Short Course On
Phase-Locked Loops and Their Applications
Day 4, AM Lecture

Digital Frequency Synthesizers

Michael Perrott

August 14, 2008

Copyright © 2008 by Michael H. Perrott

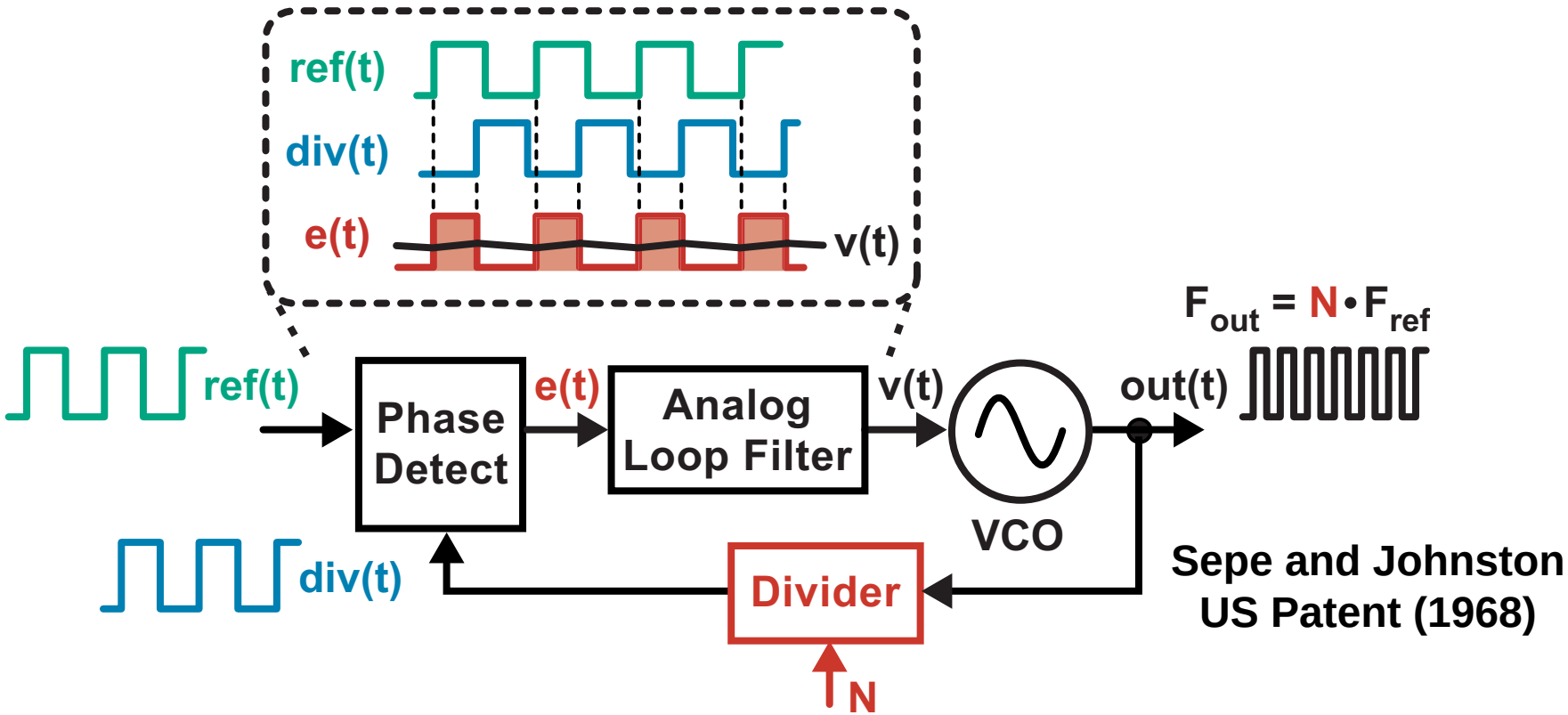
All rights reserved.

Why Are Digital Phase-Locked Loops Interesting?

- **PLLs are needed for a wide range of applications**
  - **Communication systems (both wireless and wireline)**
  - **Digital processors (to achieve GHz clocks)**
- **Performance is important**
  - **Phase noise can limit wireless transceiver performance**
  - **Jitter can be a problem for digital processors**
- **The standard analog PLL implementation is problematic in many applications**
  - **Analog building blocks on a mostly digital chip pose design and verification challenges**
  - **The cost of implementation is becoming too high …**

Can digital phase-locked loops offer excellent performance with a lower cost of implementation?

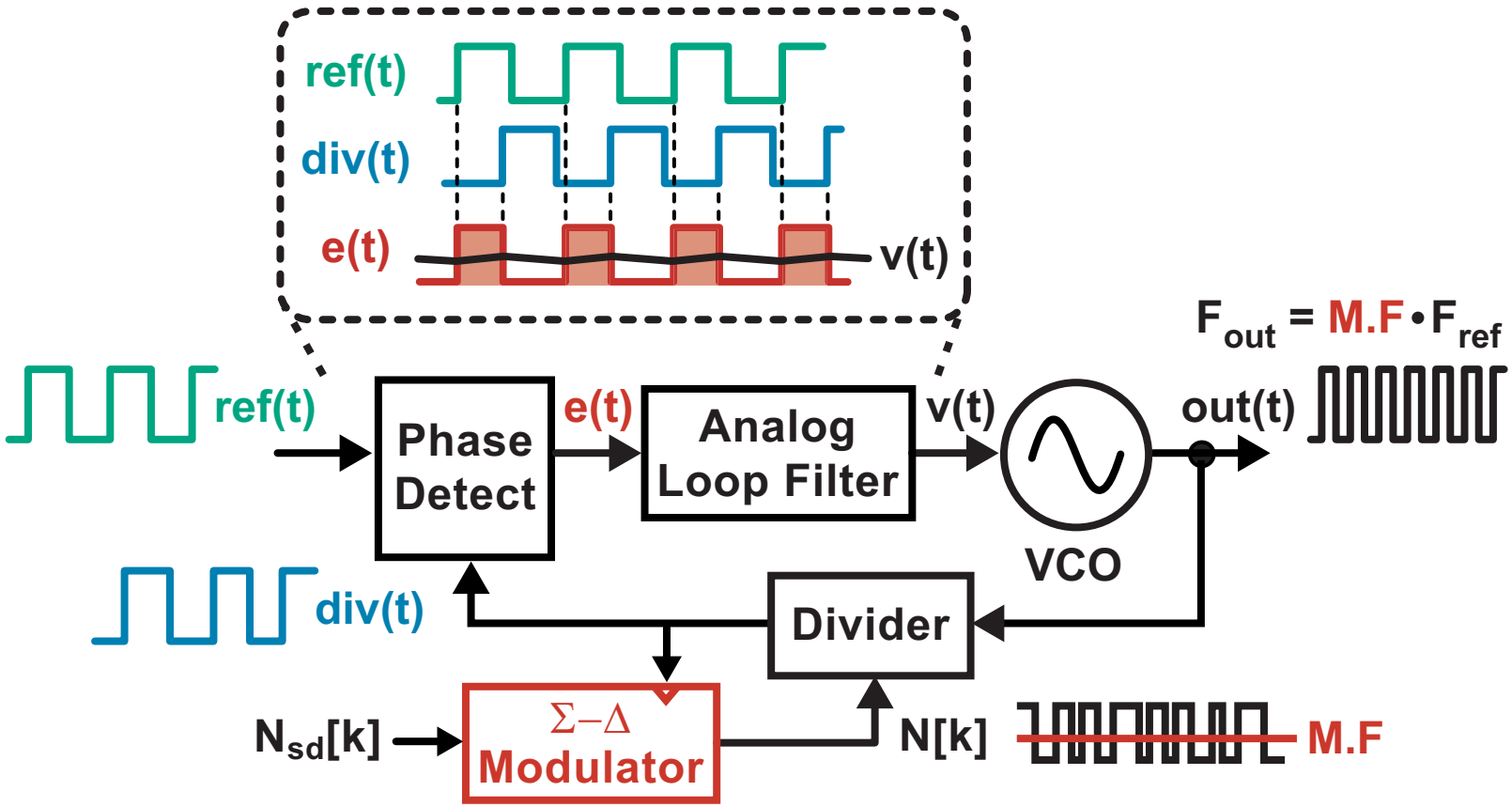
Integer-N Frequency Synthesizers



- Use digital counter structure to divide VCO frequency
  - Constraint: must divide by integer values
- Use PLL to synchronize reference and divider output

Output frequency is digitally controlled

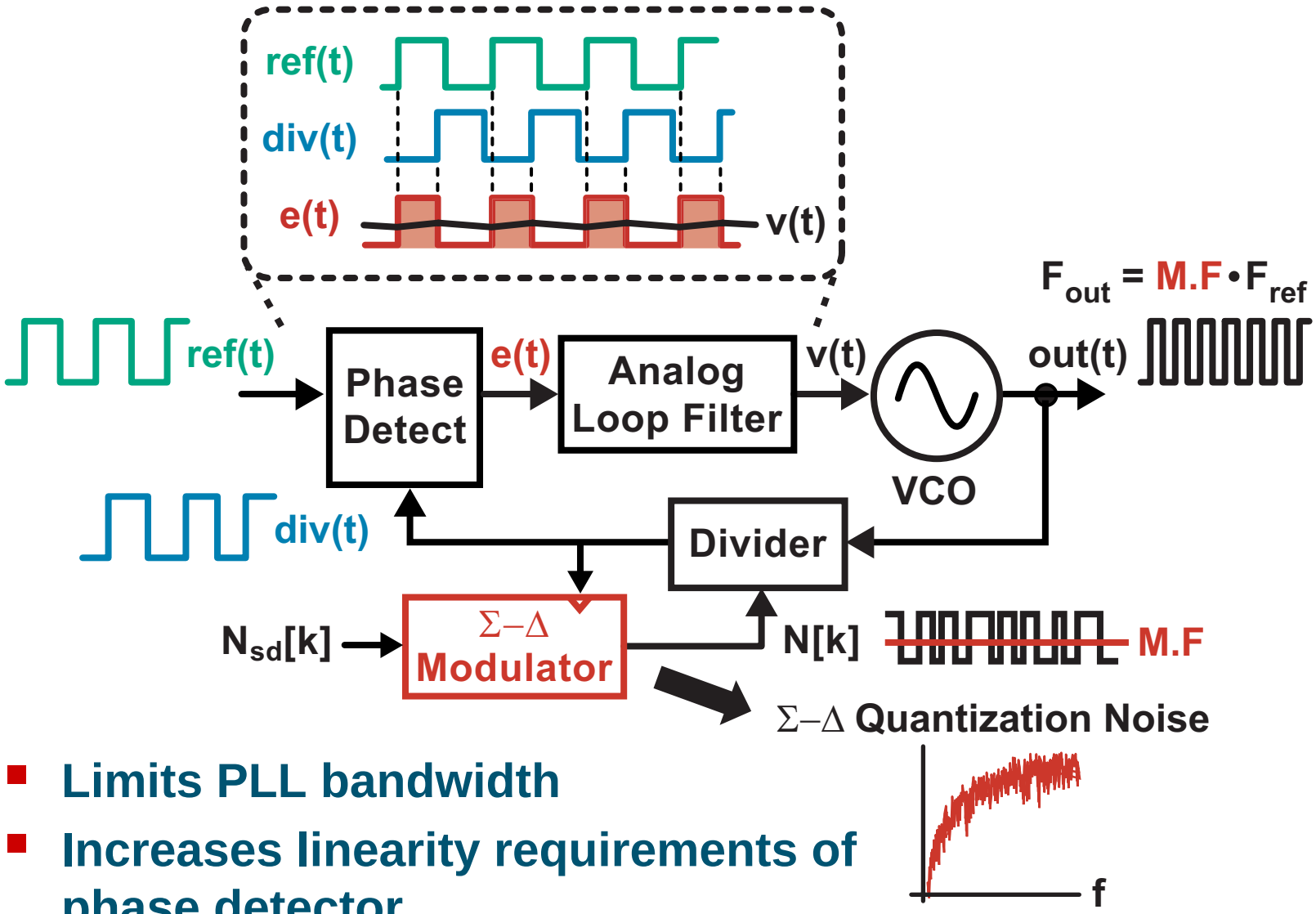
Fractional-N Frequency Synthesizers



- Dither divide value to achieve fractional divide values
  - PLL loop filter smooths the resulting variations

Very high frequency resolution is achieved

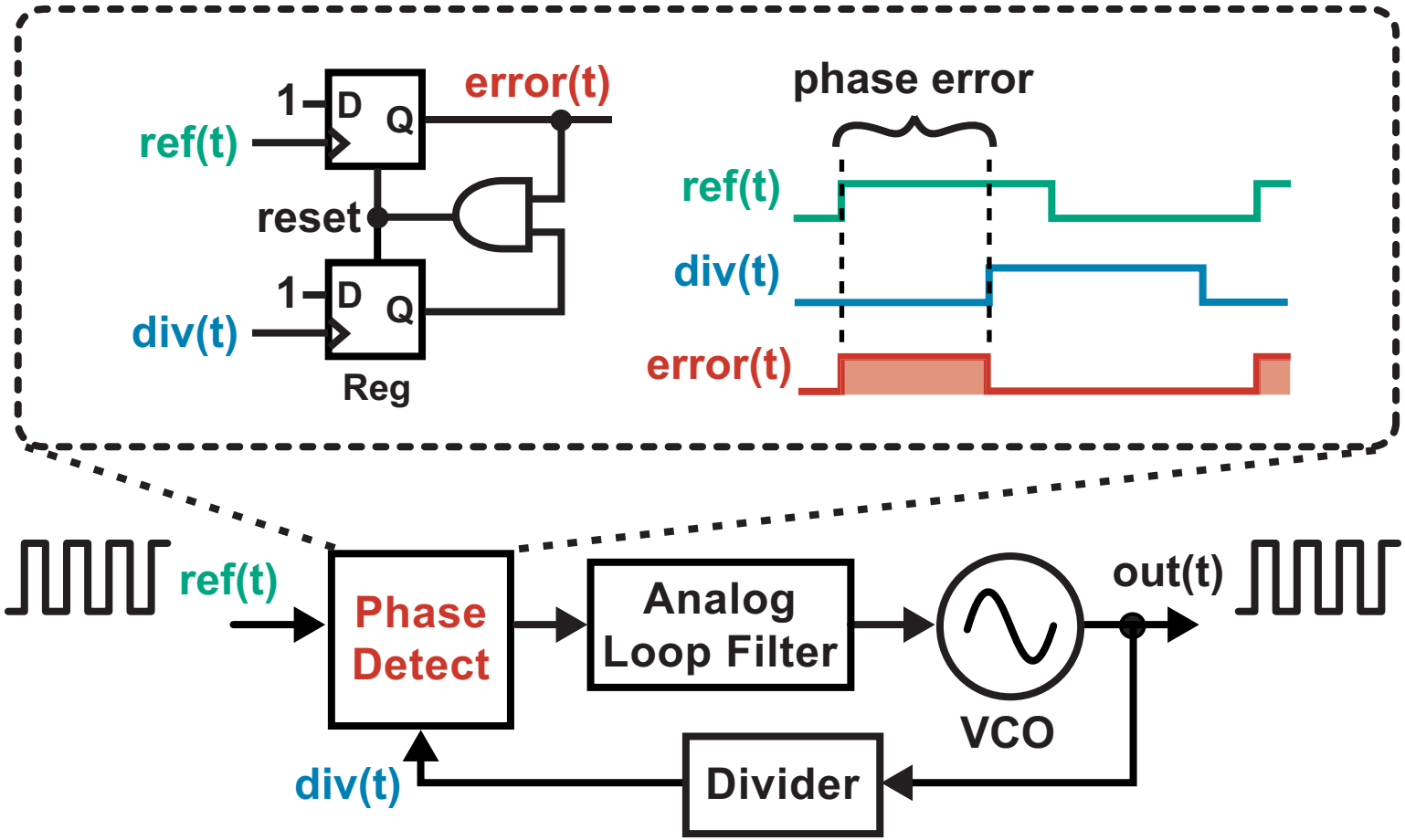
The Issue of Quantization Noise



- Limits PLL bandwidth
- Increases linearity requirements of phase detector

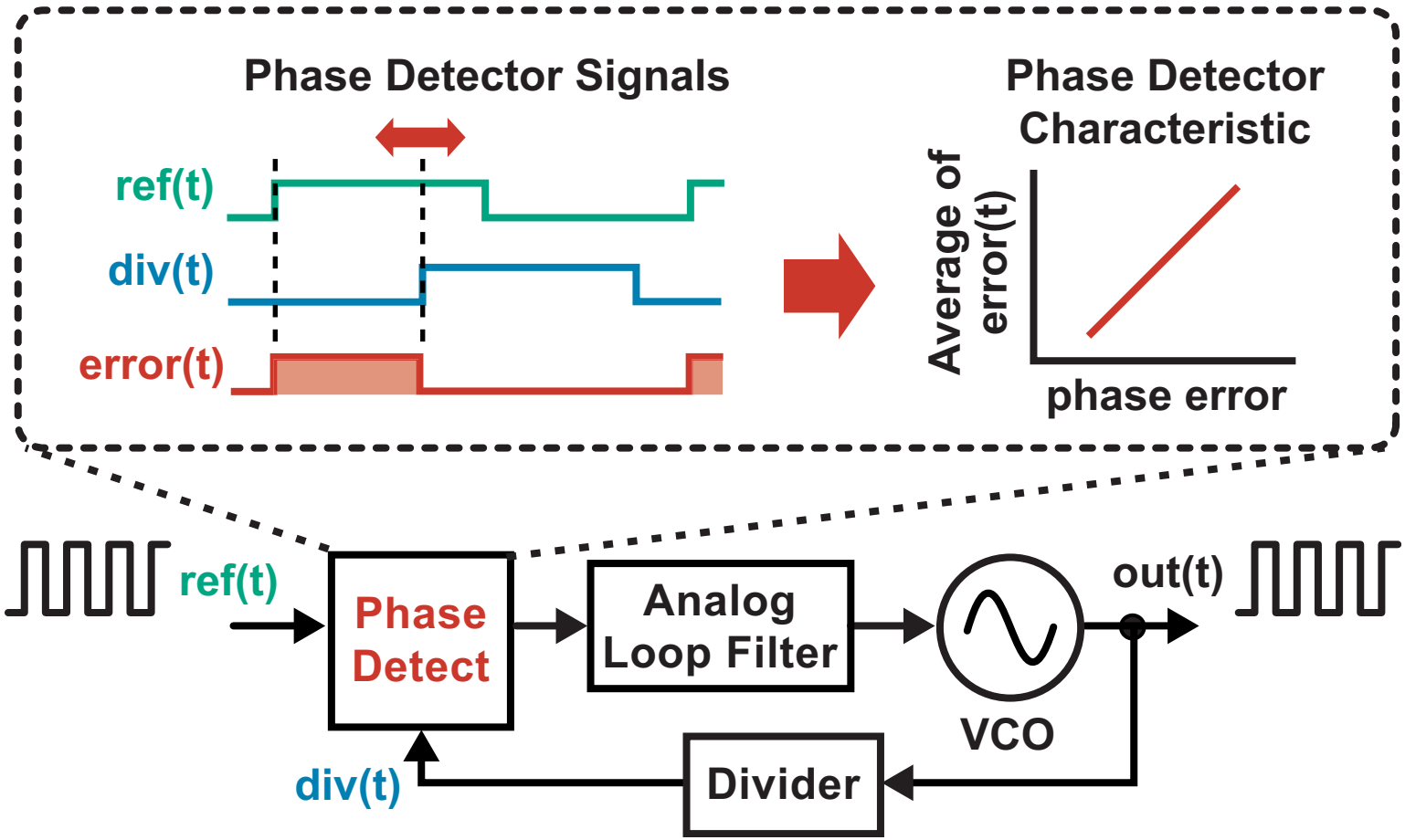
Striving for a Better PLL Implementation

Analog Phase Detection



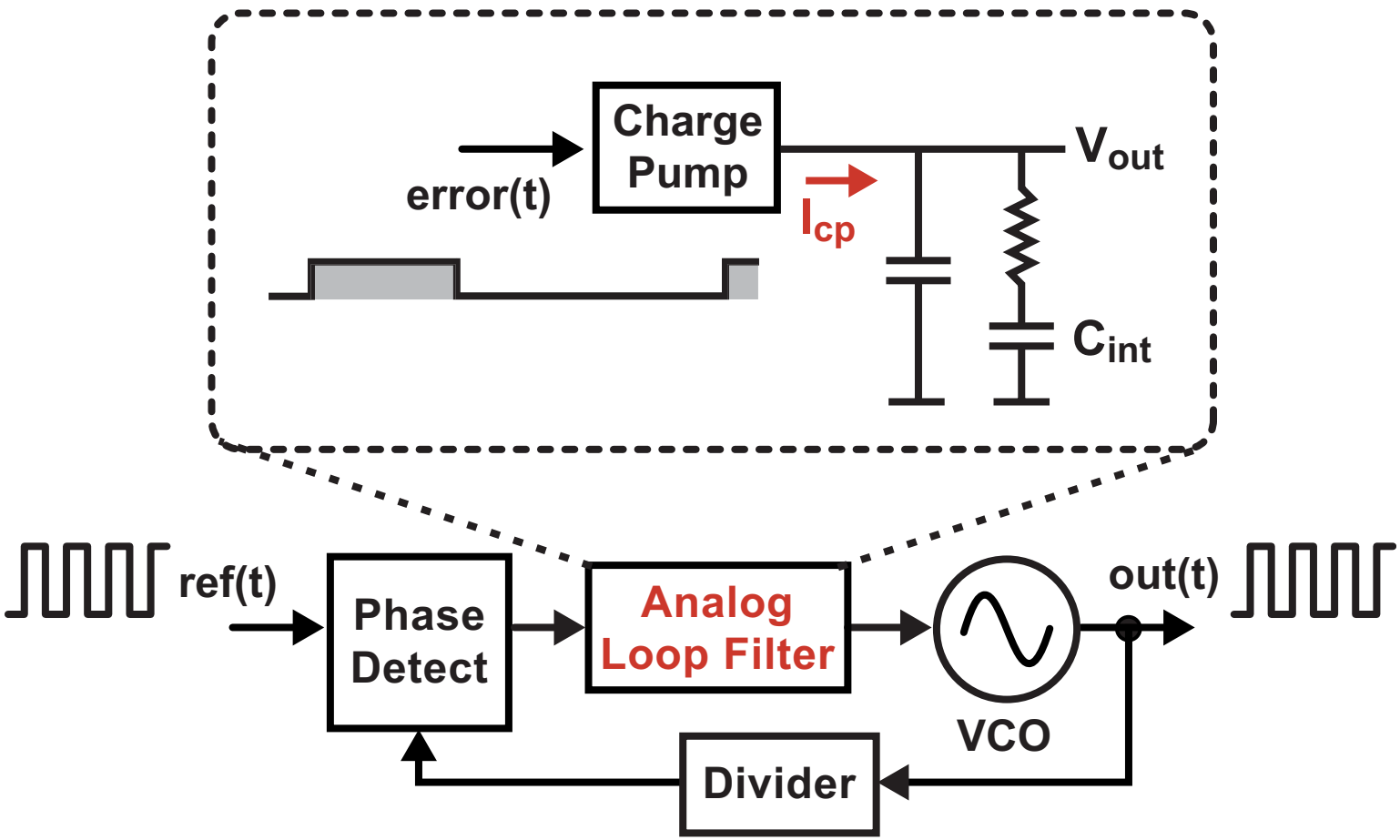
- Pulse width is formed according to phase difference between two signals
- Average of pulsed waveform is applied to VCO input

Tradeoffs of Analog Approach



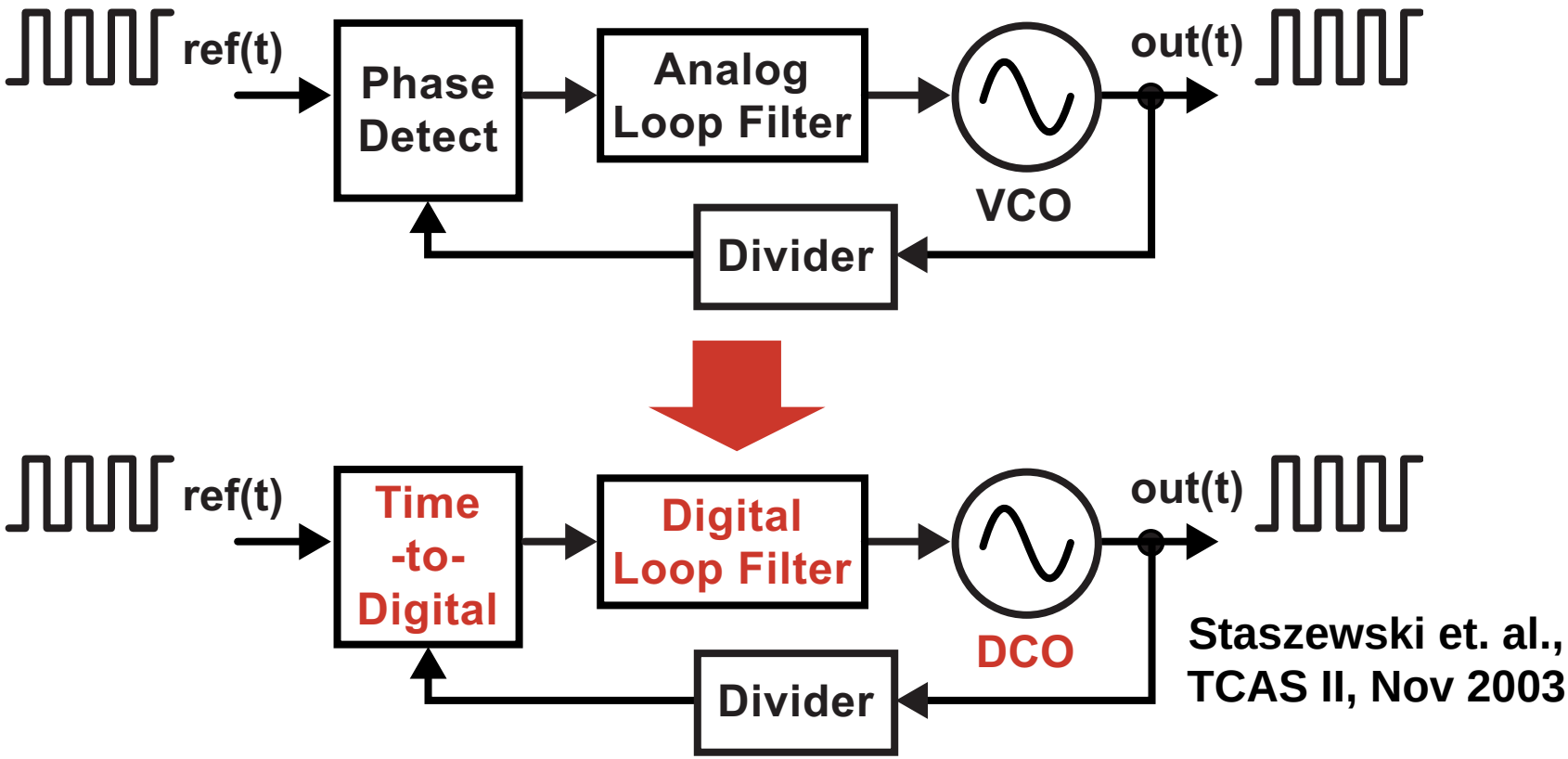
- **Benefit: average of pulsed output is a continuous, linear function of phase error**
- **Issue: analog loop filter implementation is undesirable**

Issues with Analog Loop Filter



- Charge pump: output resistance, mismatch
- Filter caps: leakage current, *large area*

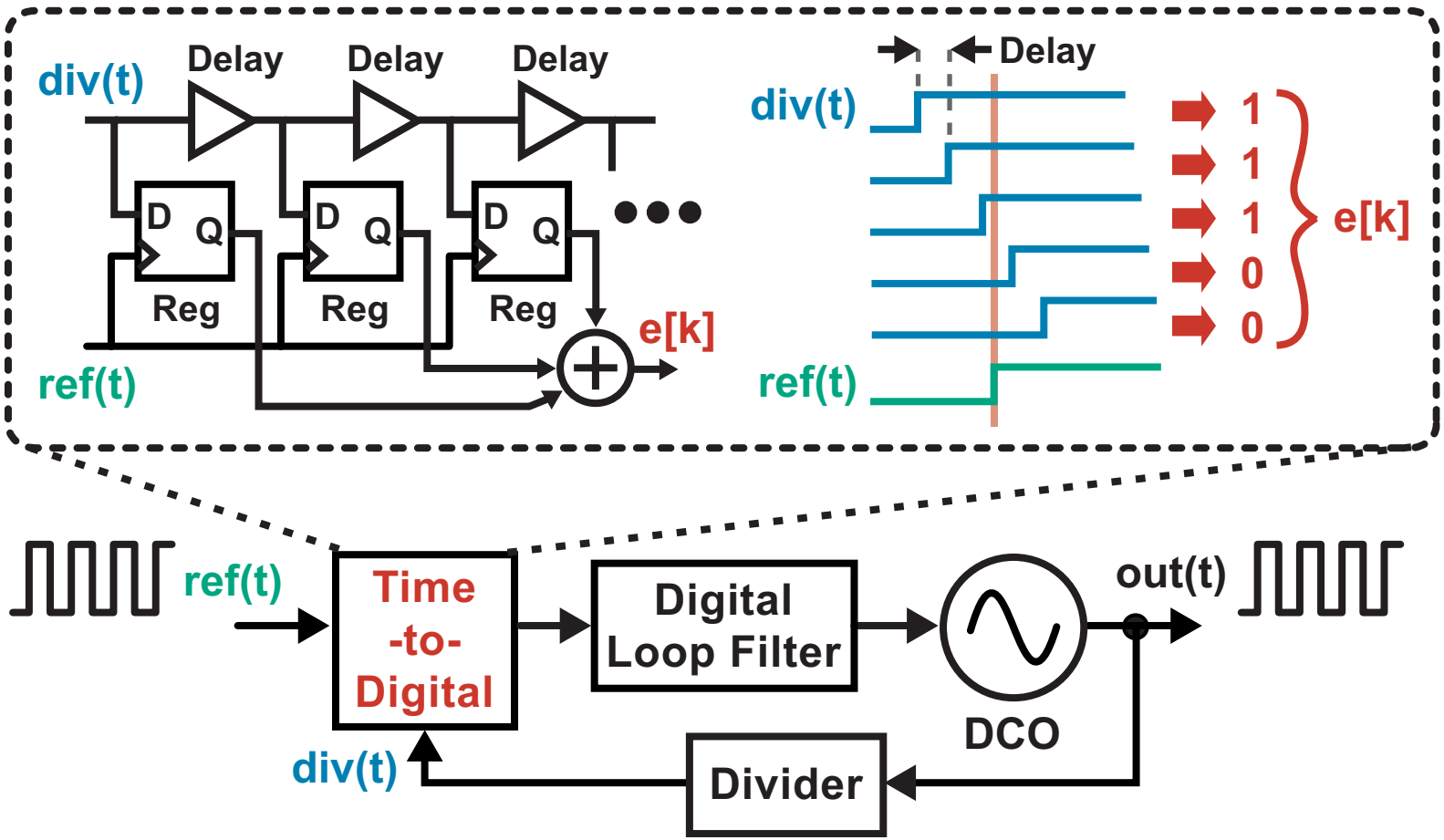
Going Digital …



- Digital loop filter: compact area, insensitive to leakage
- Challenges:
  - Time-to-Digital Converter (TDC)
  - Digitally-Controlled Oscillator (DCO)

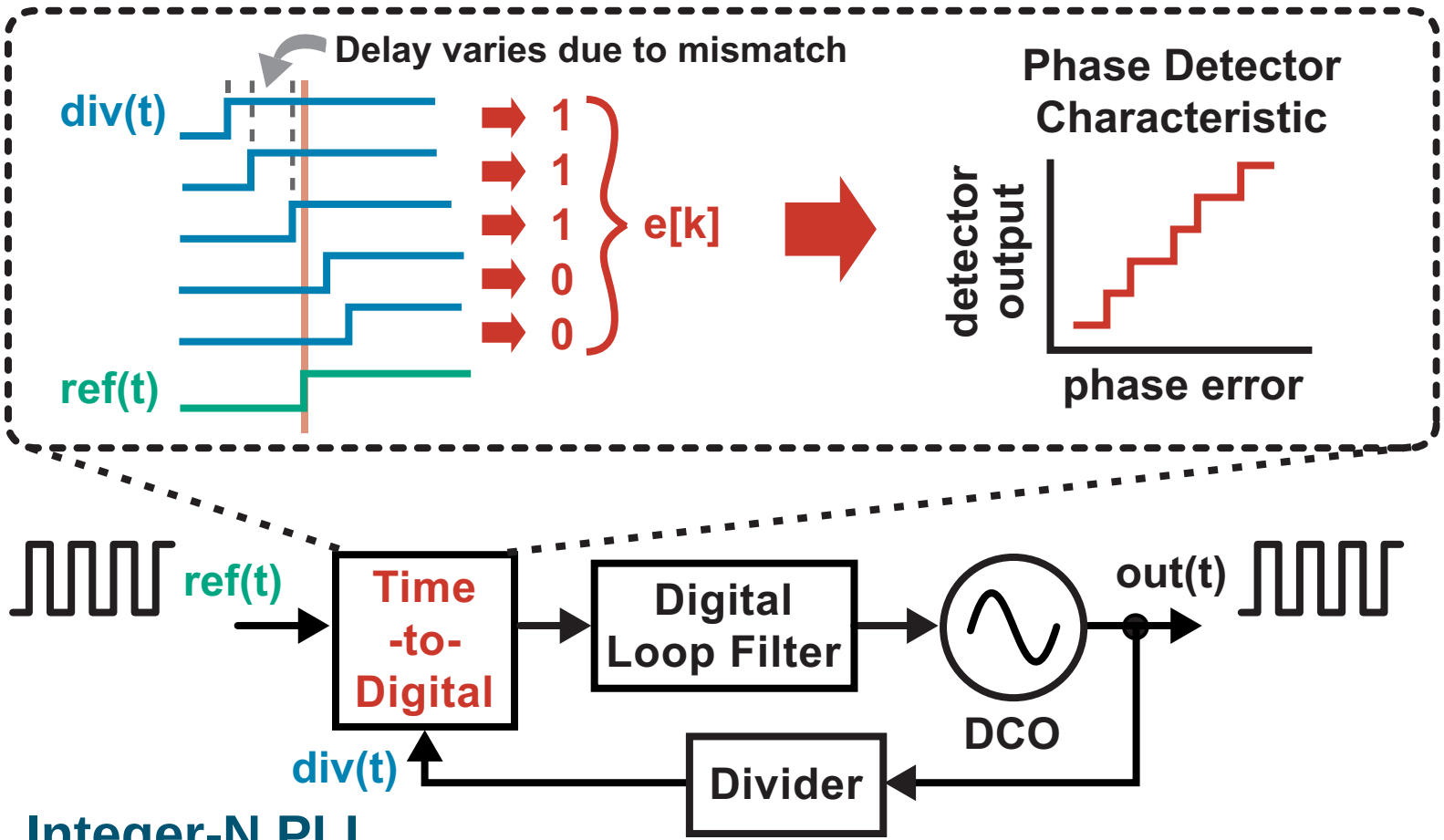
Time-to-Digital Conversion

Classical Time-to-Digital Converter



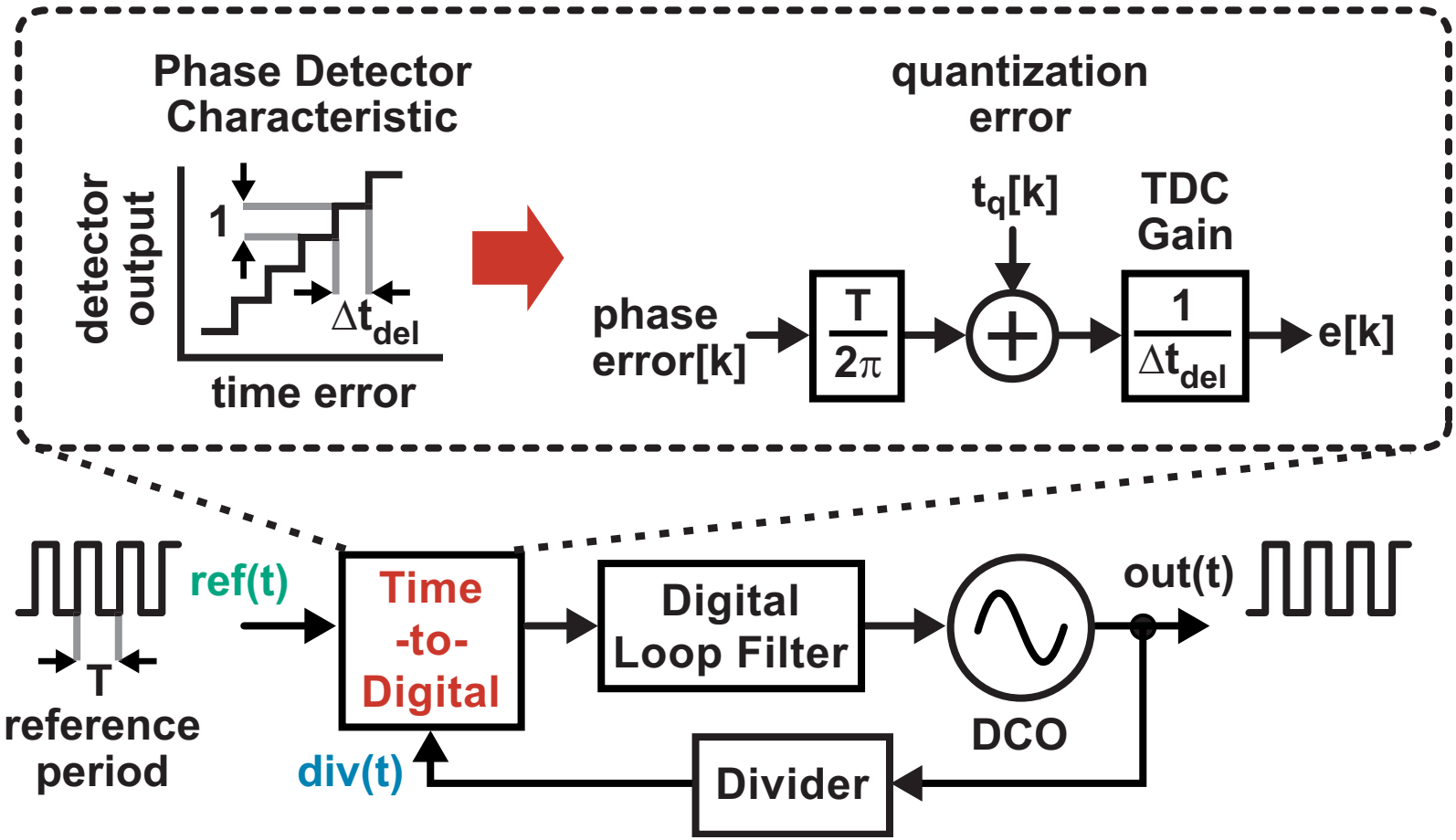
- Resolution set by a “Single Delay Chain” structure
  - Phase error is measured with delays and registers
- Corresponds to a flash architecture

Impact of Limited Resolution and Delay Mismatch



- Integer-N PLL
  - Limit cycles due to limited resolution (unless high ref noise)
- Fractional-N PLL
  - Fractional spurs due to non-linearity from delay mismatch

Modeling of TDC



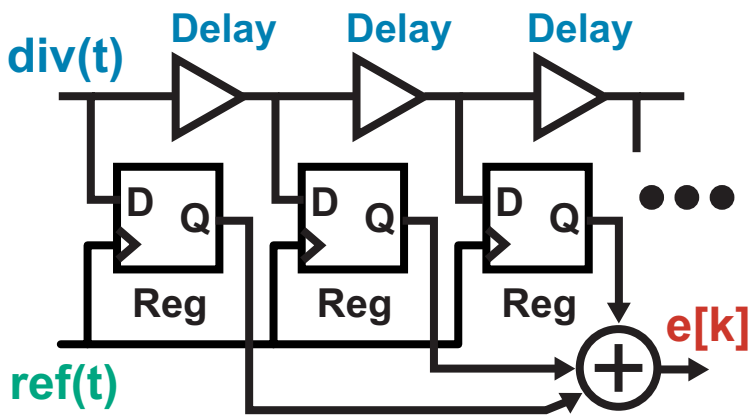
- Phase error converted to time error by scale factor: $T/2\pi$
- TDC introduces quantization error: $t_q[k]$
- TDC gain set by average delay per step: Δt_{del}

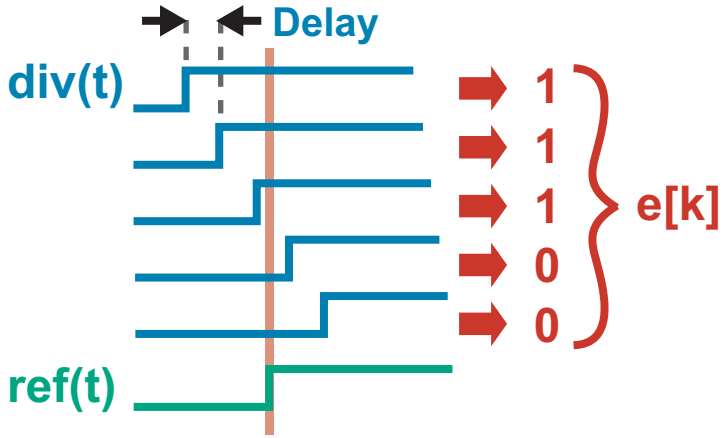
How Do We Improve Performance?

