Subsampling PLLs for Frequency Synthesis and Phase Modulation

Nereo Markulic

OUTLINE

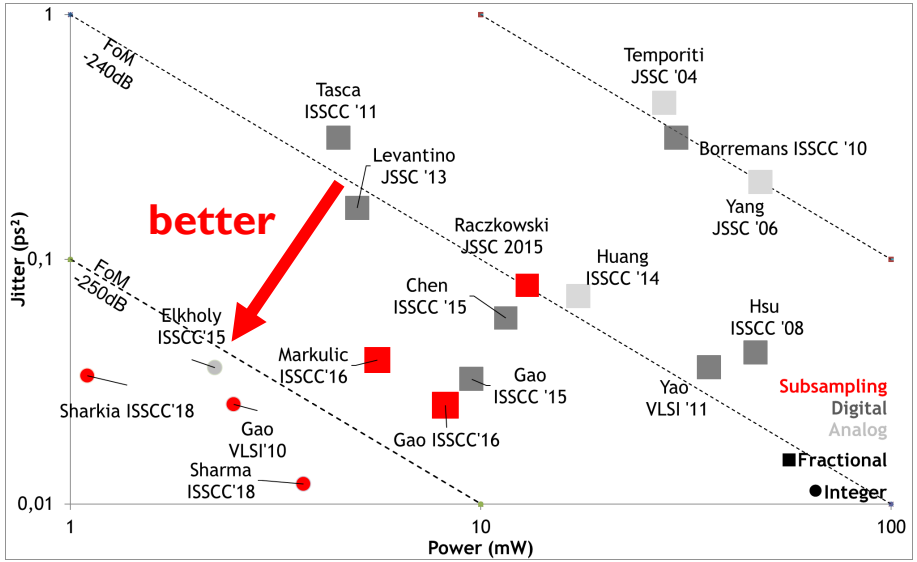
From a Subsampling PLL to a Background-Calibrated Subsampling Phase Modulator

- Subsampling PLL basics
- Fractional Synthesis in a Subsampling PLL
- Digital-to-Time Converter
- Background calibration techniques
- Subsampling PLL-based Phase Modulator

imec

SUBSAMPLING PLL BASICS

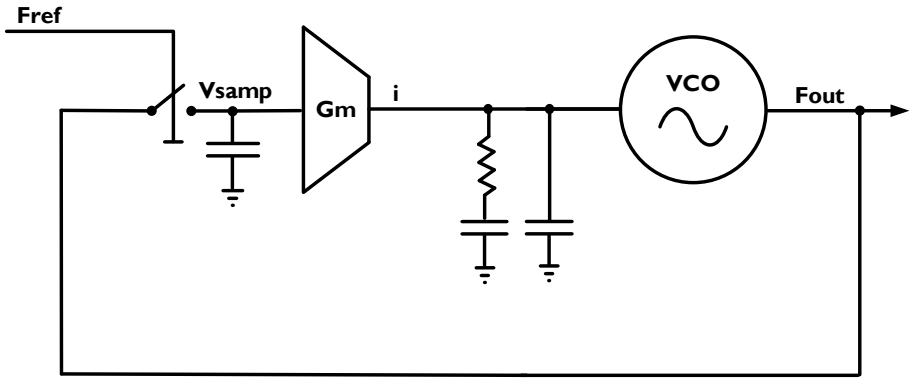
Why is subsampling PLL so popular?



- Leading state-of-the art designs
- Superior jitter versus power trade-off
  - How? We will answer that in a bit.

SUBSAMPLING PLL BASICS

Why is subsampling PLL so popular?



- Simple analog loop
  - No digitization complexity
  - Well-known equations
- Scaling friendly
  - *S&H* as a phase-error detector
  - Transconductor for *current* integration
  - *Small* loop filter
- Suitable for RF/MMW
  - No power-hungry dividers

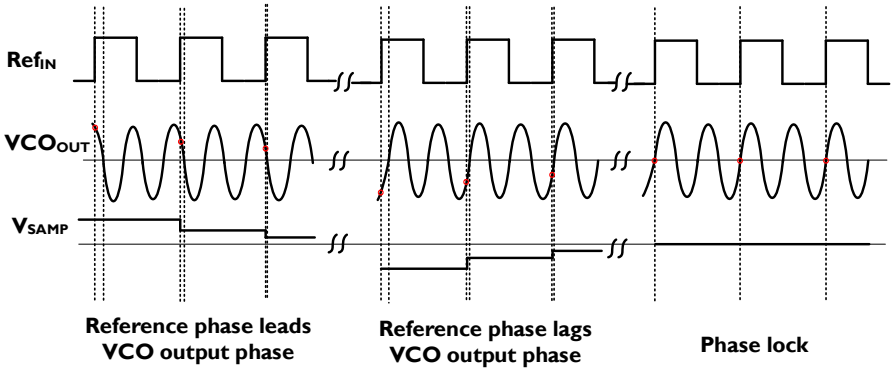
OUTLINE

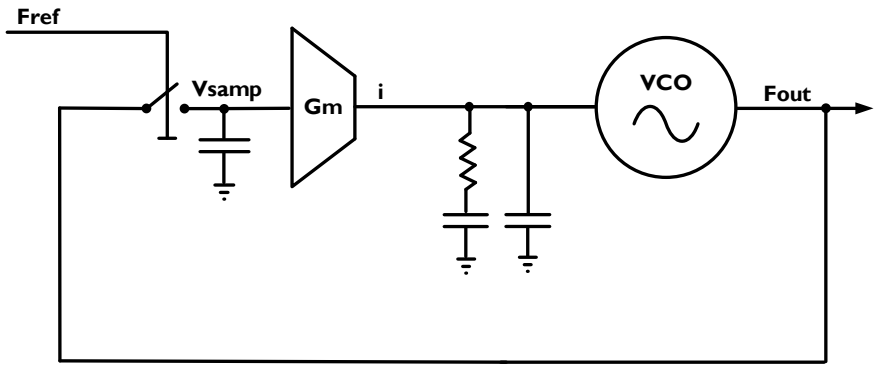
From a Subsampling PLL to a Background-Calibrated Subsampling Phase Modulator

