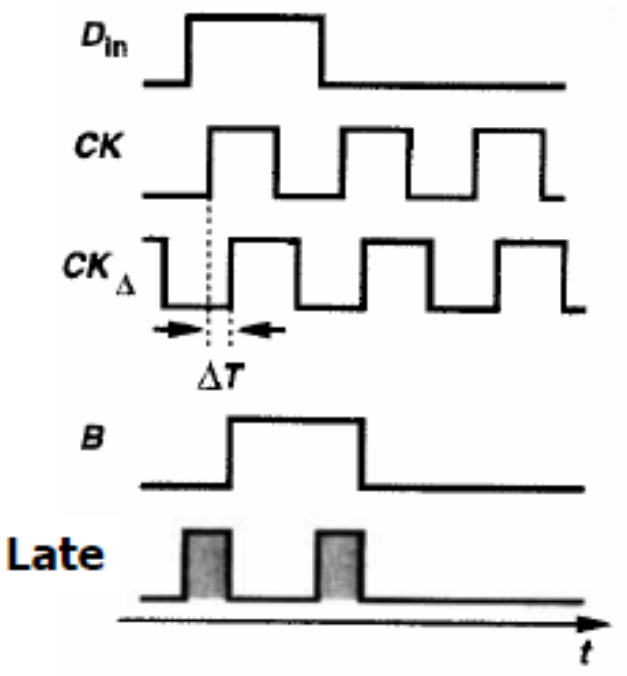


$$K_{PD} = \frac{1}{\pi}(TD)$$

- **For phase transfer 0 rad is w.r.t optimal Tb/2 (π) spacing between sampling clock and data**
  - $\phi_e = \phi_{in} - \phi_{clk} - \pi$
- **TD is the transition density – no transitions, no information**
  - A value of 0.5 can be assumed for random data.

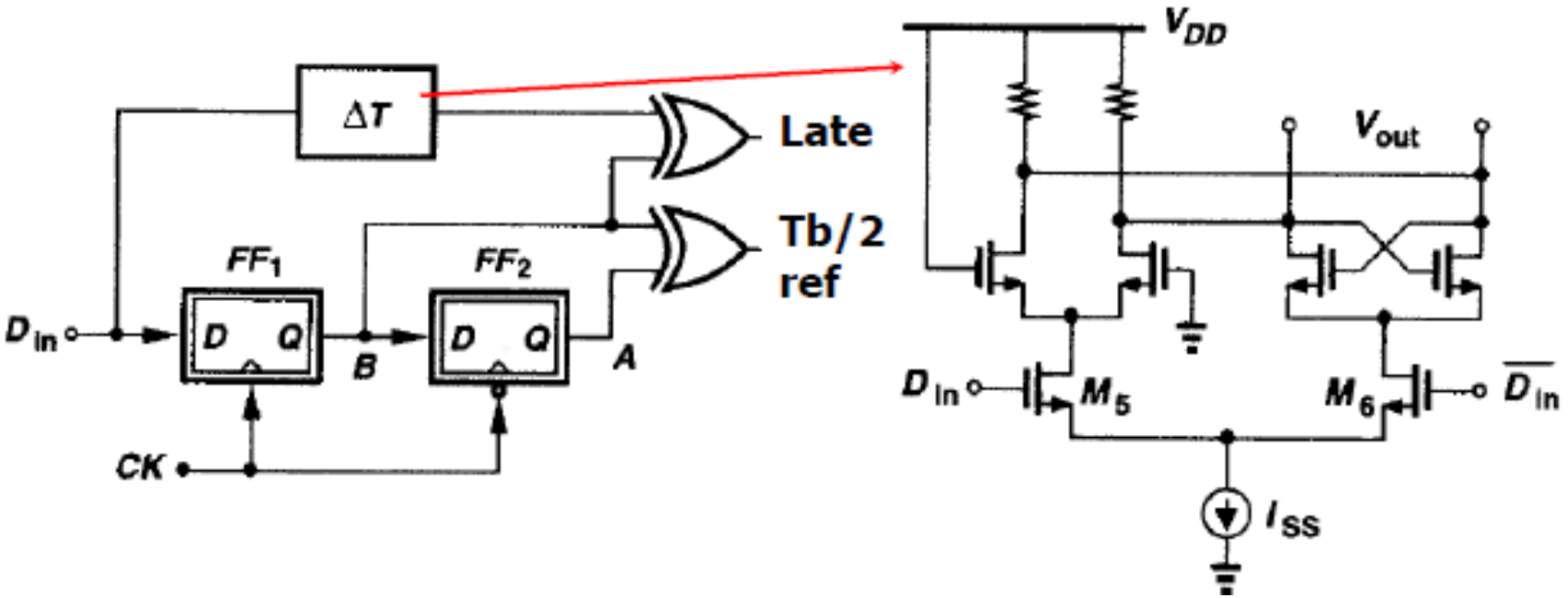
Hogge PD Nonidealities (1)





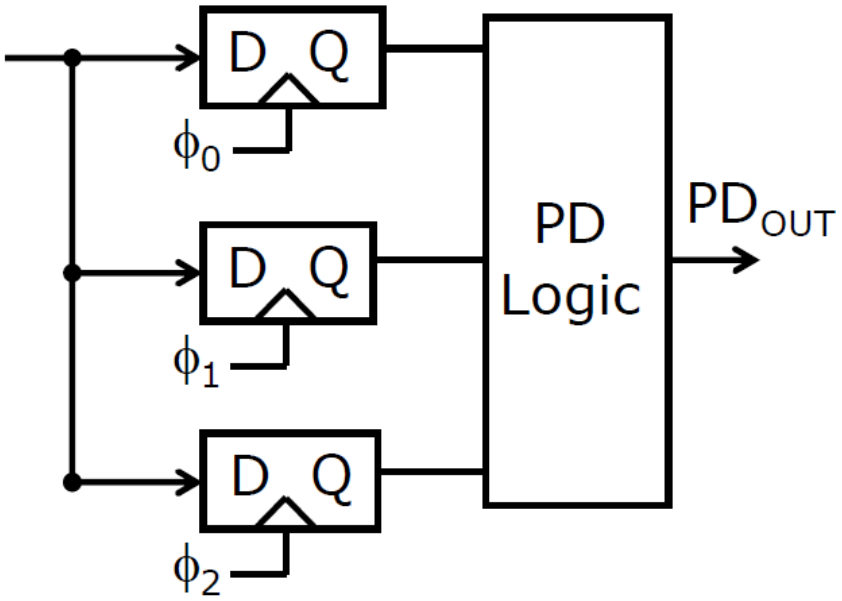
- **Flip-Flop Clk-to-Q delay widens Late pulse, but doesn't impact Tb/2 reference pulse.**
- **CDR will lock with a phase shift to equalize Tb/2 reference and Late pulse widths.**

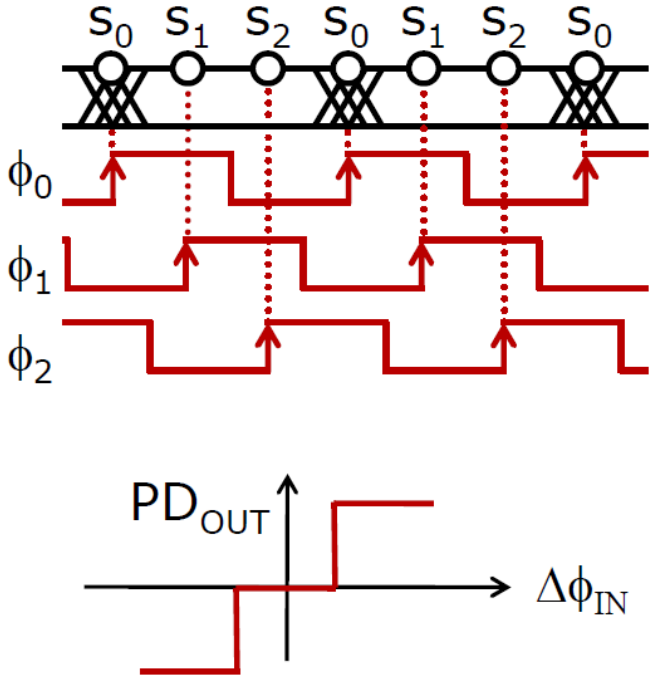
Hogge PD Nonidealities (2)



- **CDR phase shift compensated with a dummy delay element.**
- **Other issues**
  - Need extremely high-speed XOR gates
  - Phase skew between Tb/2 reference and Late signals induces a “triwave” disturbance (ripple) on the control voltage.