Two Key Issues:

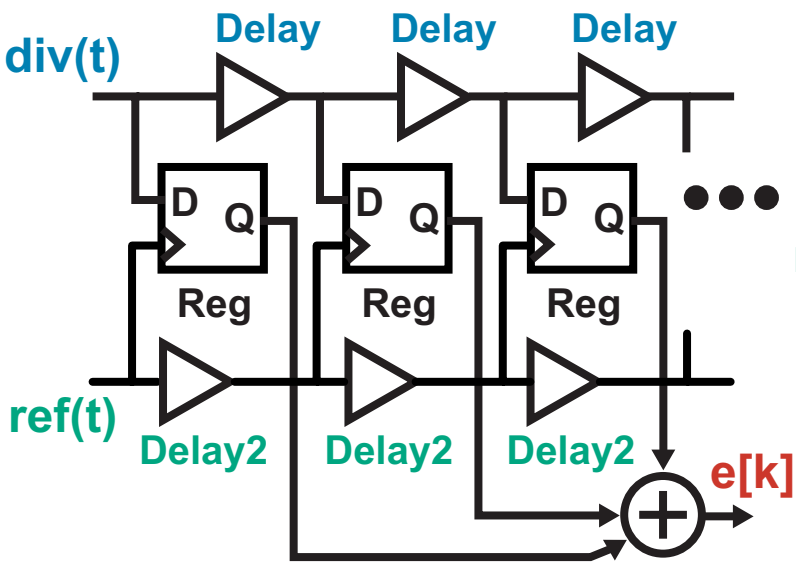
- **TDC resolution**
- **Mismatch**

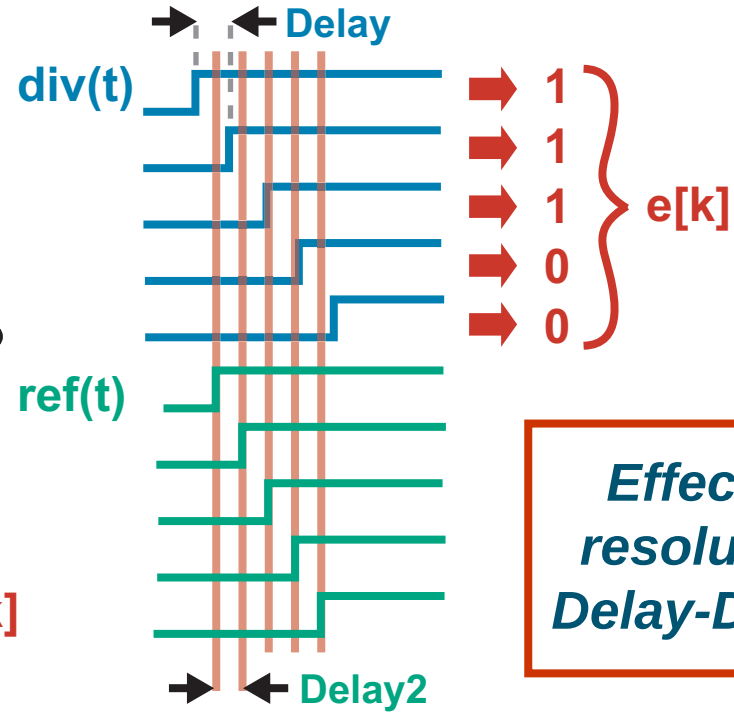
Improve Resolution with Vernier Delay Technique





Vernier

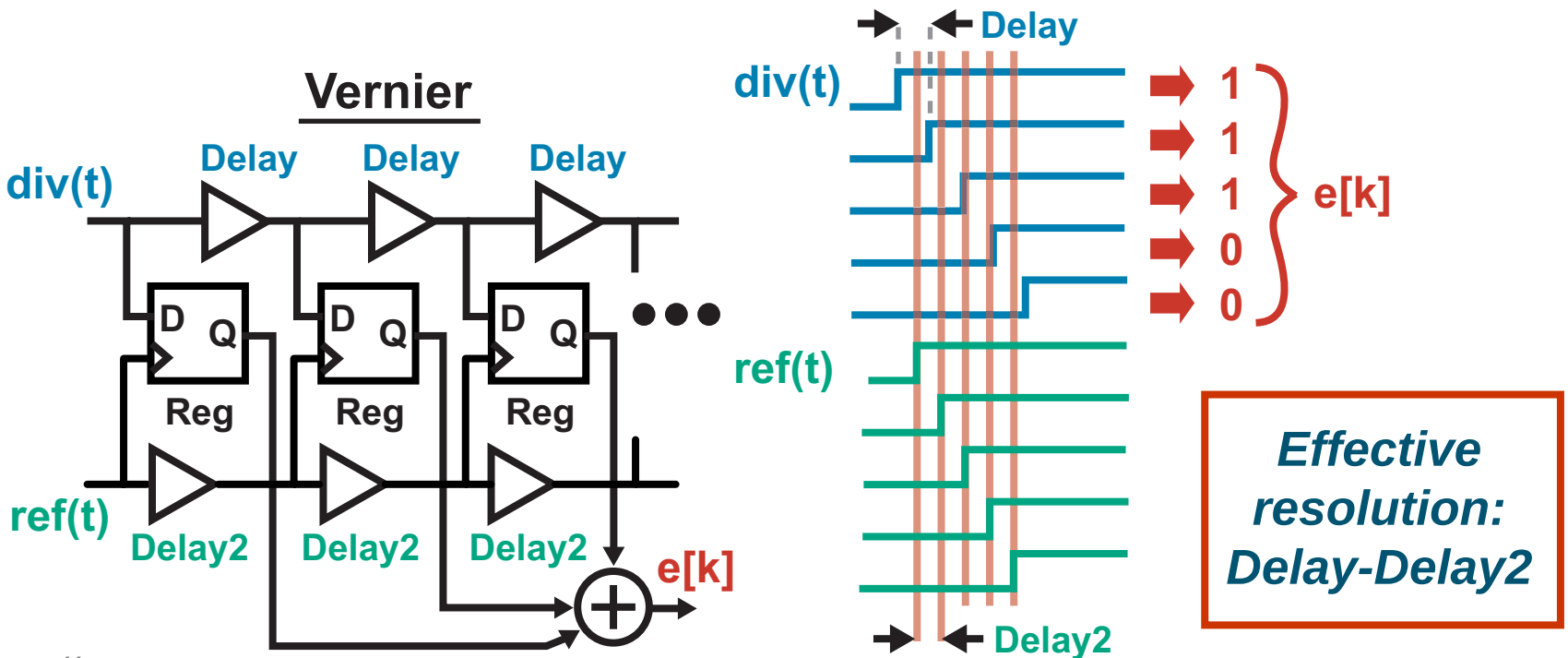




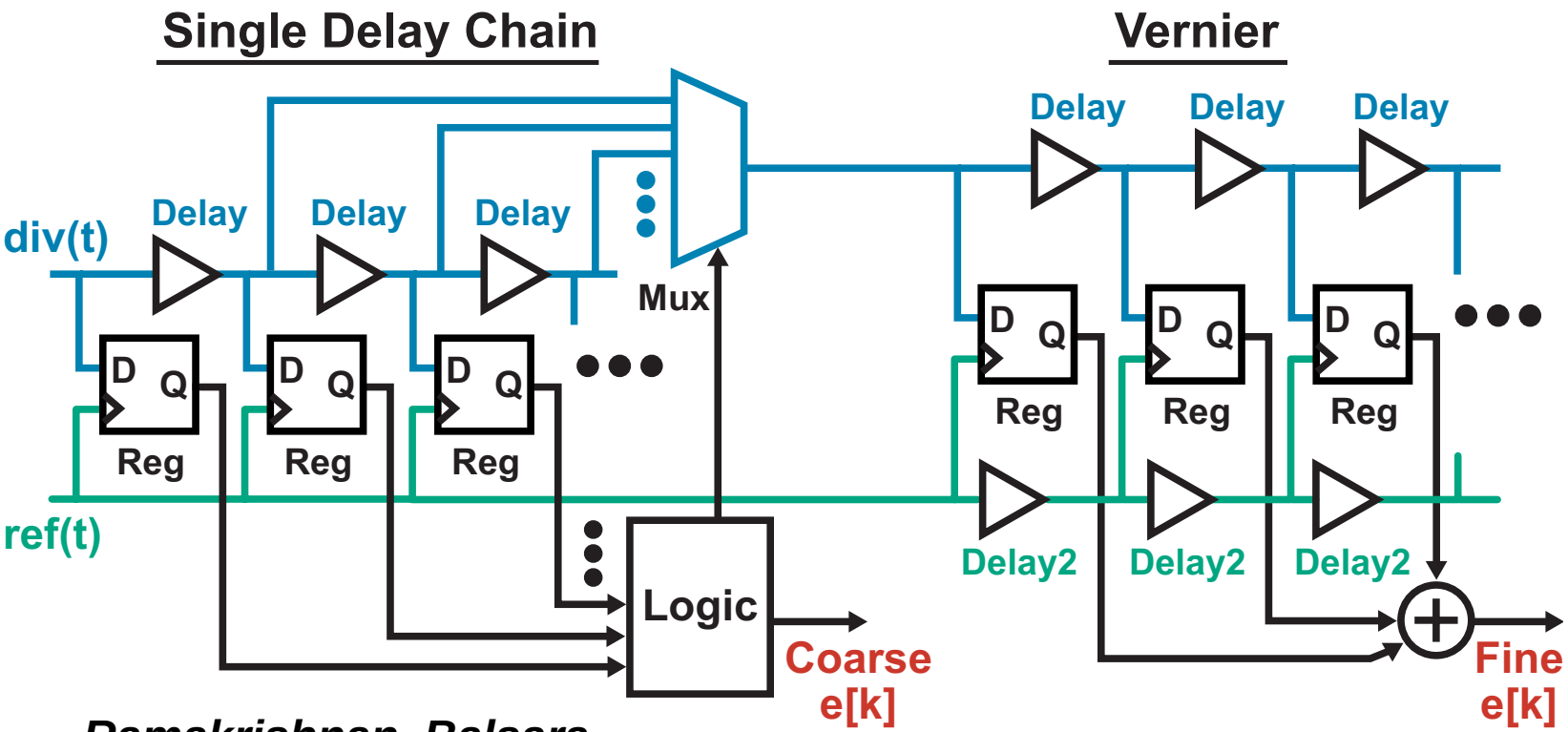
Effective resolution: Delay-Delay2

Issues with Vernier Approach

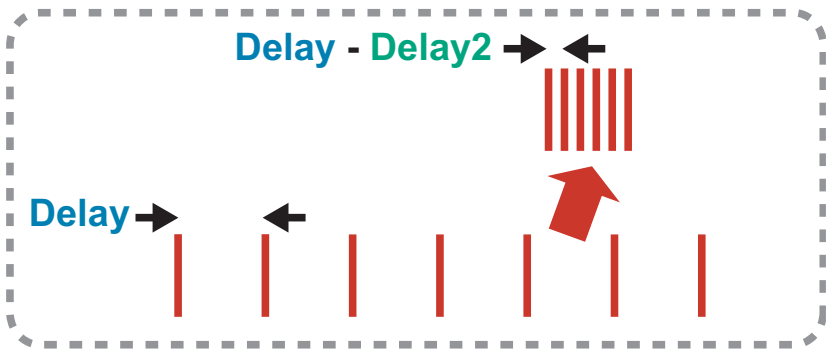
- **Mismatch issues are more severe than the single delay chain TDC**
  - **Reduced delay is formed as *difference* of two delays**
- **Large measurement range requires large area**
  - **Initial PLL frequency acquisition may require a large range**



Two-Step TDC Architecture Allows Area Reduction

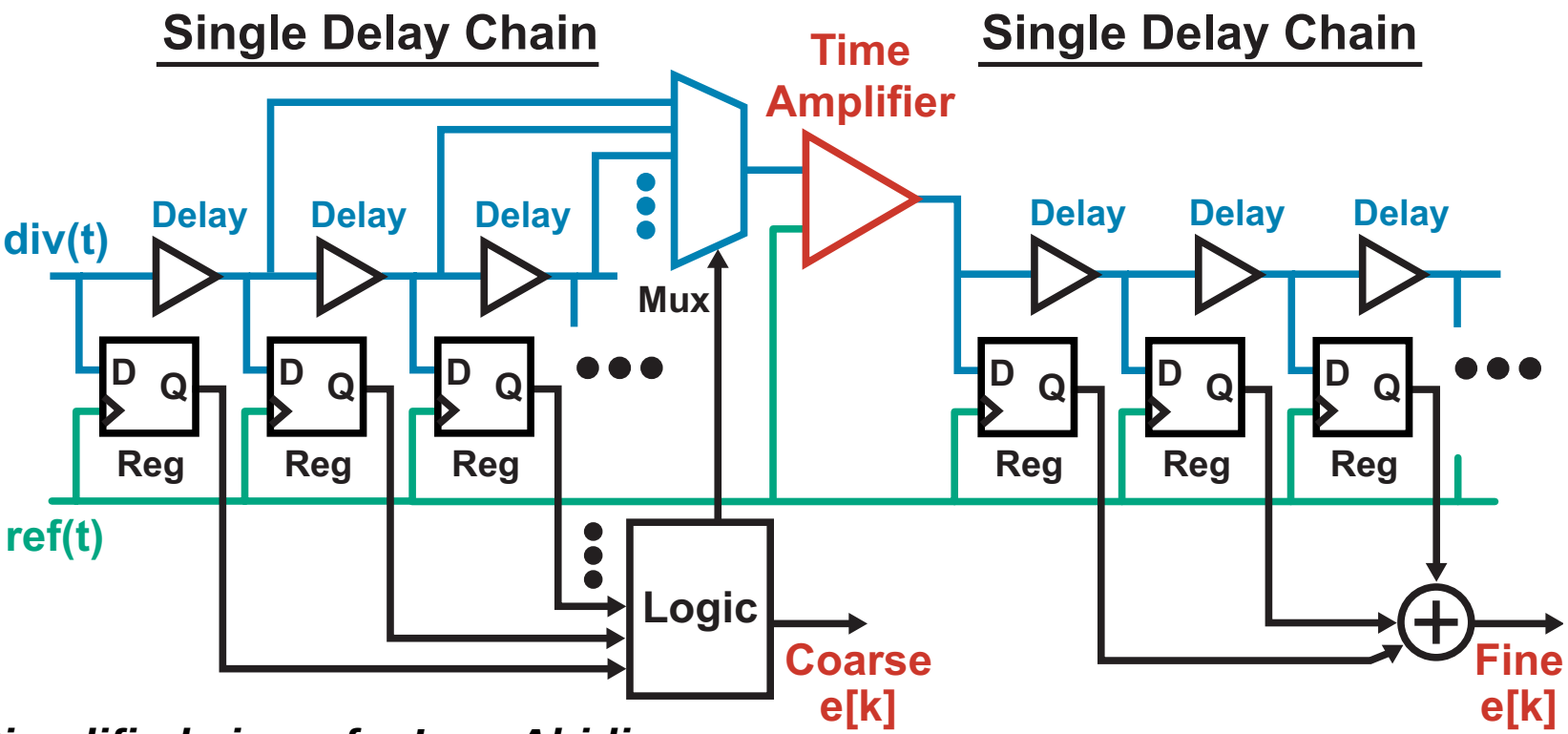


Ramakrishnan, Balsara
VLSID '06

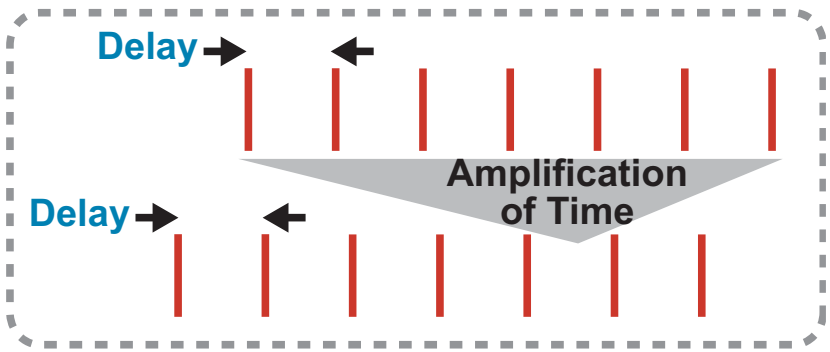


- Single delay chain provides coarse resolution
- (Folded) Vernier provides fine resolution

Two-Step TDC Using Time Amplification

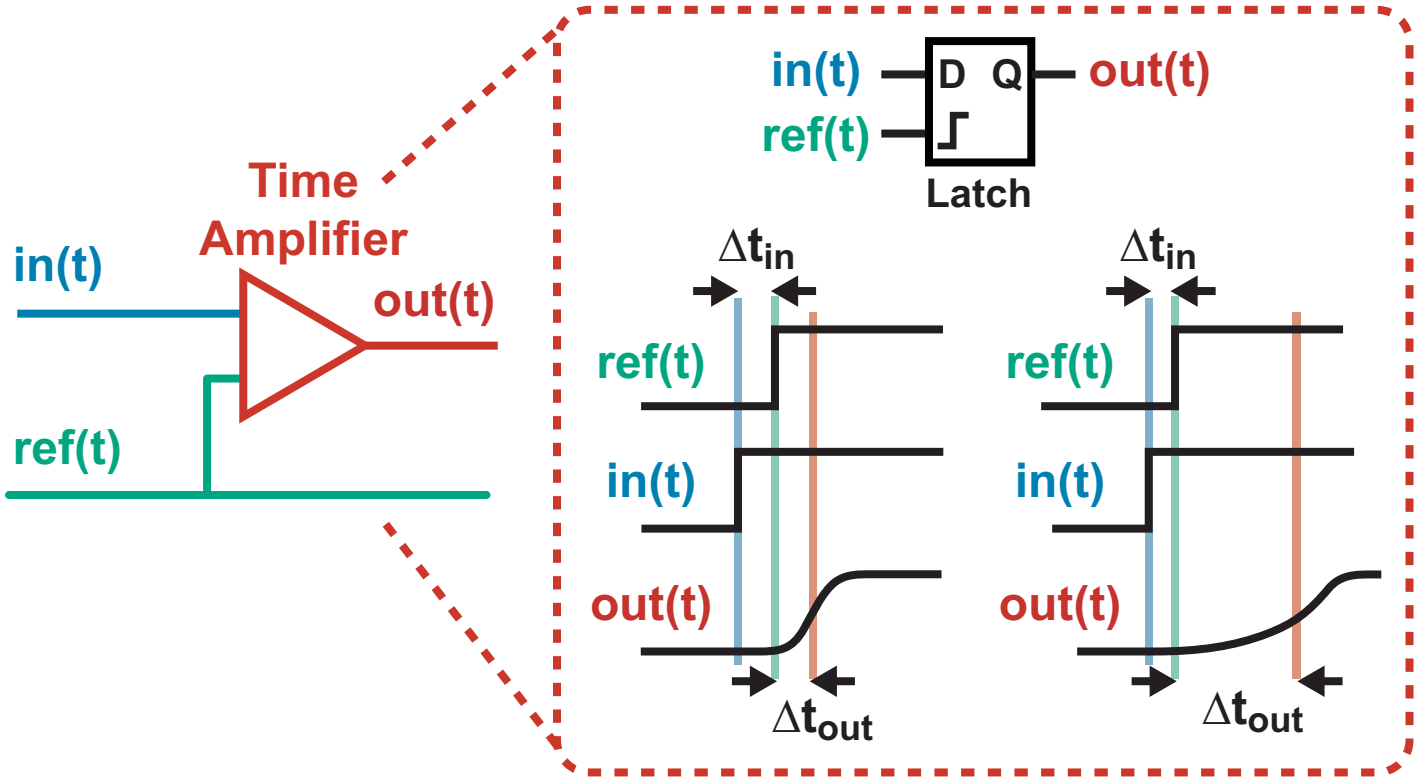


Simplified view of: *Lee, Abidi VLSI 2007*



- Single delay chain provides coarse and fine resolution
- Time amplification is used to improve resolution

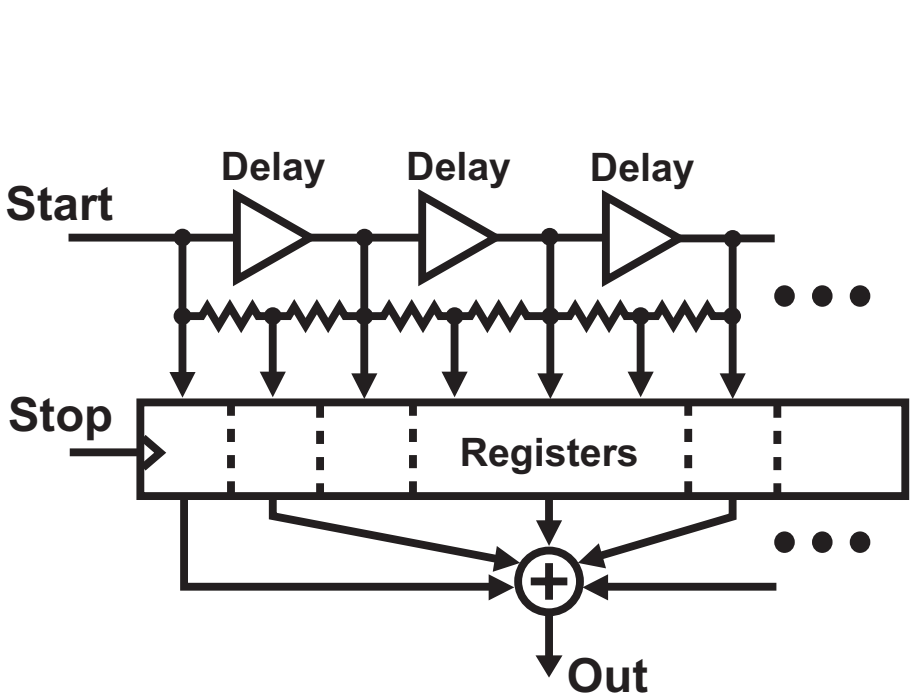
Leveraging Metastability to Create a Time Amplifier



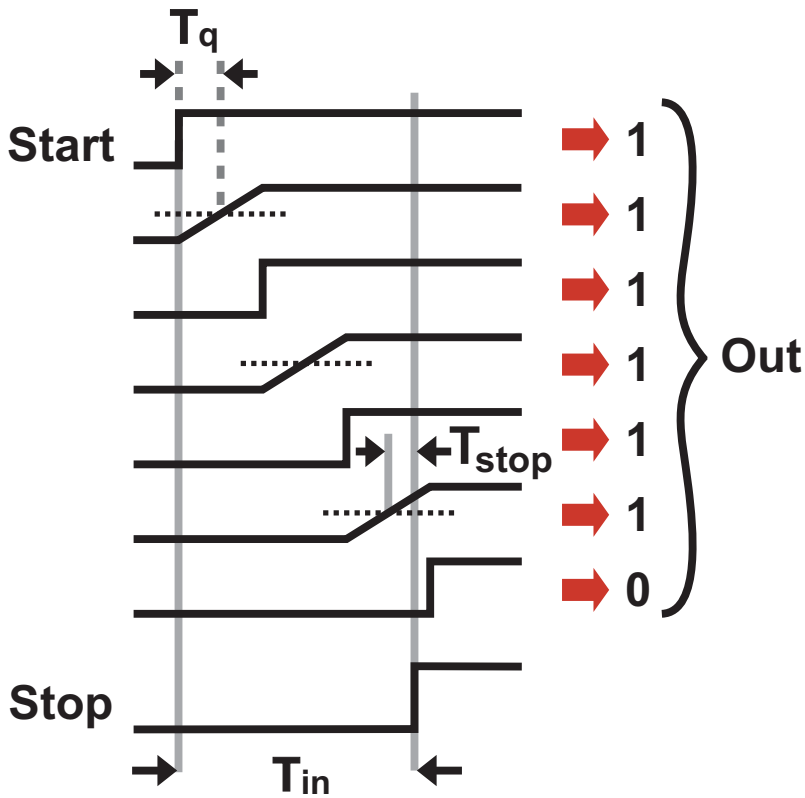
Simplified view of: Abas, et al., Electronic Letters, Nov 2002
(note that actual implementation uses SR latch)

- **Metastability leads to progressively slower output transitions as setup time on latch is encroached upon**
  - **Time difference at input is amplified at output**

Interpolating time-to-digital converter

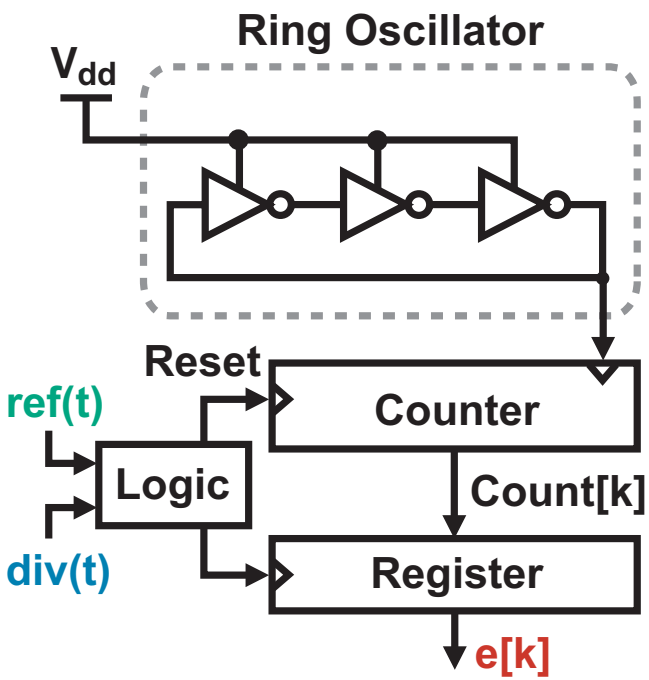


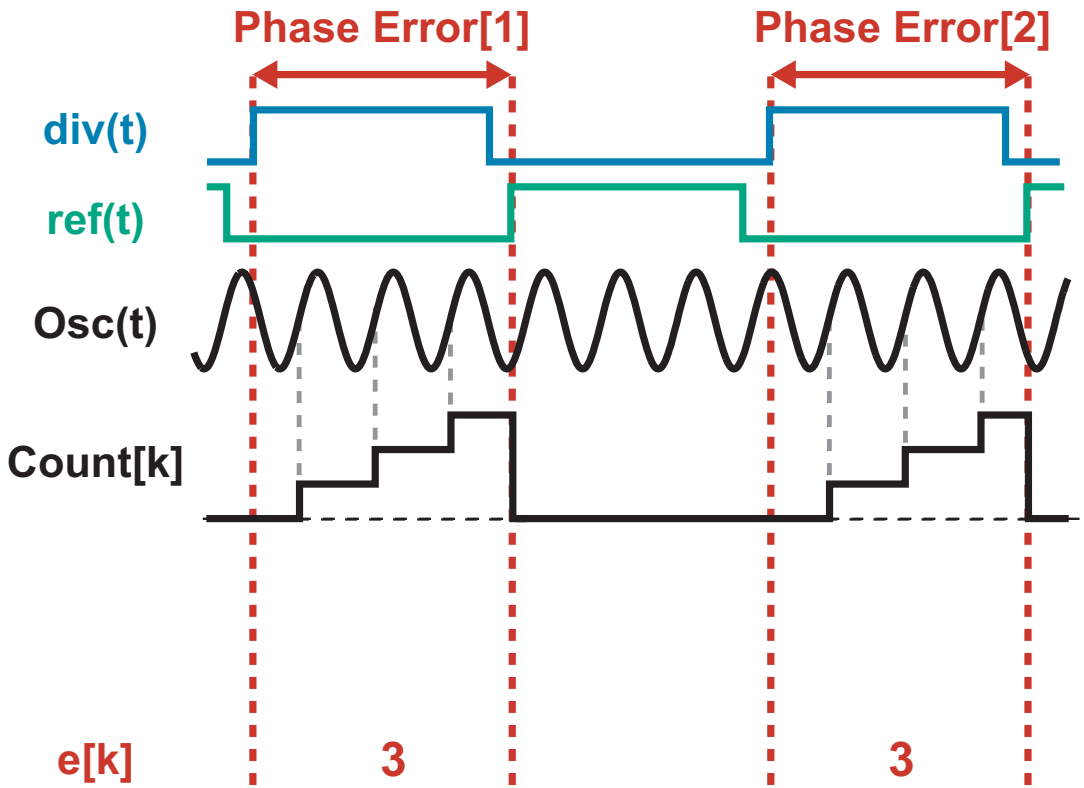
Henzler et al., ISSCC 2008



- Interpolate between edges to achieve fine resolution
- Cyclic approach can also be used for large range

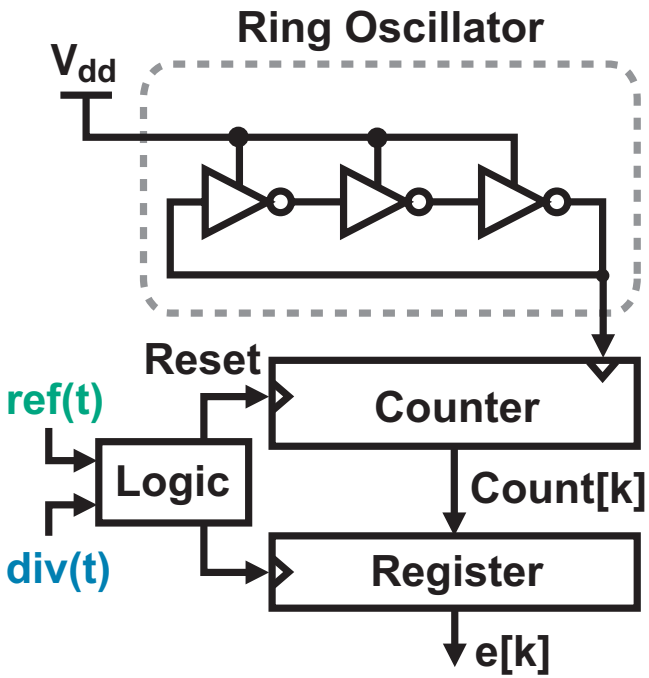
An Oscillator-Based TDC

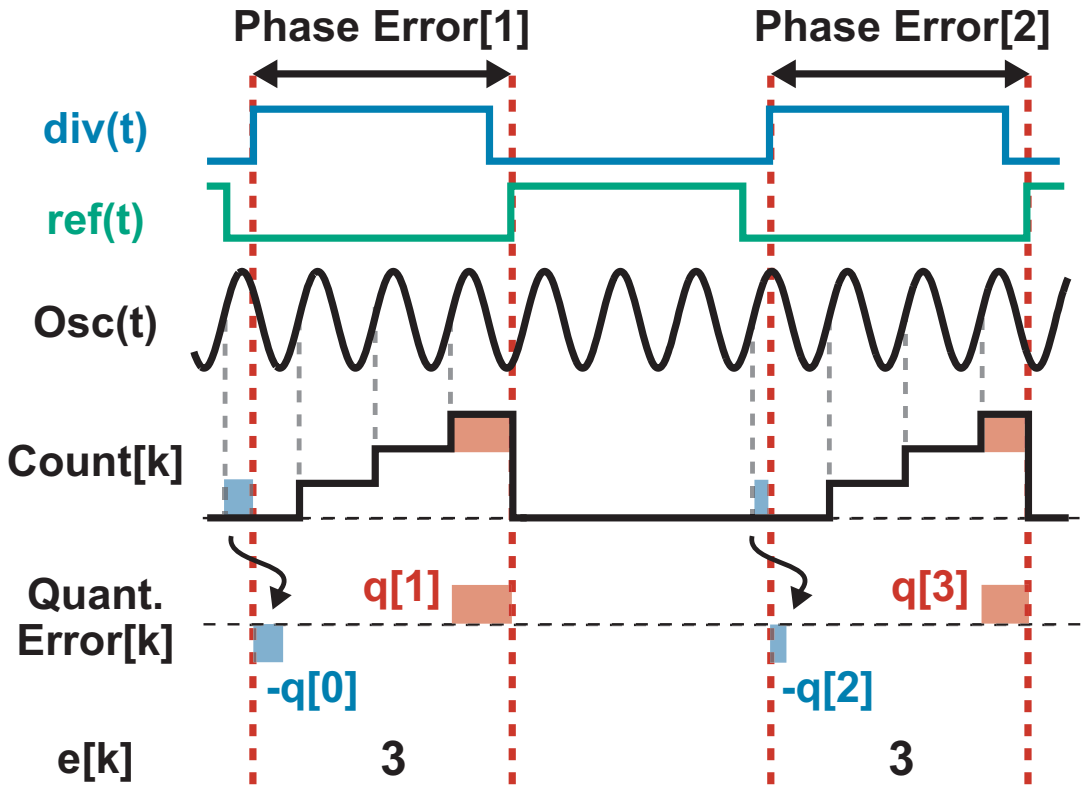




- Output *e[k]* corresponds to the number of oscillator edges that occur during the measurement time window
- Advantages
  - Extremely large range can be achieved with compact area
  - Quantization noise is scrambled across measurements

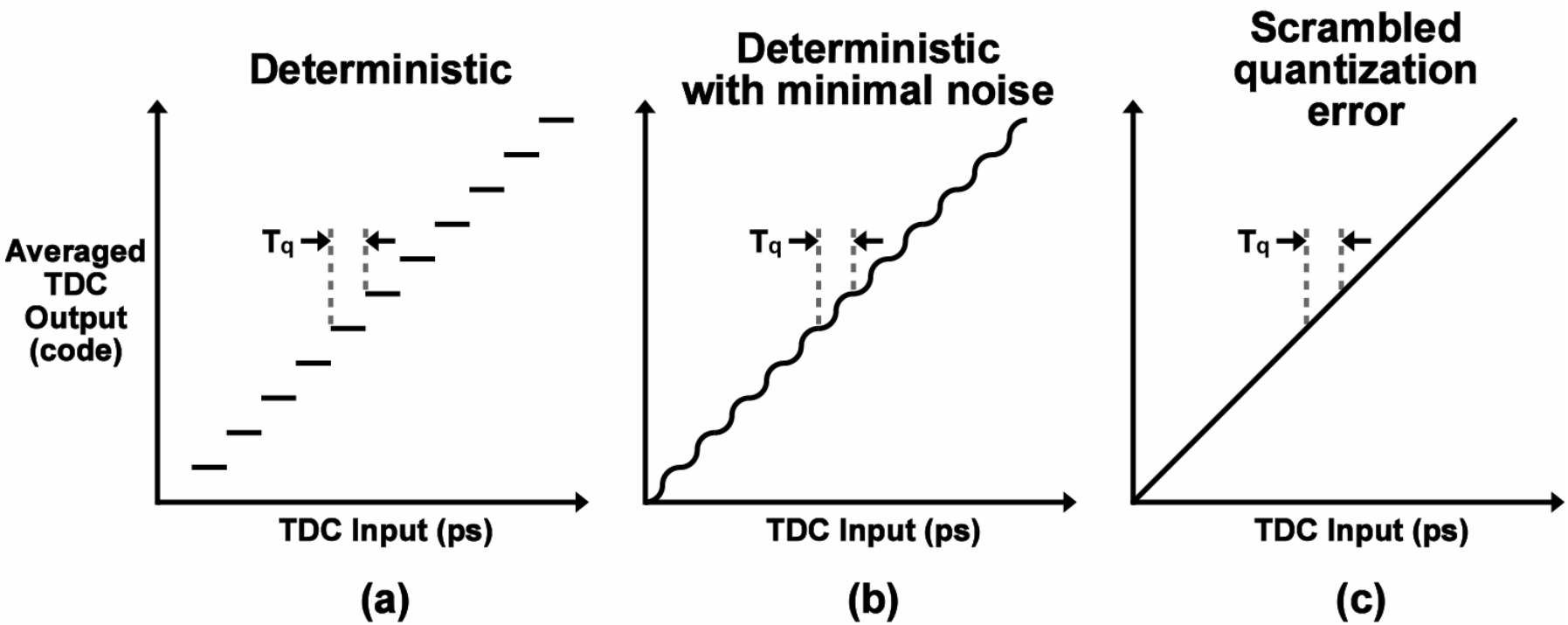
A Closer Look at Quantization Noise Scrambling





- Quantization error occurs at beginning and end of each measurement interval
- As a rough approximation, assume error is uncorrelated between measurements
  - Averaging of measurements improves effective resolution

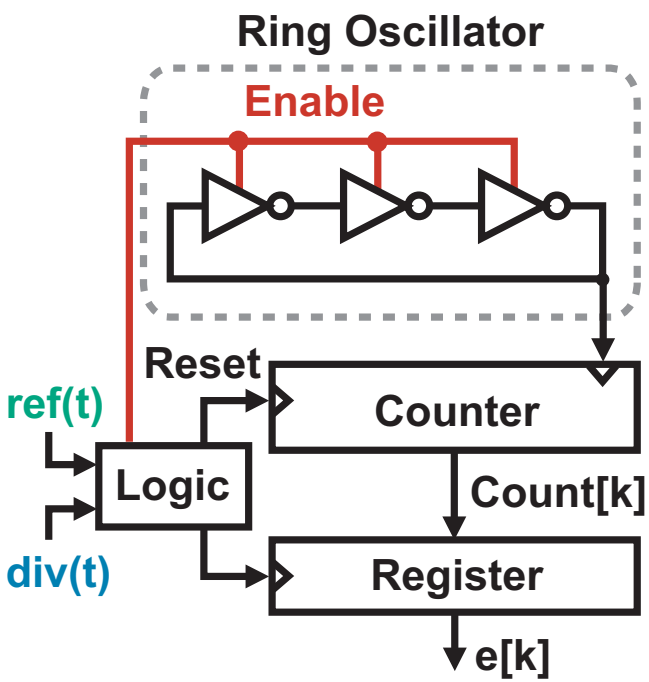
Deterministic quantizer error vs. scrambled error