- **Subsampling PLL basics**
- Fractional Synthesis in a Subsampling PLL
- Digital-to-Time Converter
- Background calibration techniques
- Subsampling PLL-based Phase Modulator

SUBSAMPLING PLL BASICS

How does it work?

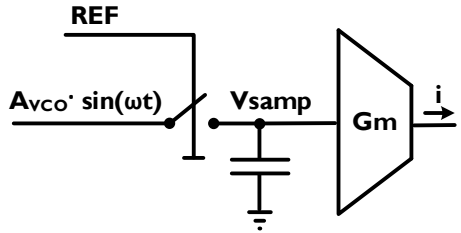


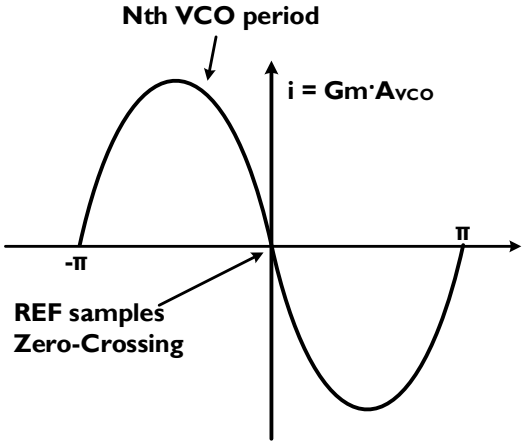


- Multiplication by 2 example
- If reference edge leads (/lags) the VCO edge, positive (/negative) voltage is sampled
  - Positive (/negative) current integrated onto the loop filter
  - Loop filter voltage increased (/decreased)
  - VCO frequency decreased (/increased)

SUBSAMPLING PLL BASICS

If it is so easy, lets build one!



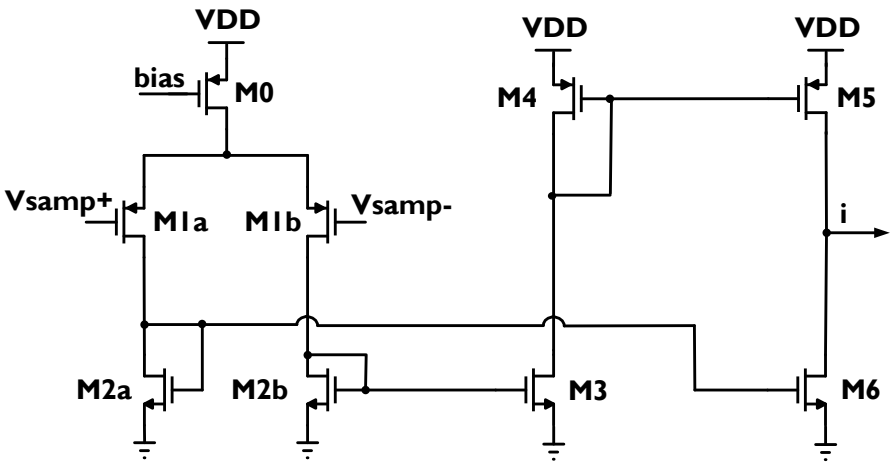


- Phase error detection gain
  - VCO zero-crossing slew rate dependent
  - (Comparably) *high* at RF

- $\mathrm{PD(s)} = \frac{\mathrm{i}}{\Delta\Phi} = \frac{\mathrm{G_m A_{vco}}\sin(\Delta\Phi)}{\Delta\Phi} = \mathrm{G_m A_{vco}}$

SUBSAMPLING PLL BASICS

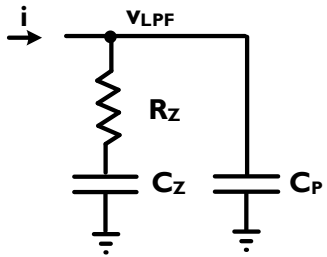
Transconductor



- Transconductor *resembles* a charge pump
  - No up/down pulses
  - Constant input voltage
- Transconductance defined by
  - Input pair Gm
  - Mirroring ratios