3- Oversampling Phase Detectors

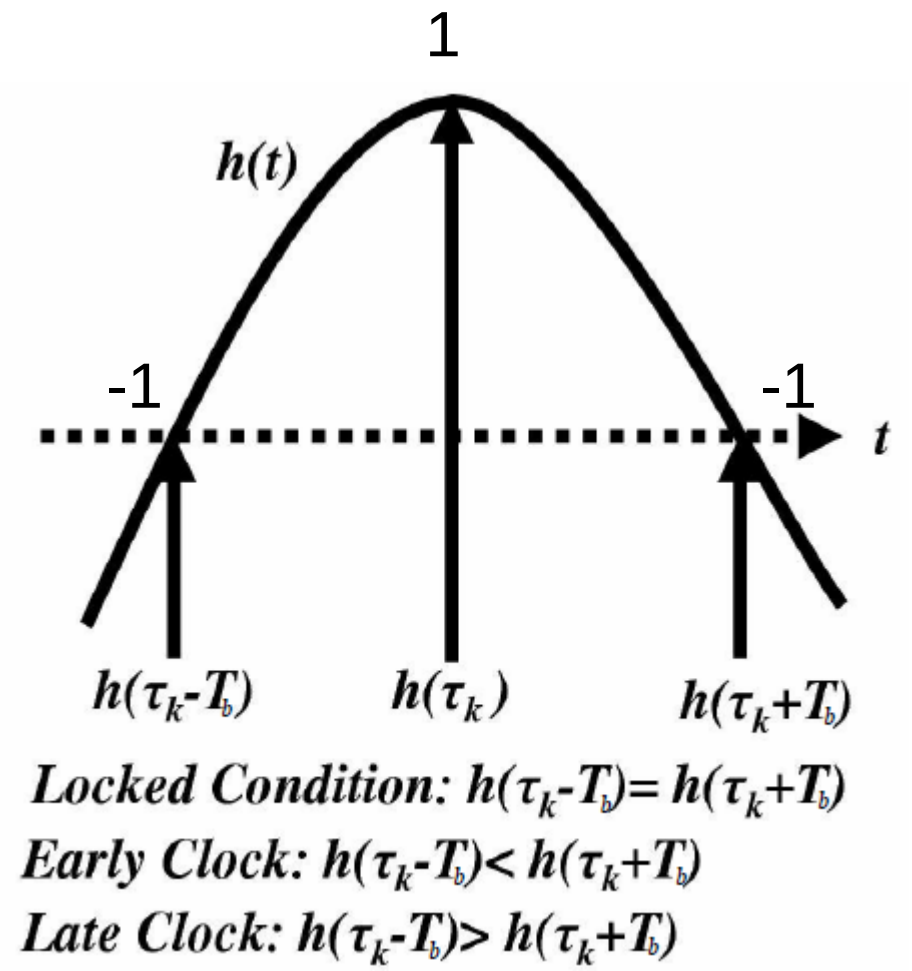




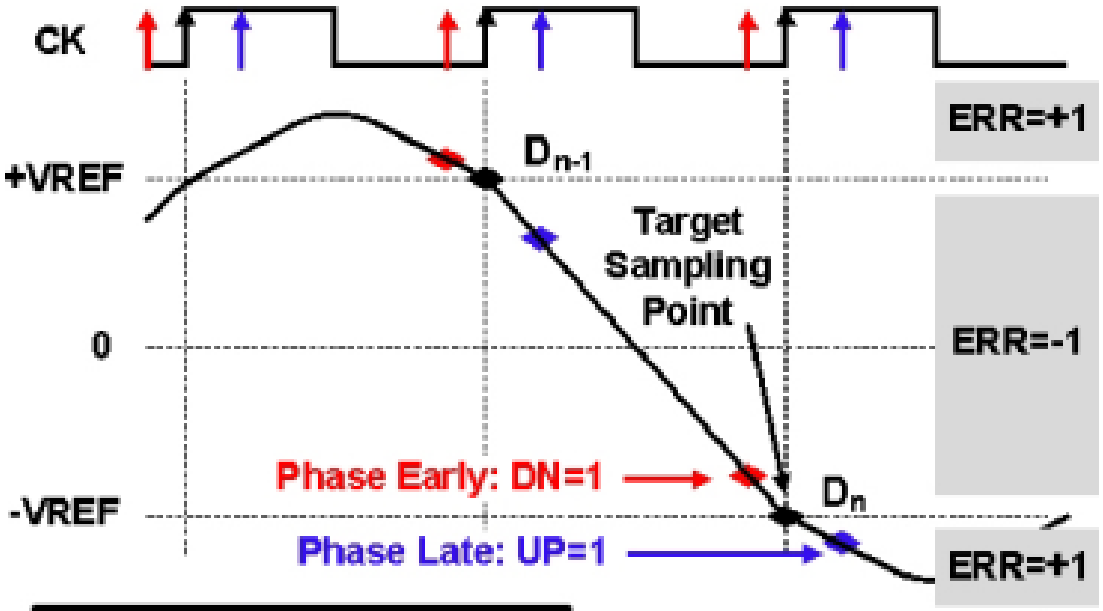
- **Multiple clock phases are used to sample incoming data bits.**
- **PD can have multiple output levels.**
- **Detects rate of phase change for frequency acquisition.**

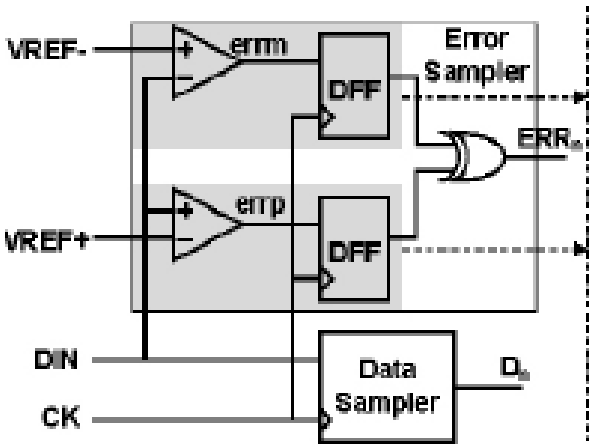
4- Baud-Rate Mueller-Muller PD (1)

- **Baud-rate phase detector only requires one sample clock per symbol (bit).**
- **Mueller-Muller phase detector commonly used.**
- **Attempting to equalize the amplitude of samples taken before and after a pulse**



4- Baud-Rate Mueller-Muller PD (2)



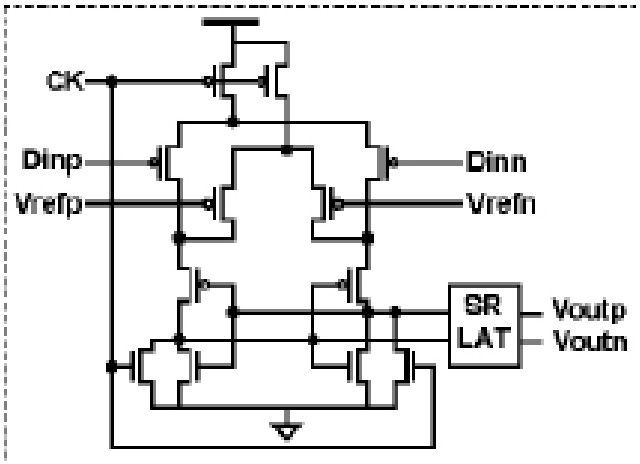


Phase Error:
$\Delta T_n = D_n \times D_{n-1} \times (ERR_n - ERR_{n-1})$

Phase detector output truth table →

ΔT_n	UP	DN
+1	1	0
0	0	0
-1	0	1

d_j	d_{j-1}	e_j	e_{j-1}	φ_{err_j}
1	-1	1	-1	LATE
-1	1	1	-1	LATE
1	-1	-1	1	EARLY
-1	1	-1	1	EARLY
All other cases				HOLD

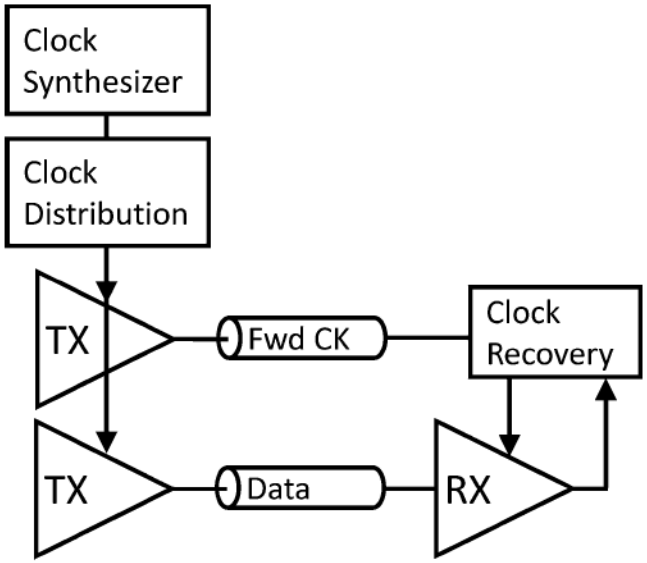


[Spagna, ISSCC 2010]

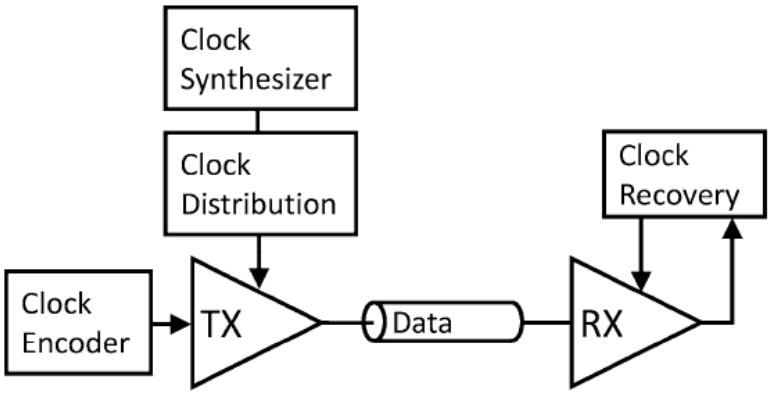
Outline

- CDR overview
- CDR Performance Metrics
- CDR Components (phase detectors)
- **CDR Topologies**
- CDR jitter properties calculations

CDR in Forwarded vs. Embedded Clock



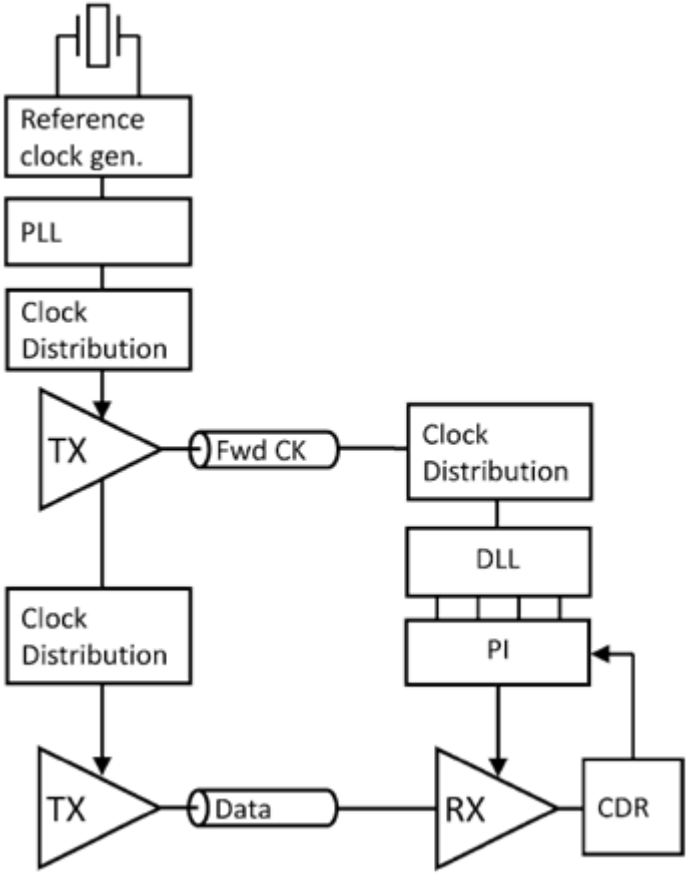
- Preferred in high aggregate bandwidth
- FC recovery unit attempts to center the receiver sample at the optimum point as measured by operating margin BER.
- One time, periodic or continuous based on matching. Coding might be necessary