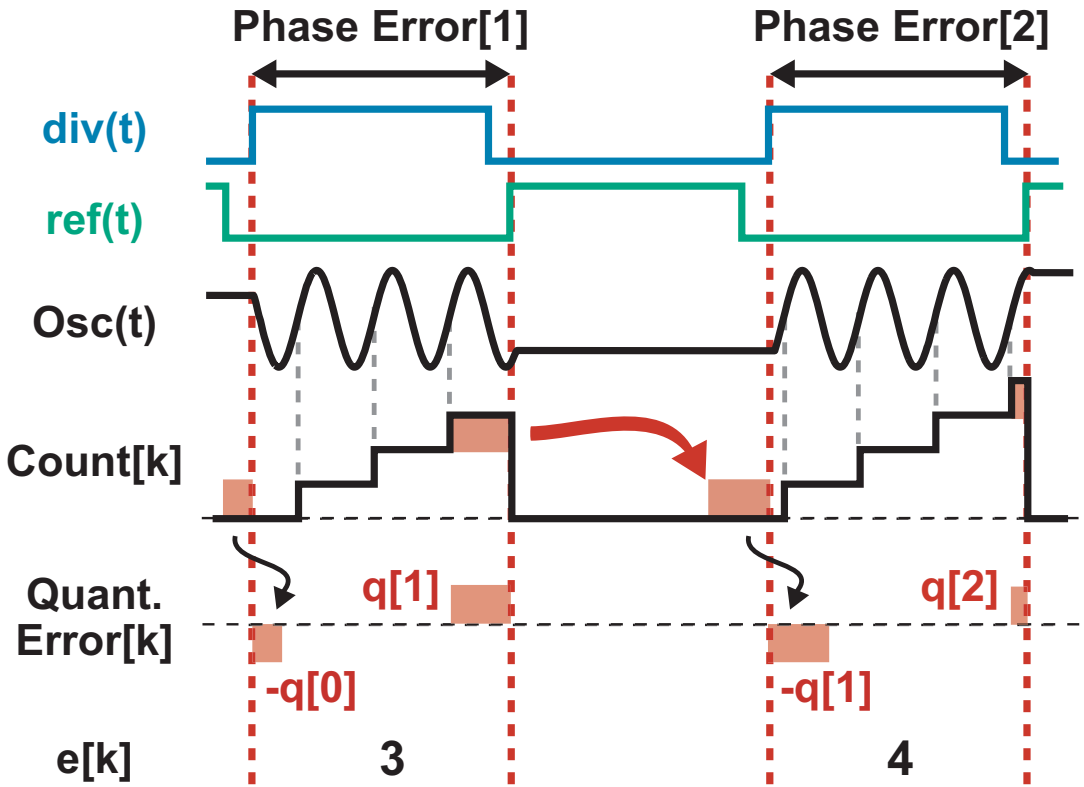


- Deterministic TDC do not provide inherent scrambling
- For oversampling benefit, *TDC error must be scrambled!*
- Some systems provide input scrambling ($\Delta\Sigma$ fractional-N PLL), while some others do not (integer-N PLL)

Proposed GRO TDC Structure

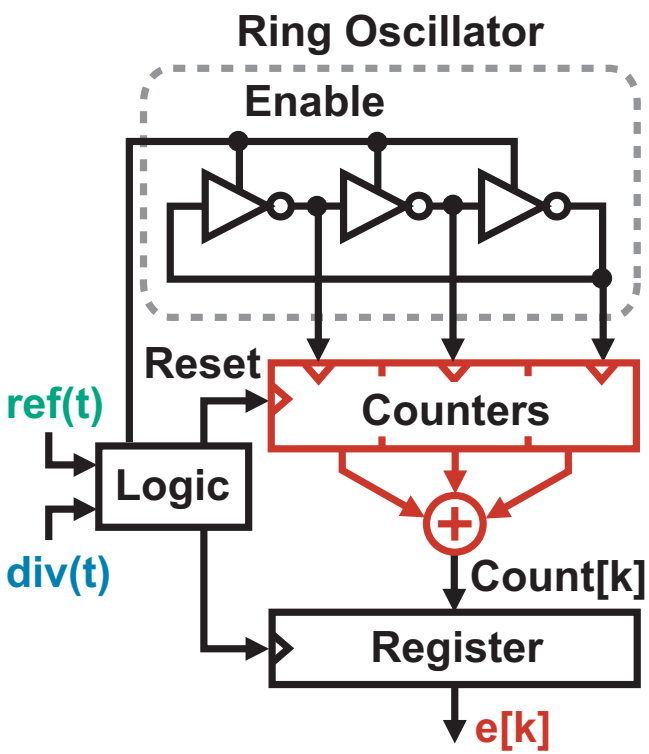
A Gated Ring Oscillator (GRO) TDC



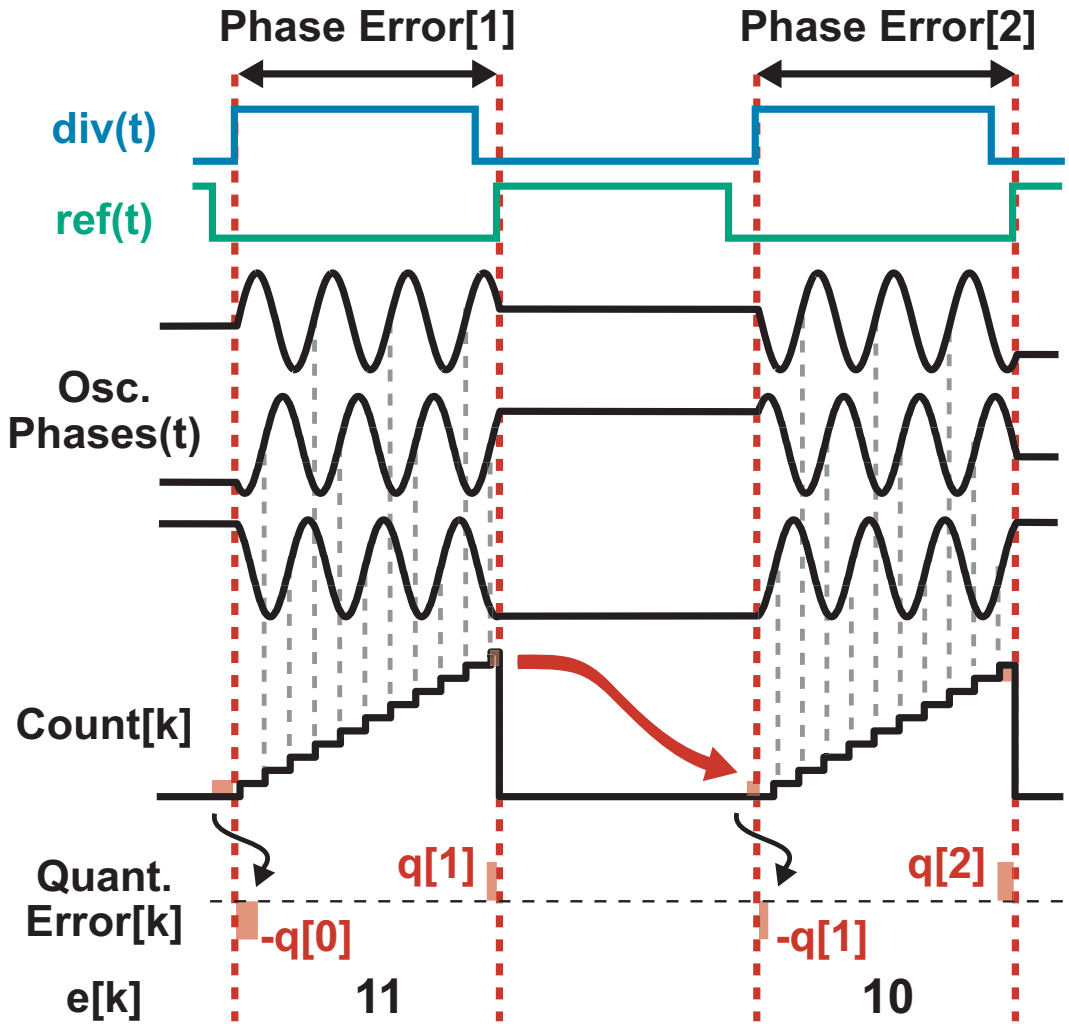


- Enable ring oscillator only during measurement intervals
  - Hold the state of the oscillator between measurements
- Quantization error becomes first order noise shaped!
  - *e[k] = Phase Error[k] + q[k] – q[k-1]*
  - Averaging dramatically improves resolution!

Improve Resolution By Using All Oscillator Phases

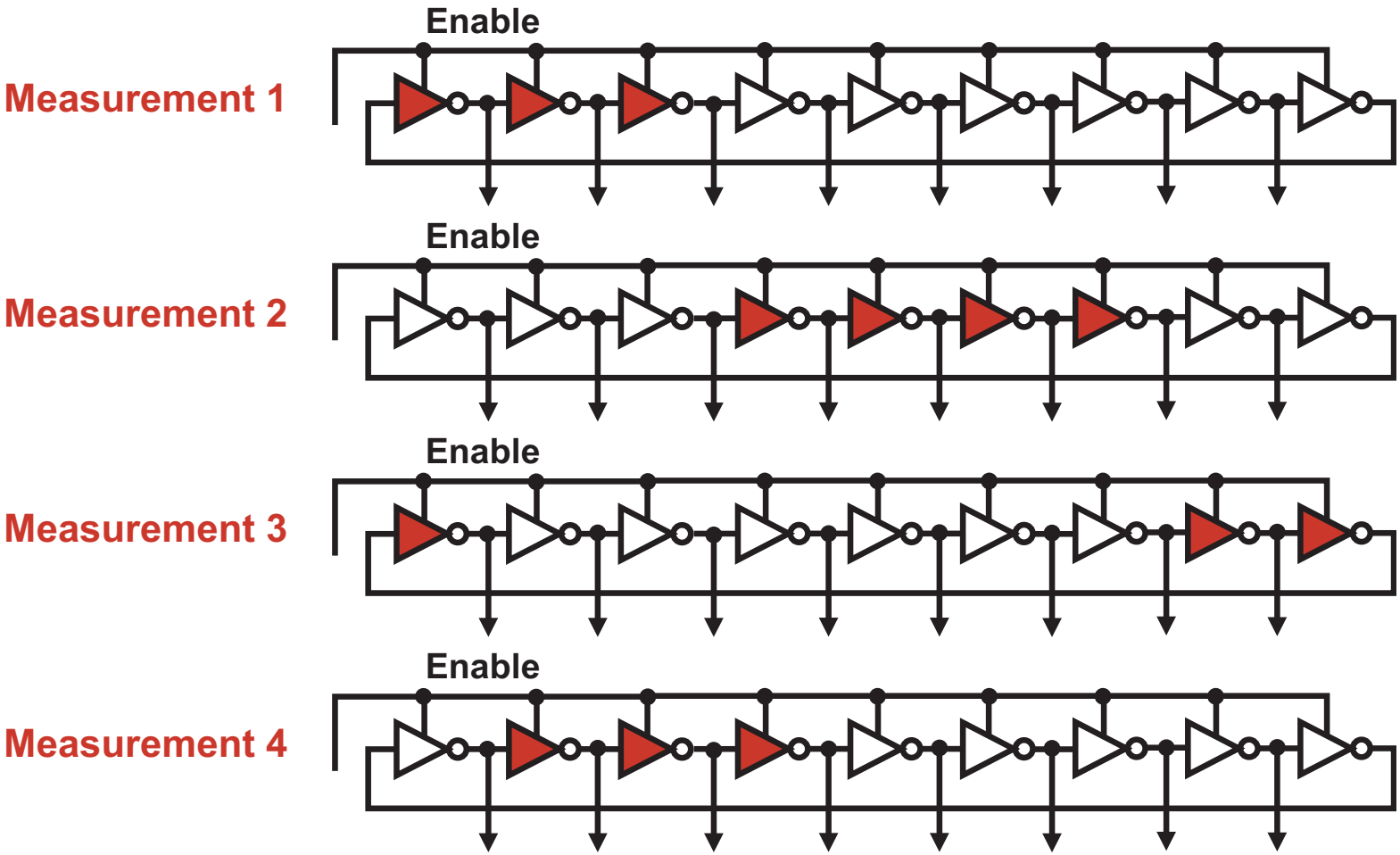


Helal, Straayer, Wei, Perrott VLSI 2007



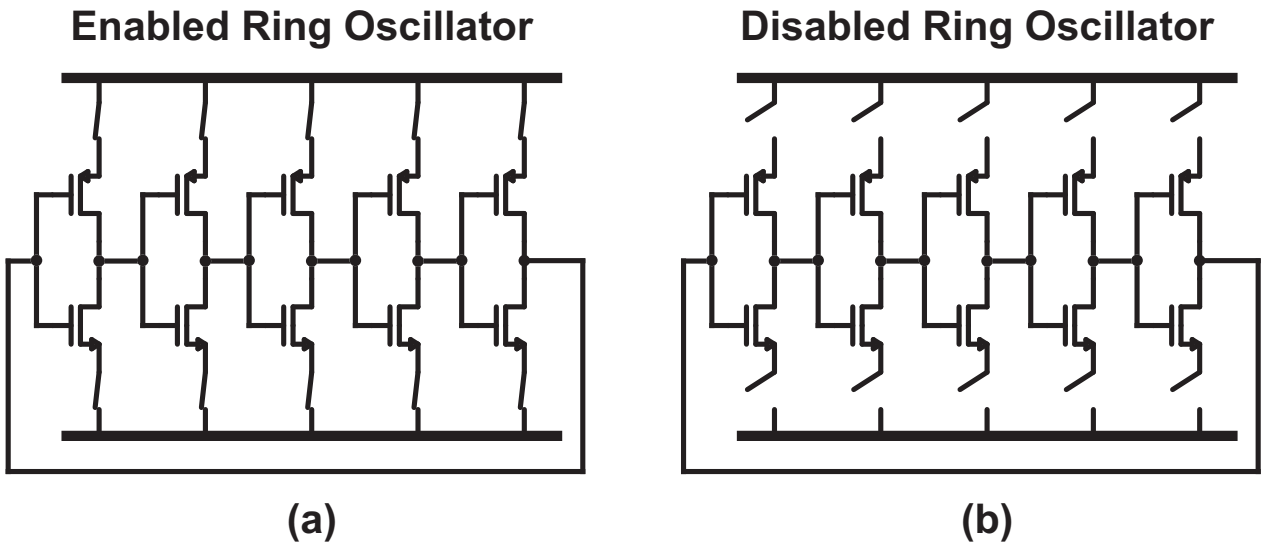
- Raw resolution is set by inverter delay
- Effective resolution is dramatically improved by averaging

GRO TDC Also Shapes Delay Mismatch

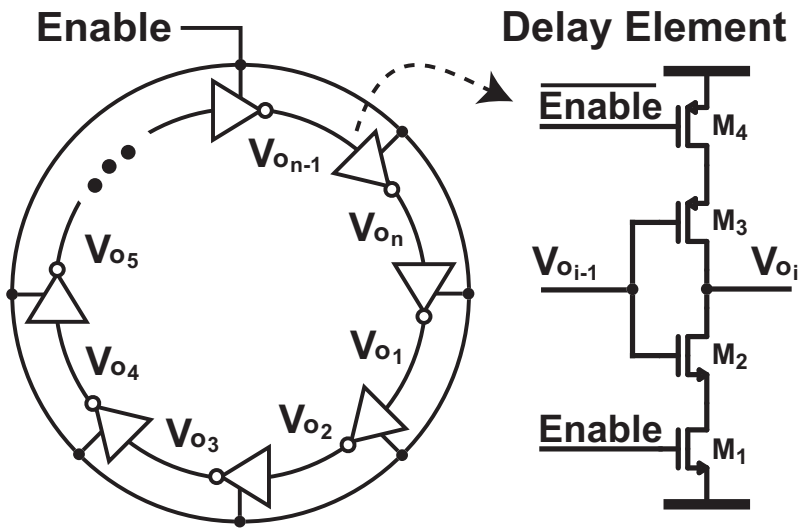


- Barrel shifting occurs through delay elements across different measurements
  - Mismatch between delay elements is first order shaped!

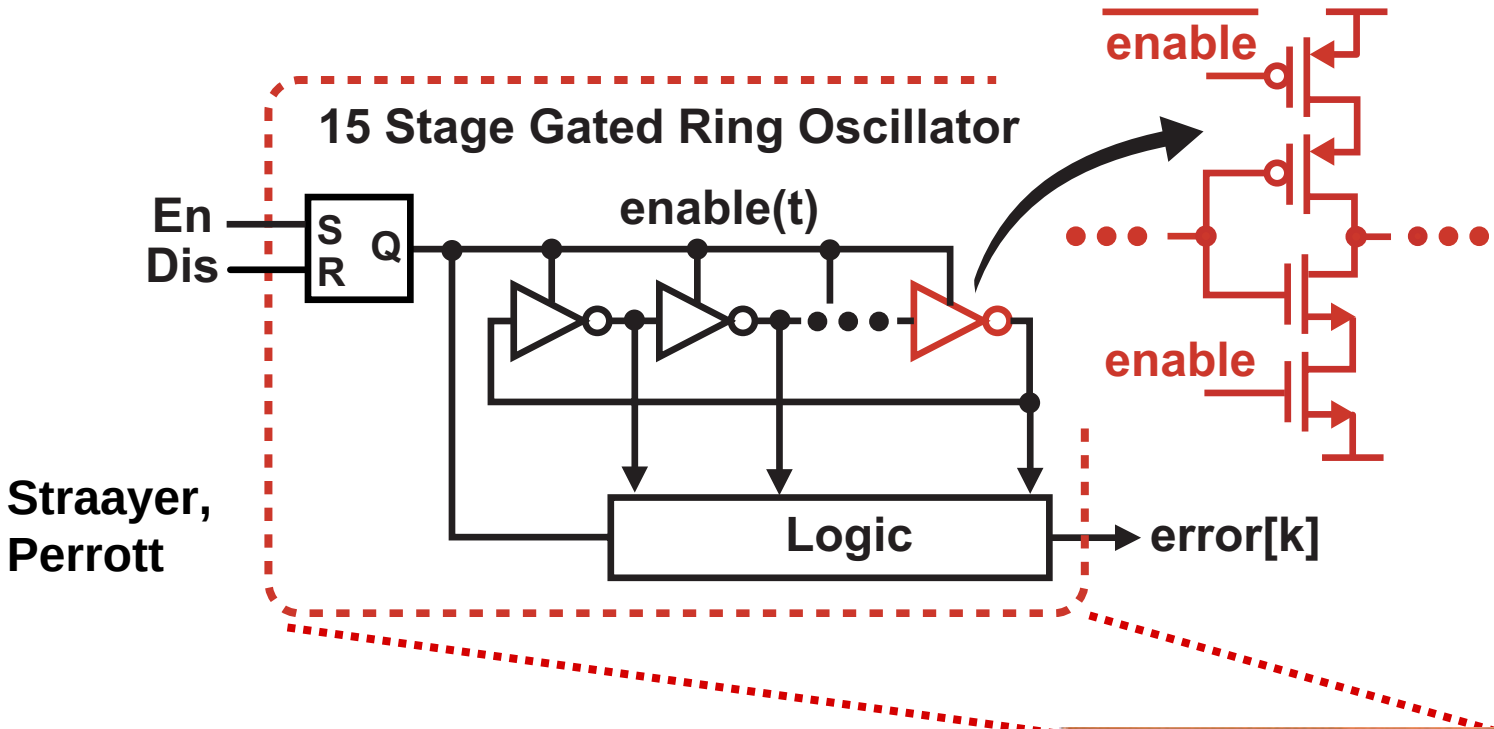
Simple gated ring oscillator inverter-based core



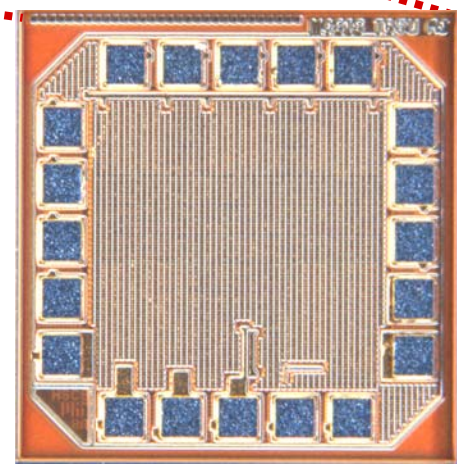
Gate the oscillator by switching the inverter cores to the power supply



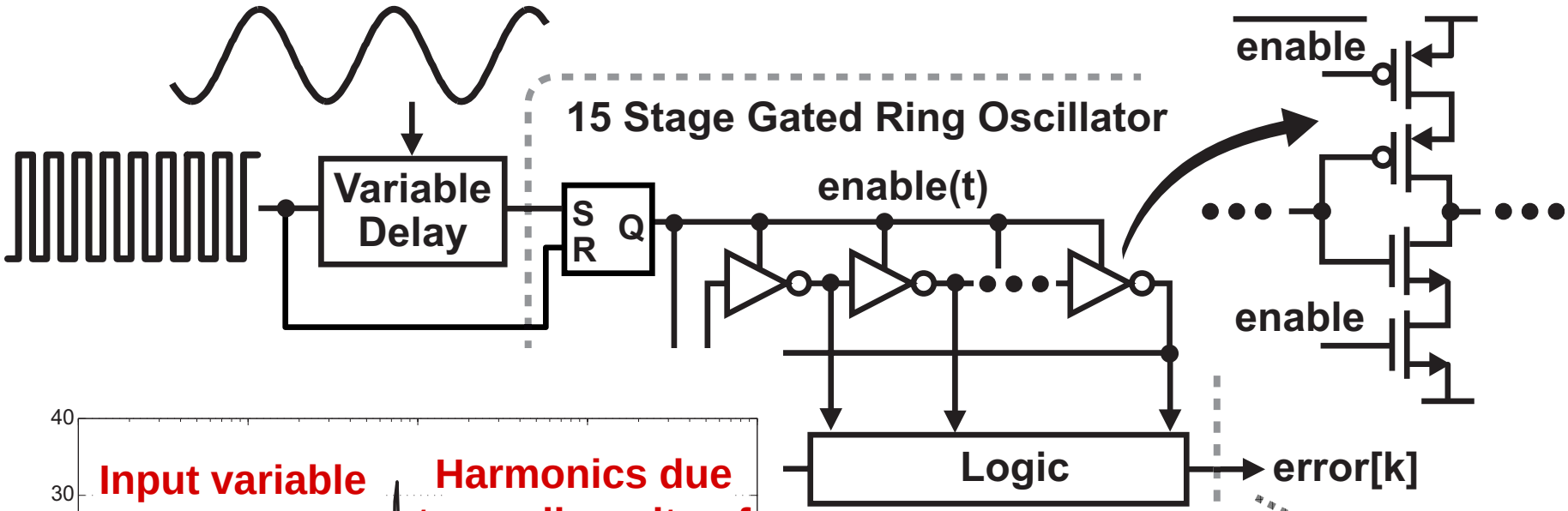
GRO Prototype

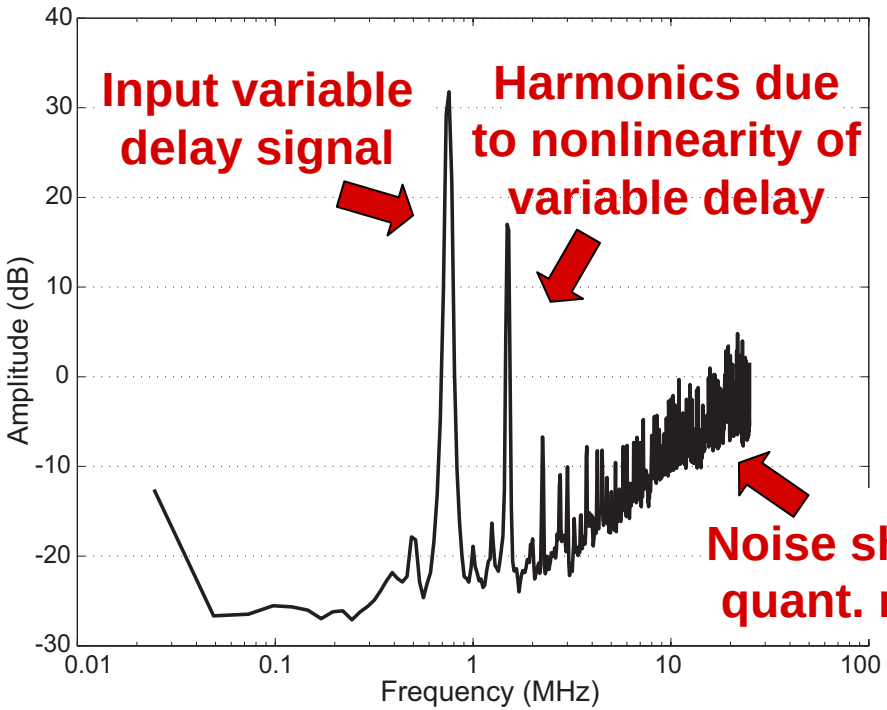


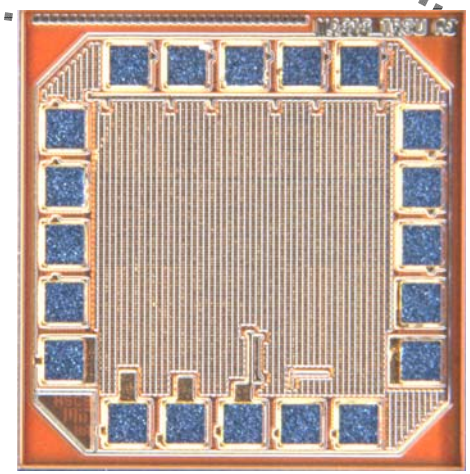
- GRO implemented as a custom 0.13 μm CMOS IC



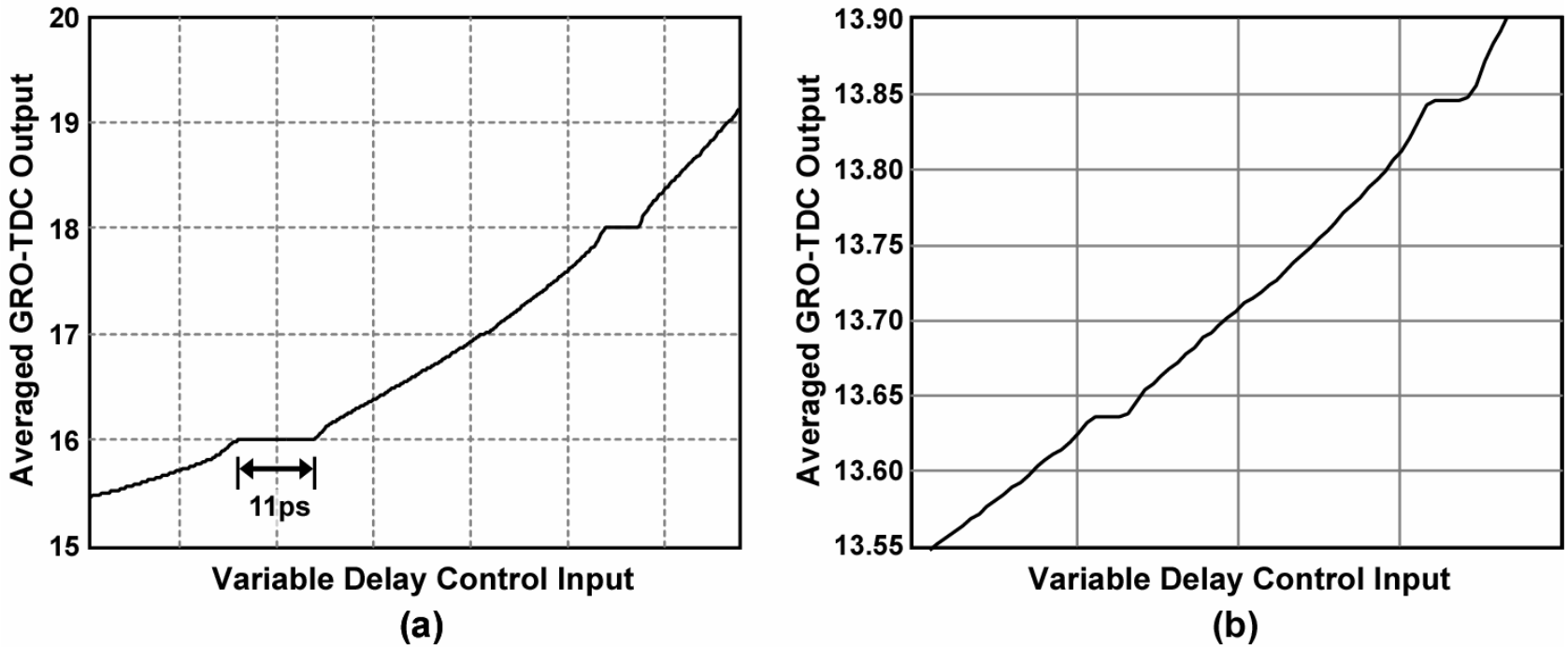
Measured GRO Results Confirm Noise Shaping





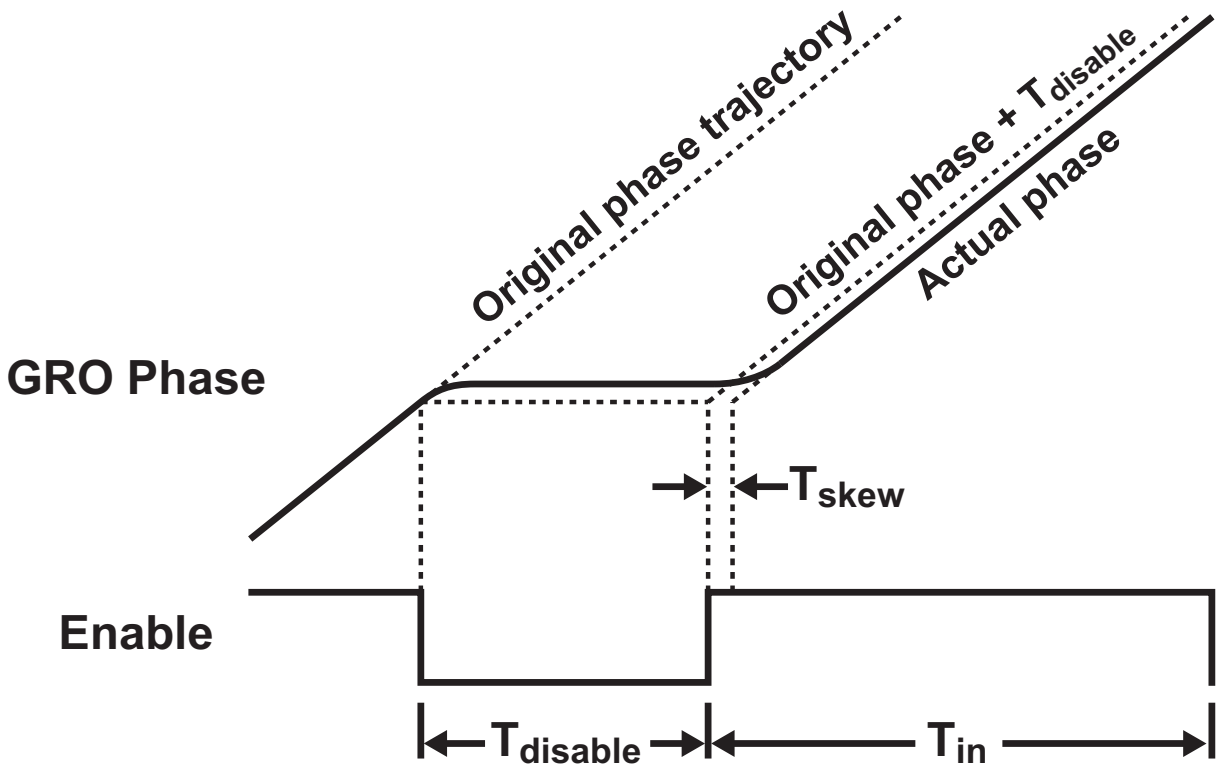


Measured deadzone behavior of inverter-based GRO



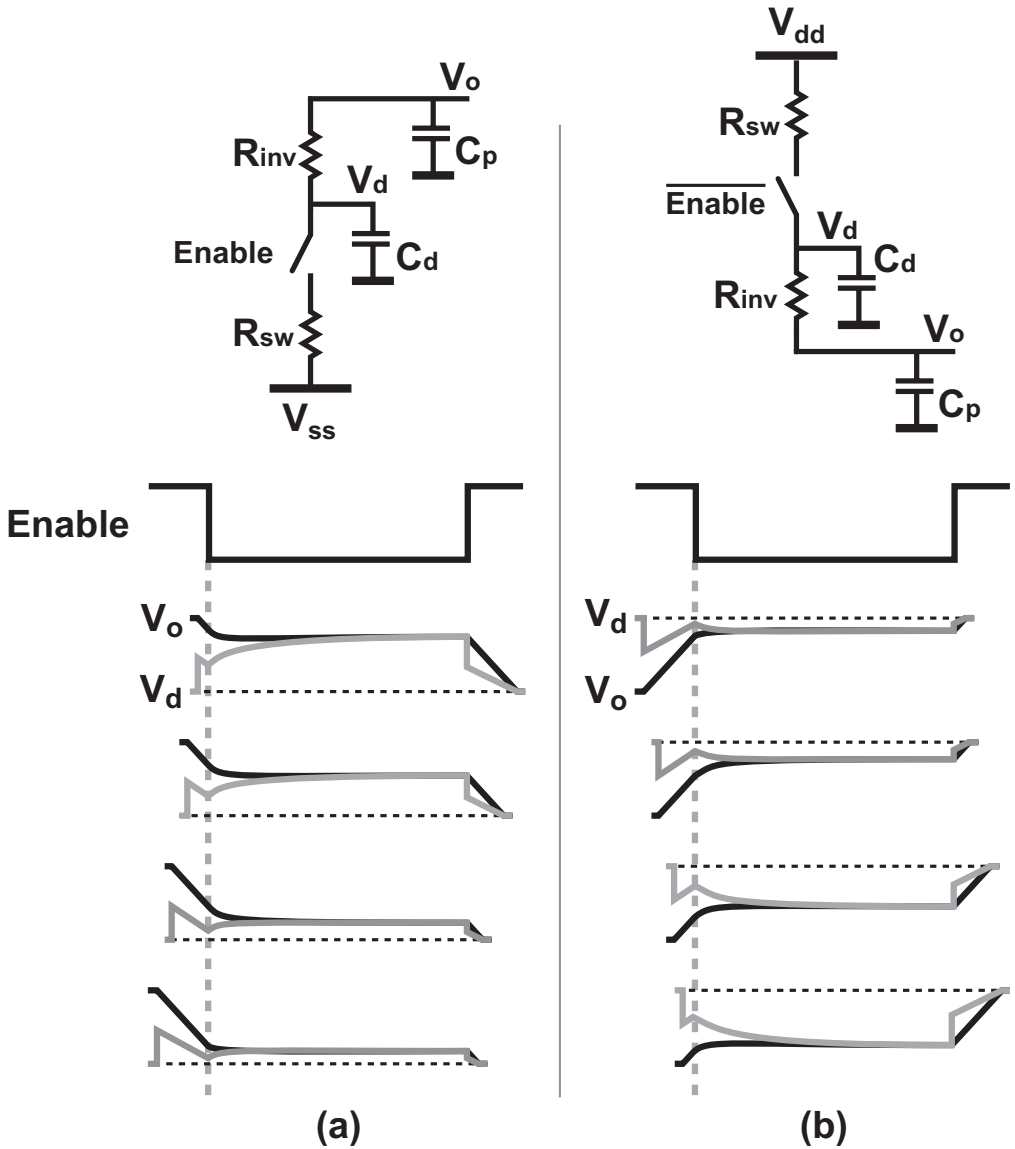
- Deadzones were caused by errors in gating the oscillator
- GRO "injection locked" to an integer ratio of F_S
- Behavior occurred for almost all integer boundaries, and some fractional values as well
- Noise shaping benefit was limited by this gating error

The issue of gating non-idealities…



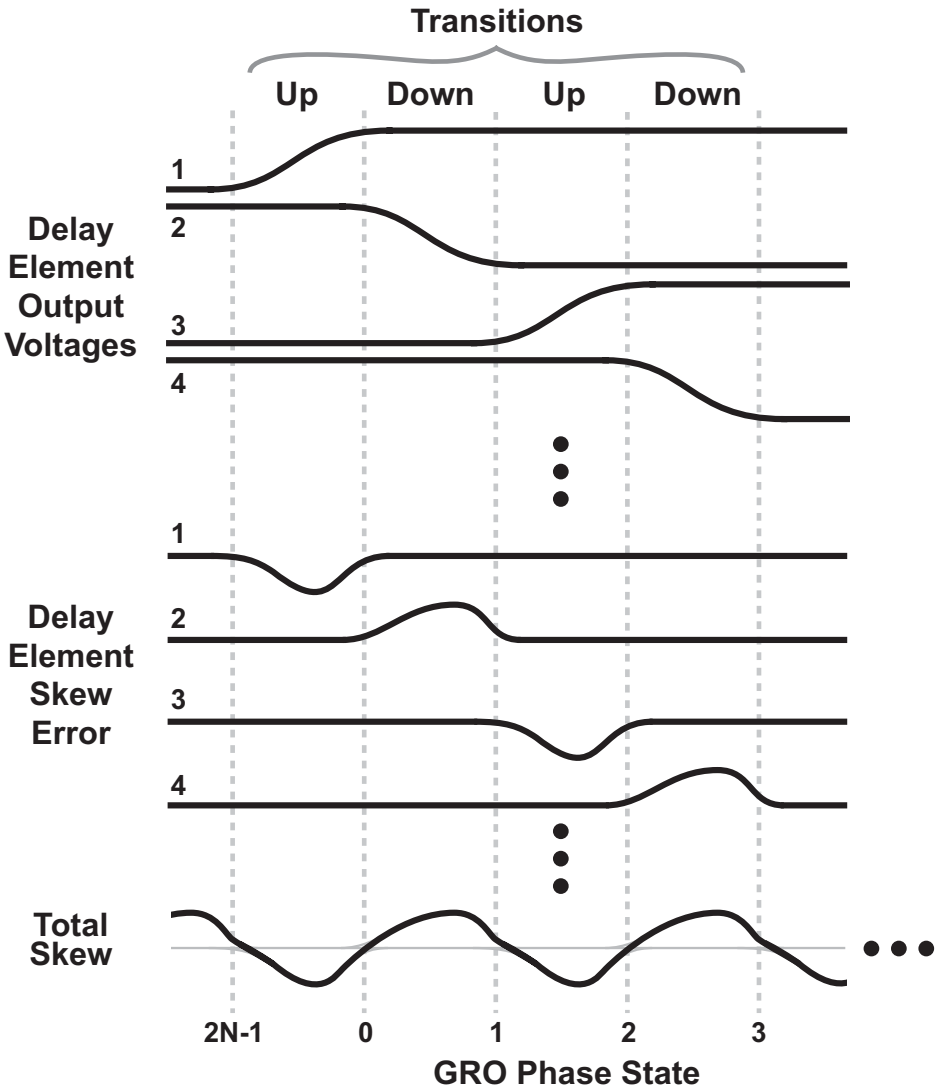
- Oscillator does not stop and start instantly
- Actual phase trajectory deviates from ideal trajectory by a time defined as "T_{skew}"