SUBSAMPLING PLL BASICS

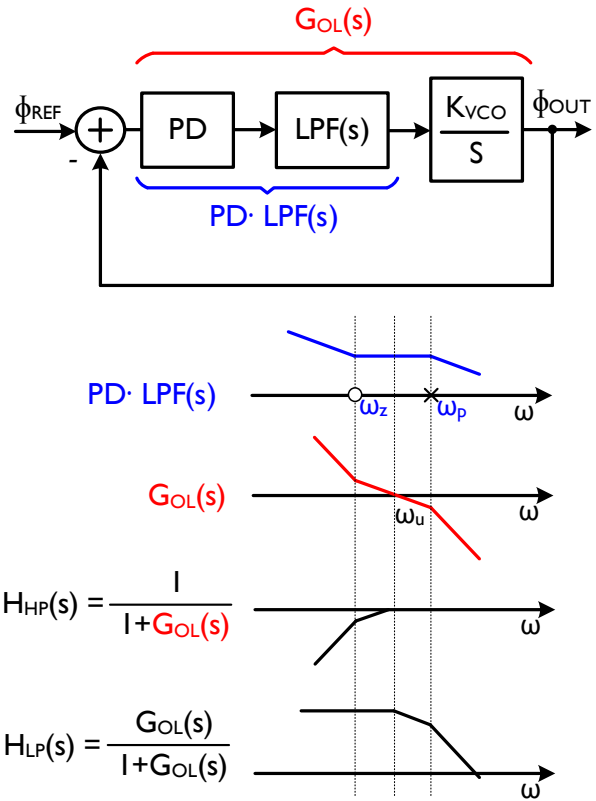
Loop filter



- Analog loop filter
  - $\mathrm{LPF}(s) = \frac{1+sR_zC_z}{s(C_Z+C_P)\left(1+s\frac{C_Z \cdot C_P}{C_Z+C_P}R_Z\right)}$
  - $= \frac{(1+s\tau_z)}{s(C_Z+C_P)(1+s\tau_p)}$

SUBSAMPLING PLL BASICS

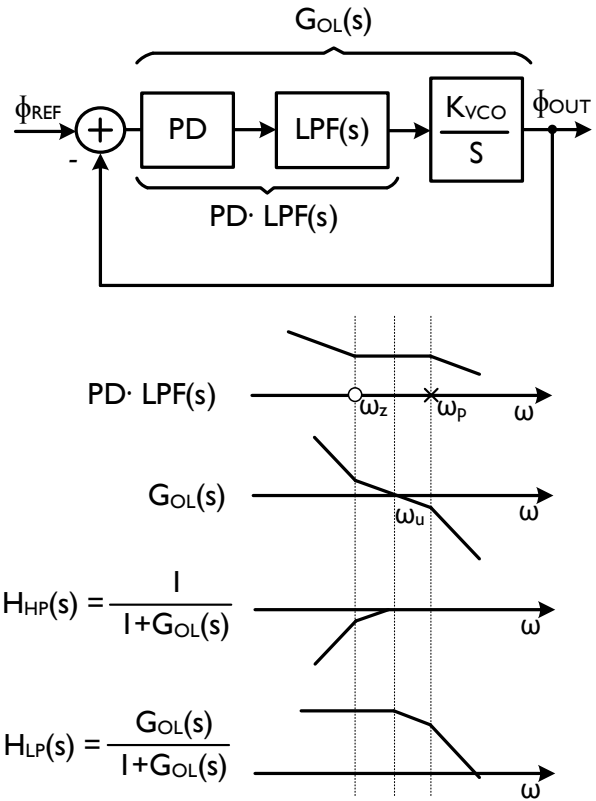
Loop stability



- Well known analog PLL equations
- $PD \cdot LPF(s) = \frac{G_m A_{vco}}{s(C_Z+C_P)} \frac{(1+s\tau_z)}{(1+s\tau_p)}$
- $G_{OL}(s) = \frac{G_m A_{vco}}{s(C_Z+C_P)} \frac{(1+s\tau_z)}{(1+s\tau_p)} \frac{K_{vco}}{s}$
- $G_{OL}(s) = 1, \text{at } \omega_u.$
- For stability, choose:
  - $\omega_Z = \frac{\omega_u}{4}$
  - $\omega_p = 4\,\omega_u$

SUBSAMPLING PLL BASICS

Loop stability



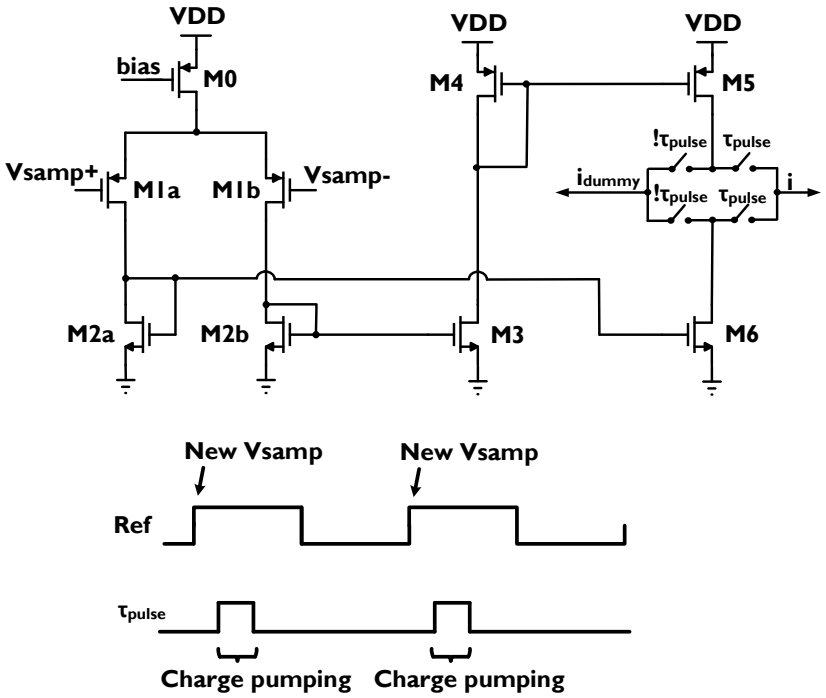
- For example, if
  - $\mathrm{Fref} = 40\ \mathrm{MHz}$
  - $\mathrm{Fvco} = 1\ \mathrm{GHz}$
  - $\omega_u = 1\ \mathrm{MHz}$
  - $G_m = 100\ \mu\mathrm{S}$
  - $K_{VCO} = 100\ \frac{\mathrm{MHz}}{\mathrm{V}}$
  - $A_{VCO} = 1\ \mathrm{V}$

  and
  - $\omega_Z = \frac{\omega_u}{4}$
  - $\omega_p = 4\ \omega_u$

- Then
  - $C_Z = 900\ \mathrm{pF}$ ← large due to *too* large PD gain
  - $C_P = 75\ \mathrm{pF}$
  - $R_Z = 700\ \Omega$