- Preferred in low aggregate bandwidth
- Can help implement modular designs if PLL not shared.
- Requires timing information to be encoded in the data to ensure a high probability of data edge transitions

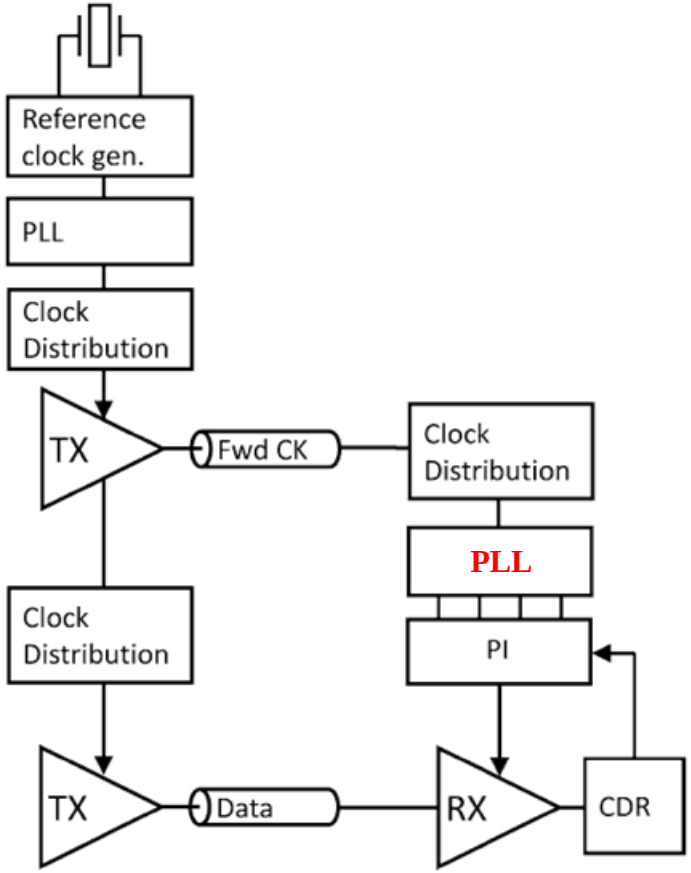
DLL-Based Forwarded Clocking

- DLL acts as a multiphase clock generator.
- The DLL and phase interpolator (PI) combination produces an adjustable phase clock generator with the same frequency as the incoming FC.
- DLLs are more simple, robust and stable than other types of multiphase clock generators.
- DLLs don't exhibit accumulation of internal phase errors.
- DLLs facilitate high bandwidth tracking of the FC.
- FC jitter is amplified (Type I DLL)

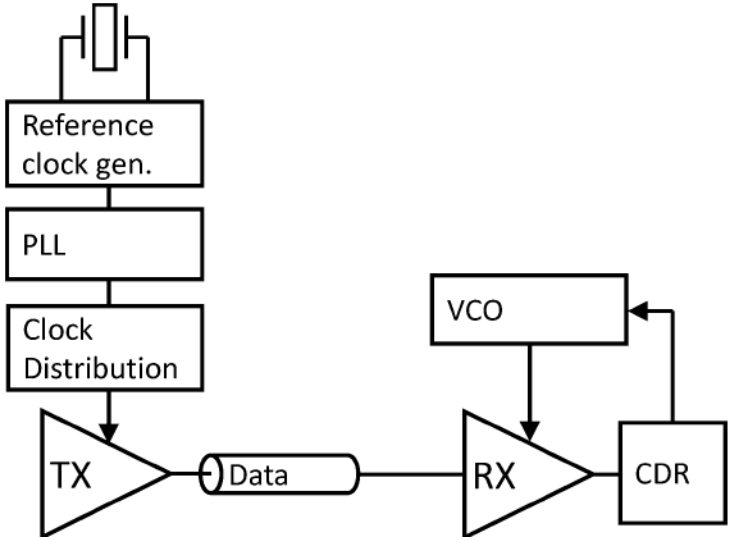


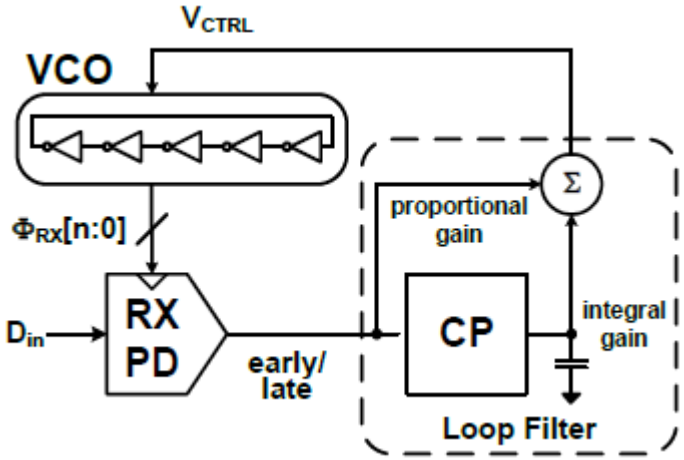
PLL-Based Forwarded Clocking

- Used for applications that are sensitive to FC jitter amplification.
- Low-pass phase transfer characteristic results in suboptimum performance and lower clock recovery bandwidth.
- PLLs also have other disadvantages such as susceptibility to jitter accumulation and stability issues.
- PLLs are usually more area intensive and complex than the equivalent DLL-based recovery solution.



Single-Loop CDR: VCO or PLL-Based EC



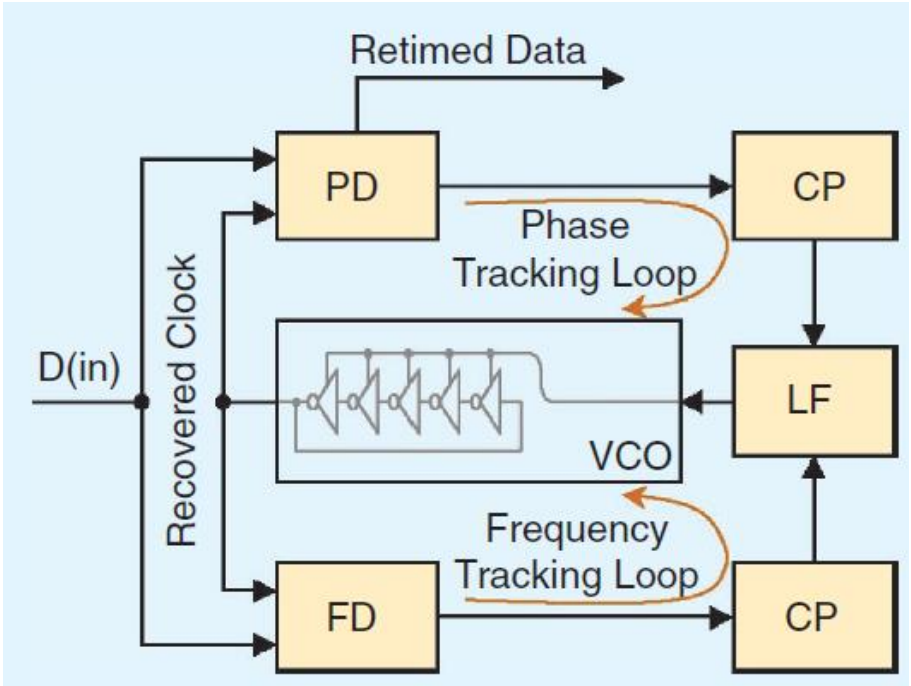


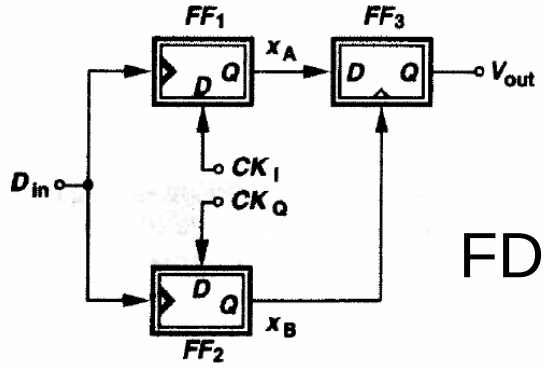
- **Dedicated RX side VCO**
- **CDR acts as the loop filter.**
- **The combination of these components forms a loop that locks to the mean phase of the incoming data.**
- **Direct locking to TX clock domain – No intermediate clock**
- **Modularity possible**

Single-Loop CDR Issues

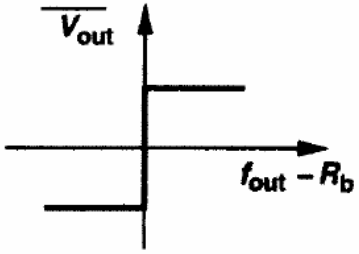
- **Large capacitors → large area for loop filters**
- **VCO-based clocking in the presence of multiple adjacent links may be subject to interference and coupling effects known as injection locking.**
- **Phase detectors have limited frequency acquisition range.**
  - Results in long lock times or not locking at all.
  - Can potentially lock to harmonics of correct clock frequency.
- **VCO frequency range variation with process, voltage, and temperature can exceed PLL lock range if only a phase detector is employed.**