Interrupted transition causes charge redistribution



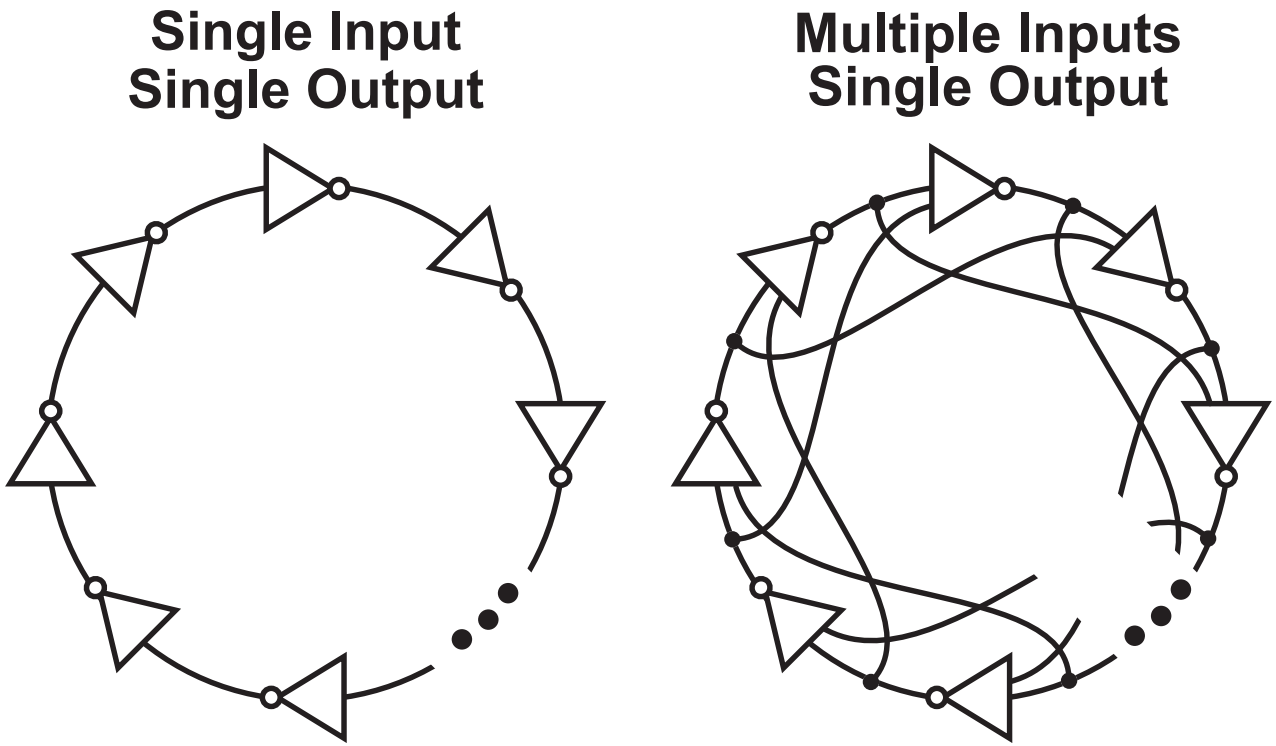
- Charge redistribution depends on when the transition is stopped
- Positive and negative transitions are not perfectly symmetric
- Gating skew (T_{skew}) then depends on GRO phase (θ_{GRO}) when Enable transitions low

Cartoon depicting the error from individual stages



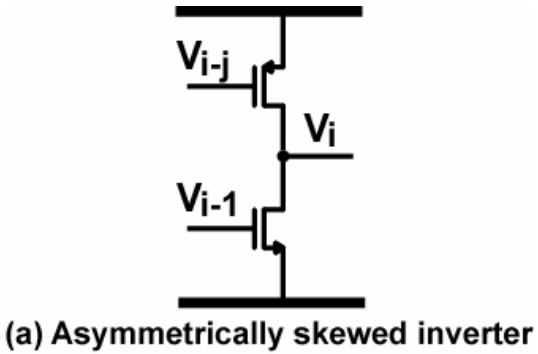
- Only one stage in transition at a time
- T_{skew} is the sum of error from each of the individual stages
- Periodic with $2T_q$ due to positive and negative transition asymmetry

Next Generation GRO: Multi-path oscillator concept



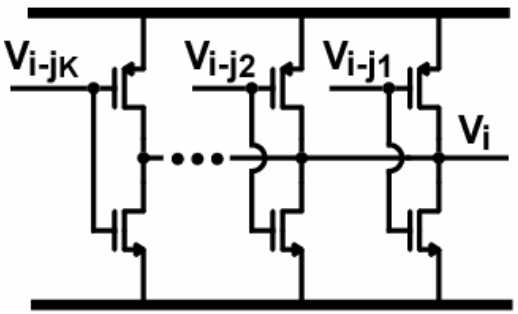
- Use multiple inputs for each delay element instead of one
- Allow each stage to optimally begin its transition based on information from the entire GRO phase state
- Key design issue is to ensure primary mode of oscillation

Multi-path inverter core



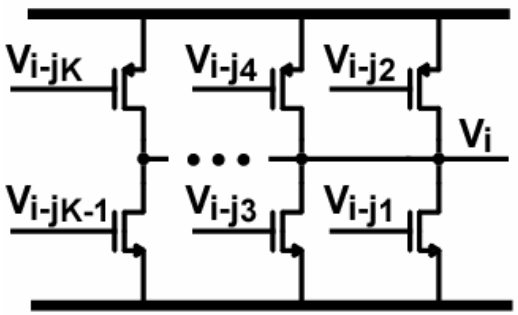
(a) Asymmetrically skewed inverter

Lee, Kim, Lee
JSSC 1997

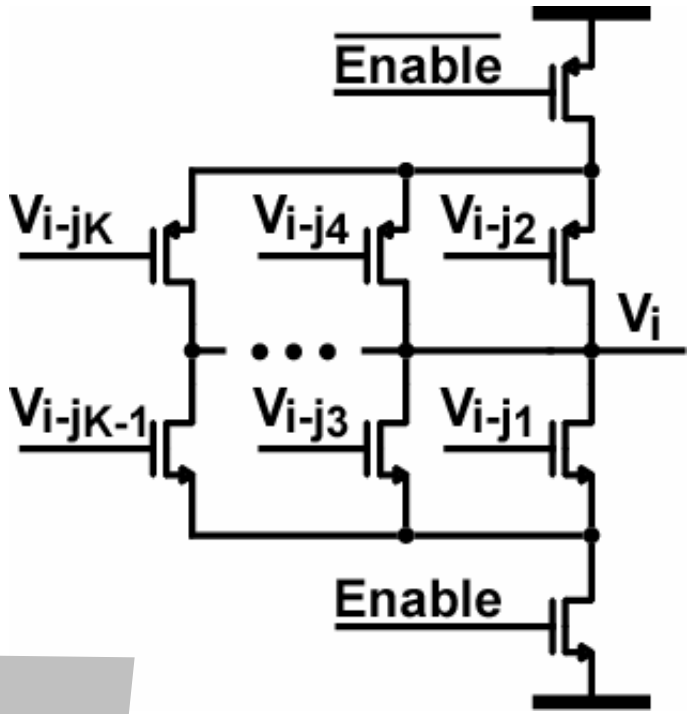


(b) Multiple skewed inverters

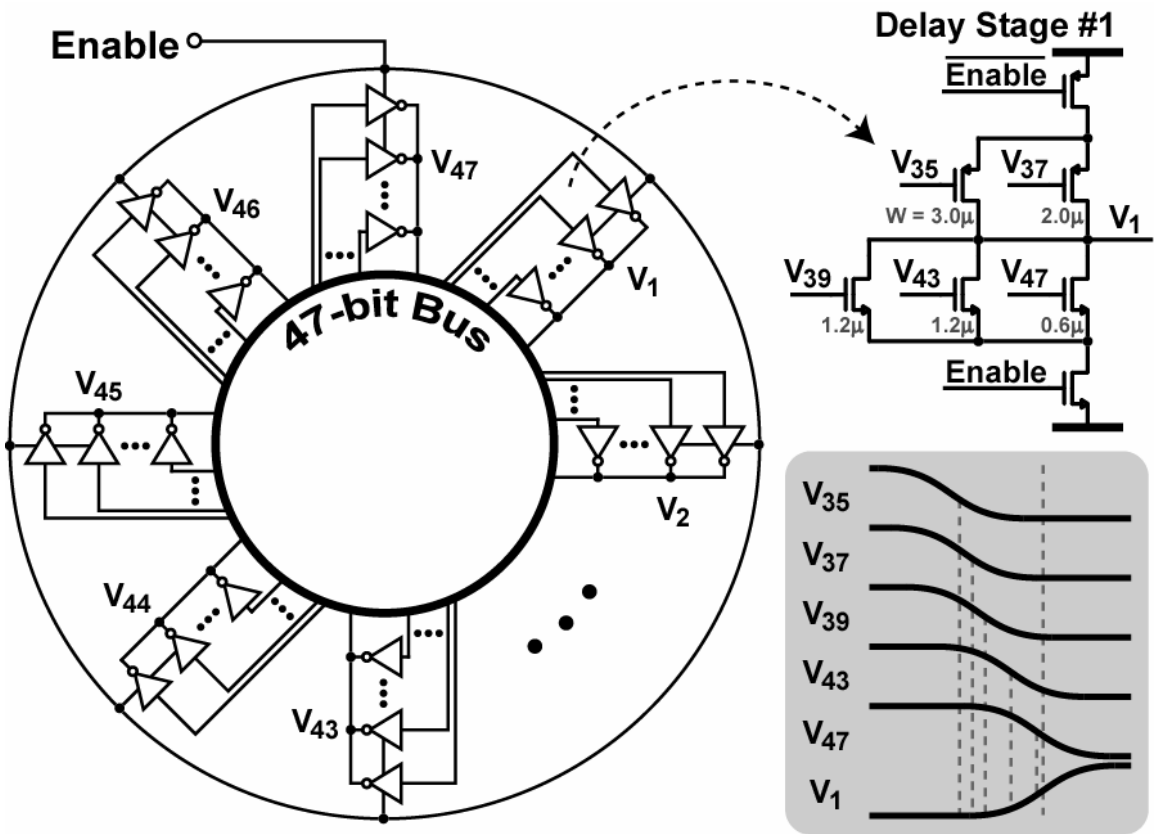
Mohan, et. al.,
CICC 2005



(c) Unrestricted connections



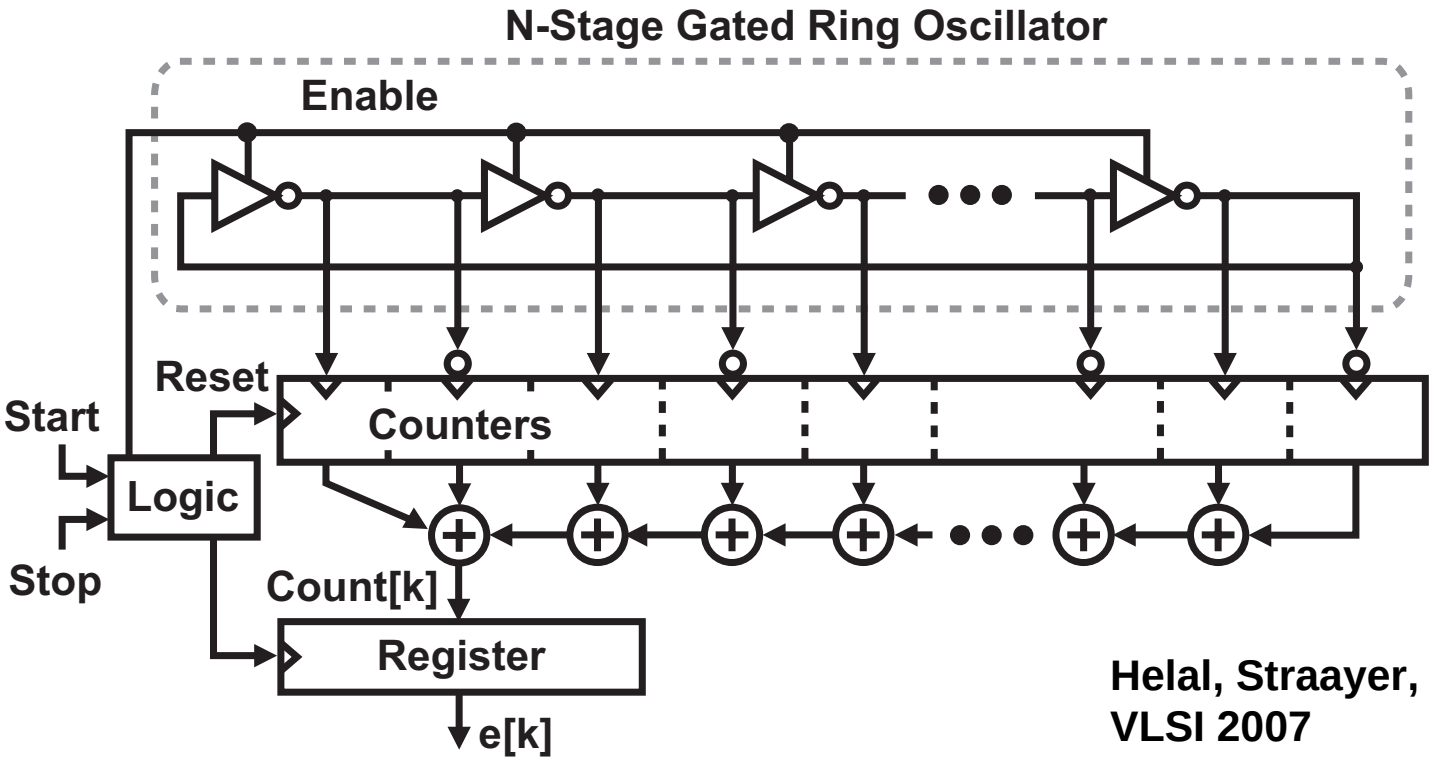
Proposed multi-path gated ring oscillator



Hsu, Straayer, Perrott
ISSCC 2008

- Oscillation frequency near 2GHz with 47 stages…
- Reduces effective delay per stage by a factor of 5-6!
- Represents a factor of 2-3 improvement compared to previous multi-path oscillators

A simple measurement approach…

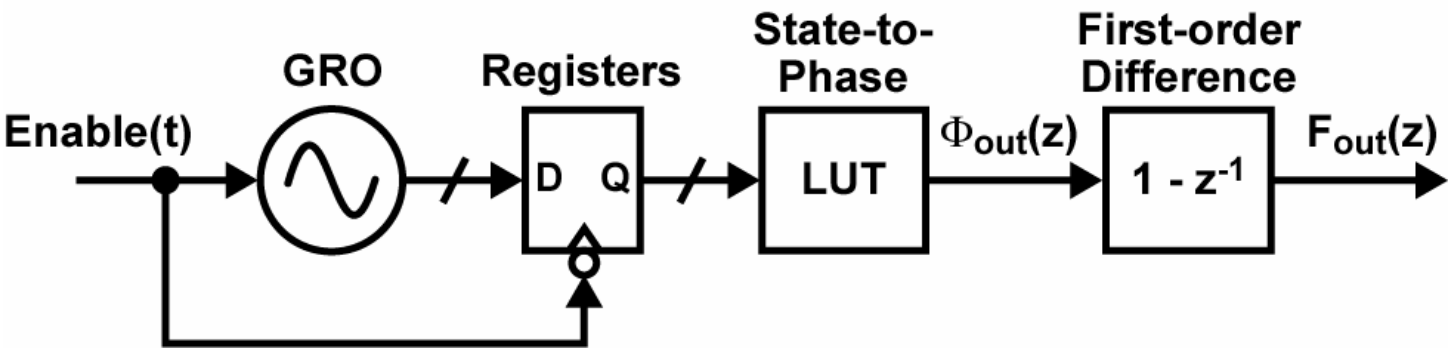


Helal, Straayer, Perrott
VLSI 2007

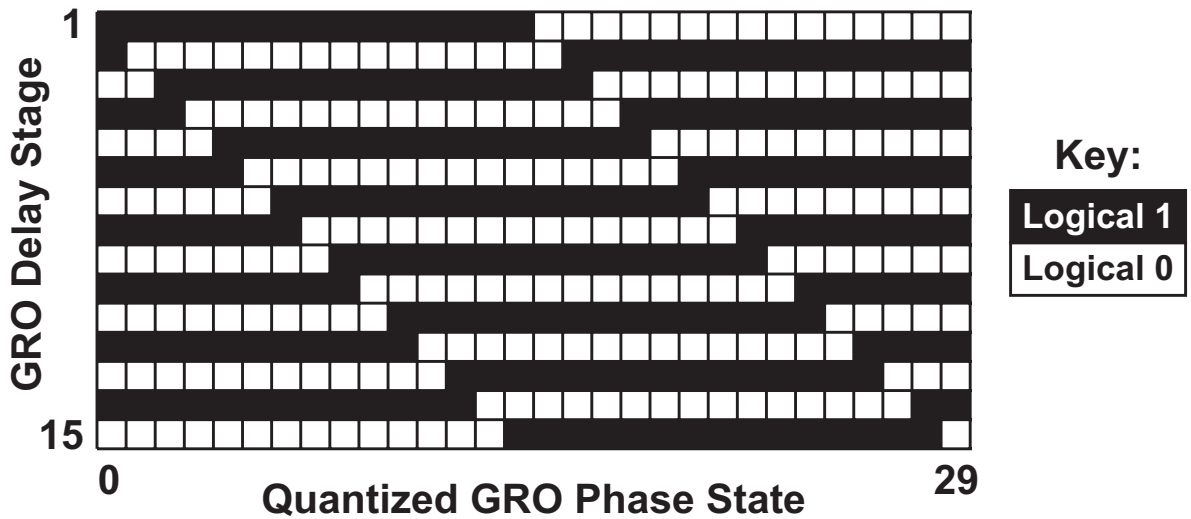
- 2 counters per stage * 47 stages = 94 counters each at 2GHz
- Power consumption for these counters is unreasonable

Need a more efficient way to measure the multi-path GRO

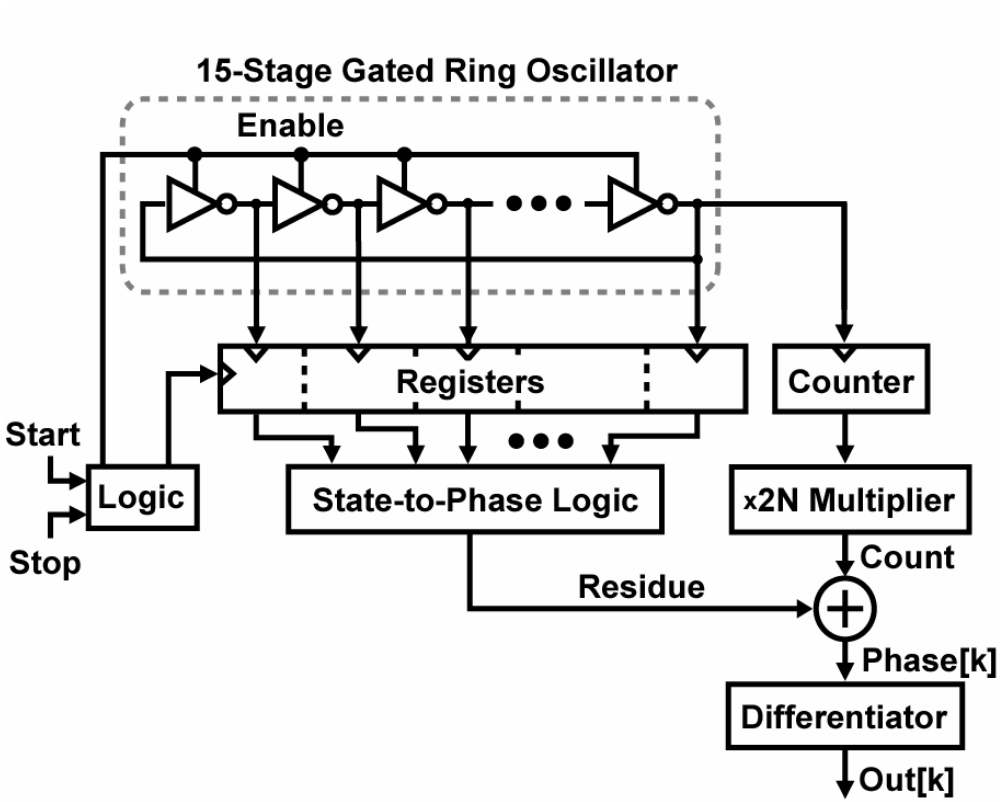
Phase-based measurement for a simple GRO

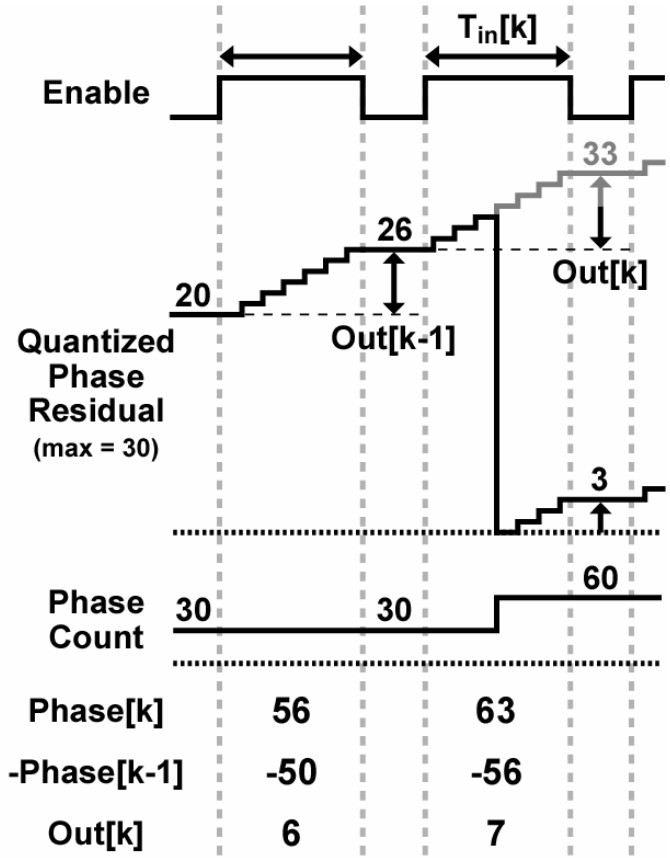


- Simple logic provides map from GRO output state to phase
- Transition sequence is predictable, unambiguous



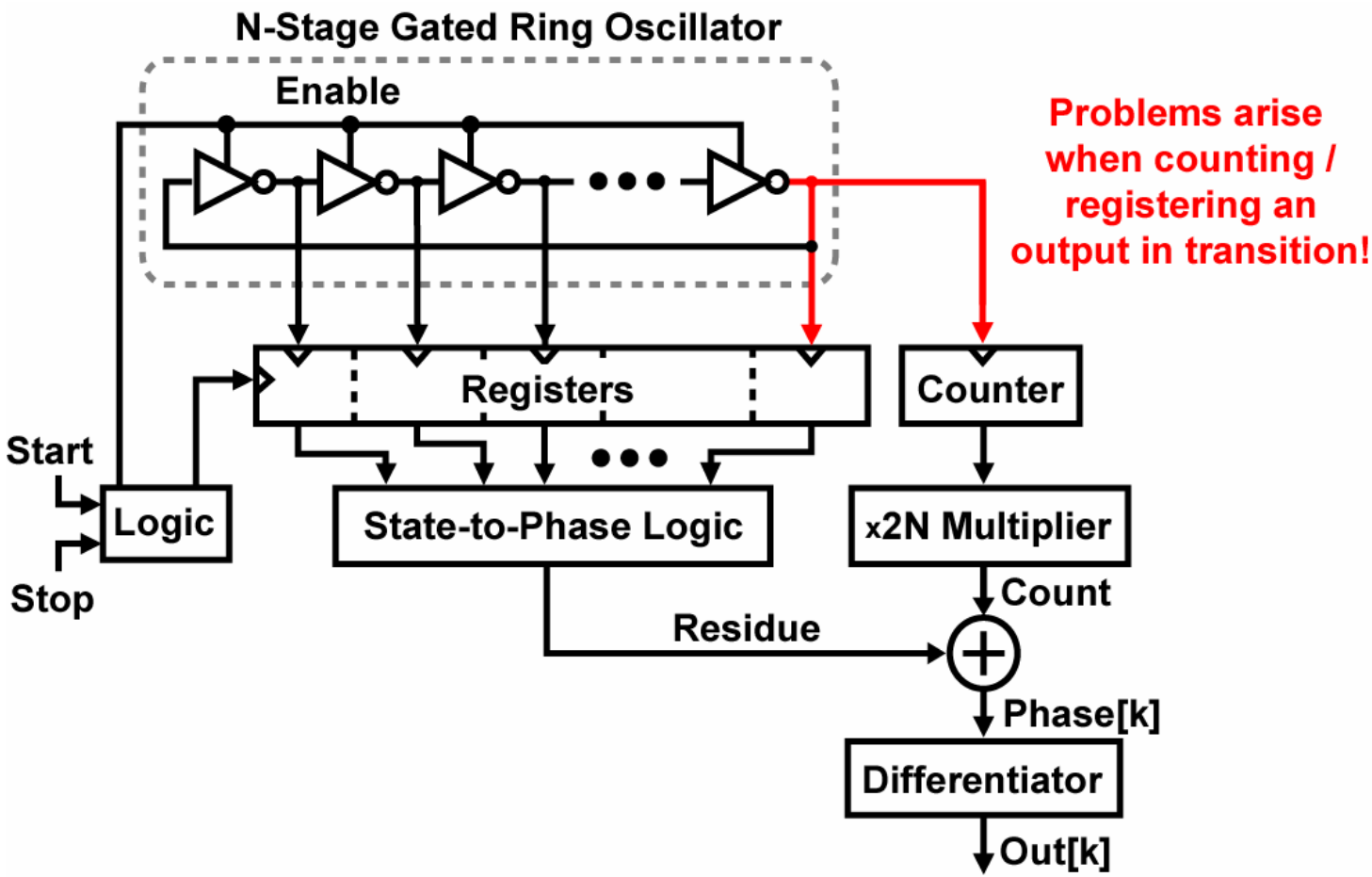
Accounting for phase wrapping…





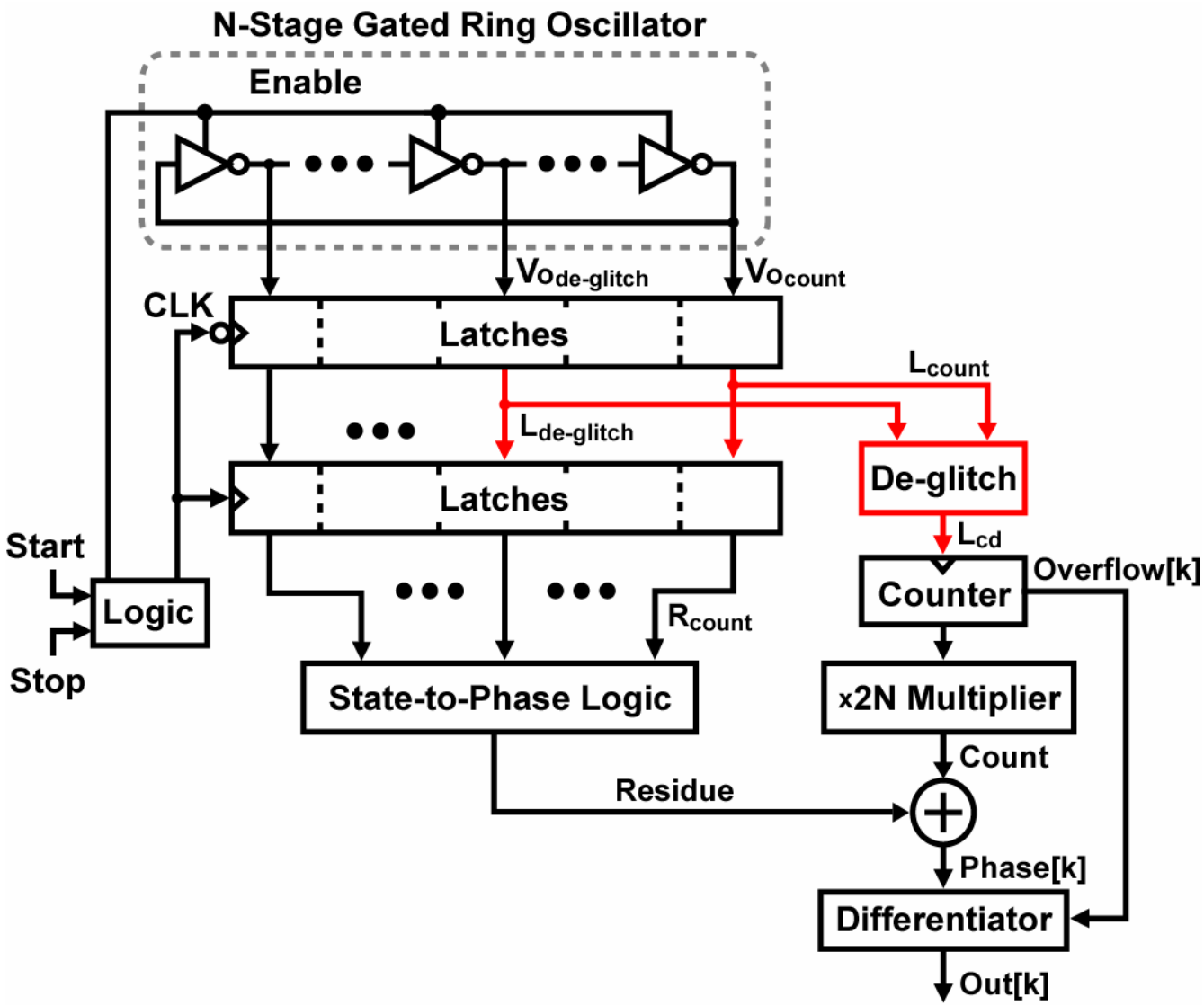
- Calculate phase from:
  - A single counter for coarse phase information
  - GRO output state for fine phase residual
- 1 counter and N registers → much more efficient

Accuracy considerations…

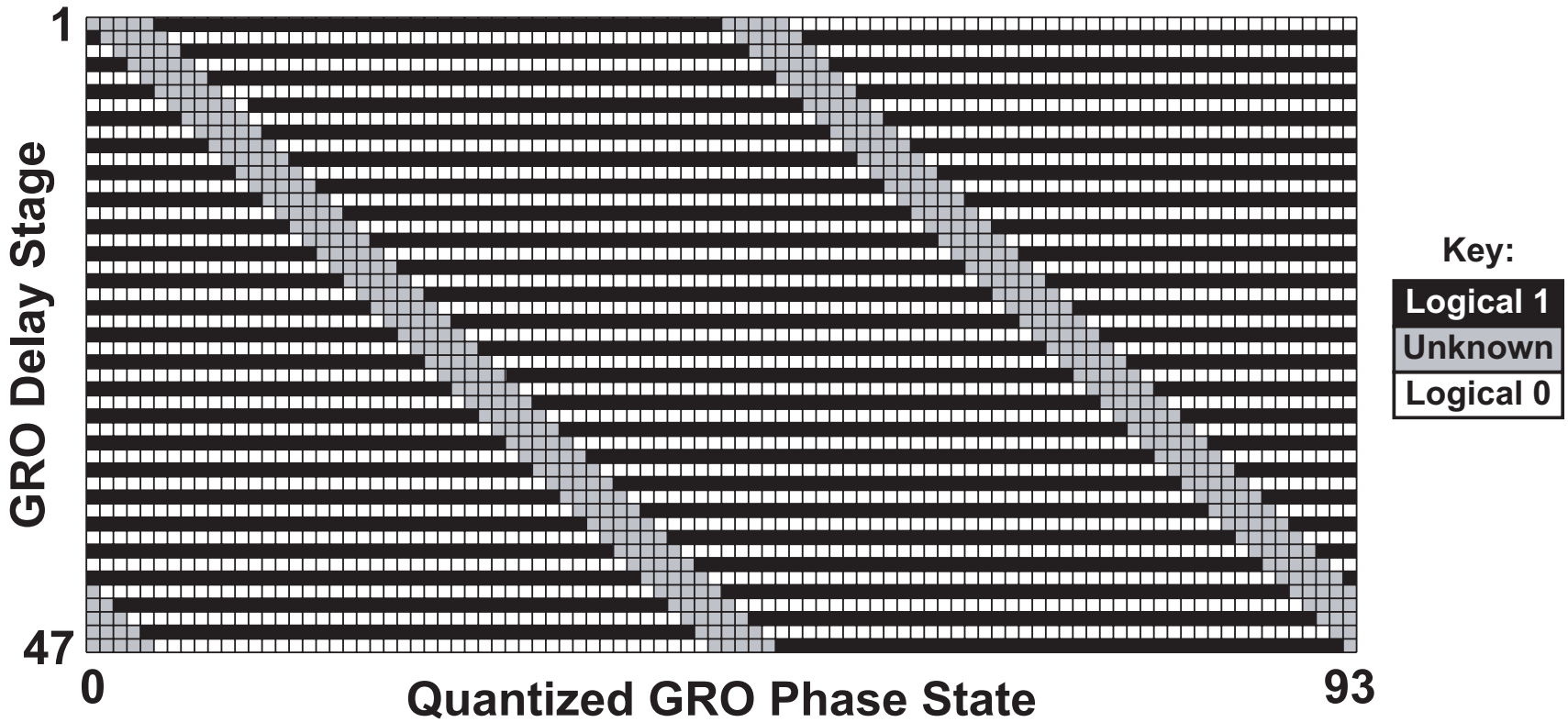


- Counter and registers need to have the same state
- Cannot allow counters to double-count a single transition

De-glitch circuits to ensure accurate measurements

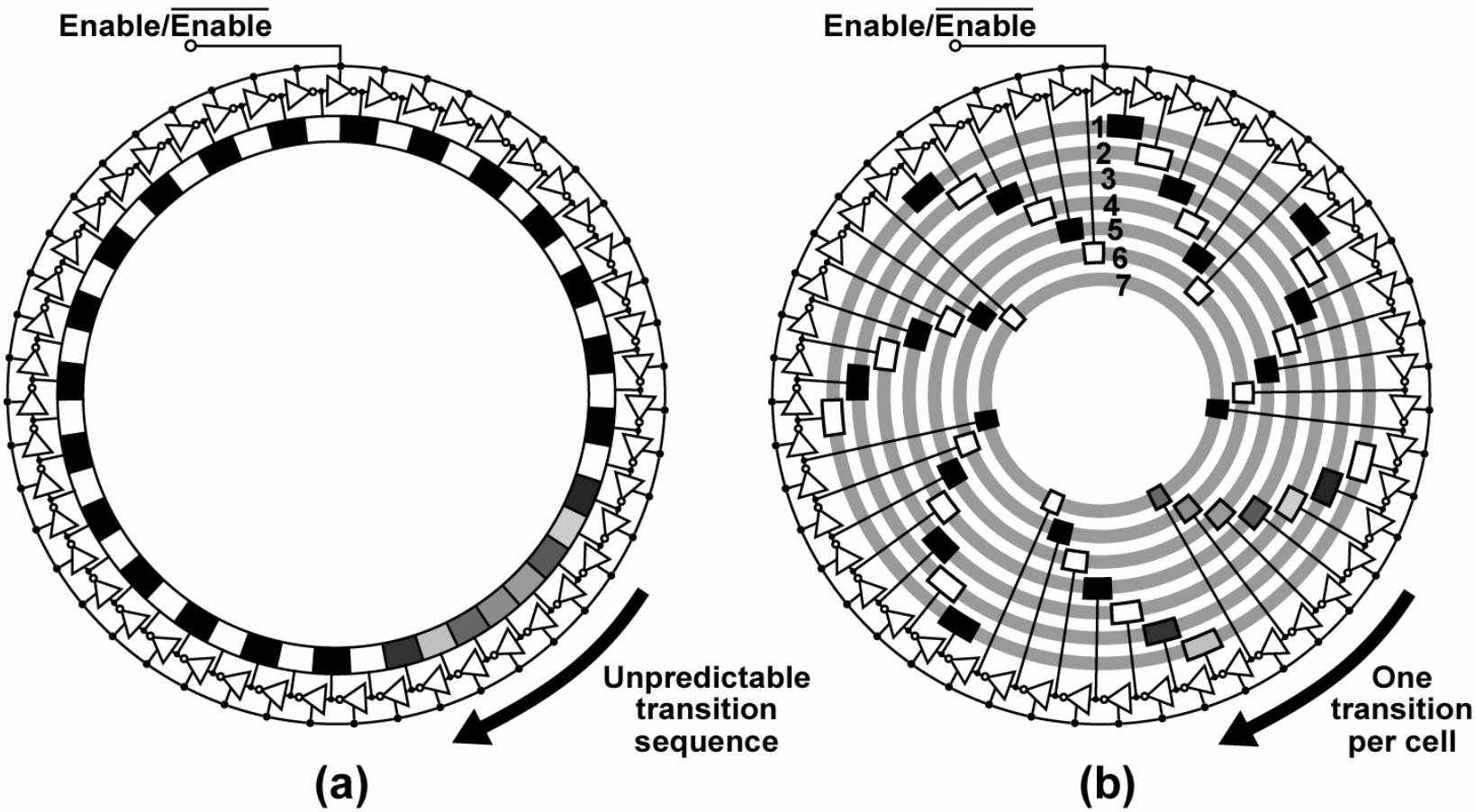


Key issue with scheme for an multi-path GRO…



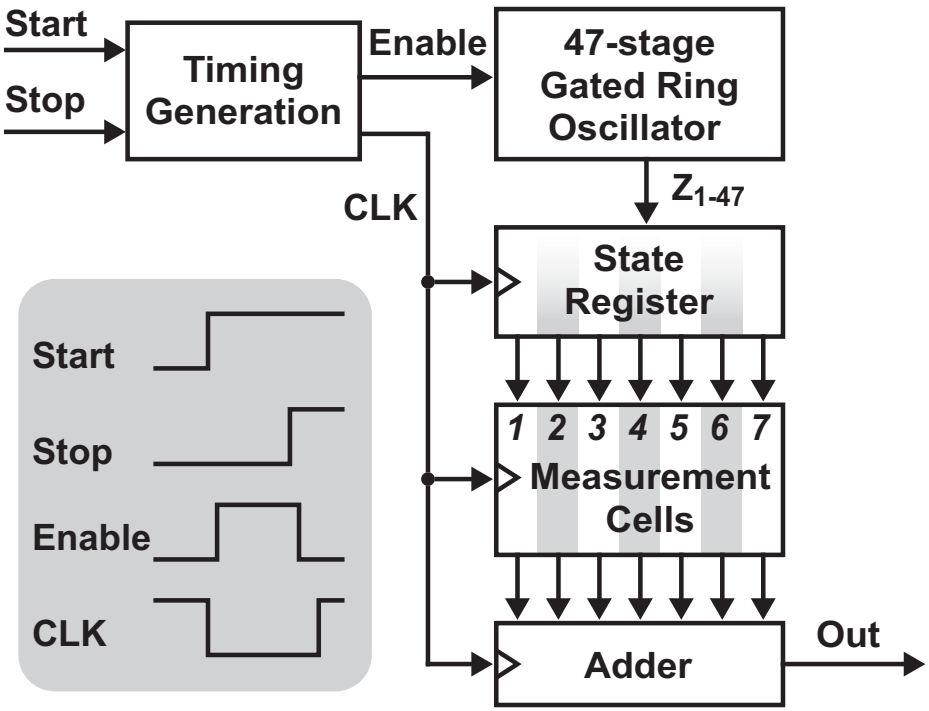
- More than one delay element output is logically uncertain
- Transition sequence is unpredictable and ambiguous
- Cannot map from entire GRO output state to phase