SUBSAMPLING PLL BASICS

Pulsed transconductor for PD gain reduction



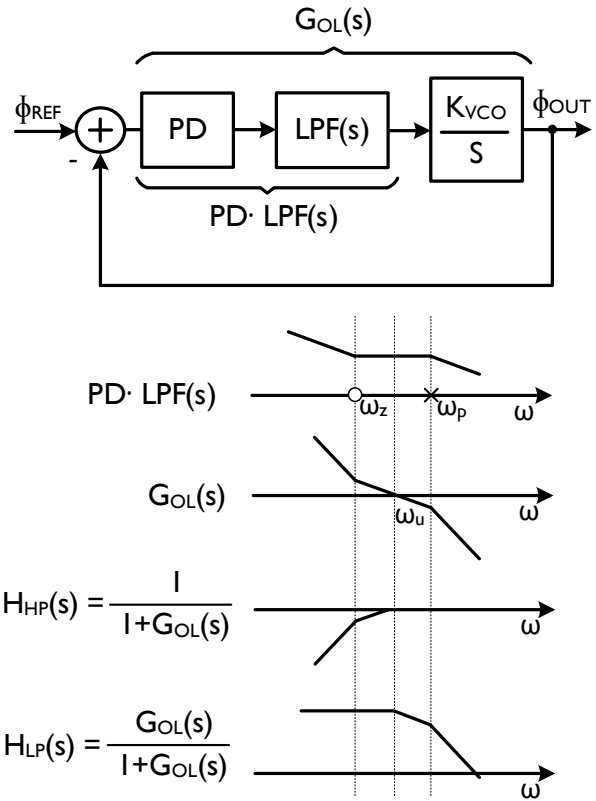
- Current pumped for a *portion* of the reference period
  - Timing not critical
  - Enables inherit sample and hold operation

- $PD(s) = G_m A_{vco} \frac{\tau_{pulse}}{T_{ref}}$

- $G_{OL}(s) = \frac{G_m A_{vco} \frac{\tau_{pulse}}{\tau_{ref}}}{s(C_Z+C_P)} \frac{(1+s\tau_z)}{(1+s\tau_p)} \frac{K_{vco}}{s}$

SUBSAMPLING PLL BASICS

Loop stability

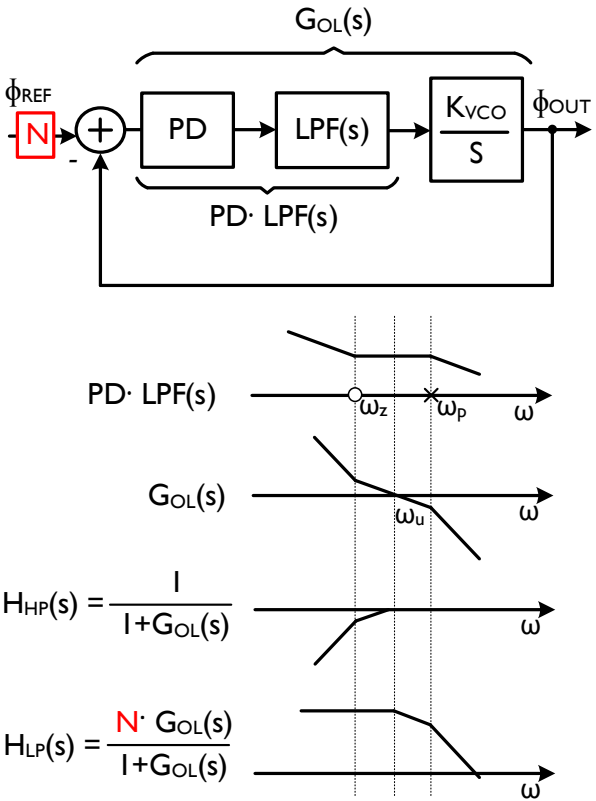


- For example, if
  - $\mathrm{Fref} = 40\ \mathrm{MHz}$
  - $\mathrm{Fvco} = 1\ \mathrm{GHz}$
  - $\omega_u = 1\ \mathrm{MHz}$
  - $G_m = 100\ \mu\mathrm{S}$
  - $K_{VCO} = 100\ \frac{\mathrm{MHz}}{\mathrm{V}}$
  - $A_{VCO} = 1\ \mathrm{V}$
  - $\tau_{pulse} = 3\mathrm{ns}$