Dual loop CDRs





FD

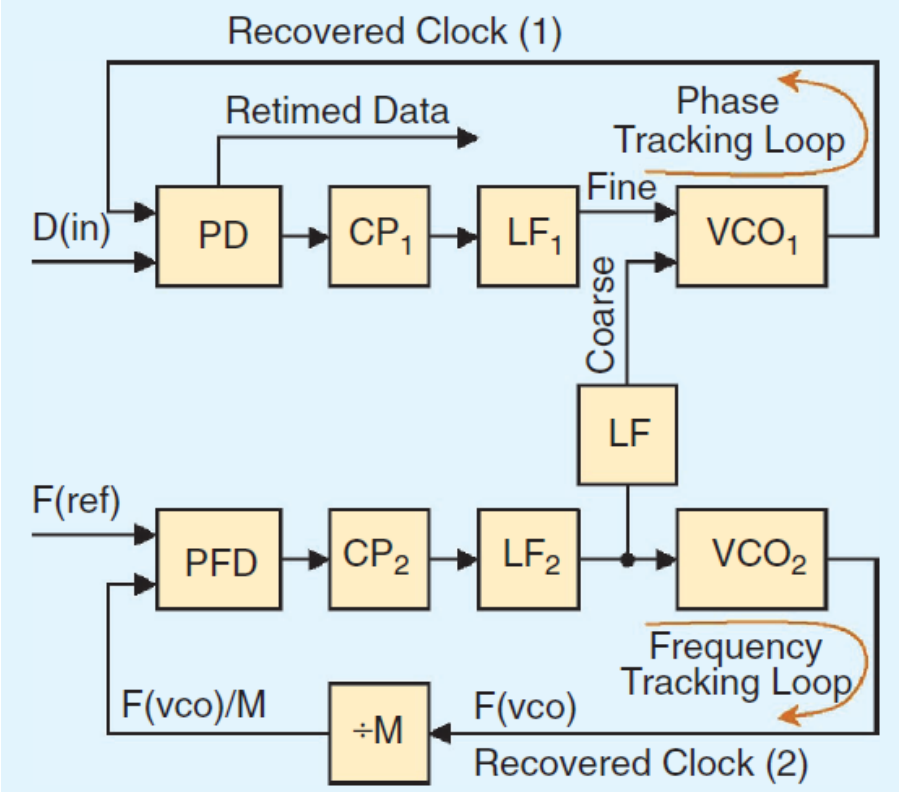


Capture range ~<15% frequency offset

- **Phase and Frequency Tracking Loops**
- **Frequency tracking loop operates during startup or loss of phase lock.**
  - Ideally should be mostly off in normal operation.
- **Frequency loop bandwidth typically much smaller than phase loop bandwidth to prevent loop interaction.**

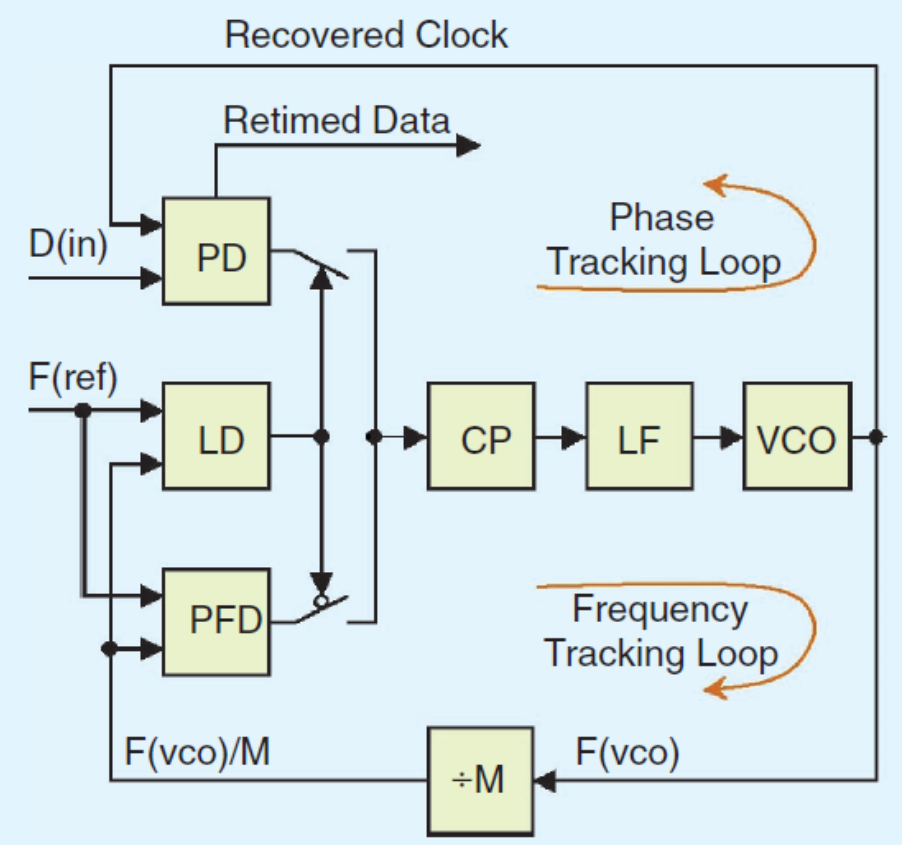
Analog Dual-Loop CDR w/ Two VCOs

- **Frequency synthesis loop with replica VCO provides a "coarse" control voltage to set phase tracking loop frequency.**
- **Frequency loop can be a global PLL shared by multiple channels.**
- **Issues**
  - VCO matching
  - VCO pulling
  - Distributing voltage long distances



Analog Dual-Loop CDR w/ One VCO

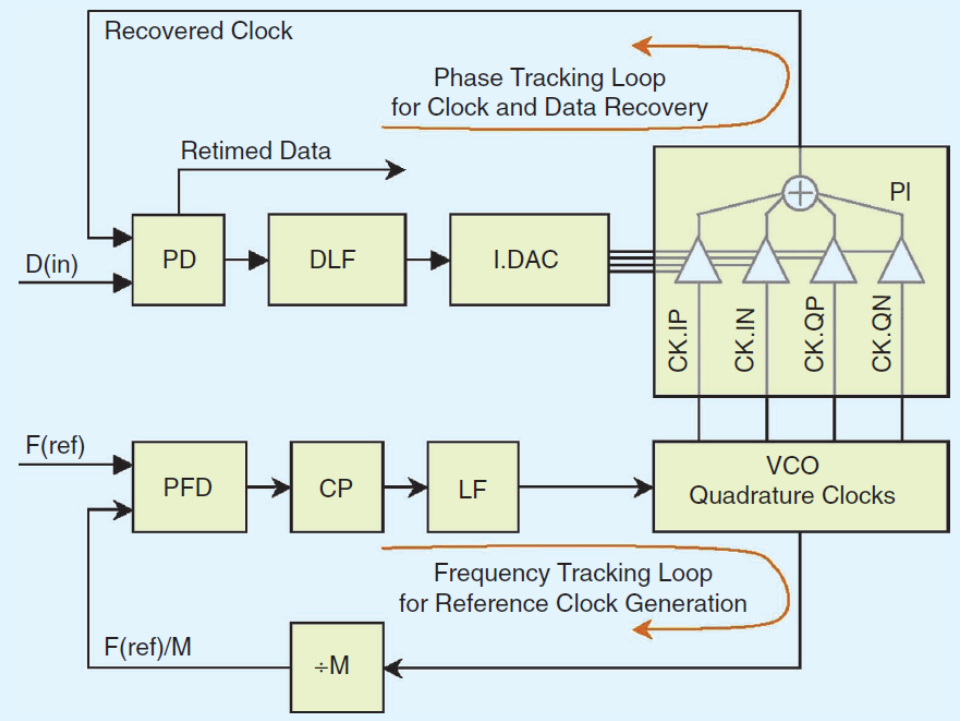
- **Frequency loop operates during startup or loss of phase lock.**
  - Ideally should be mostly off in normal operation.
- **Input reference clock simplifies frequency loop design.**
- **Care must be taken when switching between loops to avoid disturbing VCO control voltage and loose frequency lock.**



LD: Lock Detector

Phase Interpolator (PI) Based CDR (1)

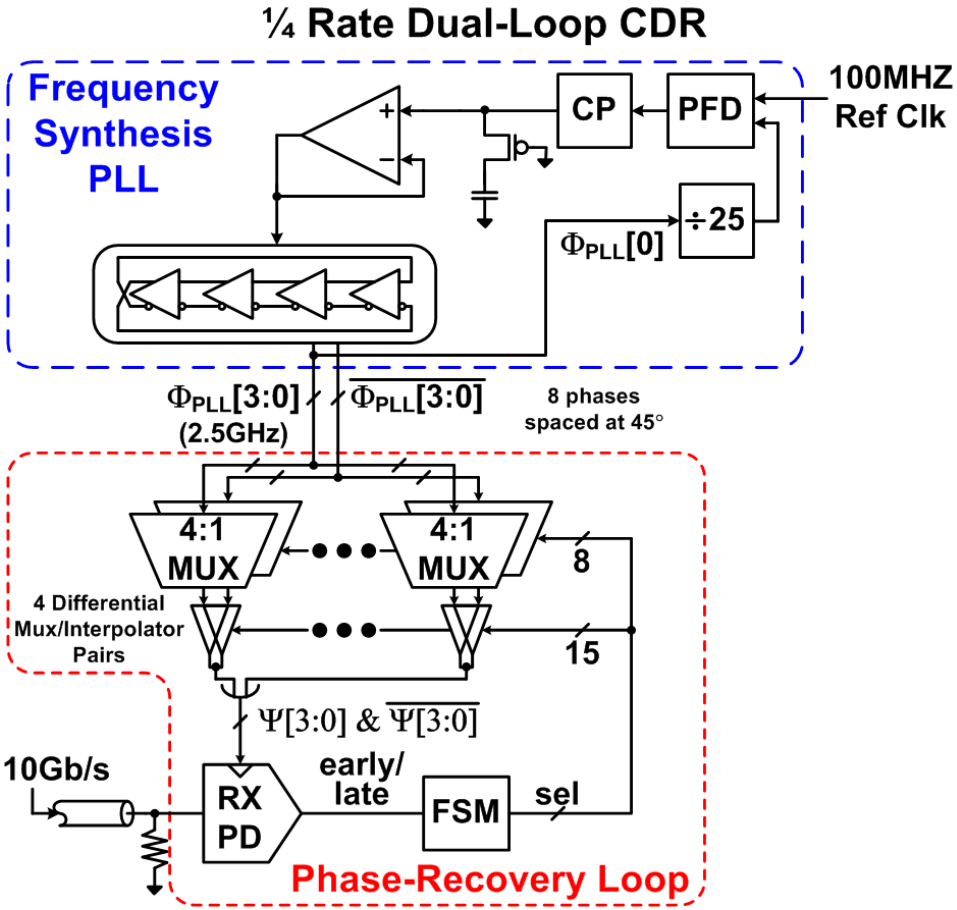
- **Frequency synthesis loop produces multiple clock phases used by the phase interpolators.**
- **Phase interpolator mixes between input phases to produce a fine sampling phase.**
  - Ex: Quadrature 90° PI inputs with 5 bit resolution provides sampling phases spaced by $90°/(2^5)=2.8°$