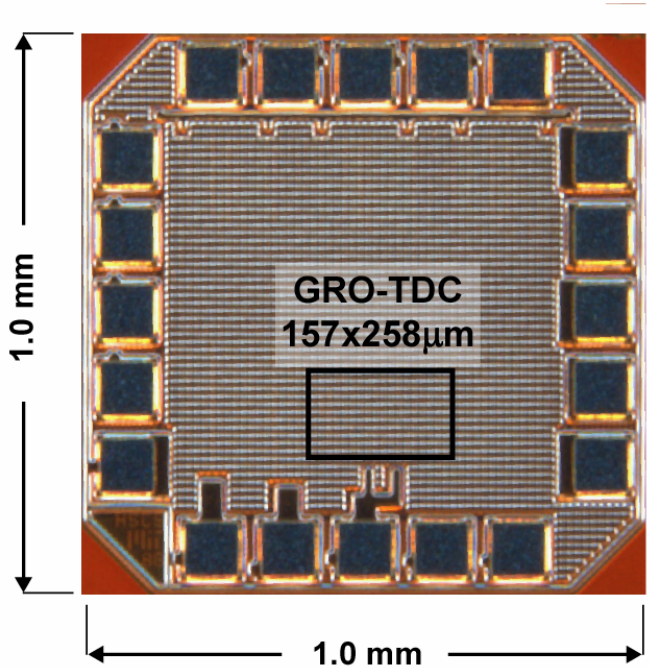
Restoring the predictable relationship…



- Calculate phase contribution from each cell independently
- Transition sequence within each cell is now predictable

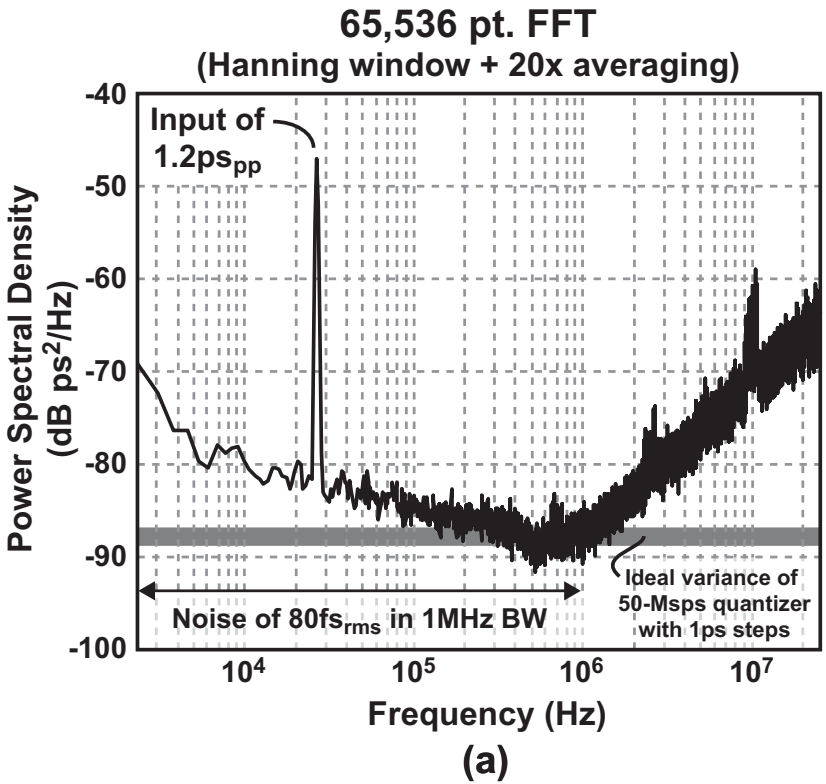
Prototype 0.13μm CMOS multi-path GRO-TDC

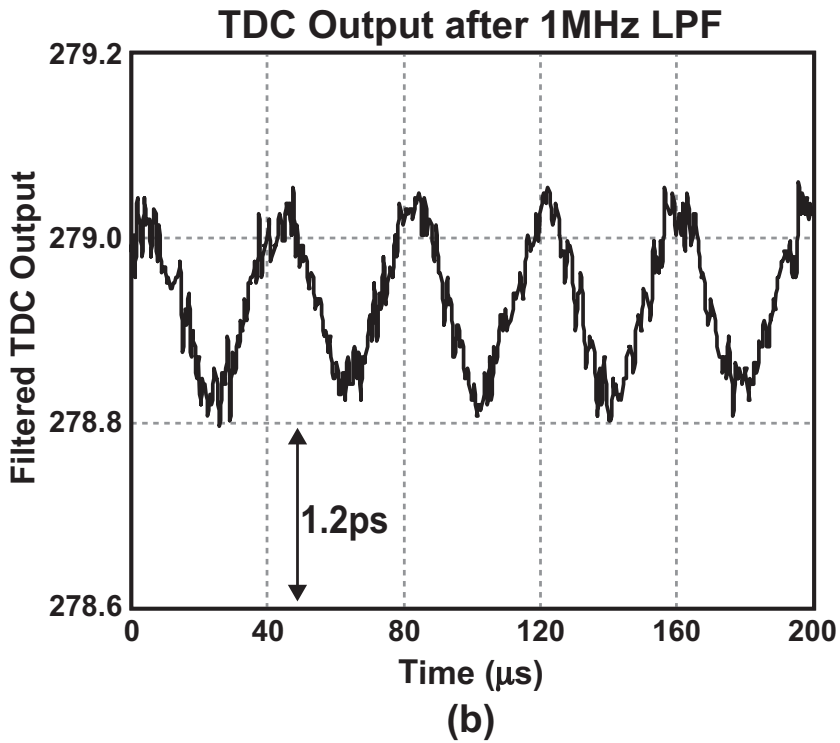




- Two implemented versions:
  - 8-bit, 500Msps
  - 11-bit, 100Msps version
- 2-21mW power consumption depending on input duty cycle

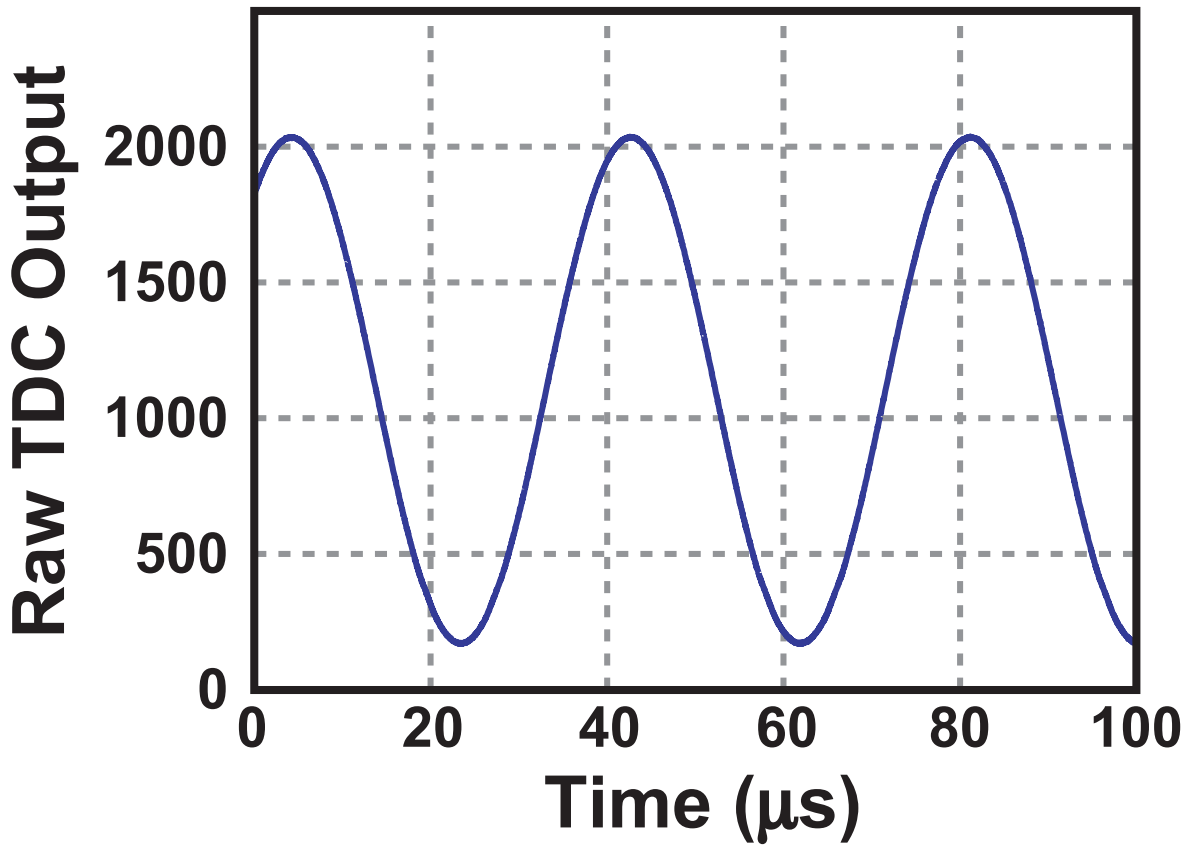
Measured noise-shaping of multi-path GRO



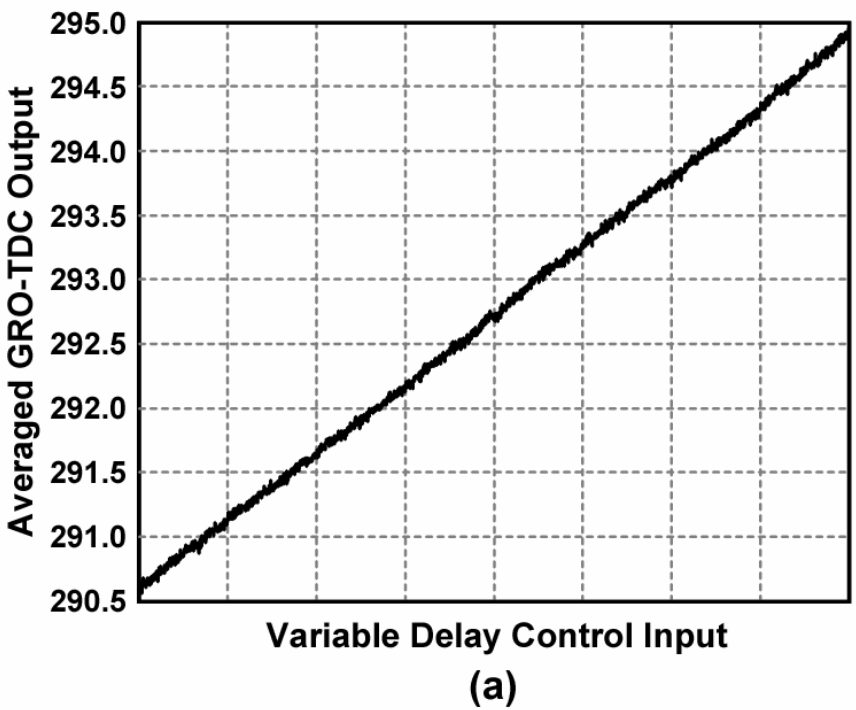


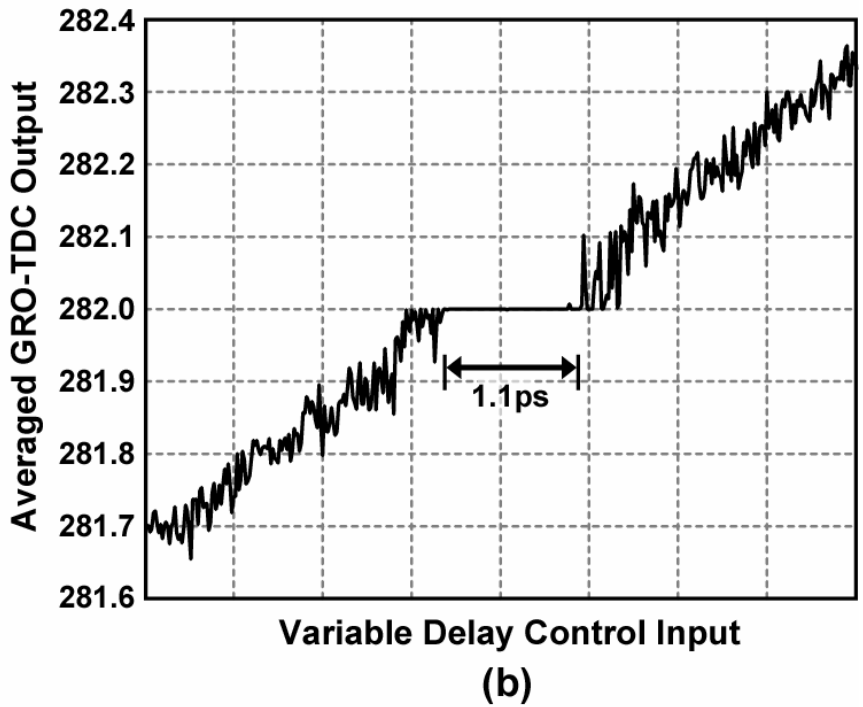
- Data collected at 50Msps
- More than 20dB of noise-shaping benefit
- 80fs$_{rms}$ integrated error from 2kHz-1MHz
- Floor primarily limited by 1/f noise (up to 0.5-1MHz)

Measured 11-bit range of multi-path GRO



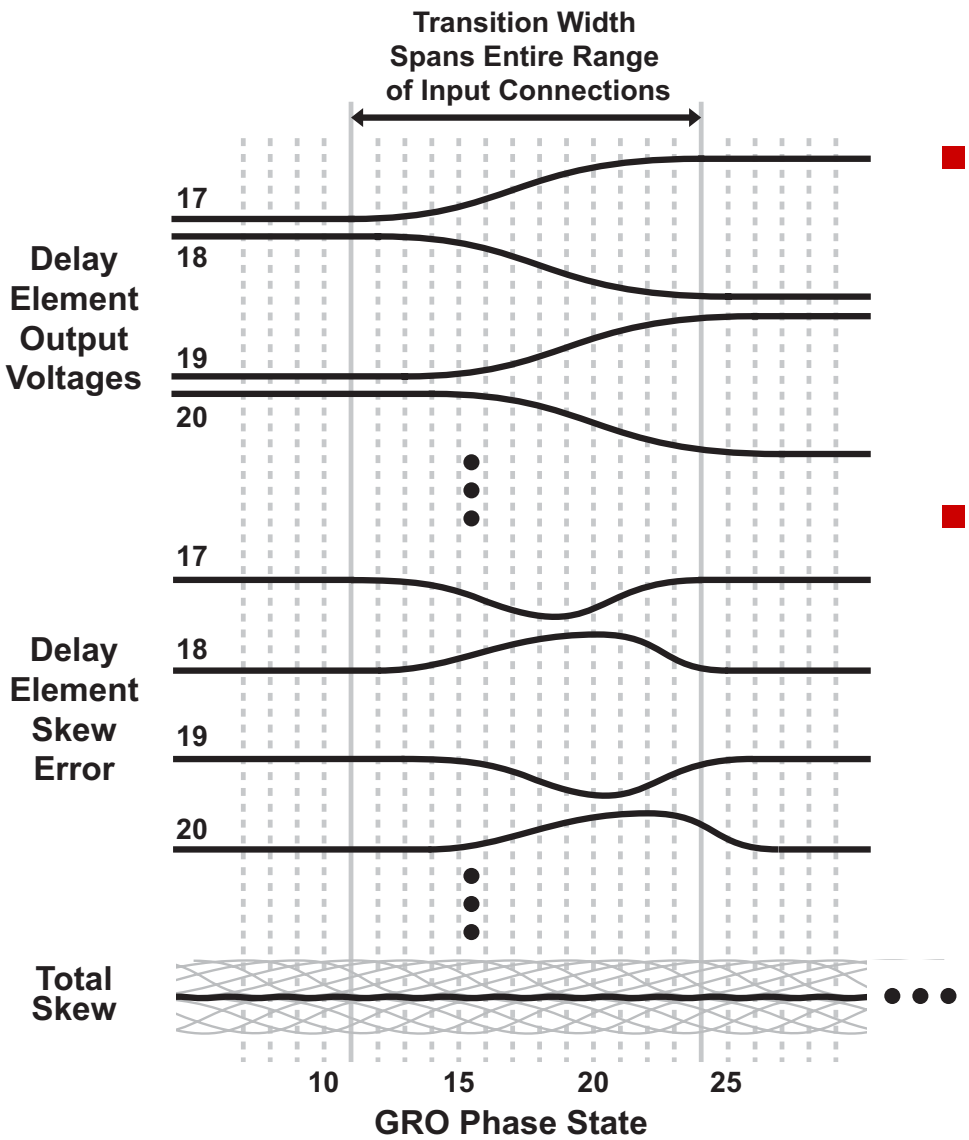
Measured deadzone behavior for multi-path GRO





- Only deadzones for outputs that are multiples of 2N
  - 94, 188, 282, etc.
  - No deadzones for other even or odd integers, fractional output
- Size of deadzone is reduced by 10x

Revised gating skew cartoon for the multi-path GRO



- At least *13* stages in transition at a time
  - Most of the mismatch from positive and negative transitions is cancelled
- T_{skew} is the *average* of error from each of the individual stages
  - GRO phase trajectory is determined by many stages, not just one

11-bit GRO-TDC performance summary

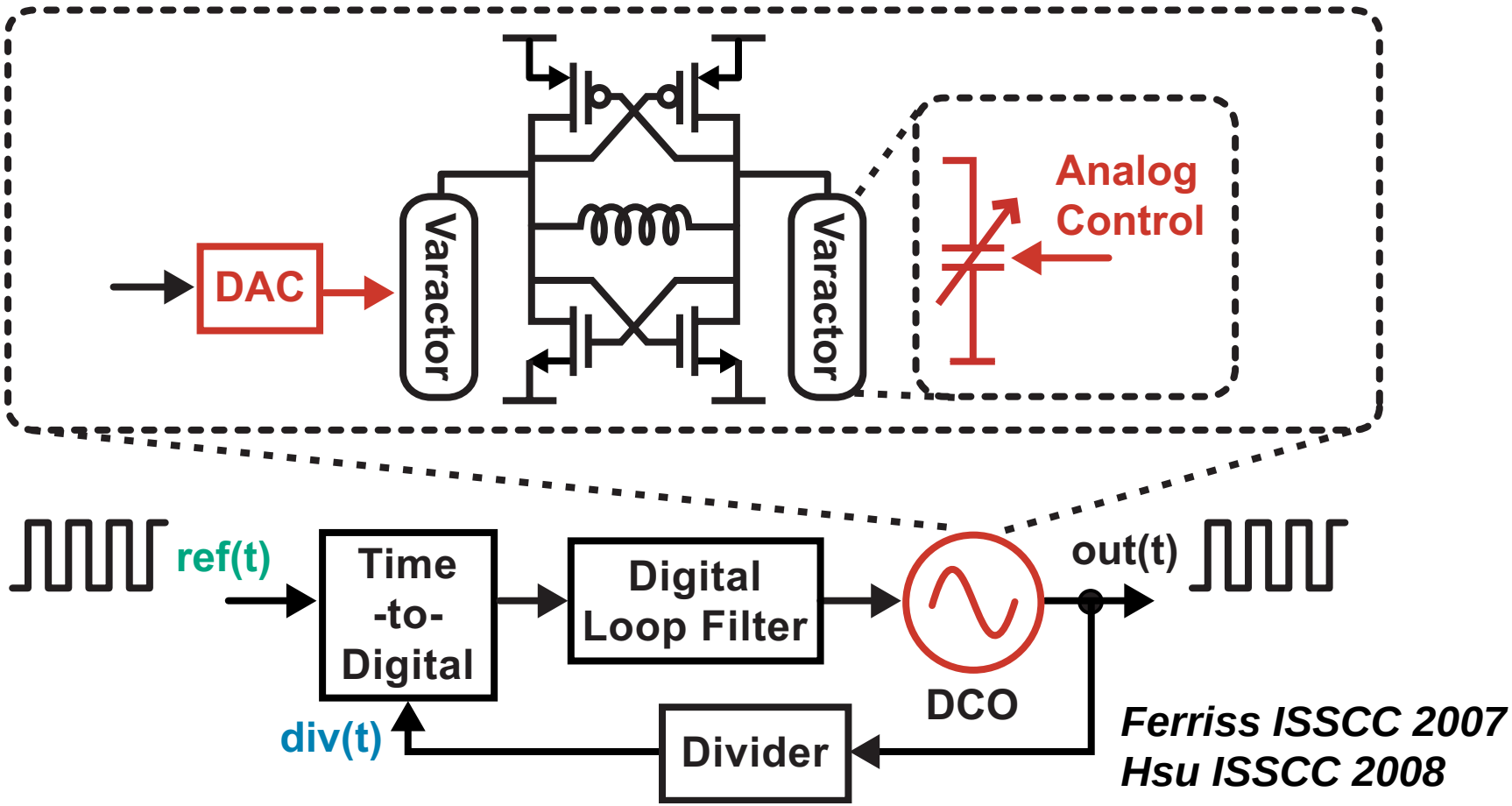
Sampling Frequency	**<100 MHz**
Raw delay resolution	**6ps**
Effective resolution	**1ps @ 50Msps**
Integrated noise	**80fs-rms, 2kHz-1MHz**
Dynamic range	**95dB, 1MHz BW**
Power	**2.2-21mW (<4mW typical)**
Area	**157 x 258μm**
Technology	**0.13μm CMOS**

Summary of Time to Digital Conversion

- **Key performance metrics are**
  - **Resolution: want low quantization noise**
  - **Mismatch: want high linearity**
  - **Power and area: want long battery life, low cost**
- **Many structures have been introduced**
  - **Classical, Vernier, Two-Step, Time Amplifiers, Re-cycling, Gated Ring Oscillator**
- **Comparable to ADCs but suffers from lack of "time memory element"**
  - **Cyclic conversion and pipeline structures have not been achieved**
- **A very promising research area!**

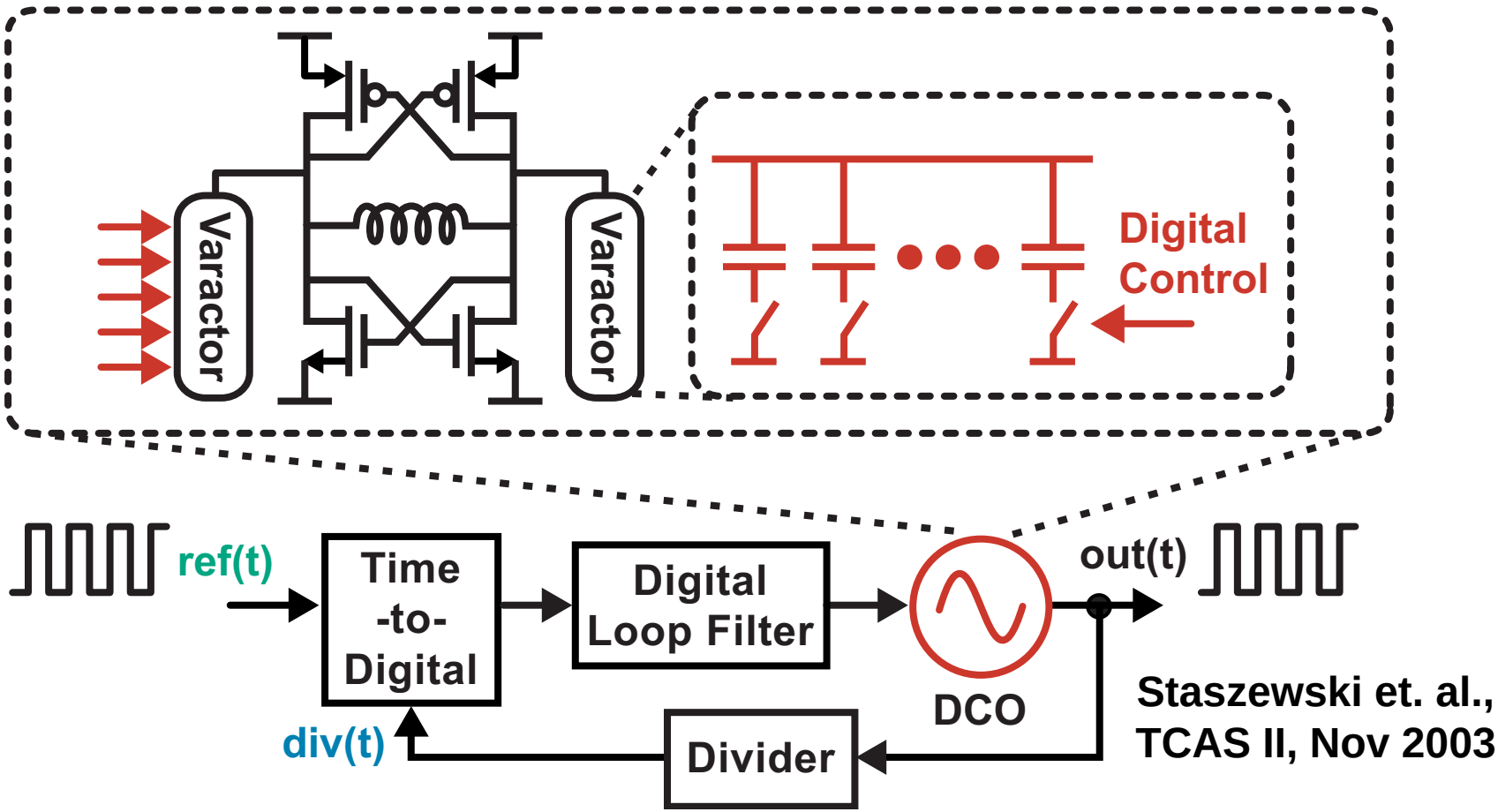
Digitally-Controlled Oscillators

A Straightforward Approach for Achieving a DCO



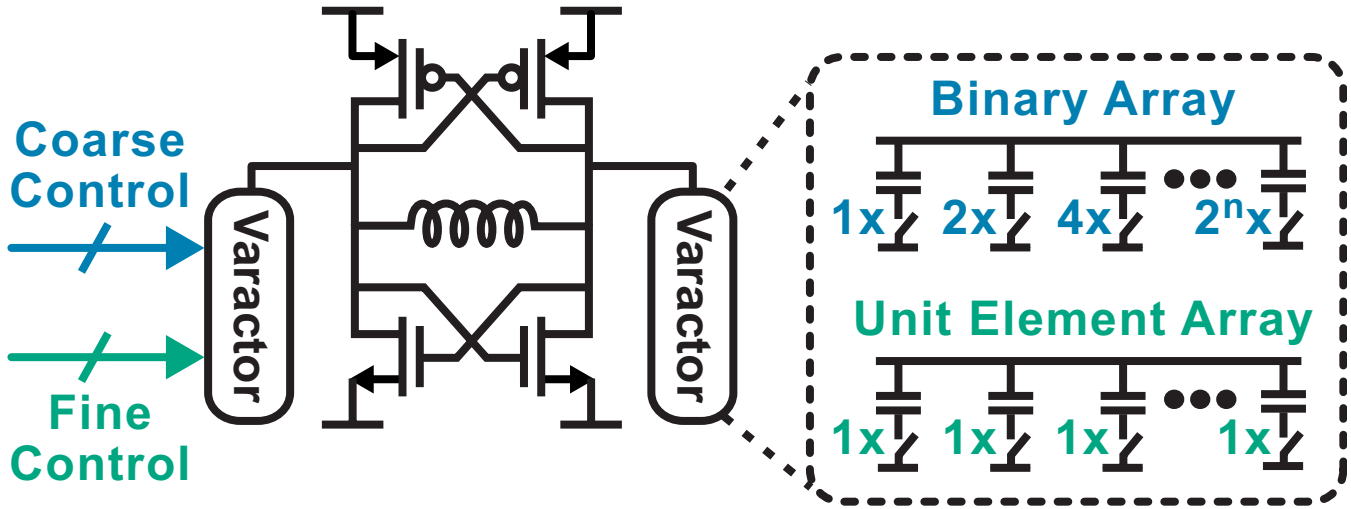
- Use a DAC to control a conventional LC oscillator
  - Allows the use of an existing VCO within a digital PLL
  - Can be applied across a broad range of IC processes

A Much More Digital Implementation



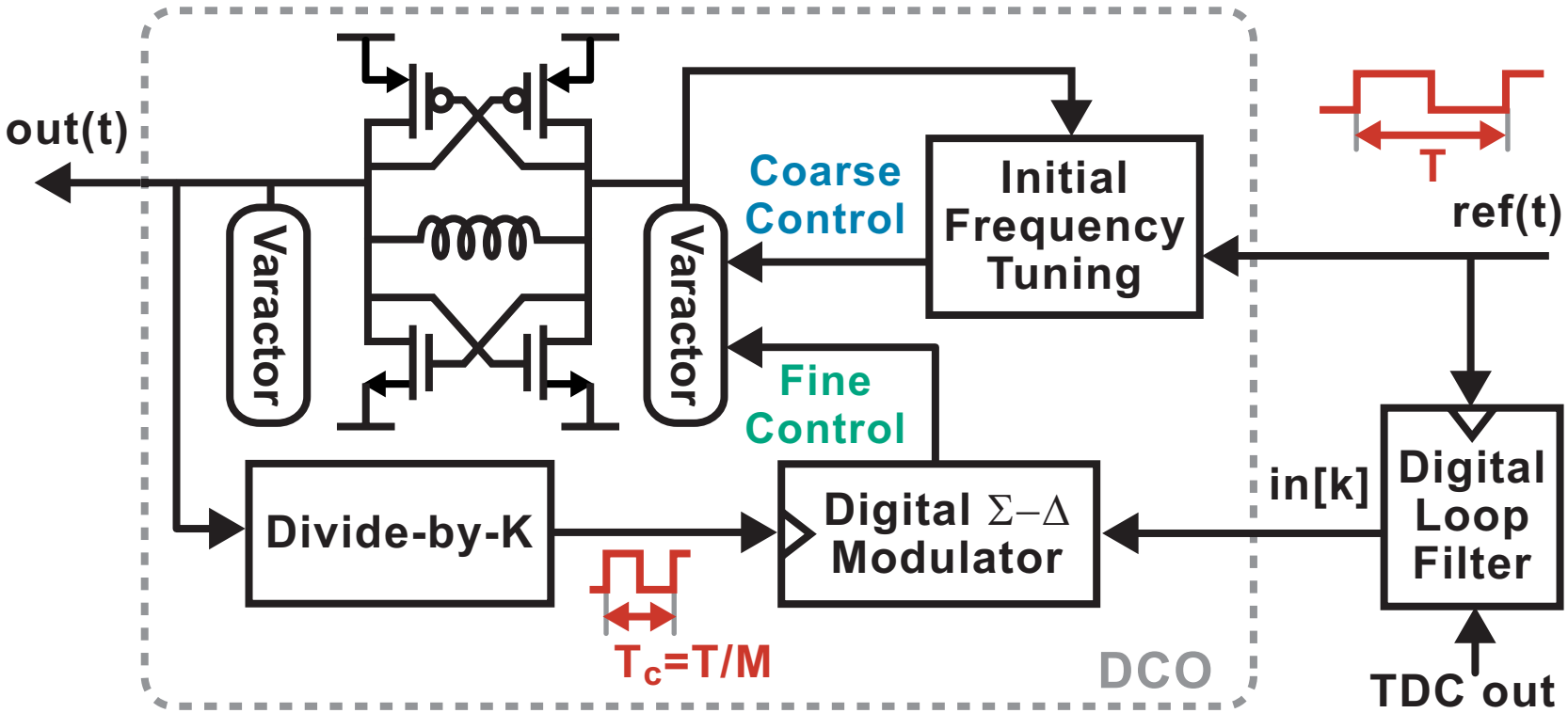
- Adjust frequency in an LC oscillator by switching in a variable number of small capacitors
  - Most effective for CMOS processes of 0.13u and below

Leveraging Segmentation in Switched Capacitor DCO



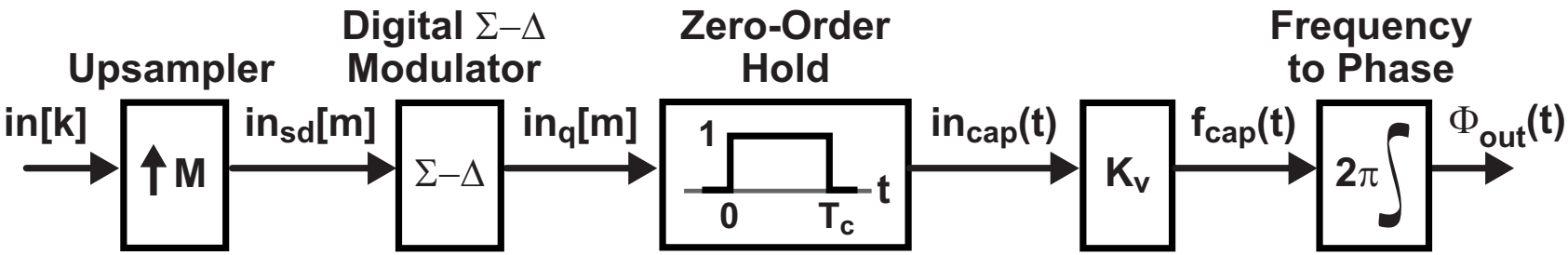
- Similar in design as *segmented* capacitor DAC structures
  - Binary array: efficient control, but may lack monotonicity
  - Unit element array: monotonic, but complex control
- Coarse and fine control segmentation of DCO
  - Coarse control: active only during initial frequency tuning (leverage binary array)
  - Fine control: controlled by PLL feedback (leverage unit element array to guarantee monotonicity)

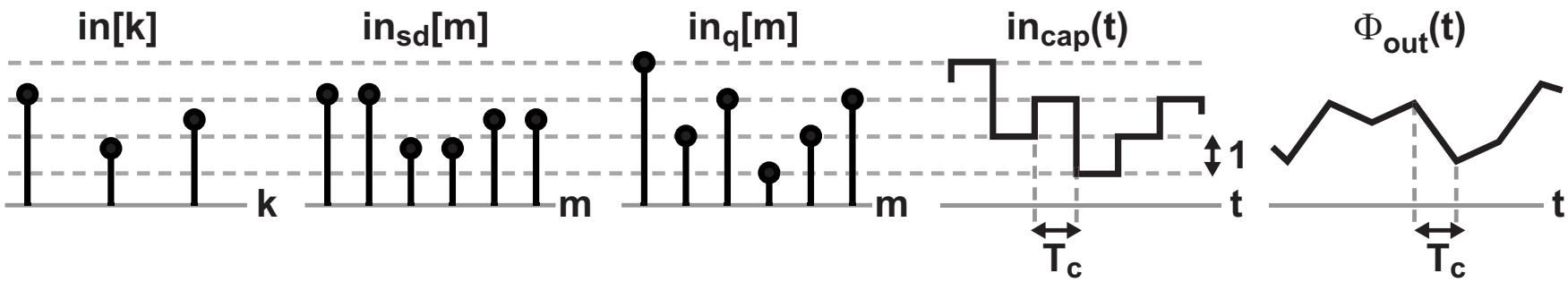
Leveraging Dithering for Fine Control of DCO



- Increase resolution by Σ–Δ dithering of fine cap array
- Reduce noise from dithering by
  - Using small unit caps in the fine cap array
  - Increasing the dithering frequency (defined as $1/T_c$)
    - We will assume $1/T_c = M/T$ (i.e. M times reference frequency)

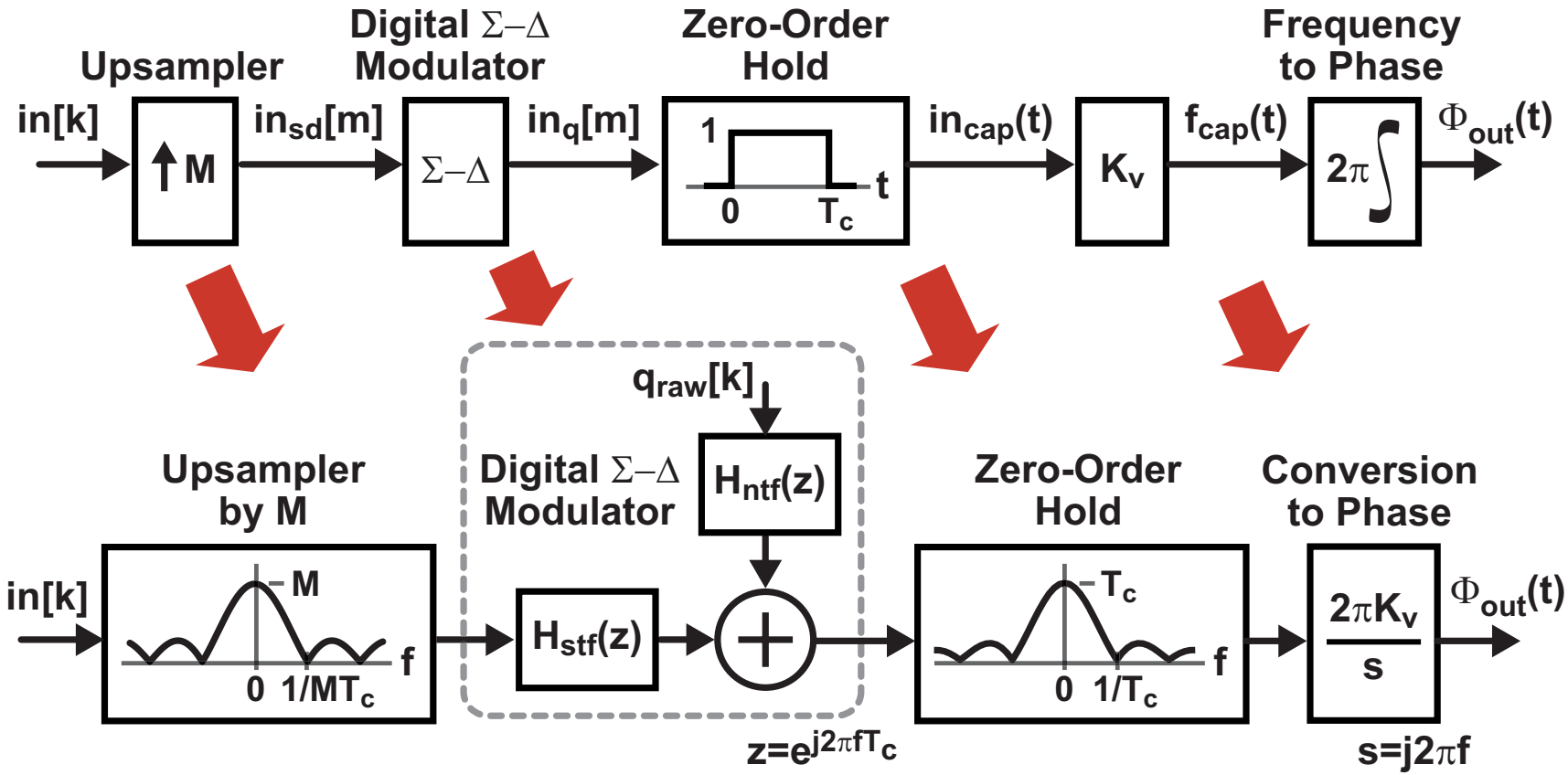
Time-Domain Modeling of the DCO





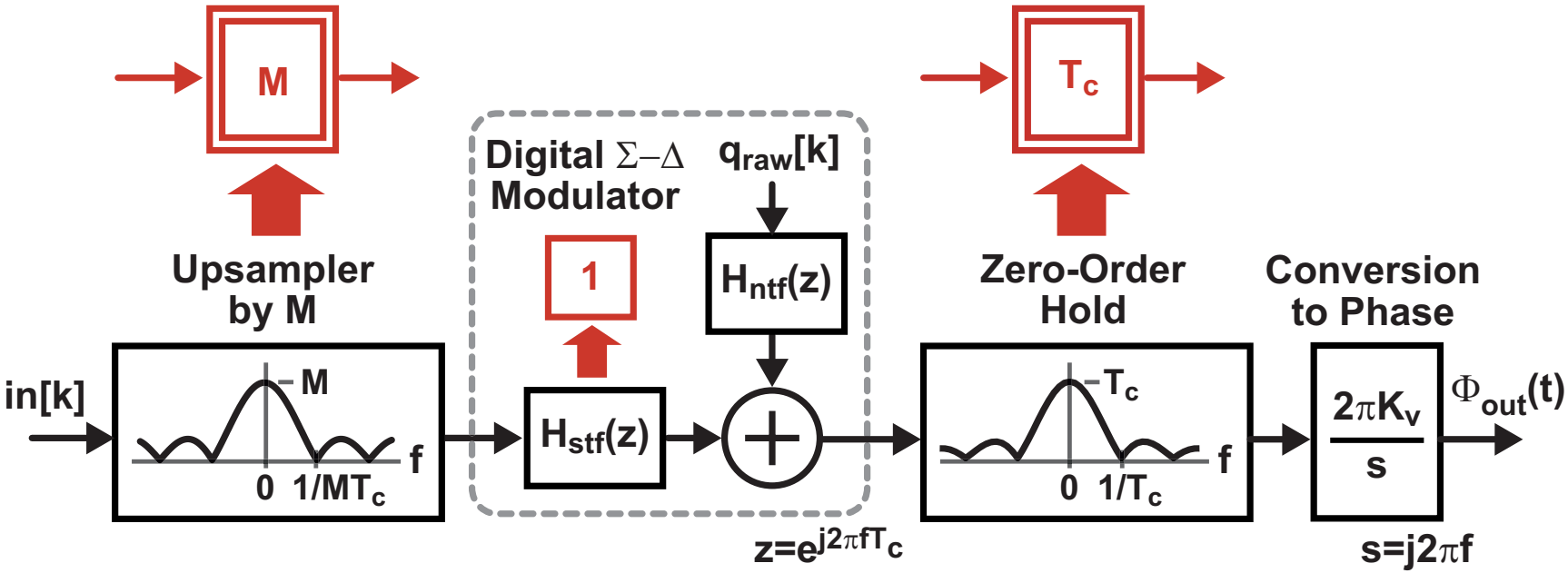
- Input to the DCO is supplied by the loop filter
  - Clocked at *1/T* (i.e., reference frequency)
- Switched capacitors are dithered by Σ–Δ at a higher rate
  - Clocked at $1/T_c = M/T$
  - Held at a given setting for duration T_c
- Fine cap element value determines K_v of VCO
  - Units of K_v are Hz/unit cap