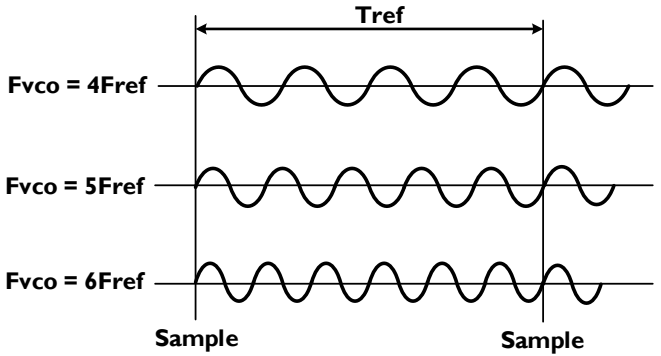
  and
  - $\omega_Z = \frac{\omega_u}{4}$
  - $\omega_p = 4\ \omega_u$
- Then
  - $C_Z = 100\ \mathrm{pF}$ ← ok
  - $C_P = 8\ \mathrm{pF}$
  - $R_Z = 6\ \mathrm{k\Omega}$

SUBSAMPLING PLL BASICS

Virtual multiplication by N

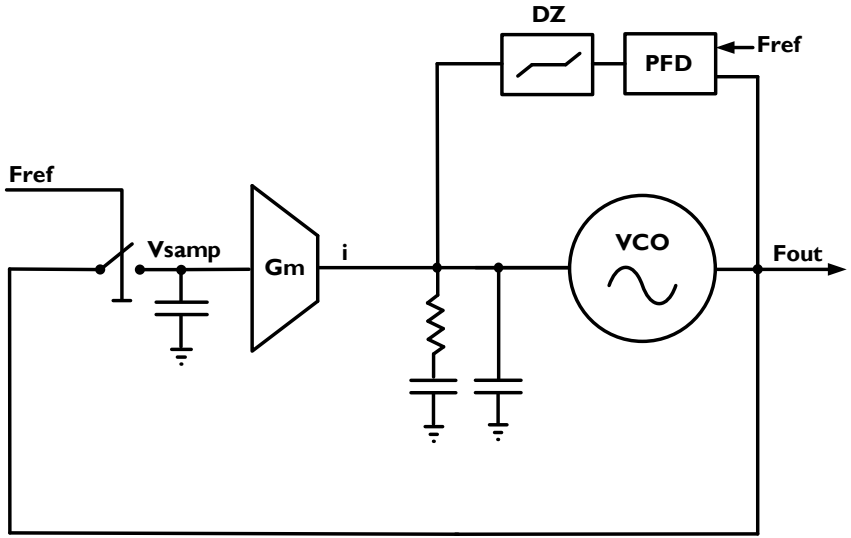


- There is no divider in the loop
  - Phase is still multiplied by N
  - Virtual multiplication
- The loop can lock to any *integer* ratio of input and output frequency
- Sampler "does not know" which Fref harmonic is sampled



SUBSAMPLING PLL BASICS

Frequency acquisition

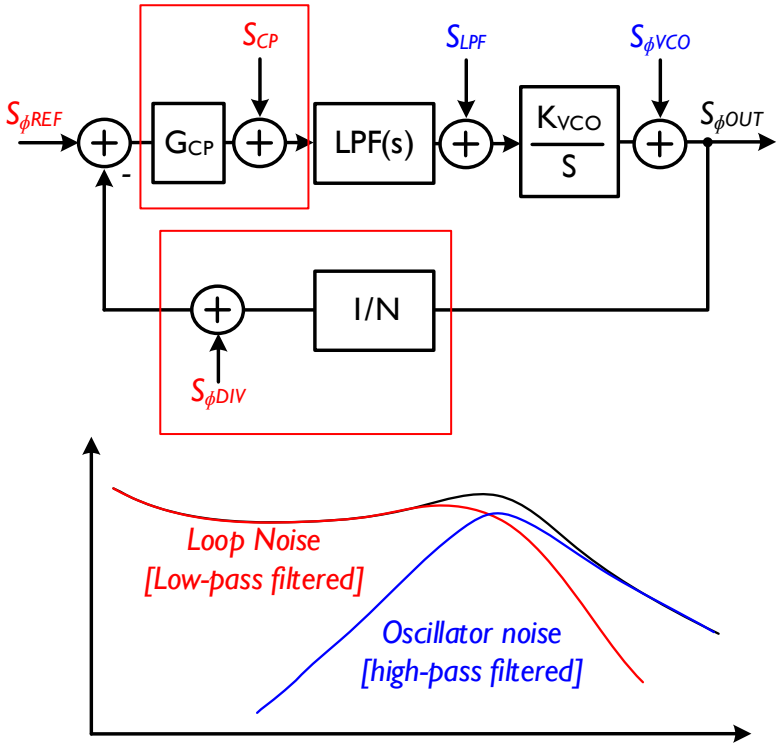


- Auxiliary acquisition loop
  - Classical PFD with a dead-zone
    - Automatically disabled
  - Largest dead zone allowed Tref
    - Subsampling has frequency tracking capabilities between +/- FREF/2
  - Low power implementation preferred
    - No noise contribution in lock

WHERE DOES THE SUPERIOR NOISE vs. POWER TRADEOFF COME FROM?

SUBSAMPLING PLL BASICS

Noise contributors in a classical PLL.