- **Digital phase tracking loop offers advantages in robustness, area, and flexibility to easily reprogram loop parameters.**

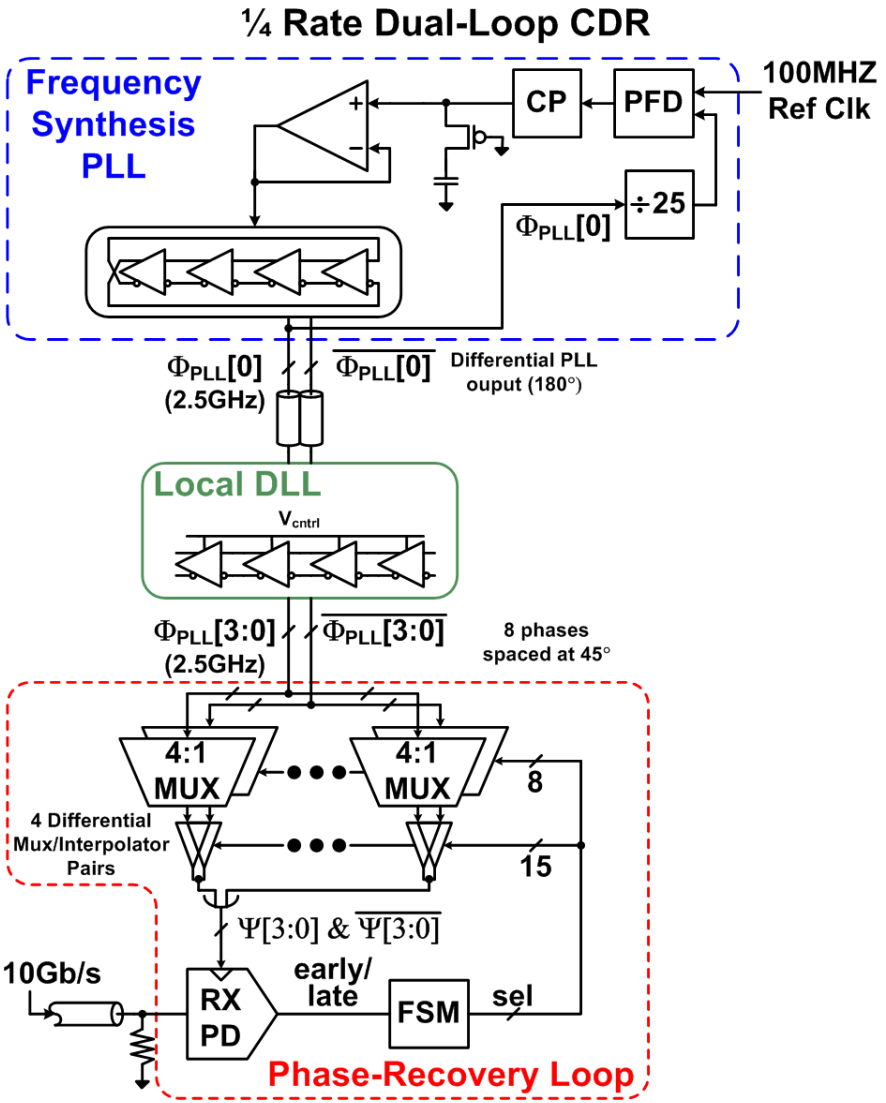
Phase Interpolator (PI) Based CDR (2)

- Frequency synthesis loop can be a global PLL.
- Can be difficult to distribute multiple phases long distance.
- Need to preserve phase spacing.
- Clock distribution power increases with phase number.
- If CDR needs more than 4 phases consider local phase generation.



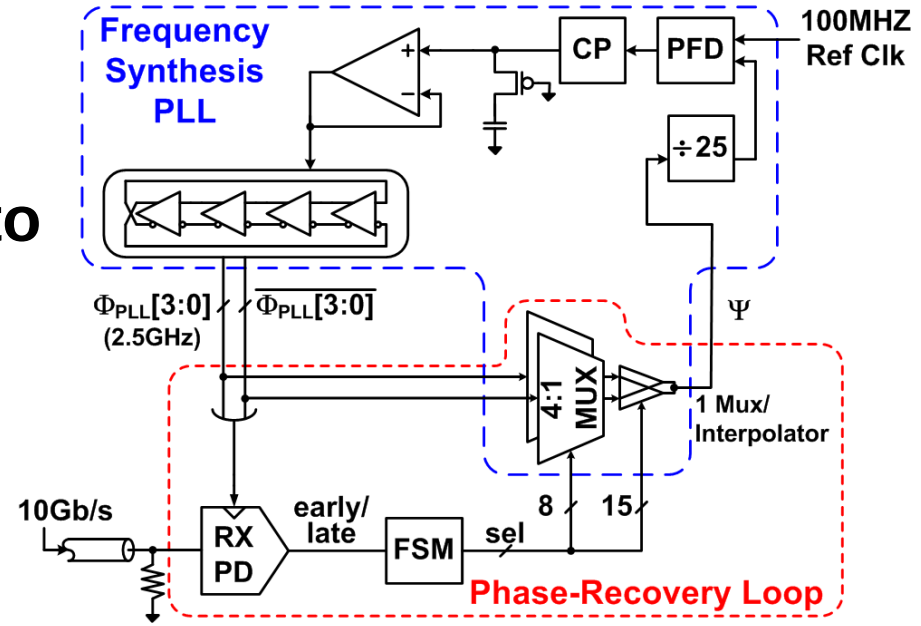
DLL Local Phase Generation

- **Only differential clock is distributed from global PLL.**
- **Delay-Locked Loop (DLL) locally generates the multiple clock phases for the phase interpolators.**
  - DLL can be per-channel or shared by a small number (4).
- **Same architecture can be used in a forwarded-clock system.**
  - Replace frequency synthesis PLL with forwarded-clock signals



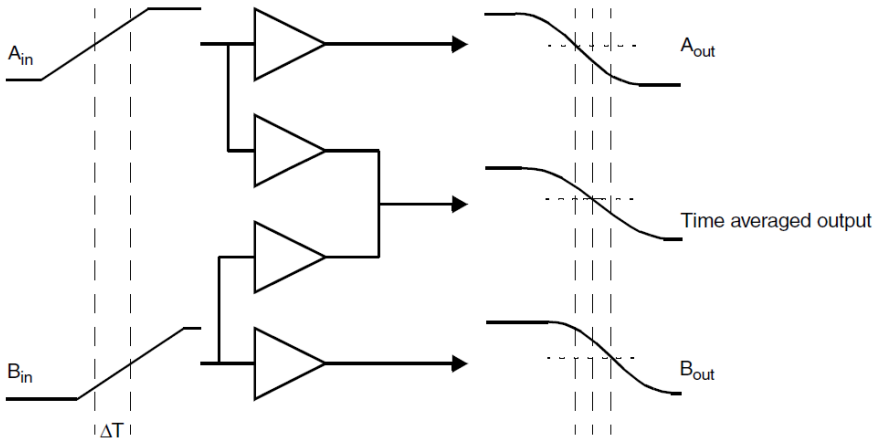
Phase Rotator PLL (PR-PLL)

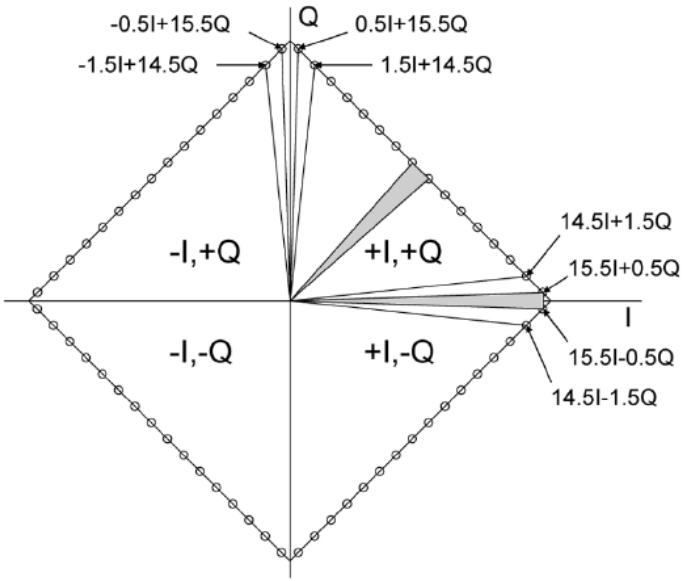
- **Phase interpolators can be expensive in terms of power and area.**
- **Phase rotator PLL places one interpolator in PLL feedback to adjust all VCO output phases simultaneously.**
- **Now frequency synthesis and phase recovery loops are coupled.**
  - Need PLL bandwidth greater than phase loop
    - Useful in filtering VCO noise