Frequency Domain Modeling of DCO



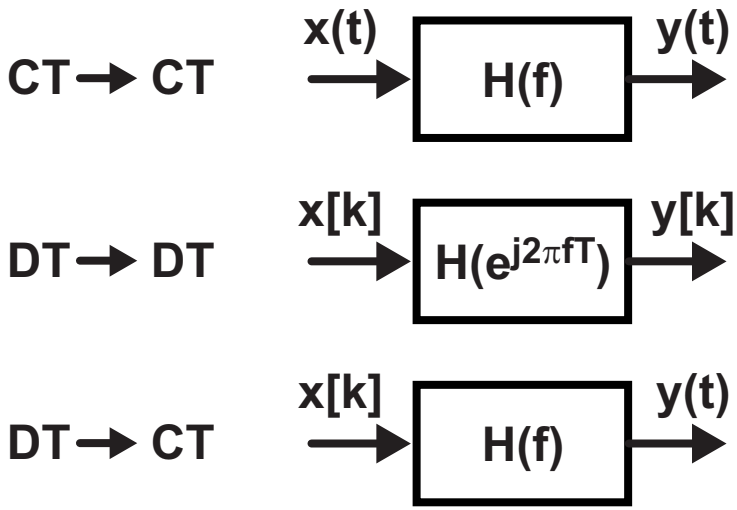
- Upsampler and zero-order hold correspond to discrete and continuous-time *sinc* functions, respectively
- Σ–Δ has signal and noise transfer functions ($H_{stf}(z)$, $H_{ntf}(z)$)
  - Note: $var(q_{raw}[k])$ = 1/12 (uniformly distributed from 0 to 1)

Simplification of the DCO Model



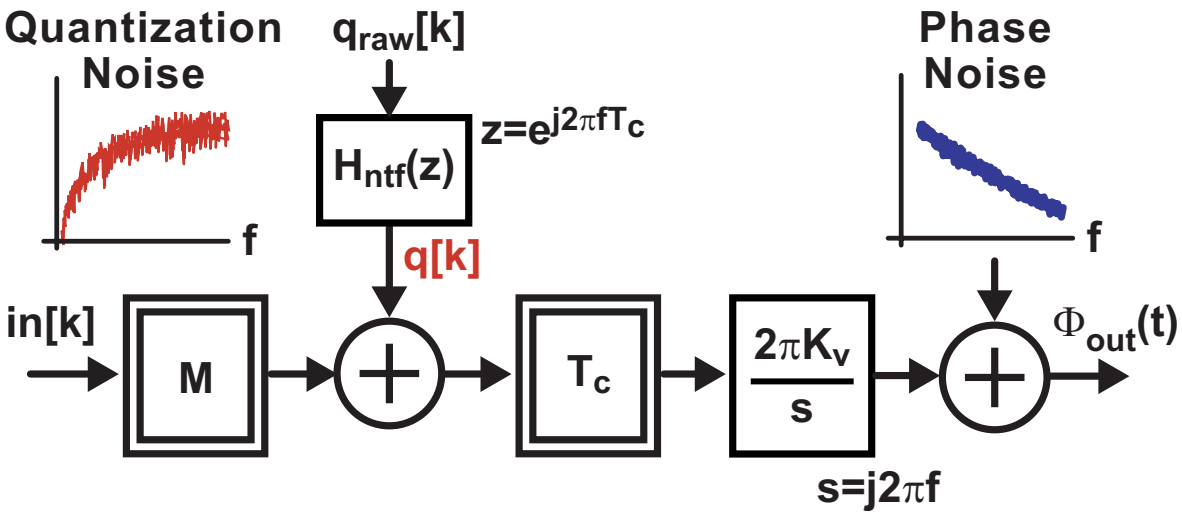
- **Focus on low frequencies for calculations to follow**
  - **Assume *sinc* functions are relatively flat at the low frequencies of interest**
    - Upsampler is approximated as a gain of *M*
    - Zero-order hold is approximated as a gain of T_c
- **Assume $H_{stf}(z) = 1$**
  - **True for Σ–Δ structures such as MASH (ignoring delays)**

Spectral Density Calculations



- **CT ➡ CT** $S_y(f) = |H(f)|^2 S_x(f)$
- **DT ➡ DT** $S_y(e^{j2\pi fT}) = |H(e^{j2\pi fT})|^2 S_x(e^{j2\pi fT})$
- **DT ➡ CT** $S_y(f) = \frac{1}{T}|H(f)|^2 S_x(e^{j2\pi fT})$

Calculation of Quantization Noise from Cap Dithering



- **DT to CT spectral calculation:**

$$S_{\Phi_{out}}(f)\Big|_{\text{dco,quant}} = \frac{1}{T_c}\left|T_c\frac{2\pi K_v}{j2\pi f}\right|^2\left|H_{ntf}(e^{j2\pi fT_c})\right|^2 S_{q_{raw}}(f)$$

$$= T_c\left|\frac{K_v}{f}\right|^2\left|H_{ntf}(e^{j2\pi fT_c})\right|^2\frac{1}{12}$$

  - $S_{q_{raw}}(f)$ = 1/12 since $q_{raw}[k]$ uniformly distributed from 0 to 1
  - $H_{ntf}(z)$ is often $1-z^{-1}$ (first order) or $(1-z^{-1})^2$ (second order)

Example Calculation for DCO Quantization Noise

- **Assumptions (Out freq = 3.6 GHz)**
  - **Dithering frequency is 200 MHz (i.e., $1/T_c$ = 200e6)**
  - **Σ–Δ has first order shaping (i.e., $H_{ntf}(z) = 1 - z^{-1}$)**
  - **Fine cap array yields 12 kHz/unit cap (i.e., K_v = 12e3)**

$$S_{\Phi_{out}}(f)\Big|_{\text{dco,quant}} = T_c \left|\frac{K_v}{f}\right|^2 \left|H_{ntf}(e^{j2\pi f T_c})\right|^2 \frac{1}{12}$$

$$= \frac{1}{200e6}\left|\frac{12e3}{f}\right|^2 \left|1 - e^{j2\pi f/200e6}\right|^2 \frac{1}{12}$$

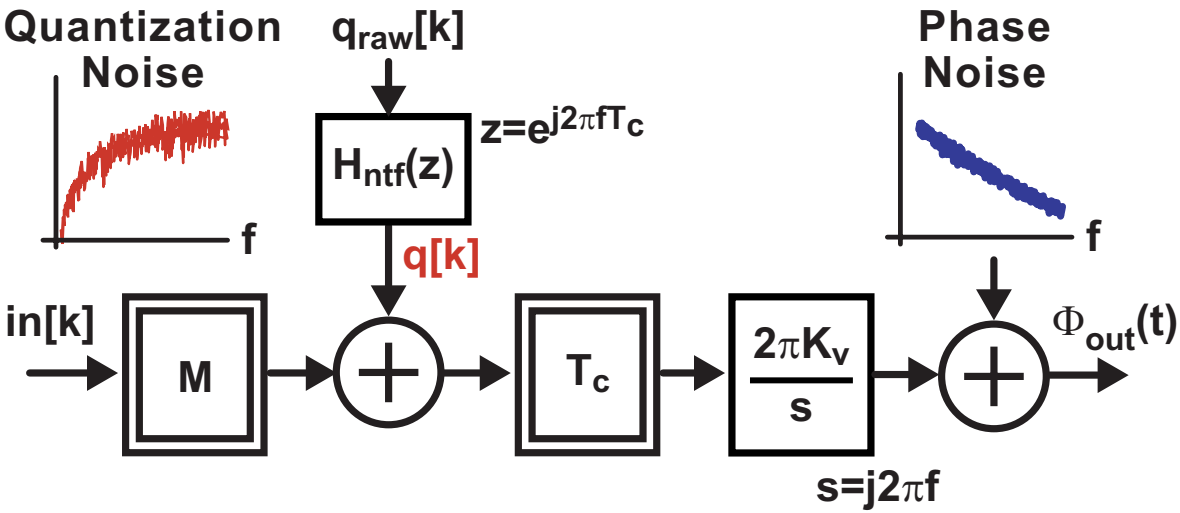
- **At a frequency offset of f = 20 MHz:**

$$= \frac{1}{200e6}\left|\frac{12e3}{20e6}\right|^2 \left|1 - e^{j2\pi 1/10}\right|^2 \frac{1}{12} = 5.73 \cdot 10^{-17}$$

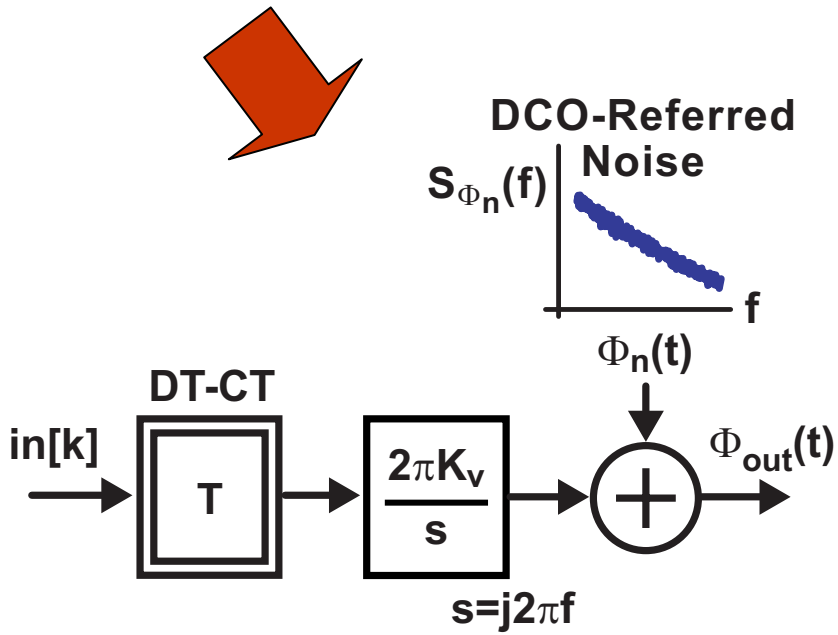
$$10\log(5.73{\cdot}10^{-17}) = -162.4\ dBc/Hz \quad \text{(at 20 MHz offset)}$$

Below the phase noise (-153 dBc/Hz at 20 MHz) in the example

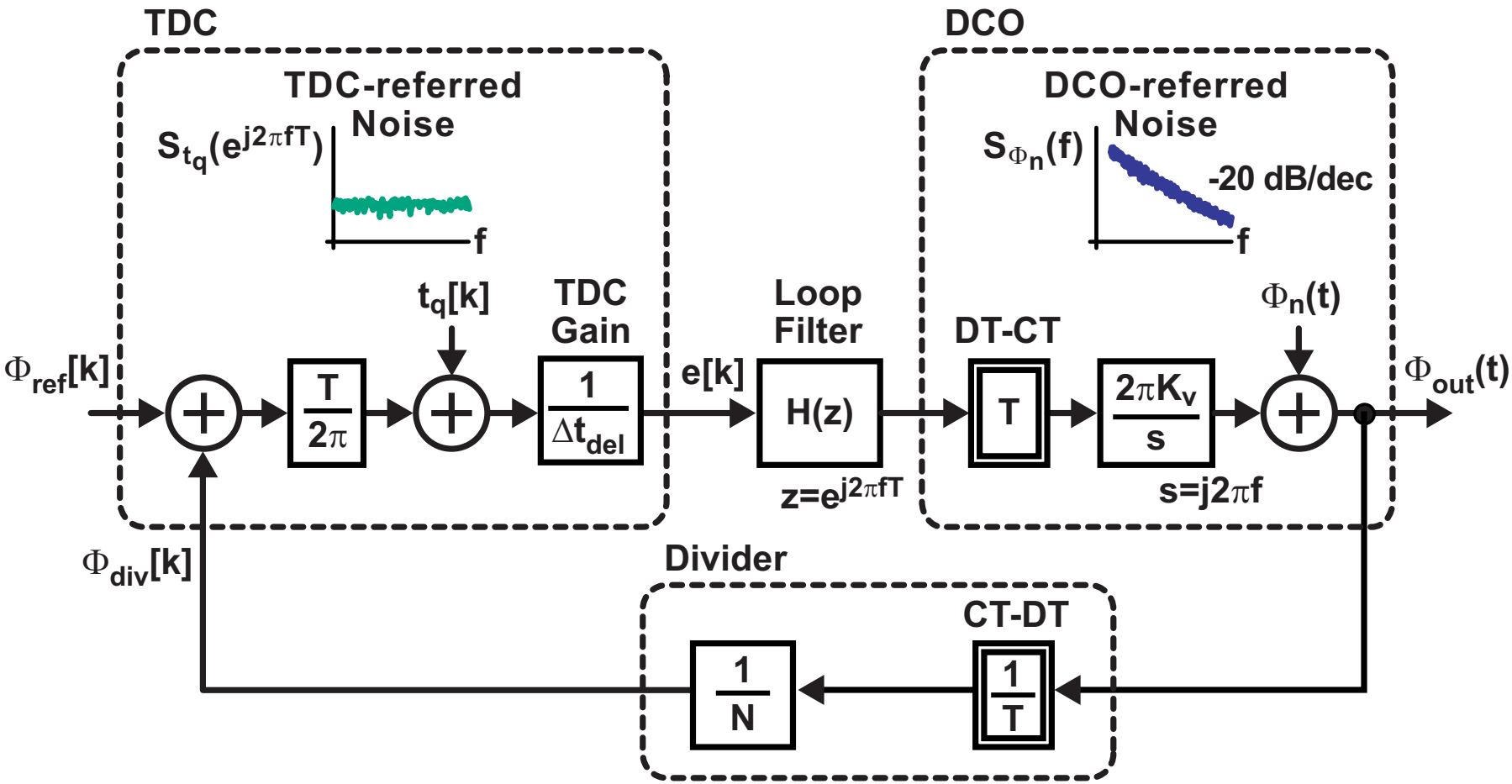
Further Simplification of DCO Model



- Proper design of DCO will yield quantization noise that is below that of the intrinsic phase noise (set by tank *Q*, etc.)
  - Assume *q[k]* = 0 for simplified model
- Note that $T = M¢T_c$

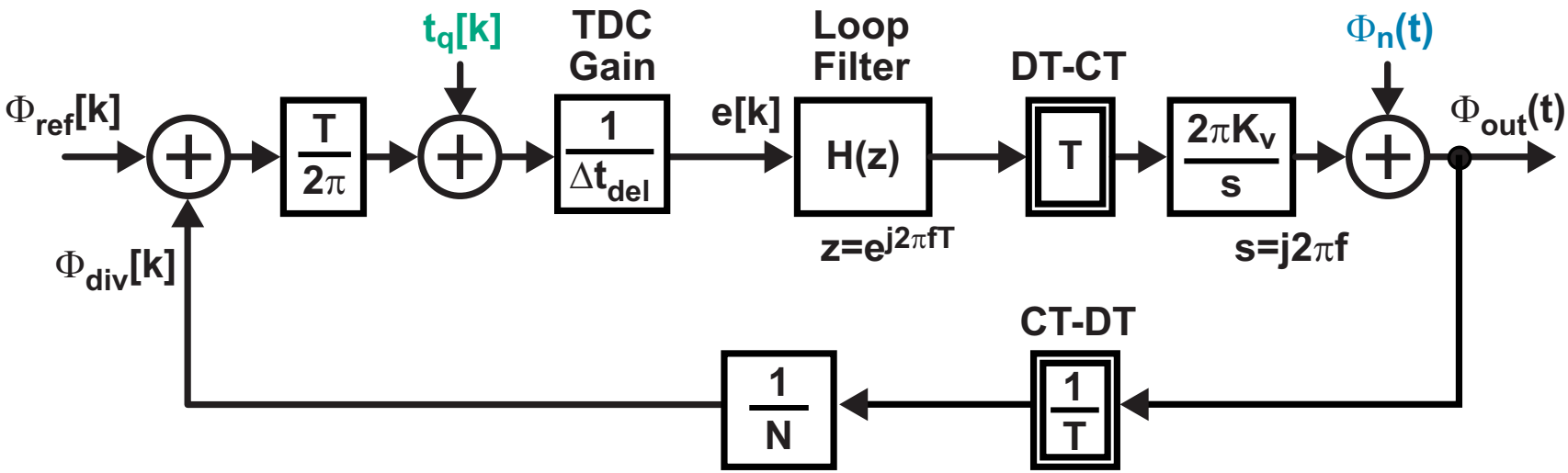


Overall Digital PLL Model



- TDC and DCO-referred noise influence overall phase noise according to associated transfer functions to output
- Calculations involve both discrete *and* continuous time

Key Transfer Functions



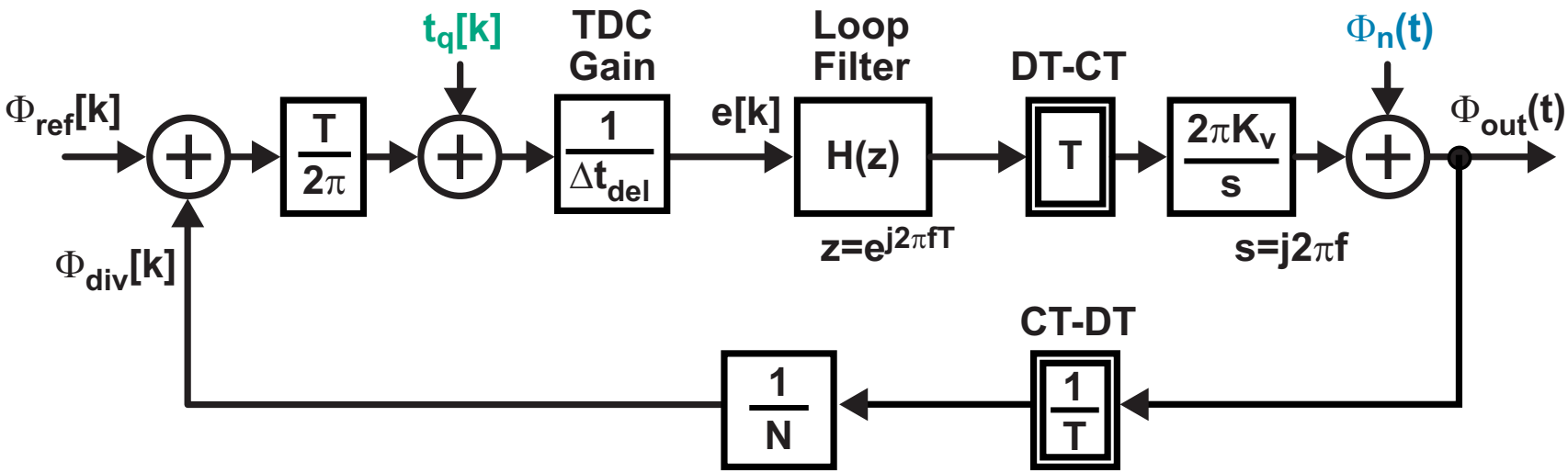
- **TDC-referred noise**

$$\frac{\Phi_{out}}{t_q} = \frac{(1/\Delta t_{del})H(e^{j2\pi fT})T2\pi K_v/(2\pi jf)}{1+(1/\Delta t_{del})H(e^{j2\pi fT})TK_v/(2\pi jf)(1/N)}$$

- **DCO-referred noise**

$$\frac{\Phi_{out}}{\Phi_n} = \frac{1}{1+(1/\Delta t_{del})H(e^{j2\pi fT})TK_v/(2\pi jf)(1/N)}$$

Introduce a Parameterizing Function



- **Define open loop transfer function *A(f)* as:**

$$A(f) = (1/\Delta t_{del}) H(e^{j2\pi fT}) T K_v / (2\pi j f)(1/N)$$

- **Define closed loop parameterizing function *G(f)* as:**

$$G(f) = \frac{A(f)}{1 + A(f)}$$

  - **Note: *G(f)* is a lowpass filter with DC gain = 1**

Transfer Function Parameterization Calculations

- **TDC-referred noise**

$$\frac{\Phi_{out}}{t_q} = \frac{(1/\Delta t_{del})H(e^{j2\pi fT})T2\pi K_v/(2\pi jf)}{1+(1/\Delta t_{del})H(e^{j2\pi fT})TK_v/(2\pi jf)(1/N)}$$

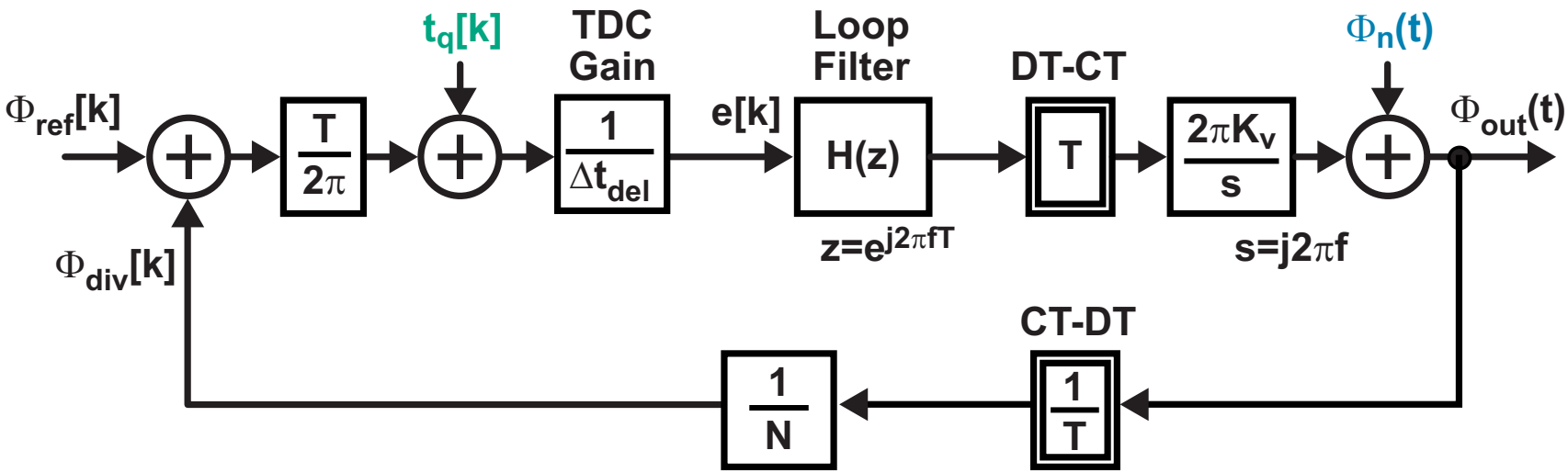
$$= \frac{2\pi NA(f)}{1+A(f)} = \boxed{2\pi NG(f)}$$

- **DCO-referred noise**

$$\frac{\Phi_{out}}{\Phi_n} = \frac{1}{1+(1/\Delta t_{del})H(e^{j2\pi fT})TK_v/(2\pi jf)(1/N)}$$

$$= \frac{1}{1+A(f)} = \frac{1+A(f)-A(f)}{1+A(f)} = \boxed{1-G(f)}$$

Key Observations



- **TDC-referred noise**

$$\frac{\Phi_{out}}{t_q} = 2\pi N G(f)$$

⬅ **Lowpass with a DC gain of $2\pi N$**

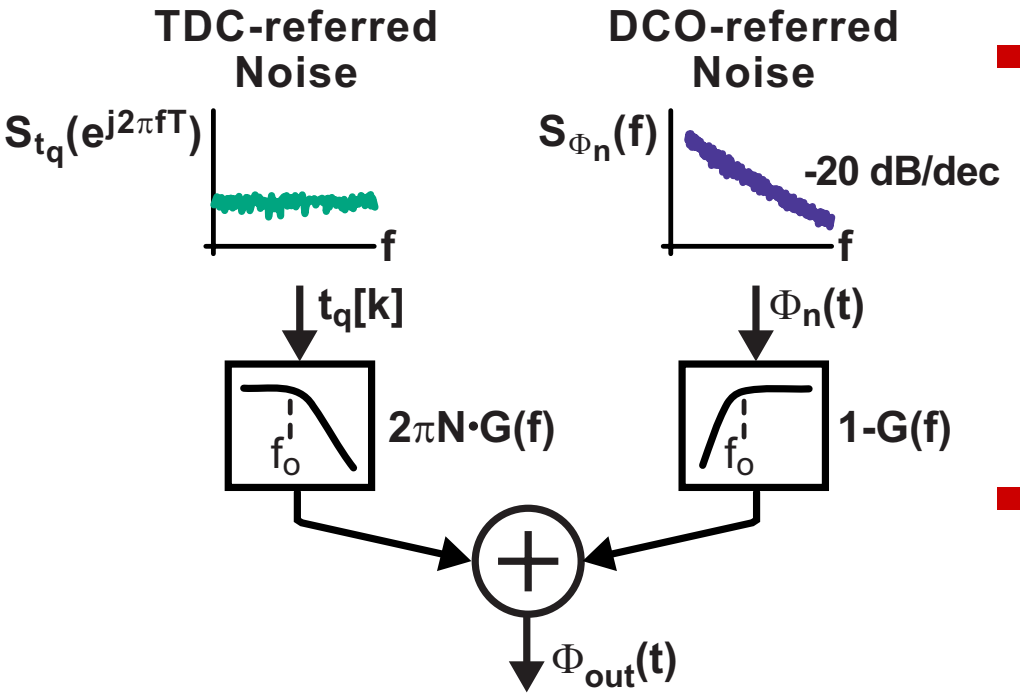
- **DCO-referred noise**

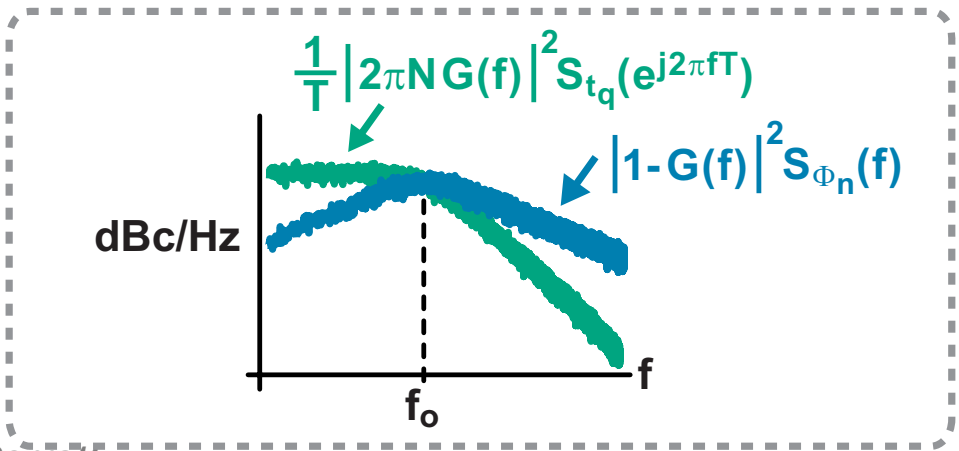
$$\frac{\Phi_{out}}{\Phi_n} = 1 - G(f)$$

⬅ **Highpass with a high frequency gain of 1**

How do we calculate the output phase noise?

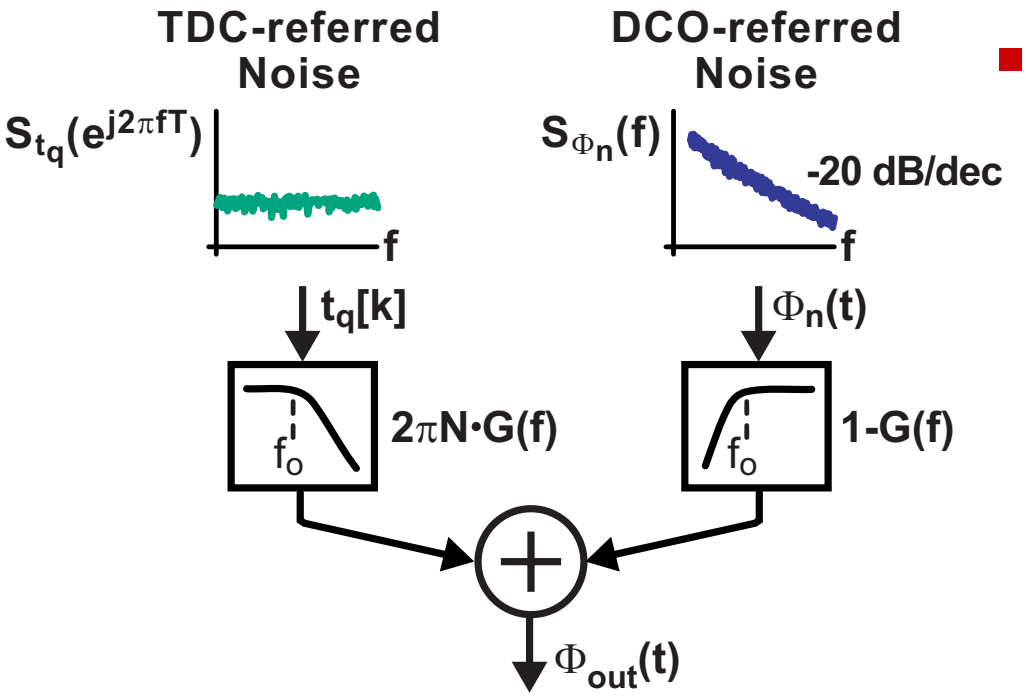
Phase Noise Calculation





- TDC noise
  - DT to CT calculation
  - Dominates PLL phase noise at low frequency offsets
- DCO noise
  - CT to CT calculation
  - Dominates PLL phase noise at high frequency offsets

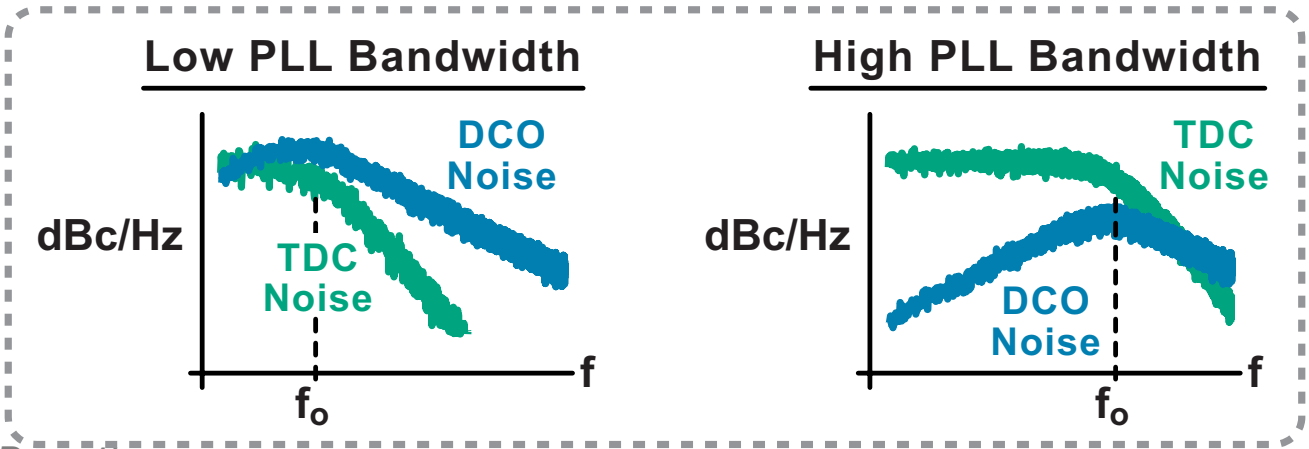
Impact of PLL Bandwidth



- **PLL bandwidth dramatically influences relative impact of TDC and VCO noise**

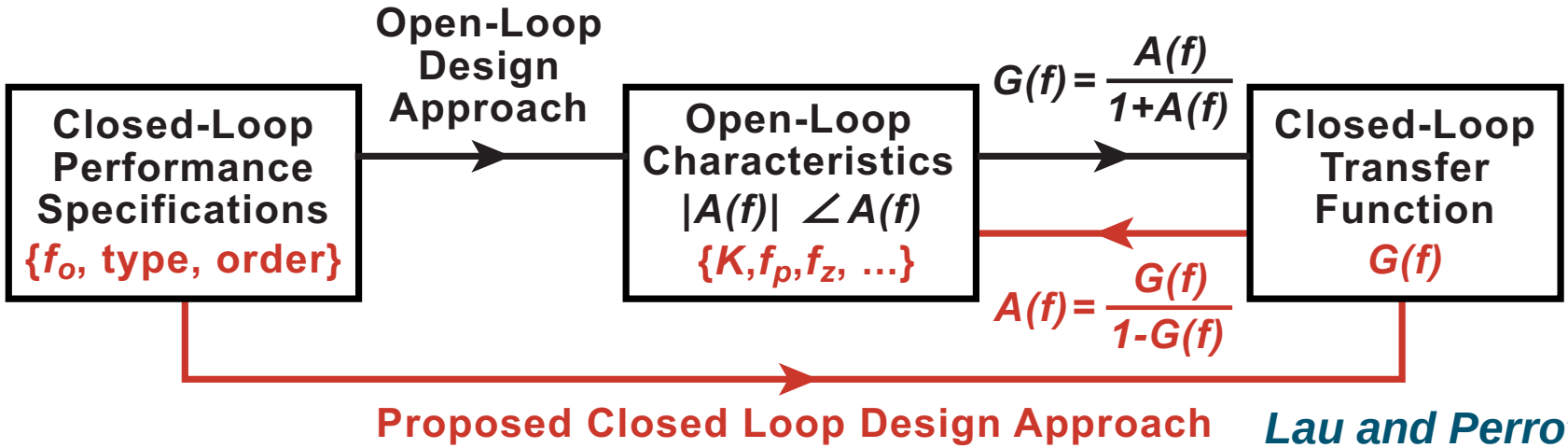
Want high PLL bandwidth?