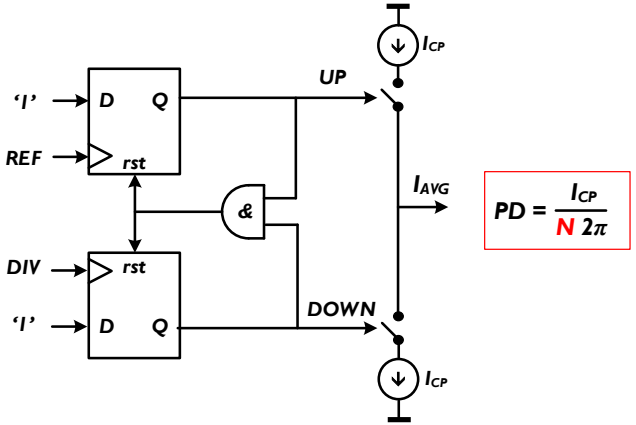


- **<u>In-band</u>** noise typically dominated by:
  1. Charge pump noise
  2. Divider noise
  3. Reference noise

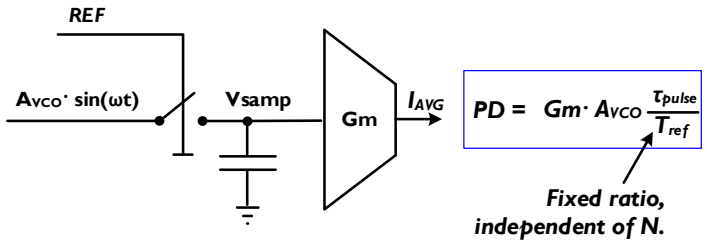
- Power consumption dominated by
  - Dividers (especially at HF)
  - Charge pump
  - VCO

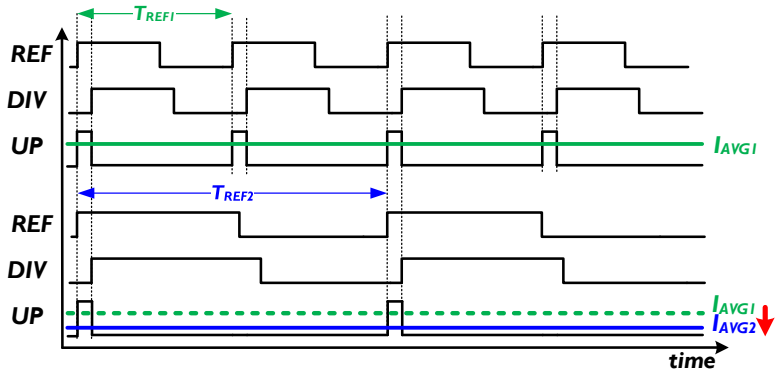
SUBSAMPLING PLL BASICS

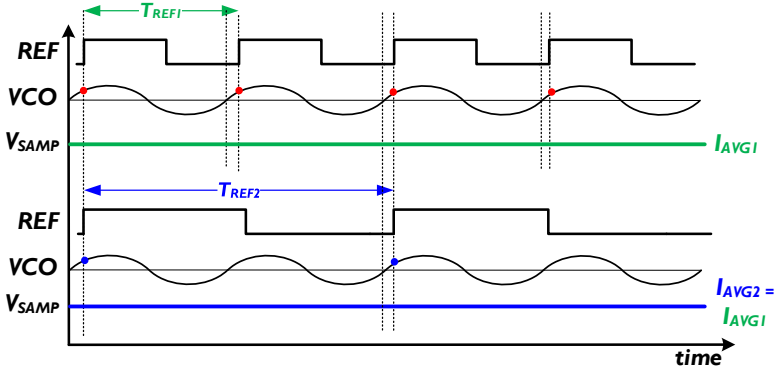
Phase error detection gain comparison: classical versus subsampling



$$PD = \frac{I_{CP}}{N\,2\pi}$$

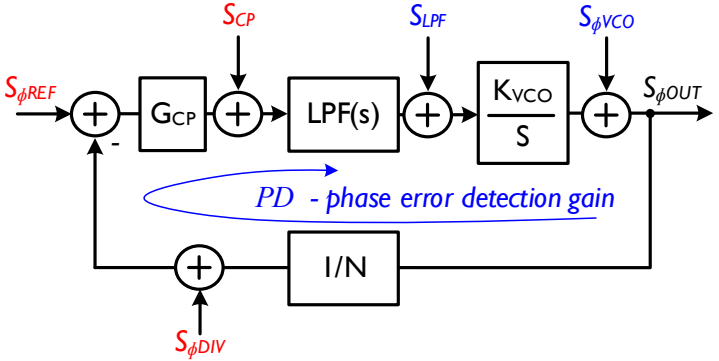


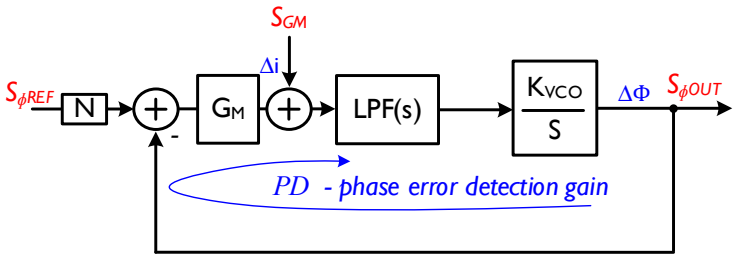




SUBSAMPLING PLL BASICS

Noise contributors in a classical PLL and in a subsampling PLL





- $S_{\Phi_{OUT}}^{IN\ BAND} = \frac{S_{CP}}{PD^2} + N^2\, S_{\Phi_{DIV}} + N^2 S_{\Phi_{REF}}$
- $PD = \frac{1}{N}\frac{I_{CP}}{2\pi}$
- PD gain is low at high frequencies
- Poor noise vs. power trade-off

- $S_{\Phi_{OUT}}^{IN\ BAND} = \frac{S_{CP}}{PD^2} + N^2 S_{\Phi_{DIV}} + N^2 S_{\Phi_{REF}}$
- $PD = G_m\, A_{vco}\, \frac{\tau_{pulse}}{T_{ref}}$
- PD gain is high (independent of N)
- Great noise vs. power trade-off