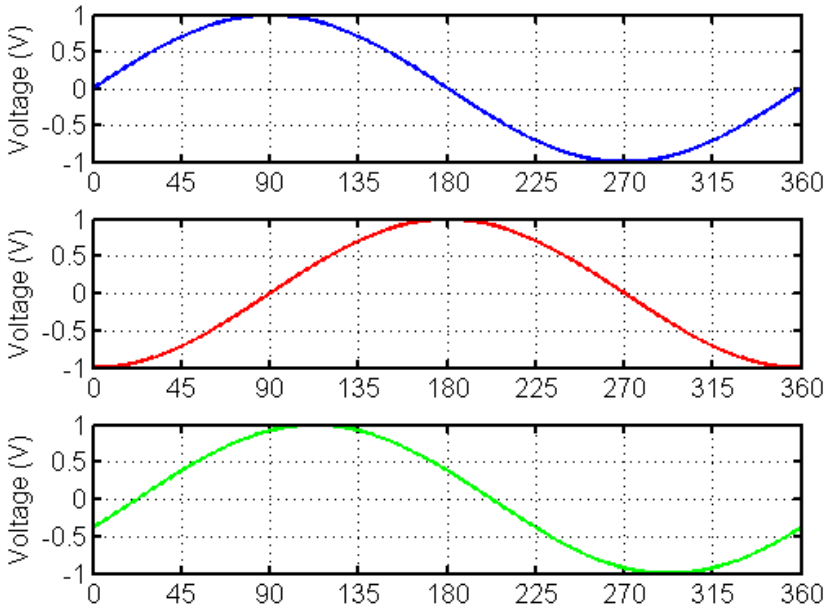
Phase Interpolators

- **Phase interpolators realize digital-to-phase conversion (DPC).**
- **Produce an output clock that is a weighted sum of two input clock phases.**
- **Common circuit structures**
  - Tail current summation interpolation
  - Voltage-mode interpolation
- **Interpolator code mapping techniques**
  - Sinusoidal
  - Linear

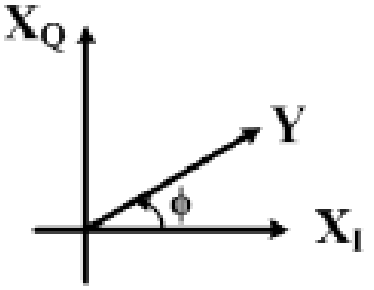




Sinusoidal Phase Interpolation



$$X_I = A\sin(\omega t)$$



$$X_Q = A\sin(\omega t - \pi/2) = -A\cos(\omega t)$$

$$Y = A\sin(\omega t - \phi)$$

$$= A\cos(\phi)\sin(\omega t) - A\sin(\phi)\cos(\omega t) \quad \left(0 \le \phi \le \frac{\pi}{2}\right)$$

$$= \cos(\phi)X_I + \sin(\phi)X_Q = a_1 X_I + a_2 X_Q$$

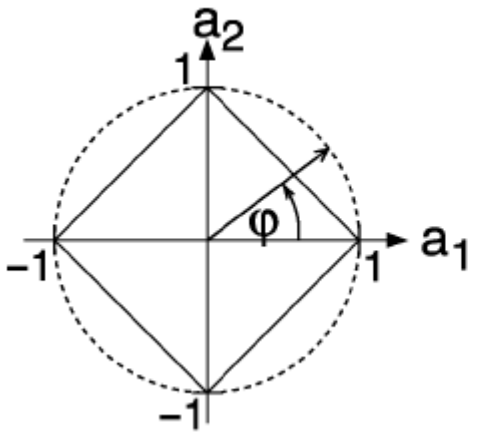
- **Arbitrary phase shift can be generated with linear summation of I/Q clock signal.**

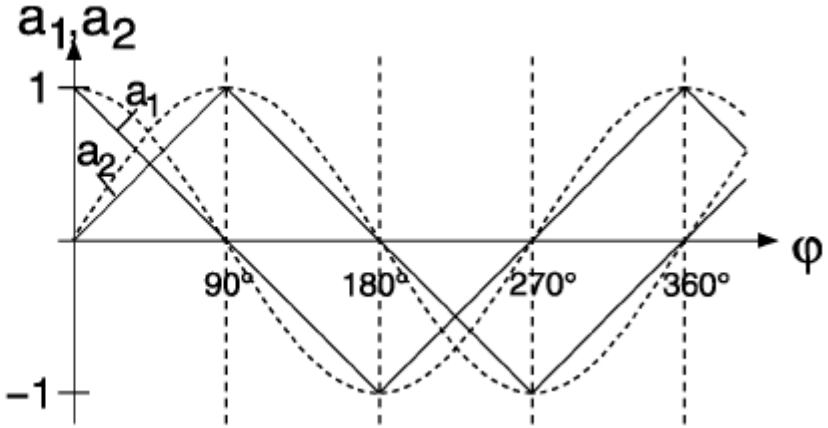
$$Y = A\sin(\omega t - \phi) = a_1 X_1 + a_2 X_Q$$

$$\textbf{where } a_1 = \cos(\phi) \textbf{ and } a_2 = \sin(\phi)$$

$$a_1^2 + a_2^2 = 1$$

Sinusoidal vs Linear Phase Interpolation



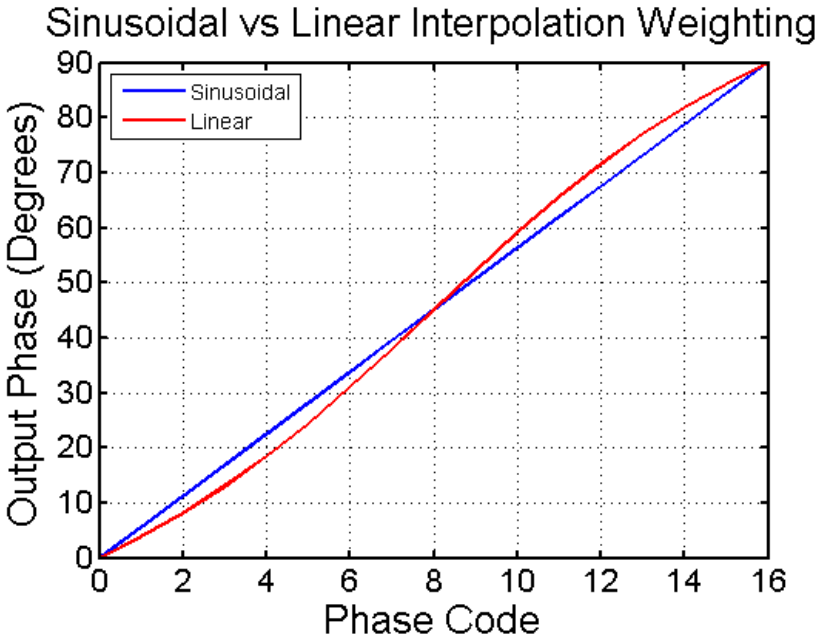


- **It can be difficult to generate a circuit that implements sinusoidal weighting.**

$$a_1^2 + a_2^2 = 1$$

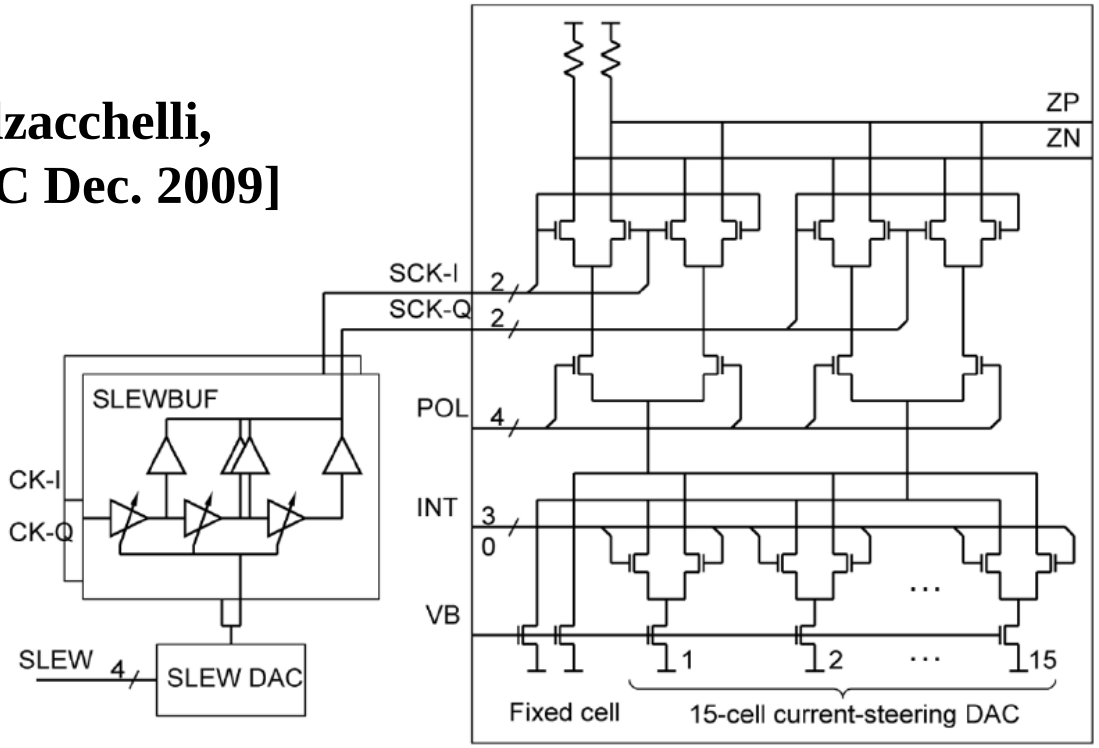
- **In practice, a linear weighting is often used.**