➡ **Need low TDC Noise**



System Level Design

Closed Loop PLL Design Approach



Lau and Perrott, DAC, June 2003

- Classical open loop approach
  - Indirectly design *G(f)* using bode plots of *A(f)*
- Proposed closed loop approach
  - Directly design *G(f)* by examining impact of its specifications on phase noise (and settling time)
  - Solve for *A(f)* that will achieve desired *G(f)*

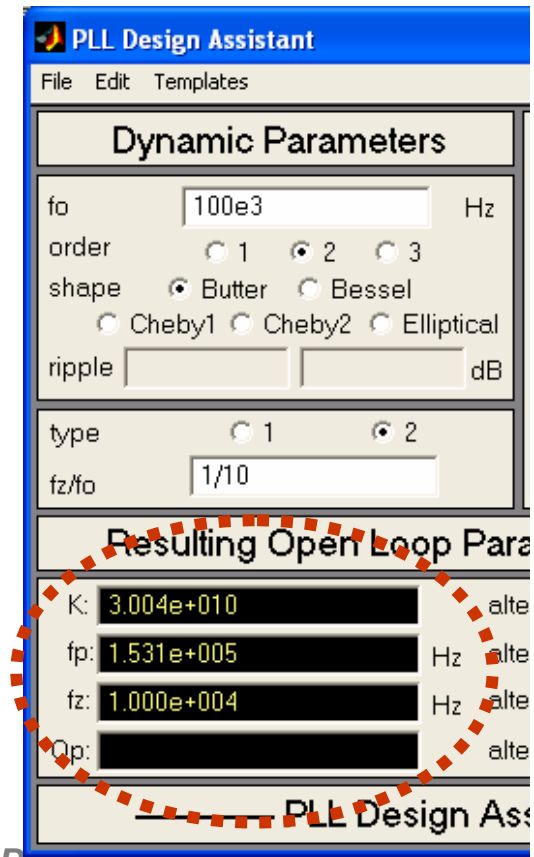
Implemented in *PLL Design Assistant* Software

http://www.cppsim.com

Transfer Function Design using PLL Design Assistant

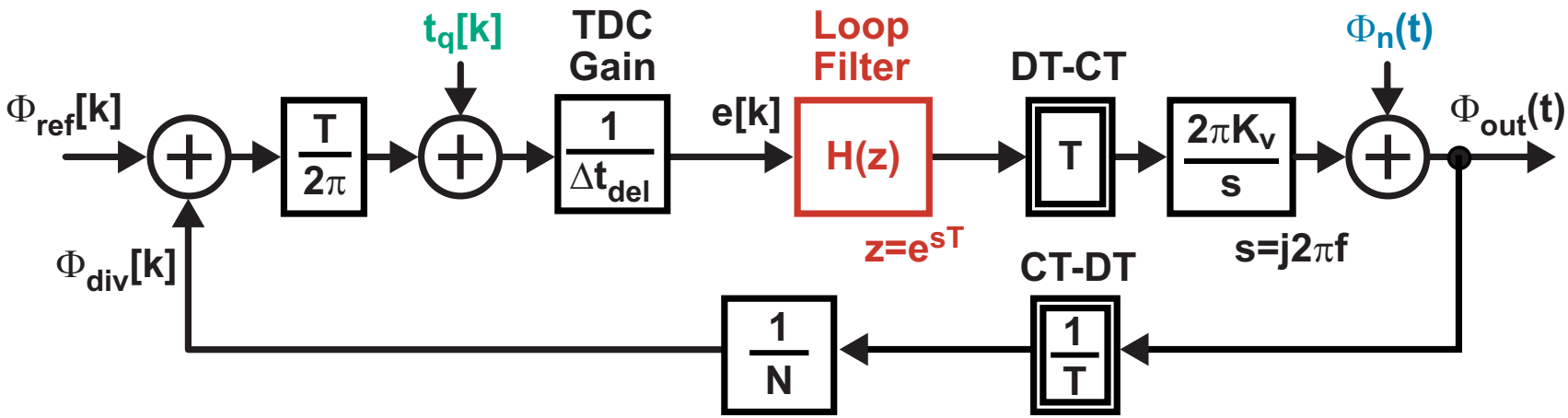
- **PLL Design Assistant assumes continuous-time open loop transfer function $A_{calc}(s)$:**

$$A_{calc}(s) = \frac{K}{s^{type}} \frac{1 + s/w_z}{1 + s/w_p}$$



- **Above parameters are calculated based on the desired closed loop PLL *bandwidth*, *type*, and *order* of rolloff (which specify *G(s)*)**
- **For 100 kHz bandwidth, *type* = 2, 2nd order rolloff, we have:**
  - $K = 3.0 \times 10^{10}$
  - $w_p = 2\pi(153\text{ kHz})$
  - $w_z = 2\pi(10\text{ kHz})$

Continuous-Time Approximation of Digital PLL



- At low frequencies (i.e., $|sT| << 1$), we can use the first order term of a Taylor series expansion to approximate

$$z^{-1} = e^{-sT} \approx 1 - sT$$

- Resulting continuous-time approximation of open loop transfer function of digital PLL:

$$A(s) \approx \frac{T}{\Delta t_{del}} \frac{K_v}{N} \frac{1}{s} H(z) \Big|_{z^{-1} \approx 1 - sT}$$

Applying PLL Design Assistant to Digital PLL Design

- **Given the continuous-time approximation of *A(s)*, we then leverage the PLL Design Assistant calculation:**

$$A(s) = A_{calc}(s)$$

  - **Also note that:**

$$z^{-1} = 1 - sT \Rightarrow \boxed{s = \frac{1 - z^{-1}}{T}}$$

- **Given the above, we obtain:**

$$\left.\frac{T}{\Delta t_{del}} \frac{K_v}{N} \frac{1}{s} H(z)\right|_{s=\frac{1-z^{-1}}{T}} = \left.\frac{K}{s^{type}} \frac{1 + s/w_z}{1 + s/w_p}\right|_{s=\frac{1-z^{-1}}{T}}$$

$$\boxed{\Rightarrow H(z) = \frac{\Delta t_{del}}{T} \frac{N}{K_v} \left(\frac{K}{s^{type-1}}\right) \left.\frac{1 + s/w_z}{1 + s/w_p}\right|_{s=\frac{1-z^{-1}}{T}}}$$

Simplified Form for Digital Loop Filter (Type II PLL)

- **From previous slide:**

$$H(z) = \frac{\Delta t_{del}}{T}\frac{N}{K_v}\left(\frac{K}{s^{type-1}}\right)\frac{1+s/w_z}{1+s/w_p}\Bigg|_{s=\frac{1-z^{-1}}{T}}$$

- **Simplified form with *type = 2* (assume *order = 2*)**

$$H(z) = K_{LF}\left(\frac{1}{1-z^{-1}}\right)\frac{1-b_1 z^{-1}}{1-a_1 z^{-1}}$$

  - **Where:**

$$a_1 = \frac{1}{1+w_p T} \qquad b_1 = \frac{1}{1+w_z T}$$

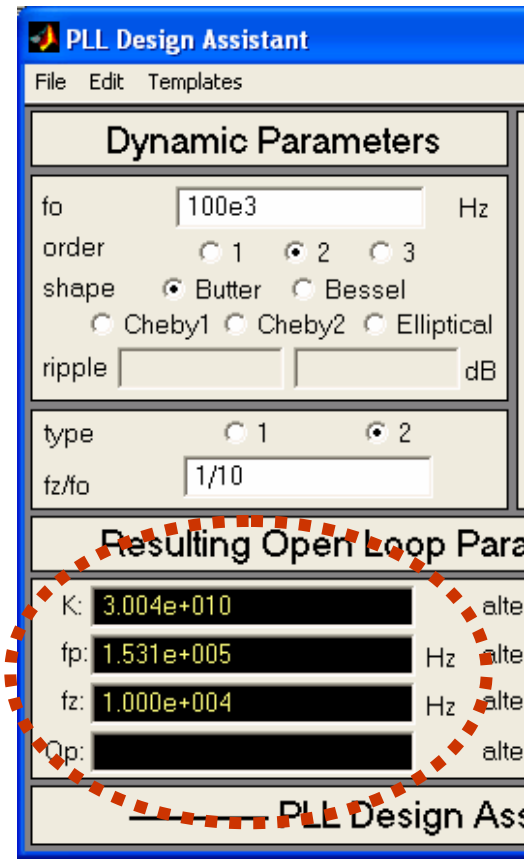
$$K_{LF} = \underbrace{\left(\frac{\Delta t_{del}}{T/N}\right)}_{*}\frac{K}{K_v}\left(\frac{w_p}{w_z}\right)\frac{a_1}{b_1}T$$

Note:
$T_{dco} = T/N$

*** Typically implemented by gain normalization circuit**

Summary of Loop Filter Design

- **PLL Design Assistant allows fast loop filter design**
  - Assumption: *Type* = 2, *2nd* order rolloff



$$H(z) = K_{LF}\left(\frac{1}{1-z^{-1}}\right)\frac{1-b_1 z^{-1}}{1-a_1 z^{-1}}$$

- **Where:**

$$a_1 = \frac{1}{1+w_p T} \qquad b_1 = \frac{1}{1+w_z T}$$

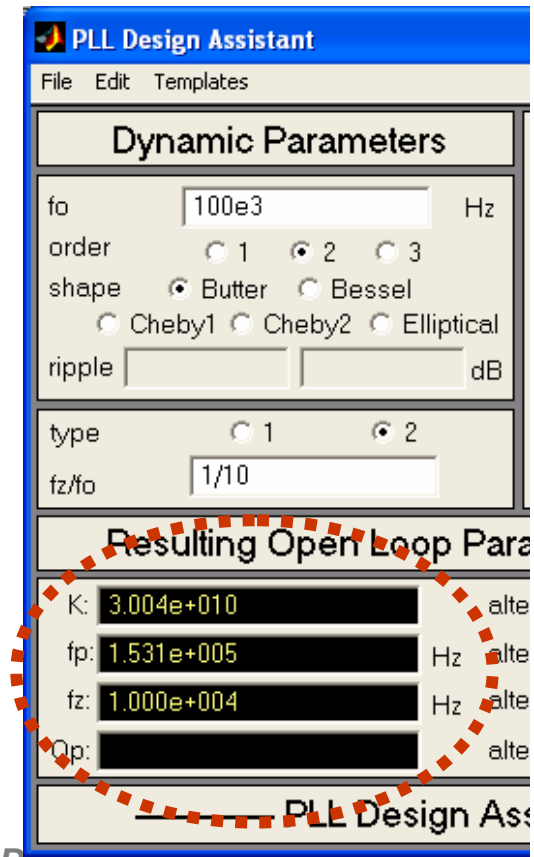
$$K_{LF} = \underbrace{\left(\frac{\Delta t_{del}}{T/N}\right)}_{*}\frac{K}{K_v}\left(\frac{w_p}{w_z}\right)\frac{a_1}{b_1}T$$

*** implemented by gain normalization circuit**

- **PLL Design Assistant provides the values of *K*, $w_p = 2\pi f_p$, $w_z = 2\pi f_z$**

Example Digital Loop Filter Calculation

- **Assumptions**
  - Ref freq (*1/T*) = 50 MHz, Out freq = 3.6 GHz (so *N* = 72)
  - Δt_{del} = 20 ps, K_v = 12 kHz/unit cap
  - 100 kHz bandwidth, *Type* = 2 , *2nd* order rolloff



$$H(z) = K_{LF}\left(\frac{1}{1-z^{-1}}\right)\frac{1-b_1 z^{-1}}{1-a_1 z^{-1}}$$

$$b_1 = \frac{1}{1+2\pi 10\text{kHz}/50\text{MHz}} = \boxed{.9987}$$

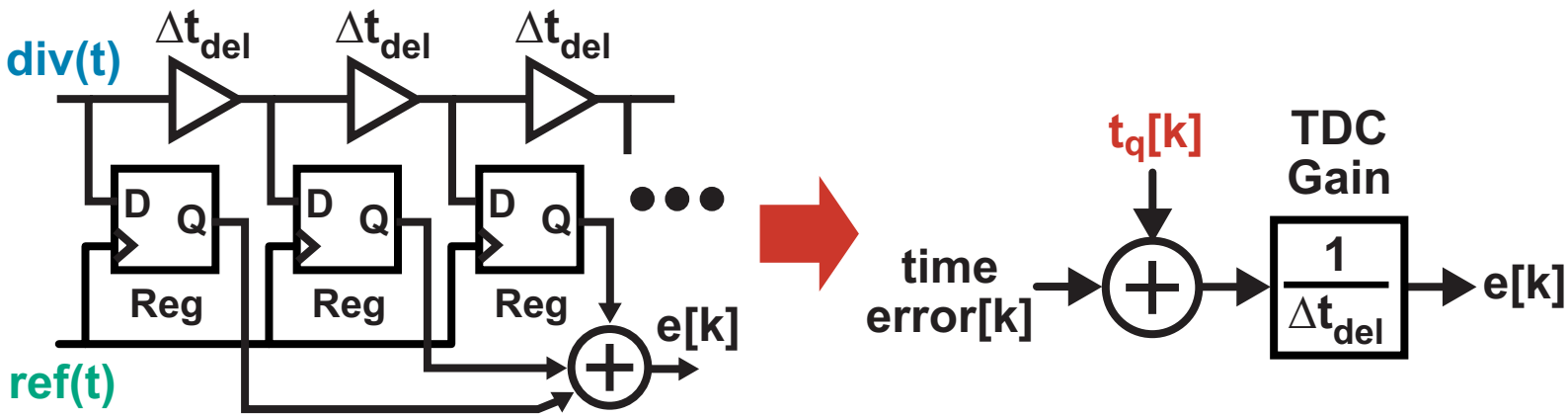
$$a_1 = \frac{1}{1+2\pi 153\text{kHz}/50\text{MHz}} = \boxed{.9811}$$

$$K_{LF} = \left(\frac{\Delta t_{del}}{T/N}\right)\frac{3\cdot 10^{10}}{12\text{kHz}}\frac{153}{10}\frac{.9811}{.9987}\frac{1}{50\text{MHz}}$$

$$= \left(\frac{\Delta t_{del}}{T/N}\right)0.75 = \boxed{\left(\frac{\Delta t_{del}}{T_{dco}}\right)0.75}$$

Overall PLL Noise Analysis

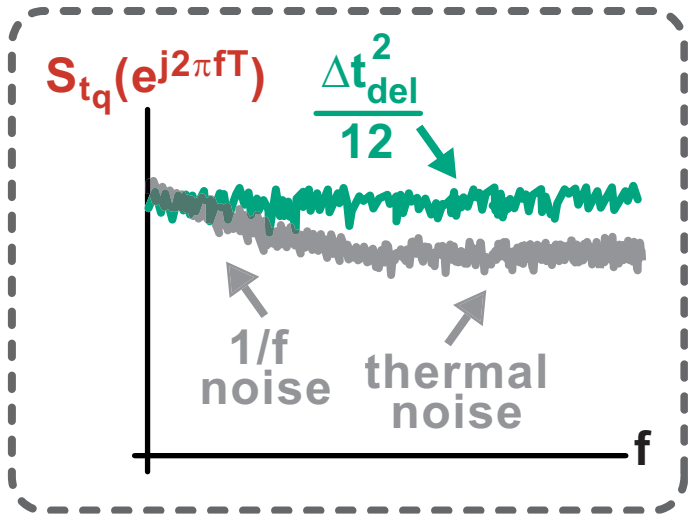
Calculation of TDC Noise Spectrum: Delay Chain TDC



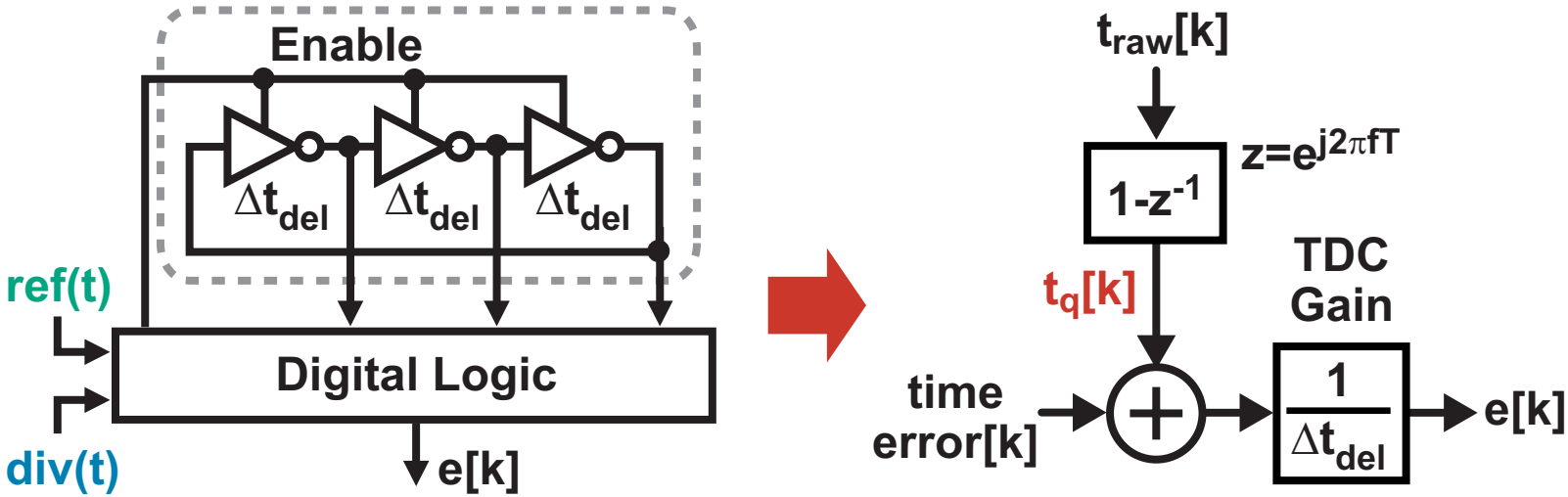
- Under the assumption that quantization error is uniformly distributed across time interval Δt_{del}:

$$S_{t_q}(e^{j2\pi fT}) = \frac{(\Delta t_{del})^2}{12}$$

- Key issue: quantization error may not be white for this TDC!
  - Use behavioral simulations to get a more accurate view
- *1/f* noise may have impact

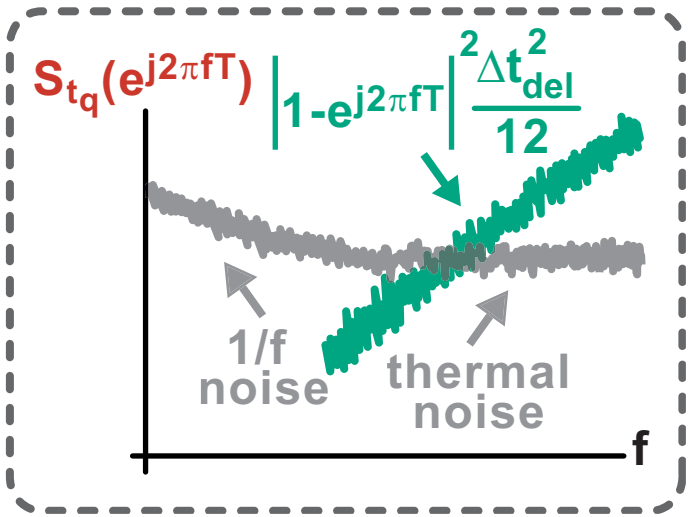


Calculation of TDC Noise Spectrum: GRO TDC



- **GRO achieves noise shaping:**

$$S_{t_q}(e^{j2\pi fT}) = \left|1 - e^{-j2\pi fT}\right|^2 \frac{(\Delta t_{del})^2}{12}$$



- ***1/f* and thermal noise limit noise performance at low frequency offsets**

Example Calculation for Delay Chain TDC

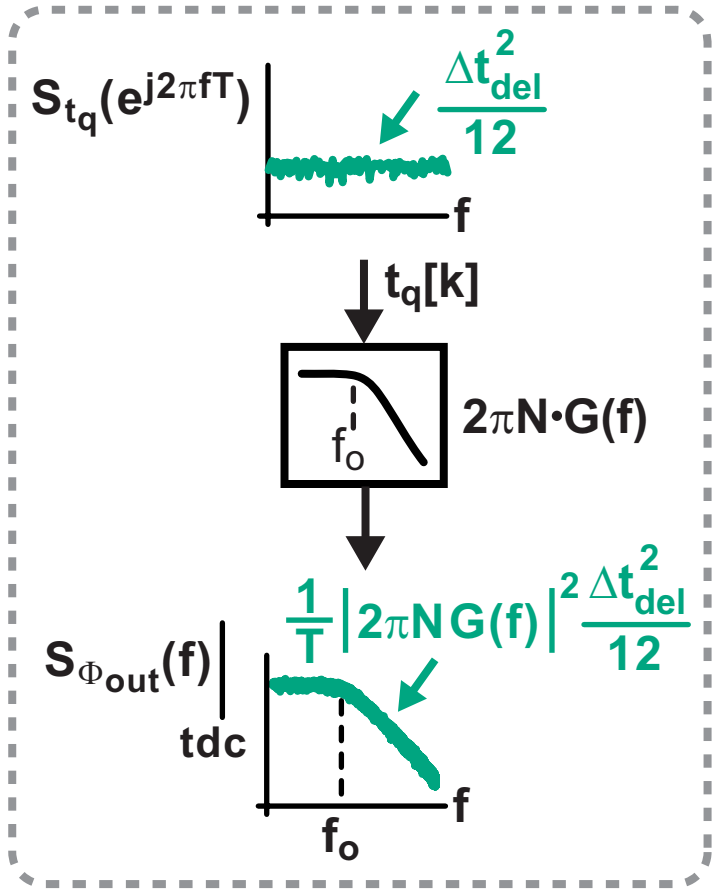
- **Ref freq = *1/T* = 50 MHz, Out freq = 3.6 GHz**

$$\Rightarrow \; N = \frac{3600}{50} = 72$$

- **Inverter delay = Δt_{del} = 20 ps**

$$S_{\Phi_{out}}(f)\Big|_{\text{tdc}} = \frac{1}{T}|2\pi N G(f)|^2 \frac{\Delta t_{del}^2}{12}$$

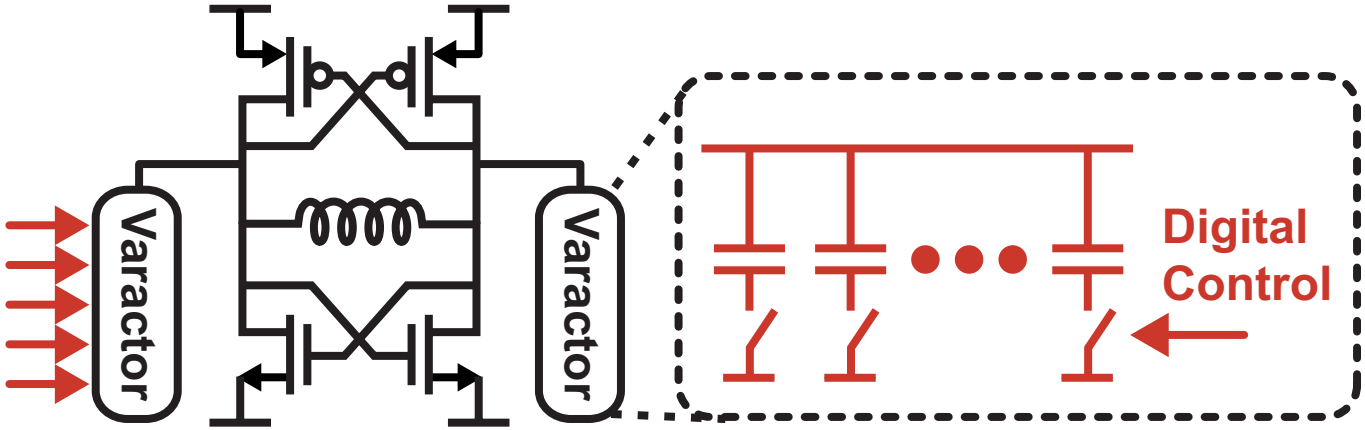
$$= 3.4 \cdot 10^{-10} |G(f)|^2$$



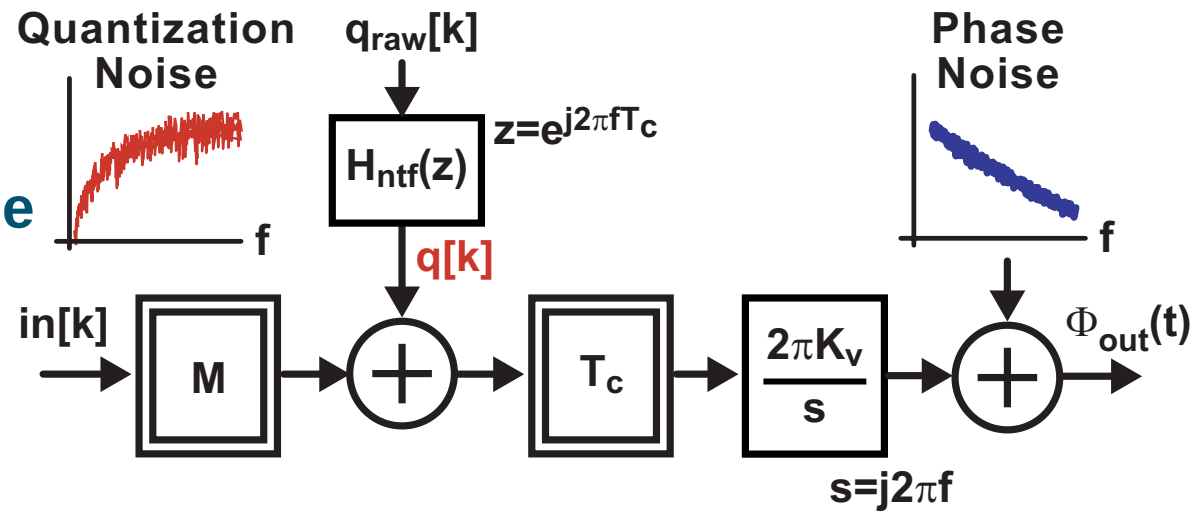
- **Note: *G(f)* = 1 at low offset frequencies**

$$10\log(3.4 \cdot 10^{-10}) = -94.7 \; dBc/Hz \;\; \text{(at low offset freq.)}$$

Calculation of Noise Spectrum: Switched Cap DCO

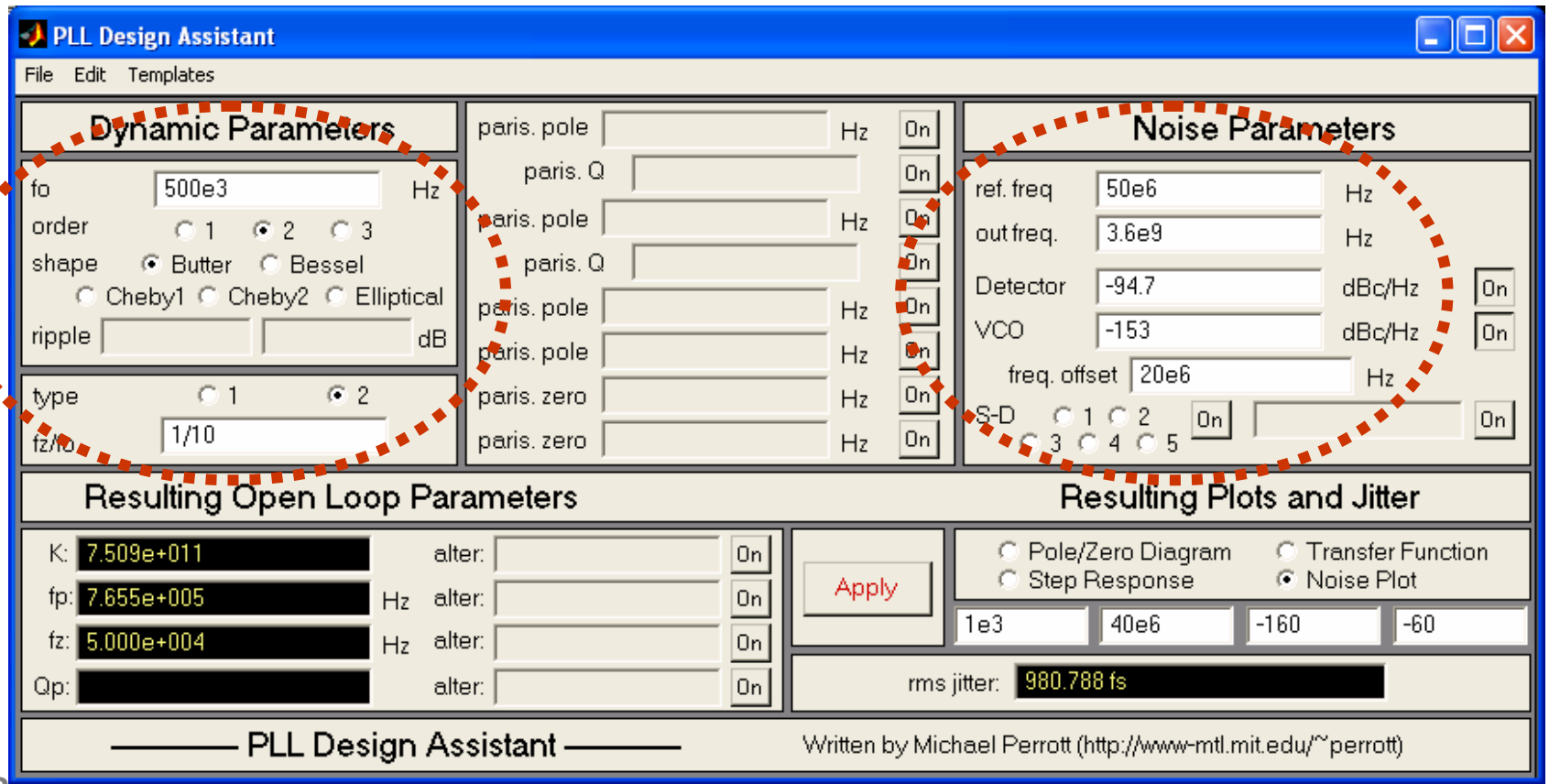


- Phase noise
  - Same as for conventional VCO (tank *Q*, etc.)
- Quantization noise from dithering
  - Design to be less than VCO phase noise

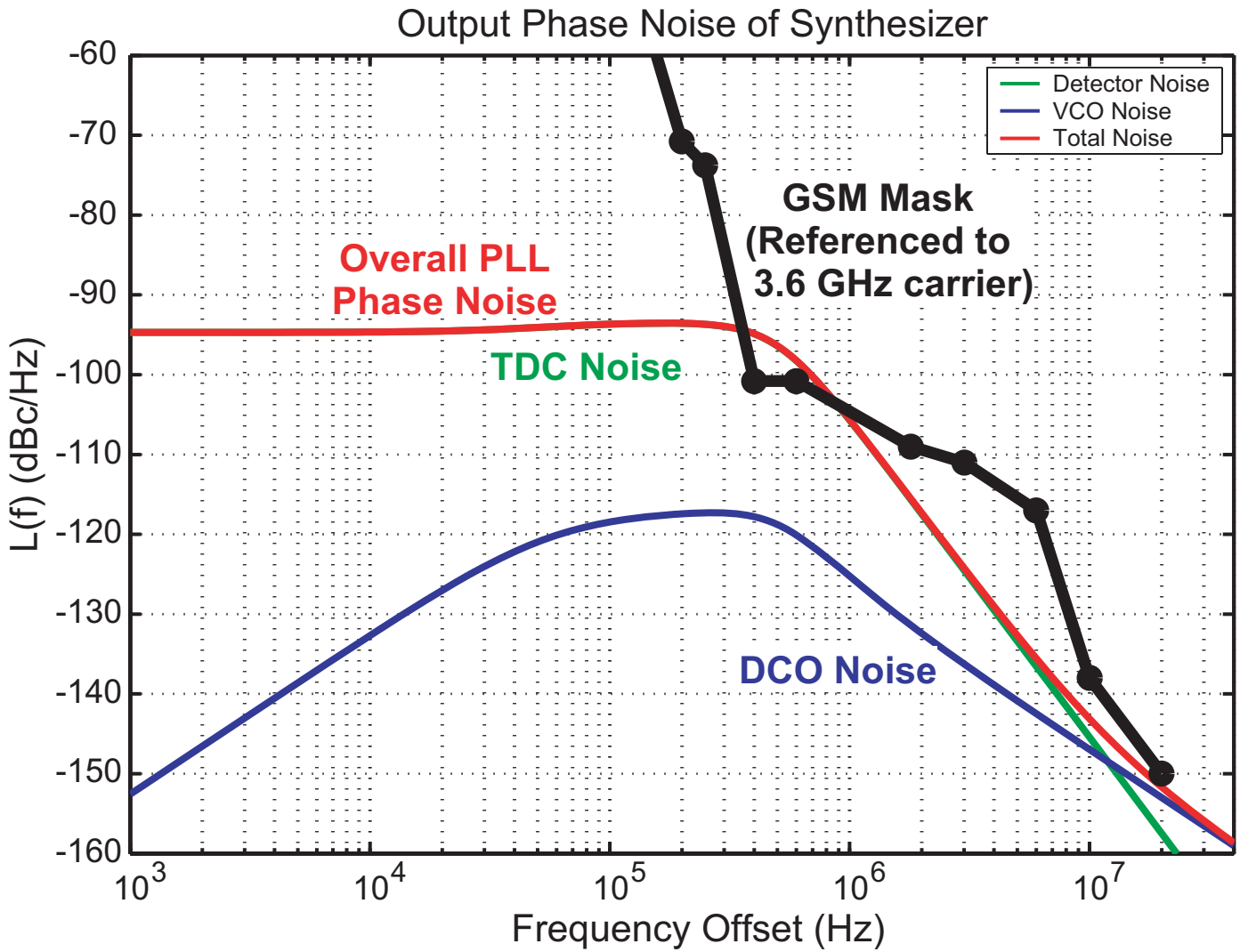


Evaluate Phase Noise with 500 kHz PLL Bandwidth

- **Key PLL parameters:**
  - ***G(f)*: 500 kHz BW, Type II, 2nd order rolloff**
  - **TDC noise: -94.7 dBc/Hz**
  - **DCO noise: -153 dBc/Hz at 20 MHz offset (3.6 GHz carrier)**



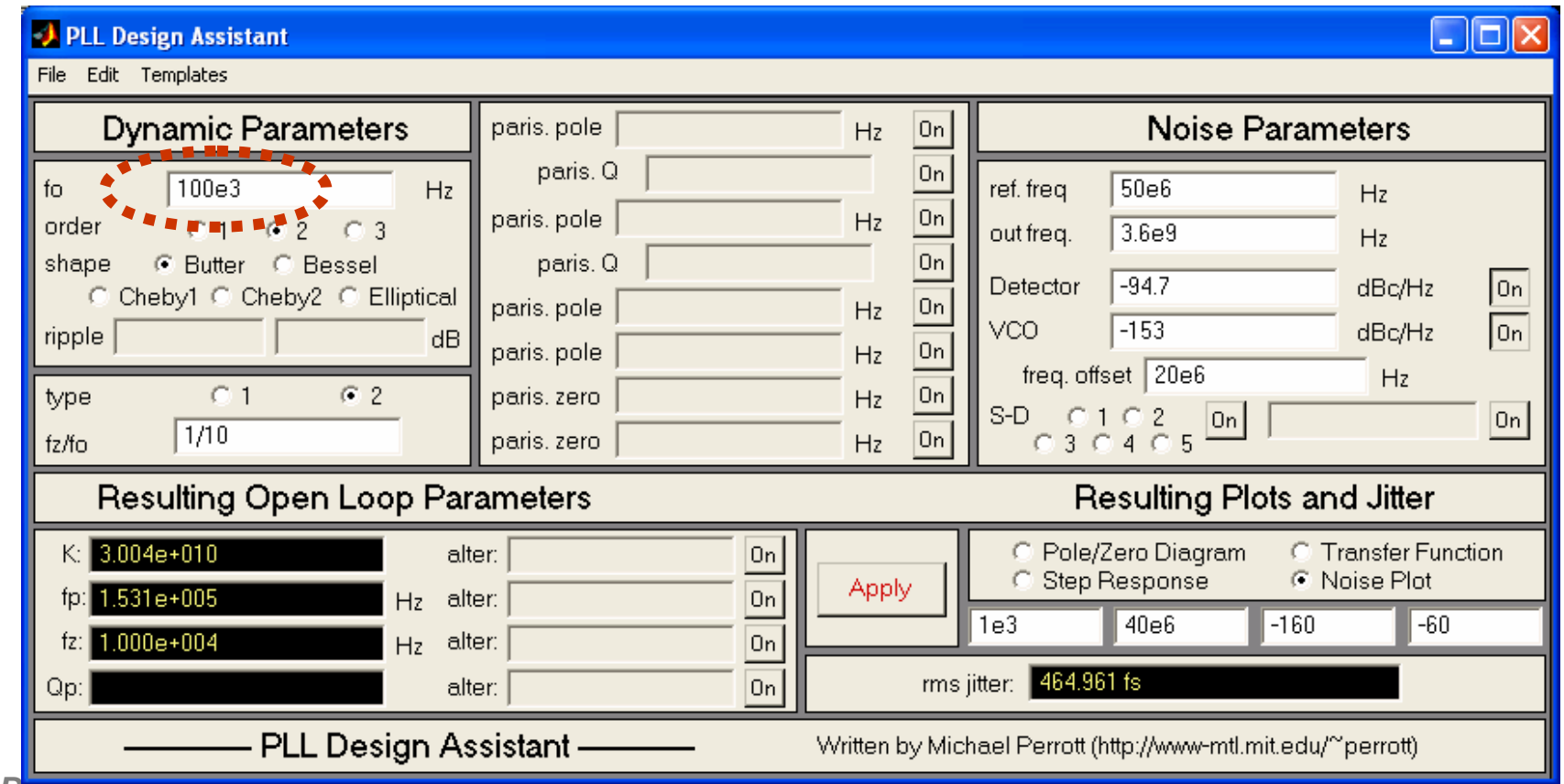
Calculated Phase Noise Spectrum with 500 kHz BW



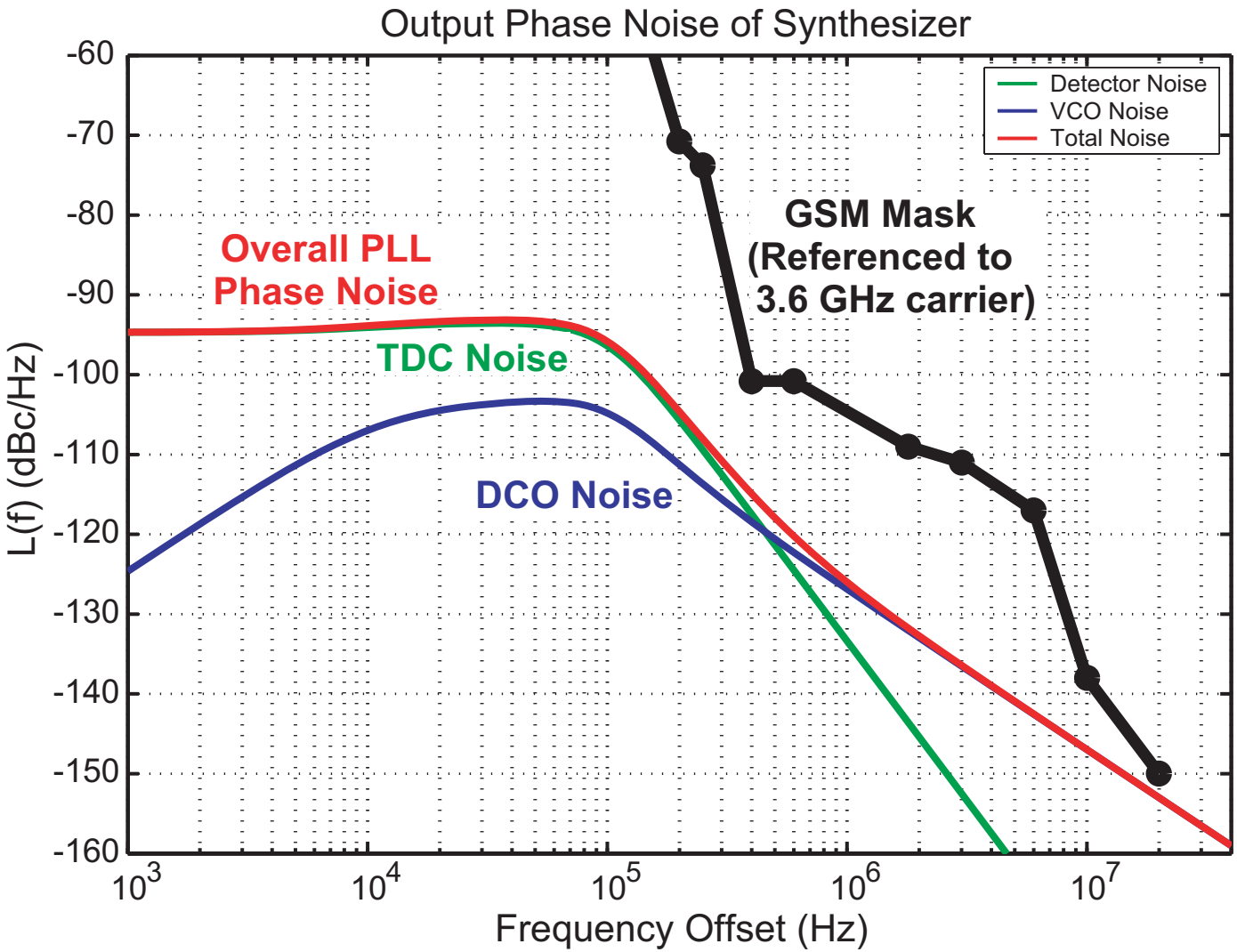
TDC noise too high for GSM mask with 500 kHz PLL bandwidth

Change PLL Bandwidth to 100 kHz

- **Key PLL parameters:**
  - **G(f): 100 kHz BW, *Type* = 2, 2^{nd} order rolloff**
  - **TDC noise: -94.7 dBc/Hz**
  - **DCO noise: -153 dBc/Hz at 20 MHz offset (3.6 GHz carrier)**