OUTLINE

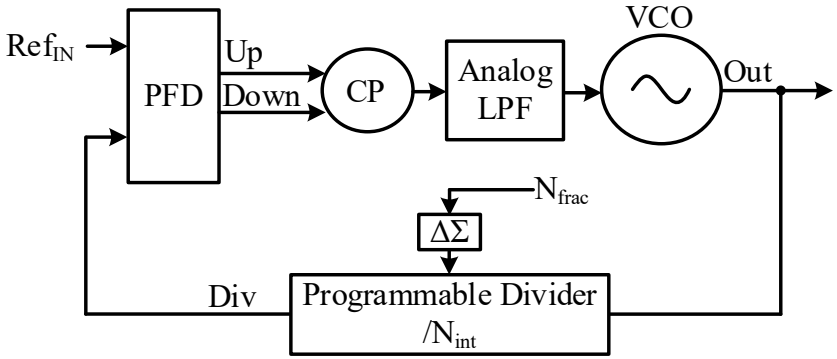
From a Subsampling PLL to a Background-Calibrated Subsampling Phase Modulator

- Subsampling PLL basics
- **Fractional Synthesis in a Subsampling PLL**
- Digital-to-Time Conversion
- Randomization & Background calibration techniques
- Subsampling PLL-based Phase Modulator

imec

FRACTIONAL SYNTHESIS IN A SUBSAMPLING PLL

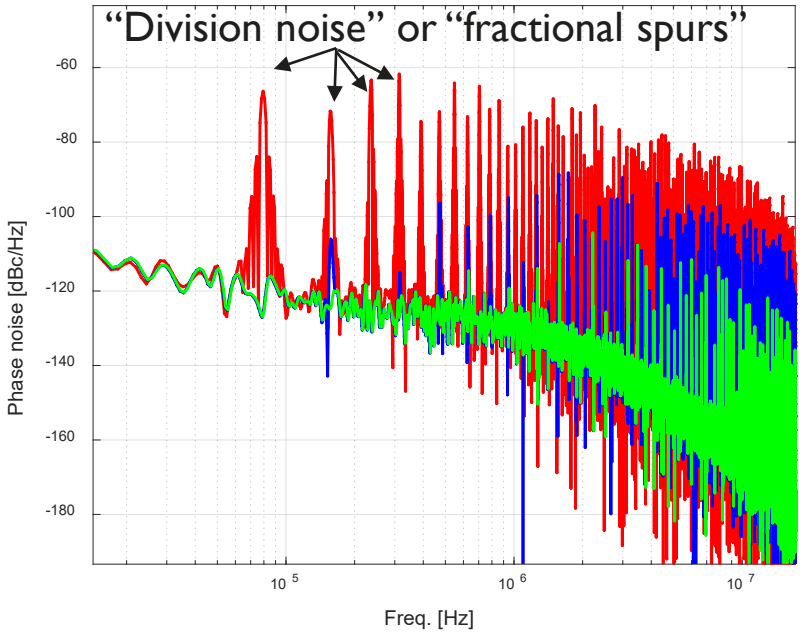
How is it done in a classical PLL?



- Divider "counts" VCO cycles
- Fractional synthesis → Programmable integer divider
  - e.g. Nfrac = 100.25,
    - divider e.g. picks between 100 / 101
- Division correct *on average*
  - instantaneous errors (~Tvco)
  - *Periodic* modulation for a fixed Nfrac → "Fractional spurs" :(
    - ΔΣ modulated division number
    - shapes the periodicity in division

FRACTIONAL SYNTHESIS IN A SUBSAMPLING PLL

How is it done in a classical PLL?



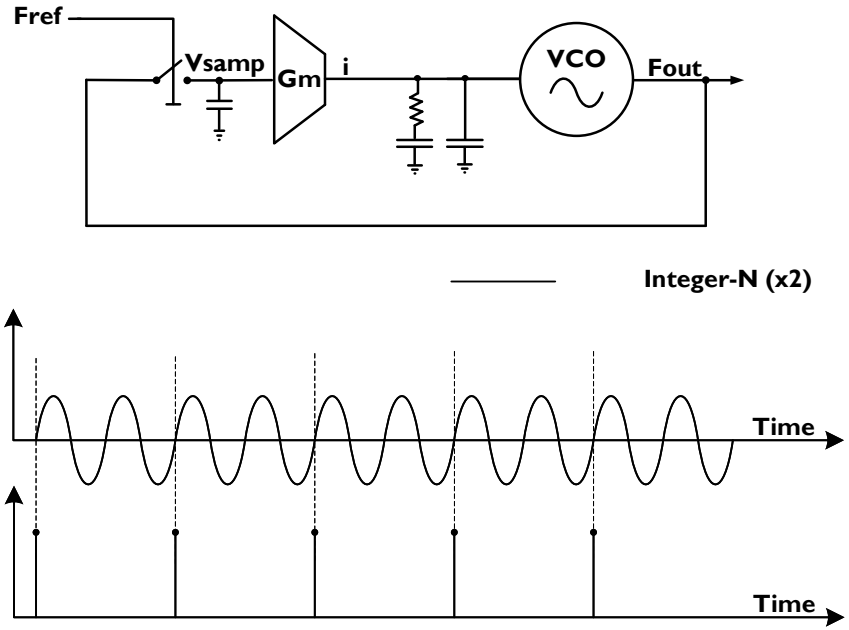
- Fractional spurs are typically a problem
  - Appear at fractional offset and its multiples
  - E.g. N = 100.00125
    - Spur is at the offset of 0.00125 x Fref
- Food for thought:
  "a true fractional divider"
  - 1 TVCO based divider (counter)
  - 0.1 TVCO based divider (counter)
  - 0.01 TVCO based divider (counter)

NO DIVIDER IN A SUBSAMPLING PLL.
NO PHASE MODULATION CAPABILITIES IN A SUBSAMPLING PLL.