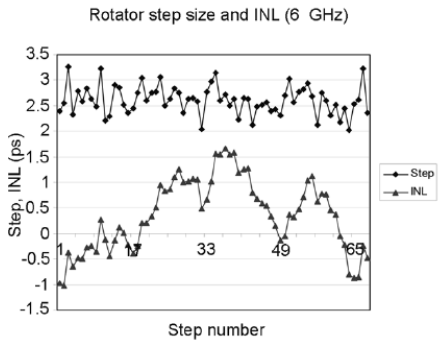
$$a_1 + a_2 = 1$$

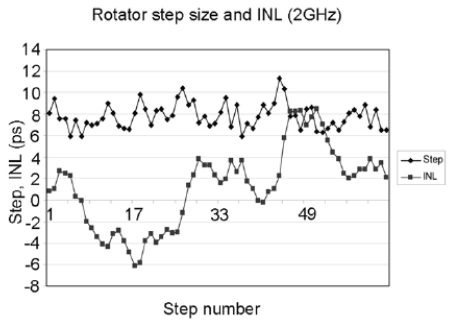


Tail-Current Summation PI

[Bulzacchelli, JSSC Dec. 2009]

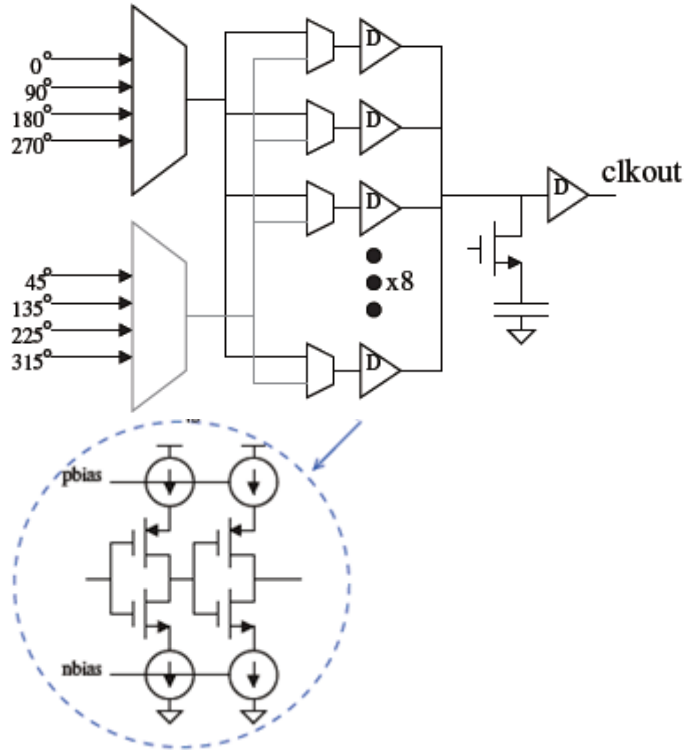


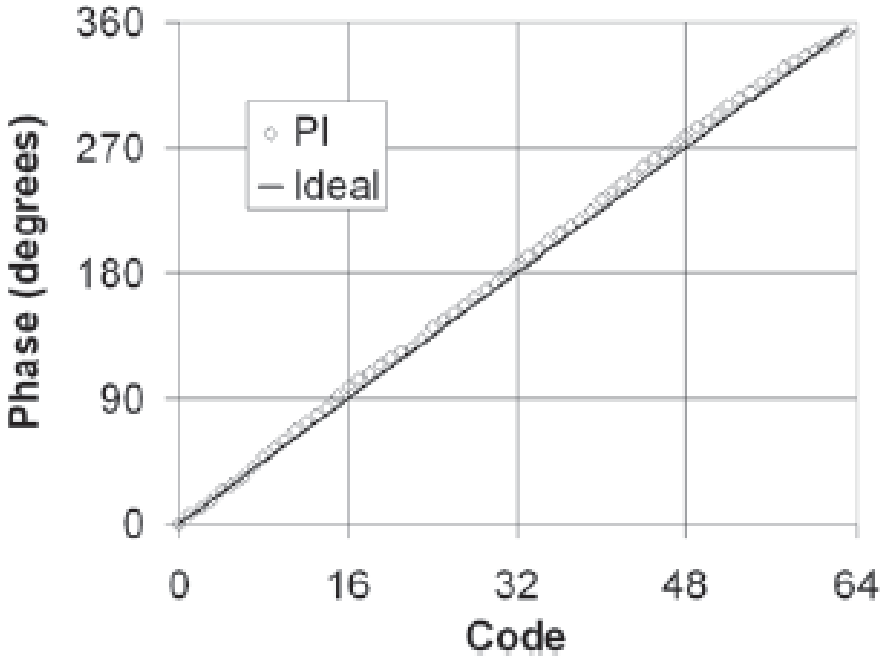




- **Control of I/Q polarity allows for full 360° phase rotation with phase step determined by resolution of weighting DAC.**
- **For linearity over a wide frequency range, important to control either input or output time constant (slew rate).**

Voltage-Mode Summation PI

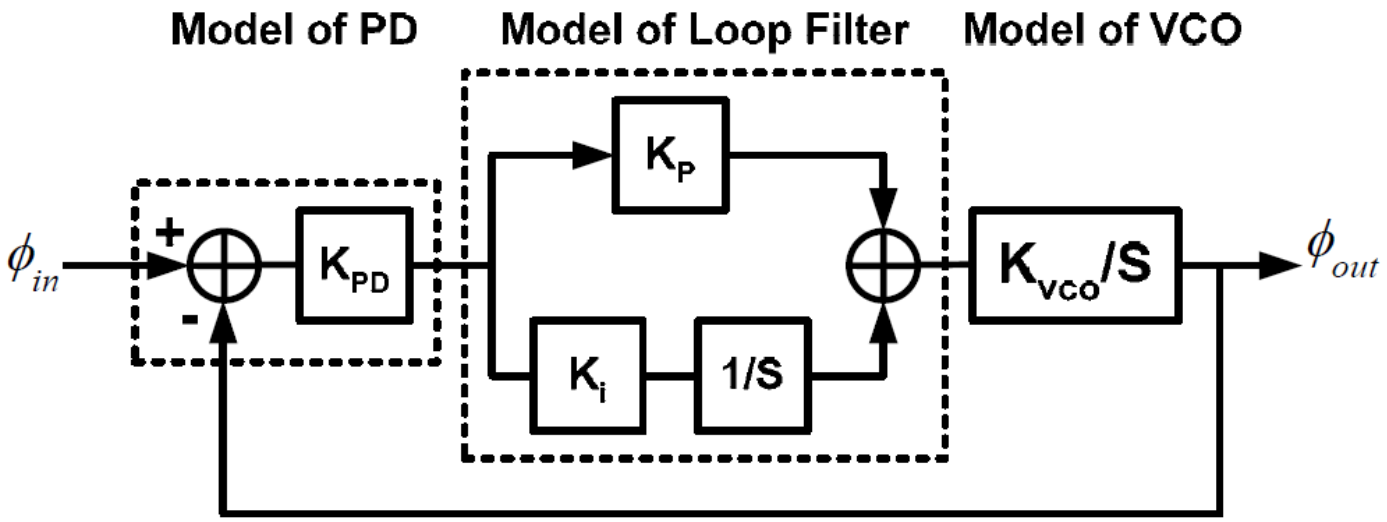




[Joshi, VLSI Symp. 2009]

- **For linearity over a wide frequency range, important to control either input or output time constant (slew rate).**

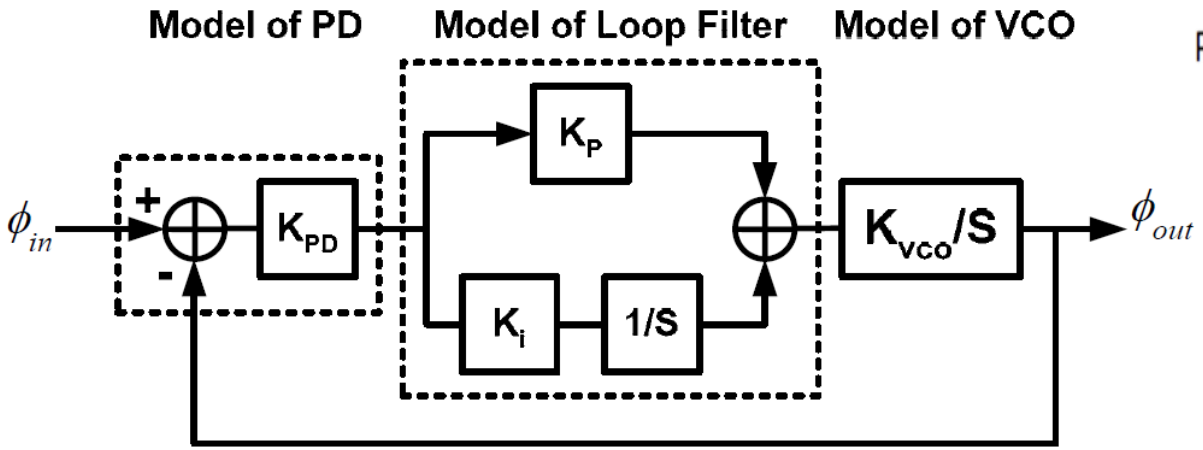
CDR Jitter Model

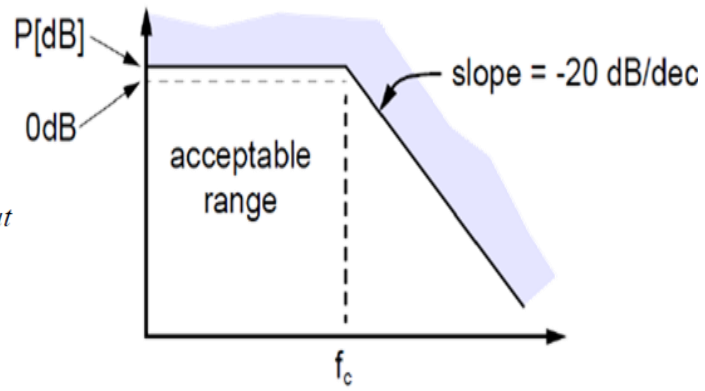


$$\frac{\phi_{out}}{\phi_{in}} = \frac{s \cdot K_P \cdot K_{PD} \cdot K_{VCO} + K_i \cdot K_{PD} \cdot K_{VCO}}{s^2 + s \cdot K_P \cdot K_{PD} \cdot K_{VCO} + K_i \cdot K_{PD} \cdot K_{VCO}}$$