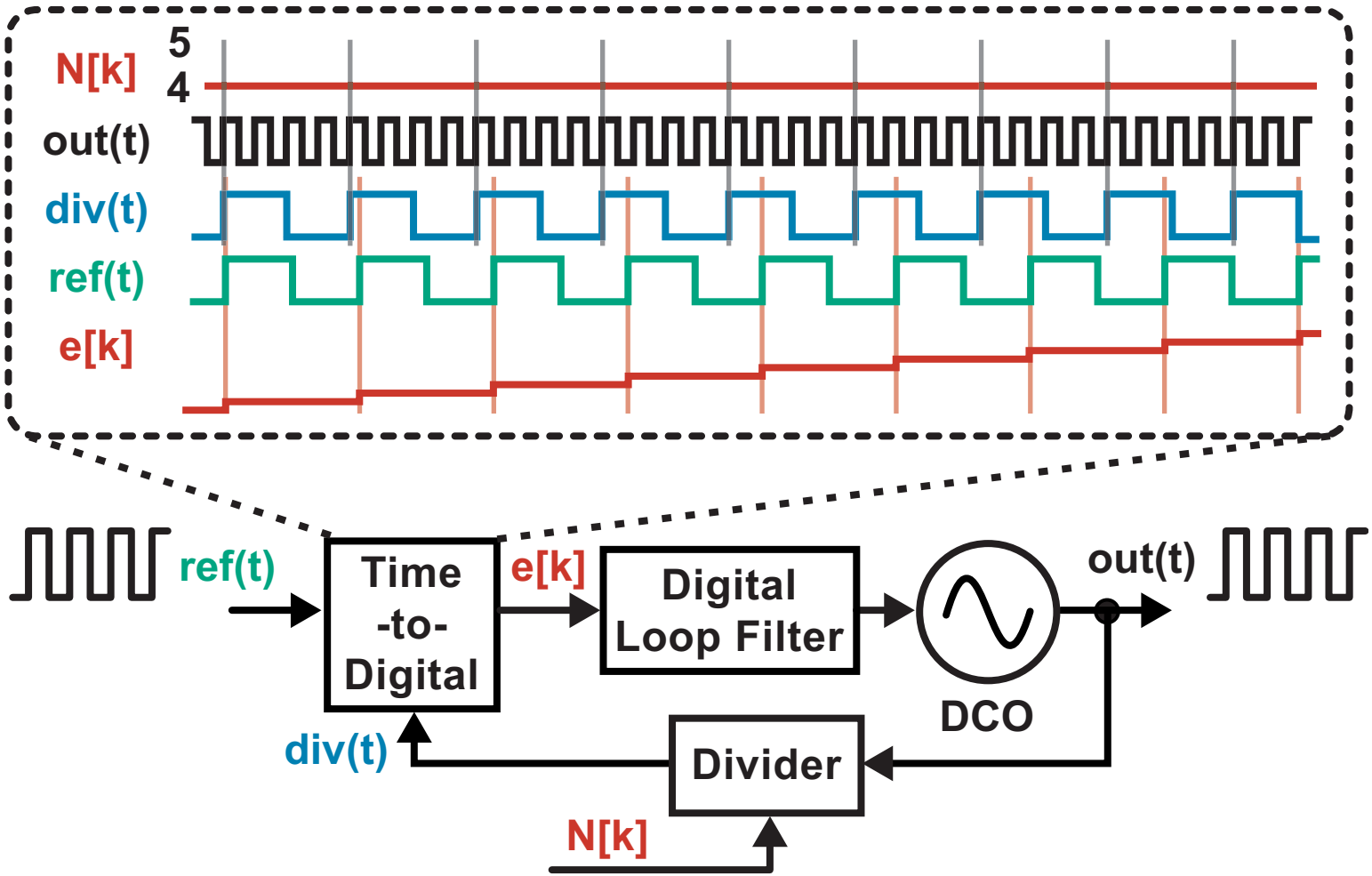
Calculated Phase Noise Spectrum with 100 kHz BW



GSM mask is met with 100 kHz PLL bandwidth

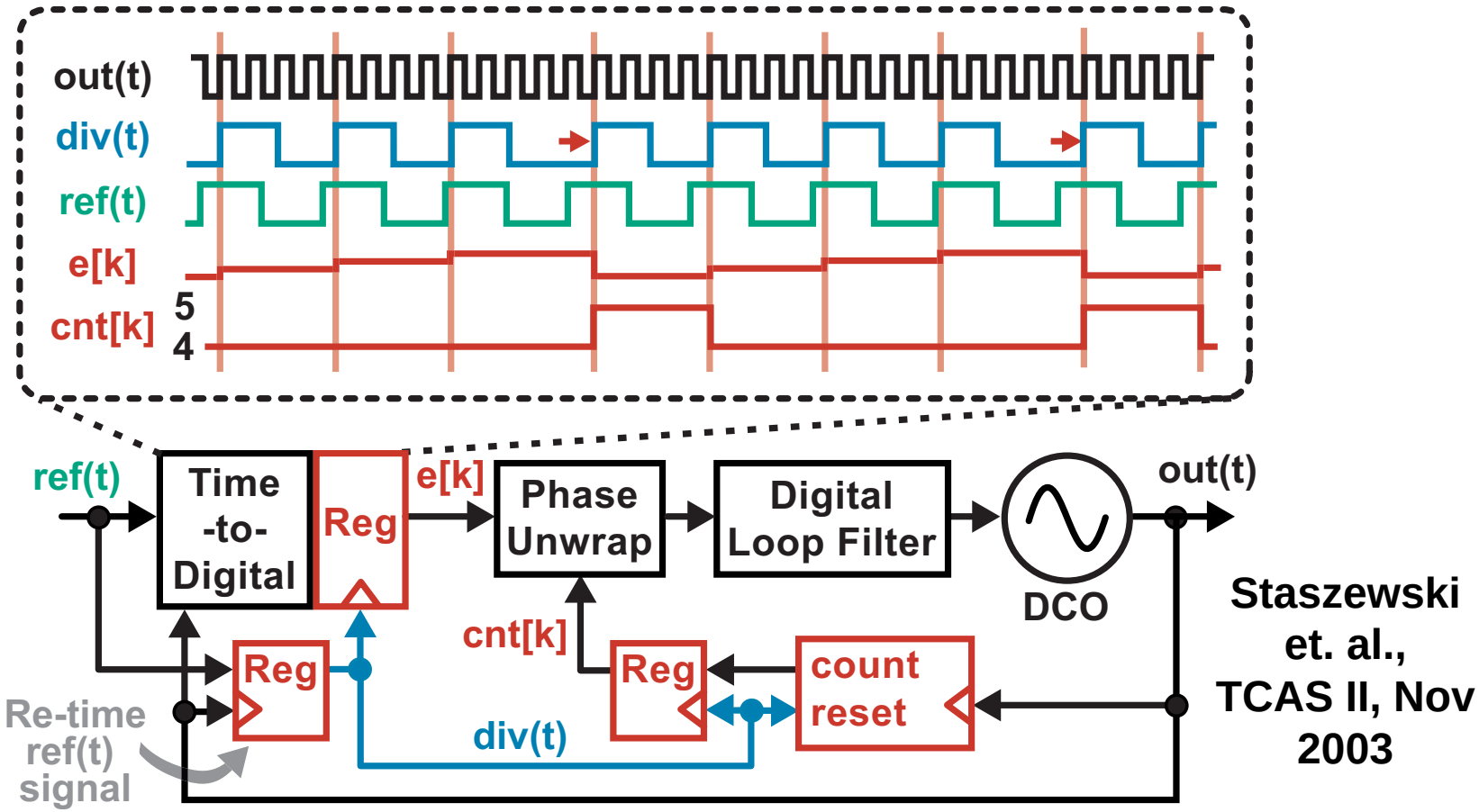
Digital Fractional-N Synthesis

A First Glance at Fractional-N Signals ($F_{out} = 4.25F_{ref}$)



- Constant *divide value* of $N = 4$ leads to frequency error
  - Phase error accumulates in unbounded manner

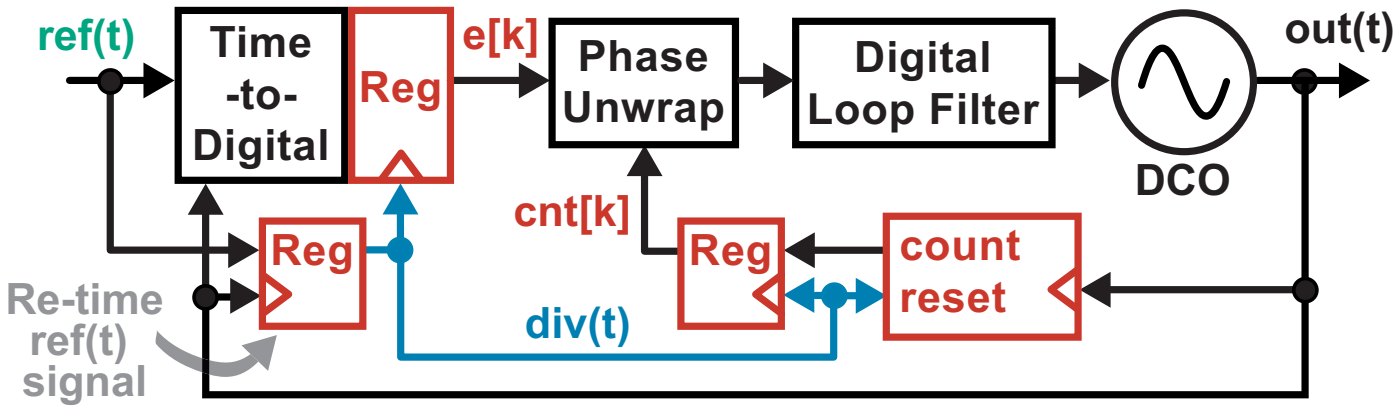
TI Approach to Fractional Division



Staszewski et. al., TCAS II, Nov 2003

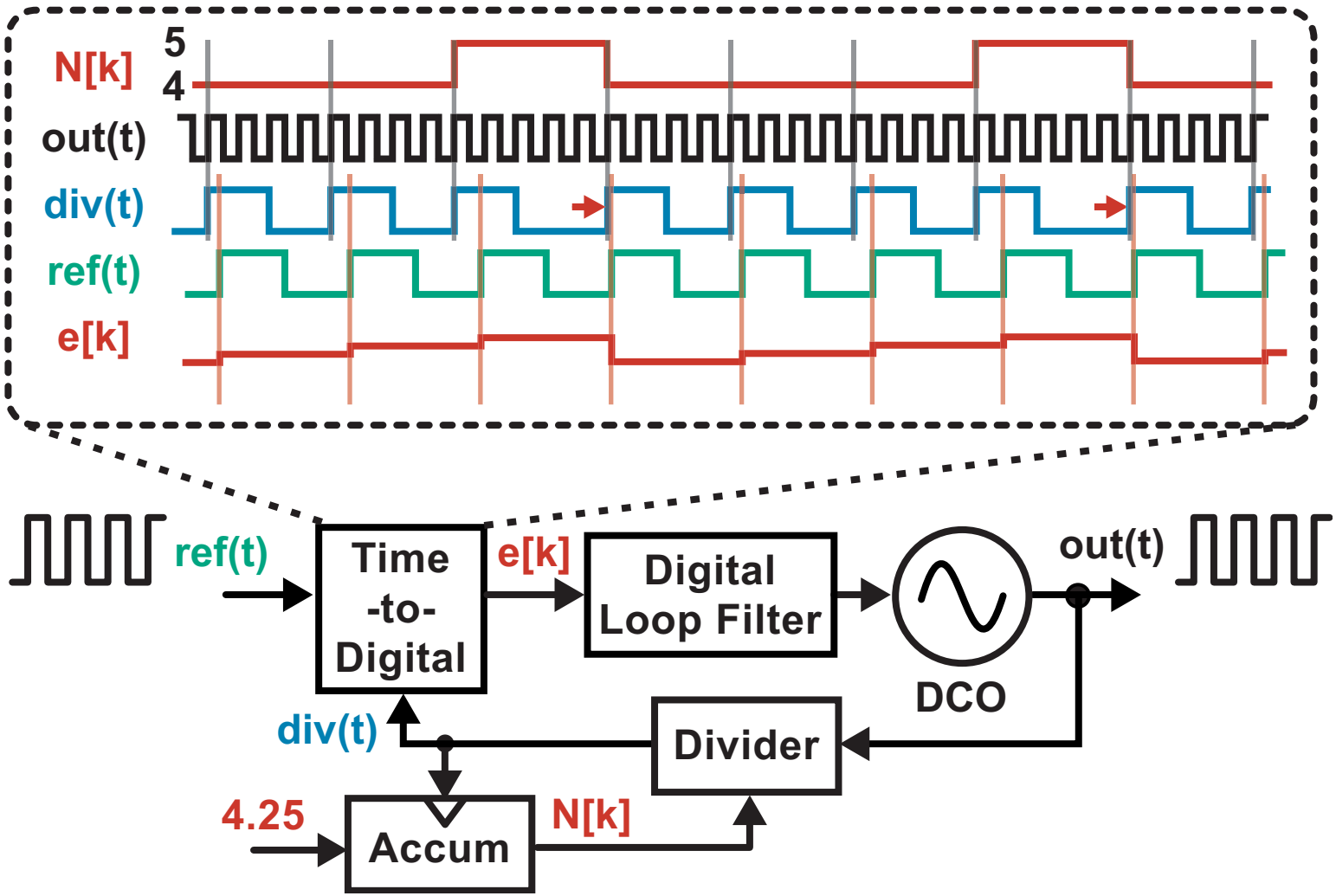
- Wrap *e[k]* by feeding delay chain in TDC with *out(t)*
- Counter provides information of *when* wrapping occurs

Key Issues



- Counter, re-timing register, and delay stages of TDC must operate at very high speeds
  - Power consumption can be an issue
- Calibration of TDC scale factor required to achieve proper unwrapping of *e[k]*
  - Can be achieved continuously with relative ease
    - See Staszewski et. al, JSSC, Dec 2005

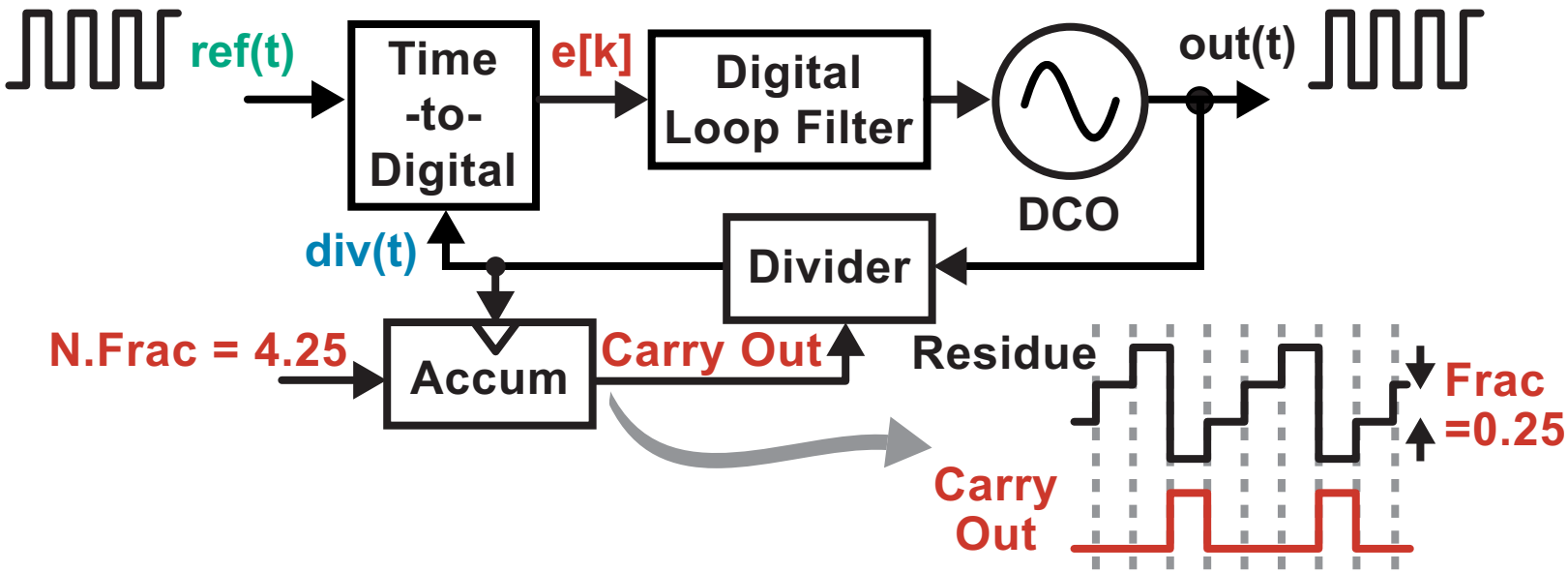
Fractional-N Synthesizer Approach ($F_{out} = 4.25F_{ref}$)



- Accumulator guides the "swallowing" of VCO cycles
  - Average divide value of $N = 4.25$ is achieved in this case

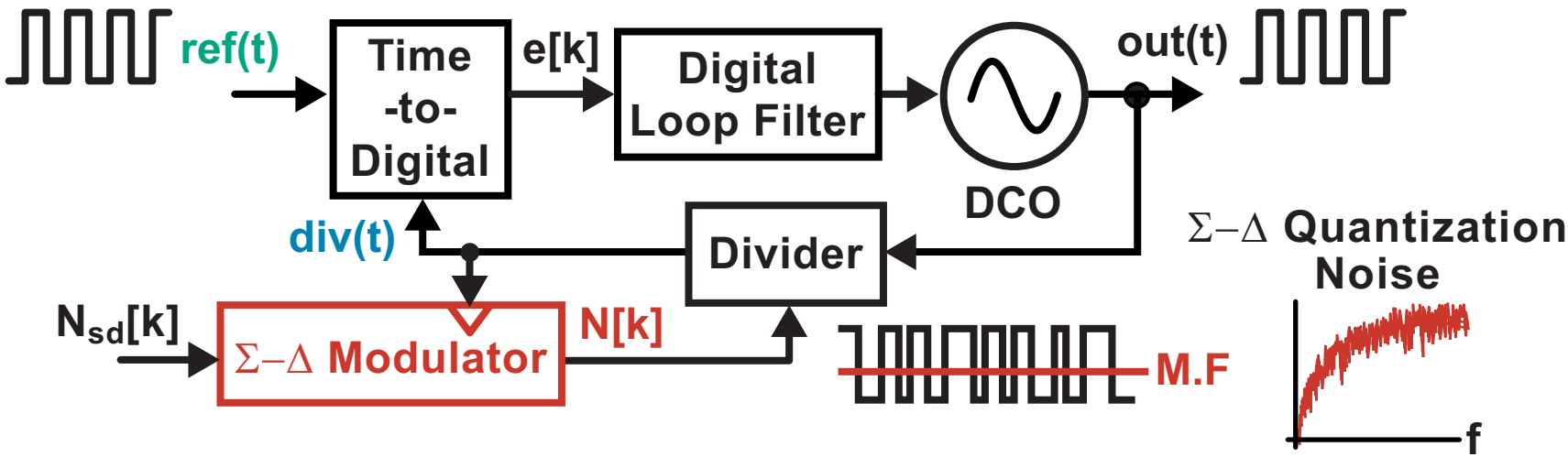
The Accumulator as a Phase "Observer"

- Accumulator residue corresponds to an estimate of the instantaneous phase error of the PLL
  - Fractional value (i.e., 0.25) yields the slope of the residue
- Carry out signal is asserted when the phase error deviation (i.e. residue) exceeds one VCO cycle
  - Carry out signal accurately predicts when a VCO cycle should be "swallowed"

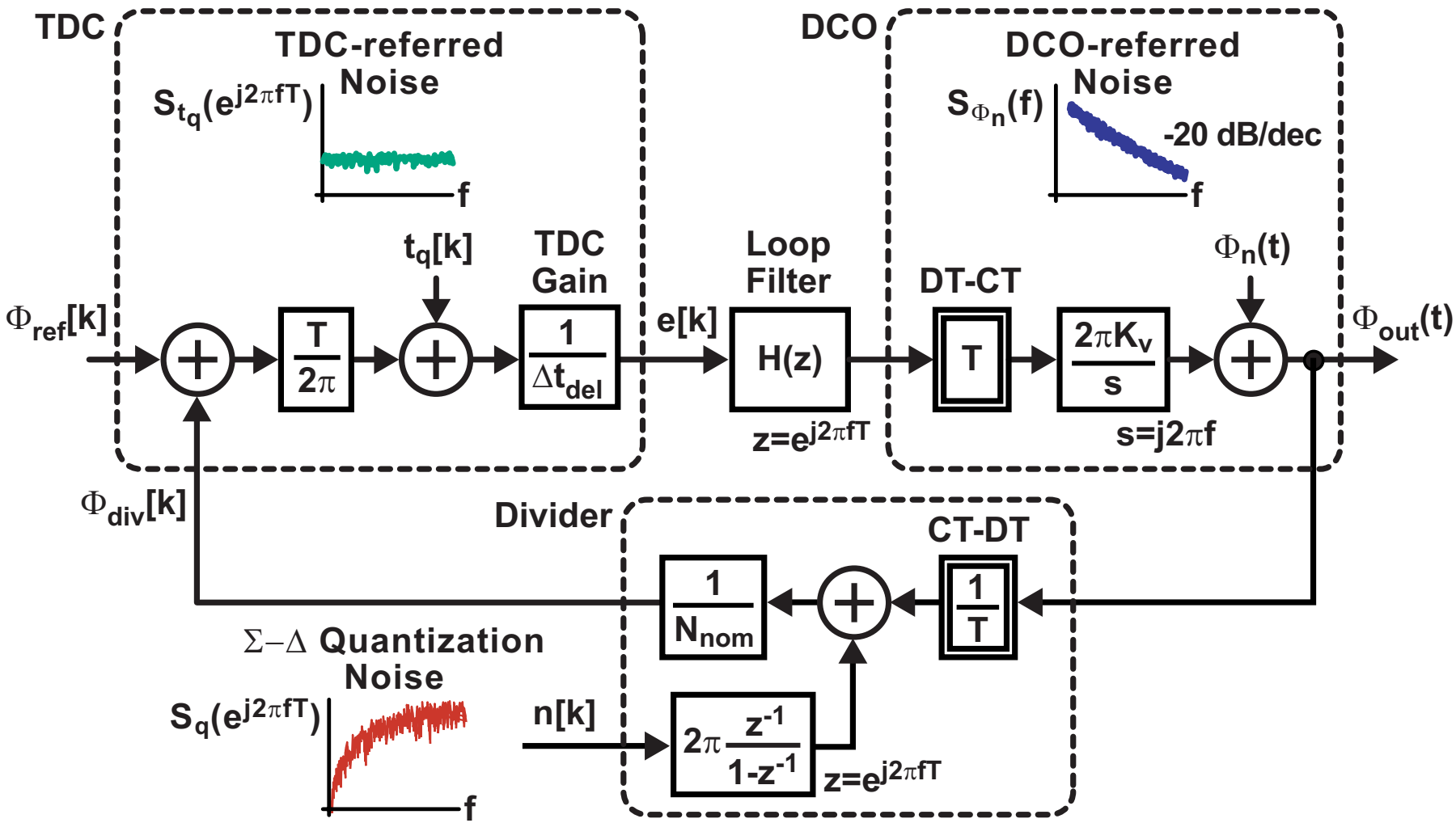


Improve Dithering Using Sigma-Delta Modulation

- **Provides improved noise performance over accumulator-based divide value dithering**
  - **Dramatic reduction of spurious noise**
  - **Noise shaping for improved in-band noise**
  - **Maintains bounded phase error signal**
- **Digital Σ–Δ fractional-N synthesizer architecture is directly analogous to analog Σ–Δ fractional-N synth.**

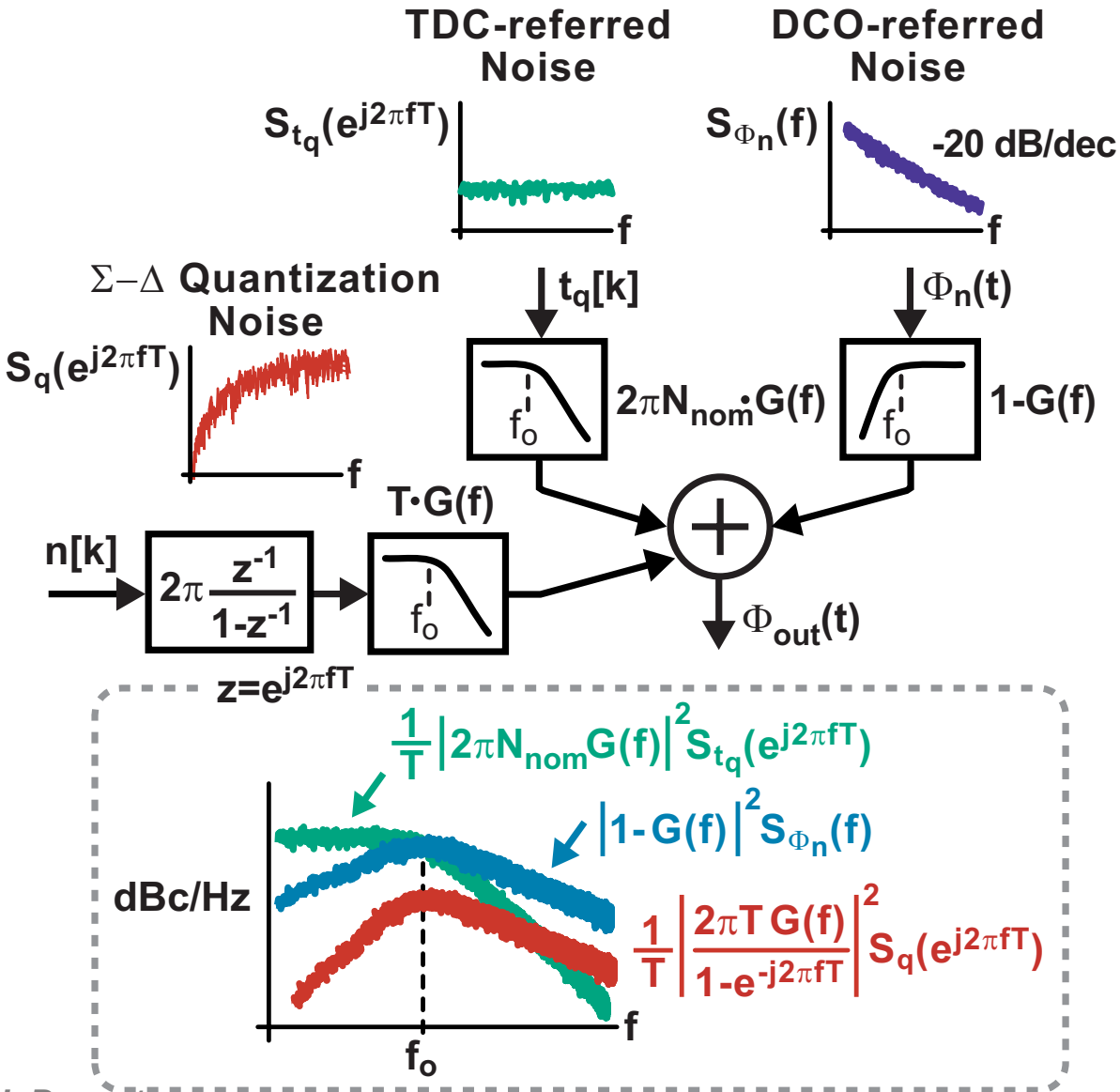


Model of Digital Σ–Δ Fractional-N PLL



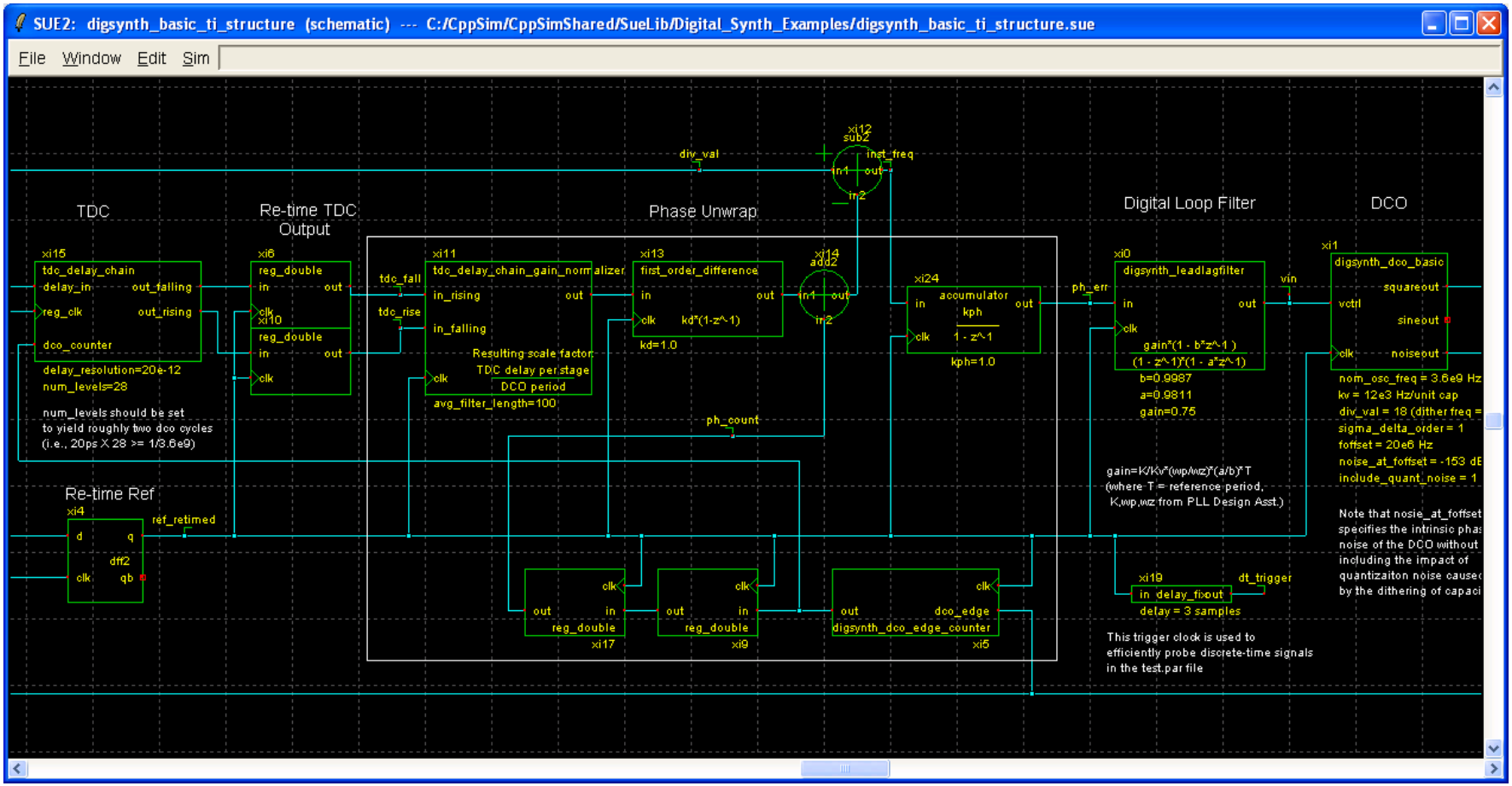
- Divider model is expanded to include the impact of divide value variations

Transfer Function View of Digital Σ–Δ Fractional-N PLL



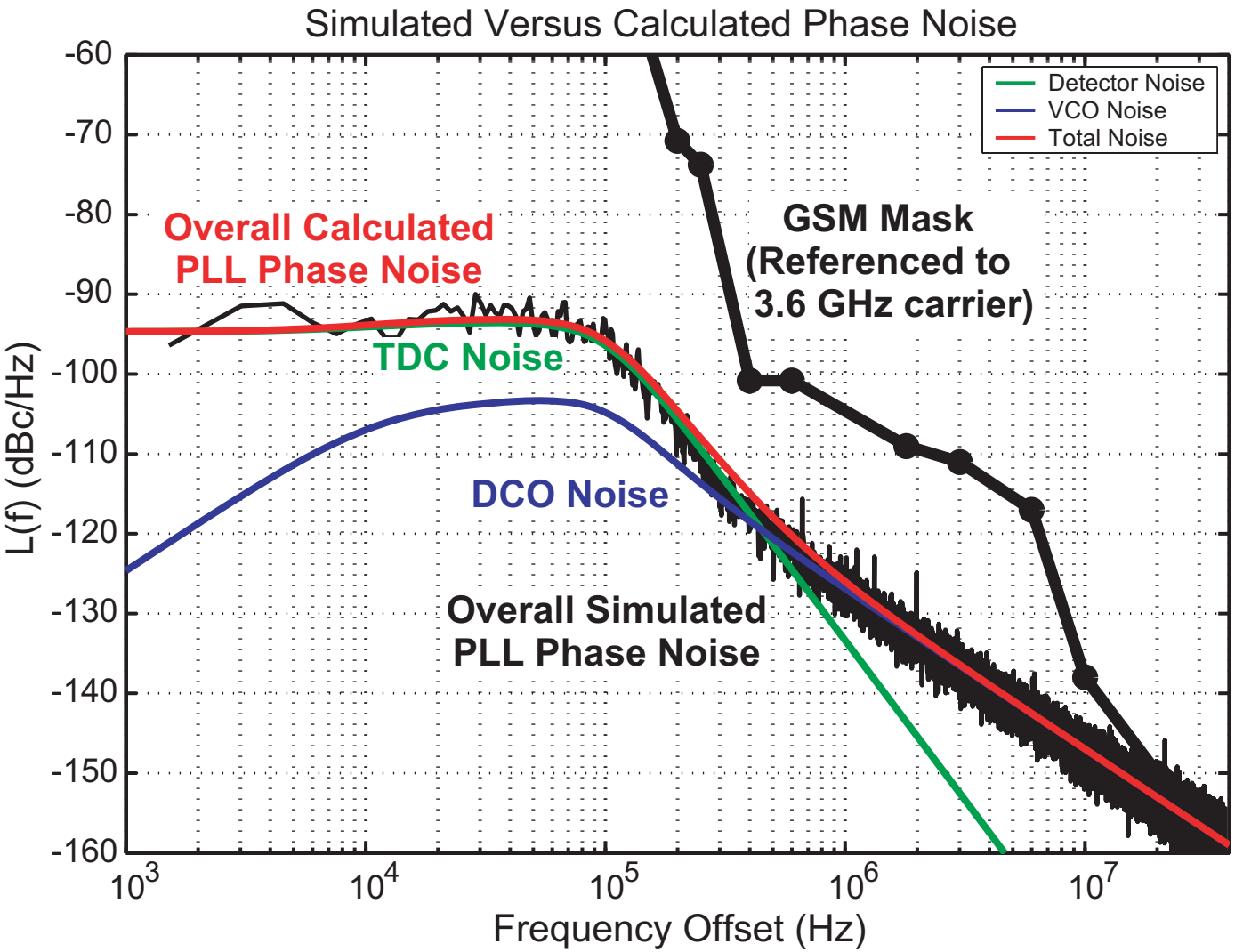
- Σ–Δ quantization noise now impacts the overall PLL phase noise
  - High PLL bandwidth will increase its impact
- Digital PLL implementation simplifies quantization noise cancellation

CppSim Behavioral Model of TI Digital Synthesizer



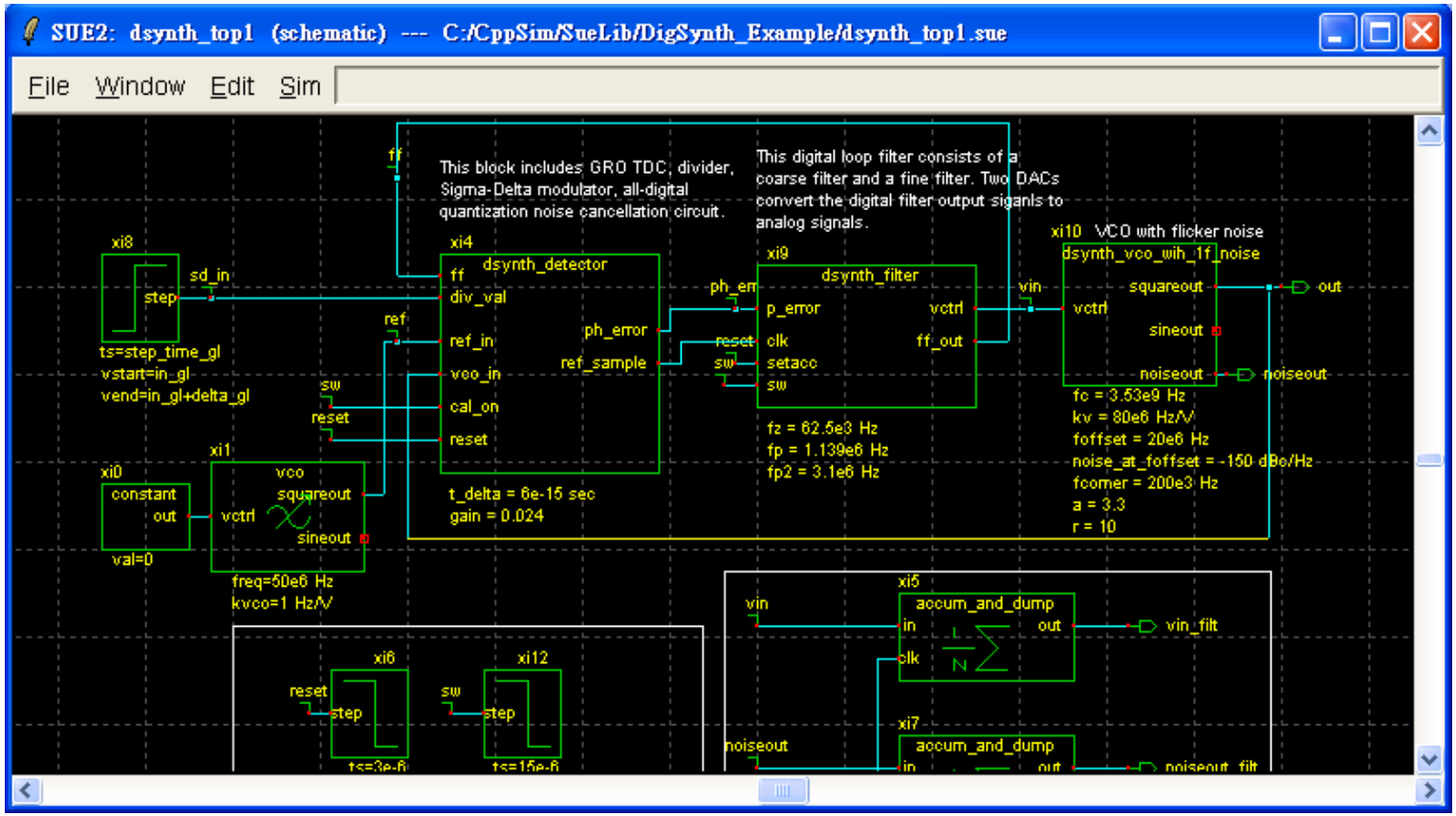
- **Implements basic version of TI "all-digital" synthesizer with parameters we calculated in this tutorial**

Comparing Behavioral Simulation to Calculations



- **Calculations validated by simulation results!**

Behavioral Simulation of a Digital Fractional-N PLL



- **Check out the CppSim tutorial:**
  - **Design of a Low-Noise Wide-BW 3.6GHz Digital Σ–Δ Fractional-N Frequency Synthesizer Using the PLL Design Assistant and CppSim**

http://www.cppsim.com

Summary of Digital Frequency Synthesizers

- **Digital Phase-Locked Loops look extremely promising for future applications**
  - **Very amenable to future CMOS processes**
  - **Excellent performance can be achieved**
- **TDC structures are an exciting research area**
  - **Ideas from A-to-D conversion can be applied**
- **Analysis of digital PLLs is similar to analog PLLs**
  - **PLL bandwidth is often chosen for best noise performance**
    - TDC (or Ref) noise dominates at low frequency offsets
    - DCO noise dominates at high frequency offsets
- **Behavioral simulation tools such as CppSim allow architectural investigation and validation of calculations**

Innovation of future digital PLLs will involve joint circuit/algorithm development