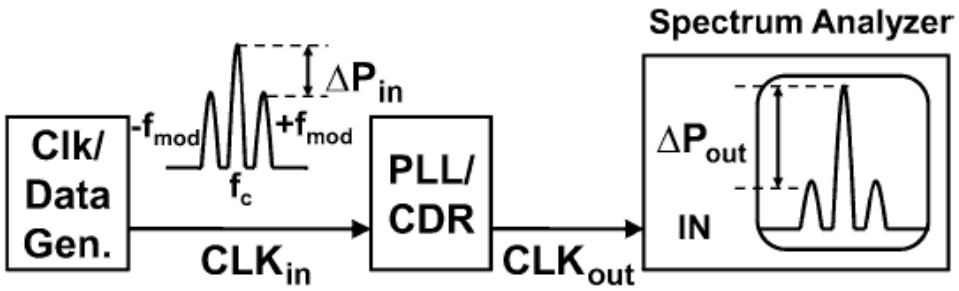
$$K_P = I_C \cdot R \qquad K_i = \frac{I_C}{C} \qquad \omega_n = \sqrt{K_i \cdot K_{PD} \cdot K_{VCO}} \qquad \zeta = \frac{K_P}{K_i} \cdot \frac{\omega_n}{2}$$

Jitter Transfer Function (JTF) calculation



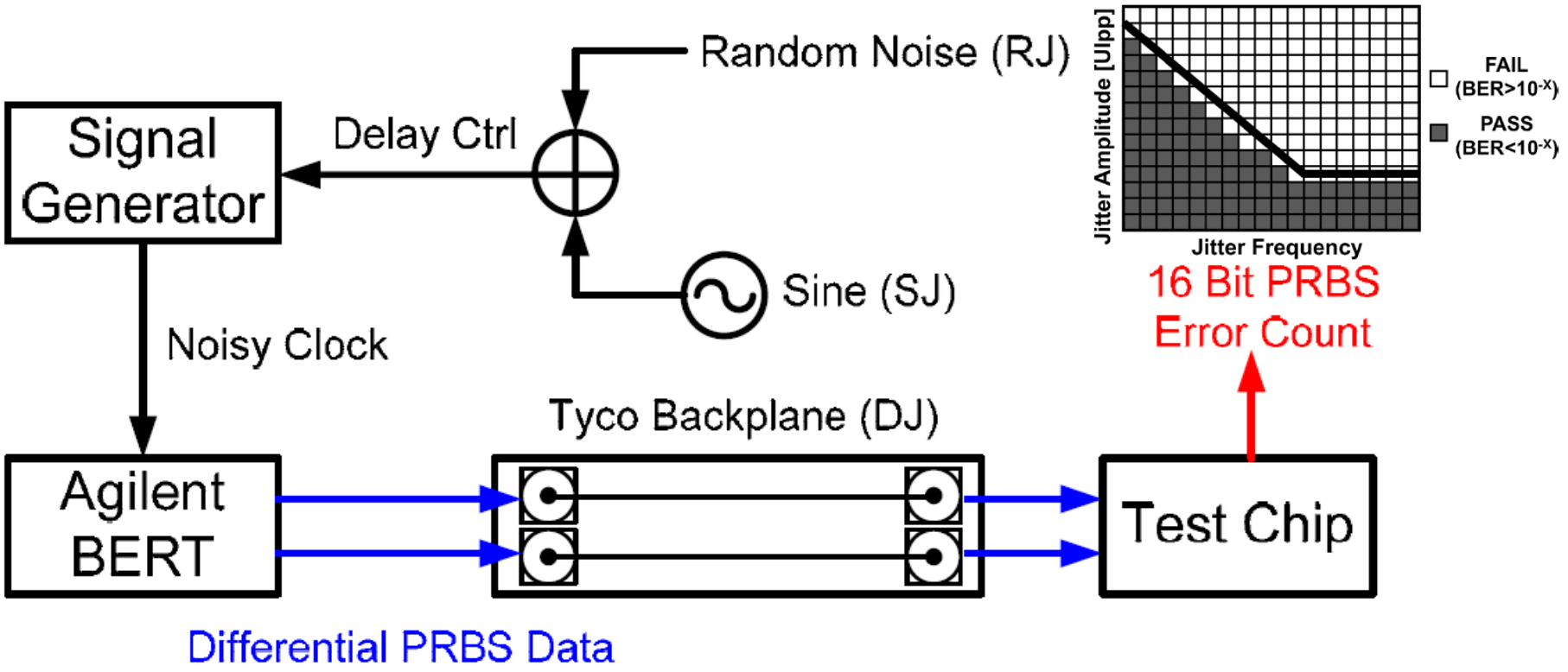


- **Jitter transfer is how much jitter "transfers" to output.**
  - If the PLL has any peaking in the phase transfer function, this jitter can actually be amplified.



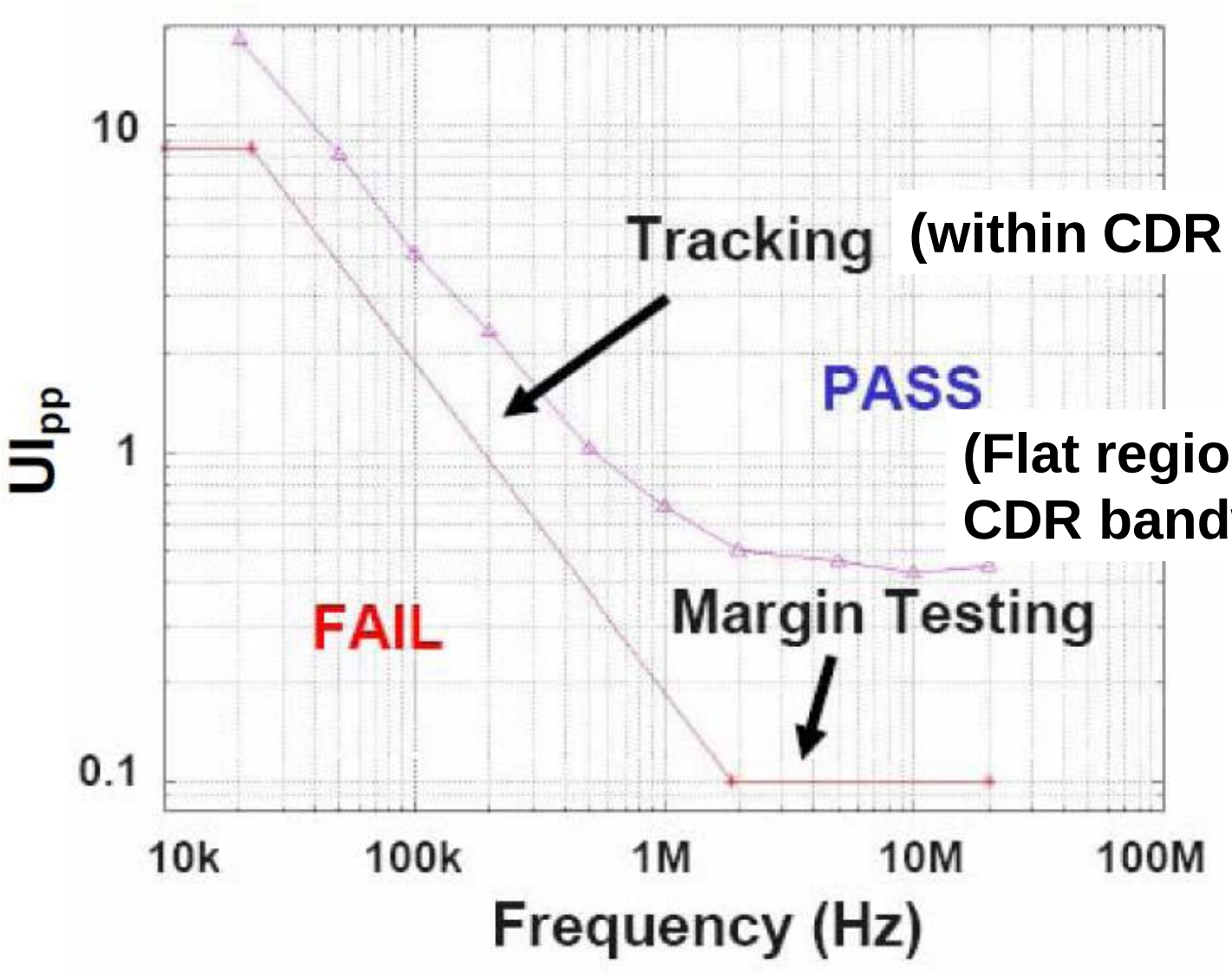
$$J_{TF}(f_{mod})=\Delta P_{out}/\Delta P_{in}$$

Jitter Tolerance Measurement (1)



- **Random and sinusoidal jitter are added by modulating BERT clock**
- **Deterministic jitter is added by passing data through the channel**
- **For a given frequency, sinusoidal jitter amplitude is increased until the minimum acceptable BER (10^{-12}) is recorded.**

Jitter Tolerance Measurement (2)



(within CDR bandwidth)

(Flat region is beyond CDR bandwidth)

CDR Summary

- **Clock frequency and optimum phase position are extracted from incoming data.**
- **Phase detection continuously running.**
- **Jitter tracking limited by CDR bandwidth.**
- **With technology scaling we can make CDRs with higher bandwidths and the jitter tracking advantages of source synchronous systems is diminished.**
- **Possible CDR implementations**
  - Stand-alone PLL
  - “Dual-loop” architecture with a PLL or DLL and phase interpolators (PI)
  - Phase-rotator PLL