HOW TO SYNTHESIZE FRACTIONAL CHANNELS?

FRACTIONAL SYNTHESIS IN A SUBSAMPLING PLL

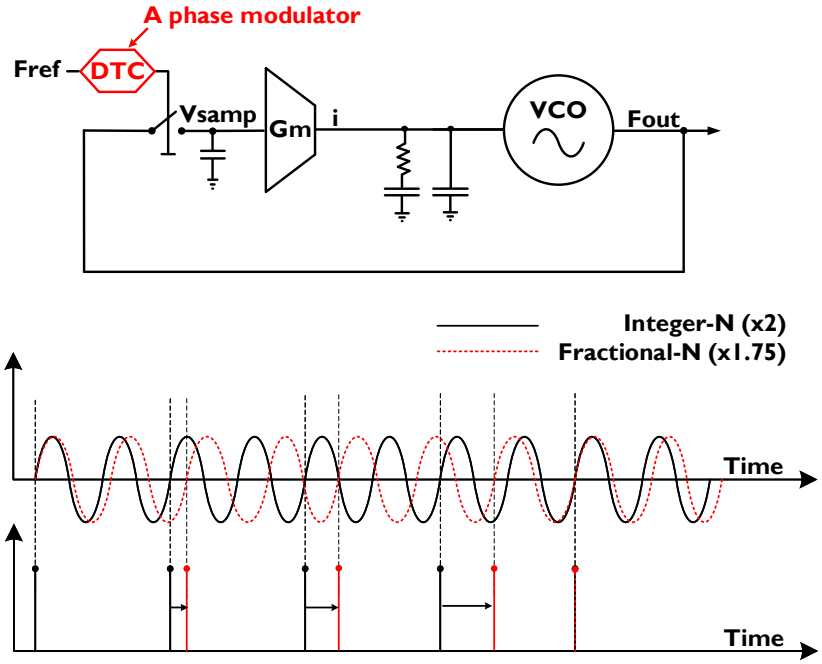
Basic principle



- Integer-N multiplication example
  - N = 2
  - 2 VCO periods in a reference period
- No phase modulation

FRACTIONAL SYNTHESIS IN A SUBSAMPLING PLL

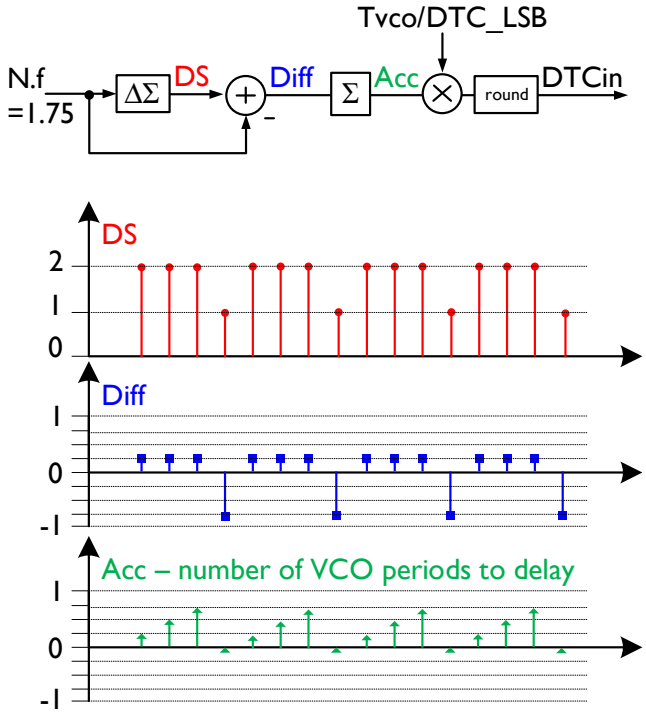
Basic principle



- Fractional-N multiplication example
  - N = 1.75
  - 1.75 VCO periods in a reference period
- A Digital-to-Time Converter (DTC)
  - Phase modulator
  - Needs to cover at least one VCO period

FRACTIONAL SYNTHESIS IN A SUBSAMPLING PLL

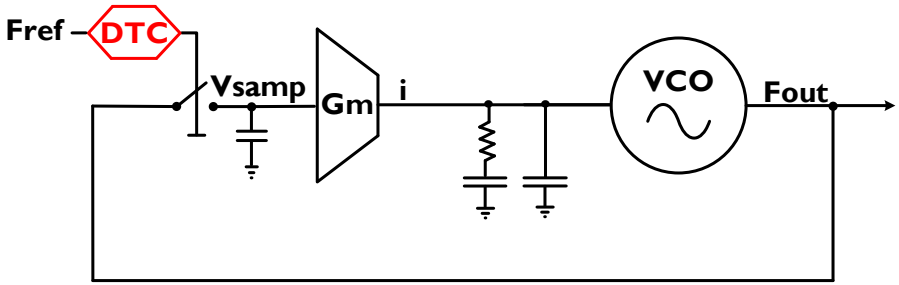
Digital computation path

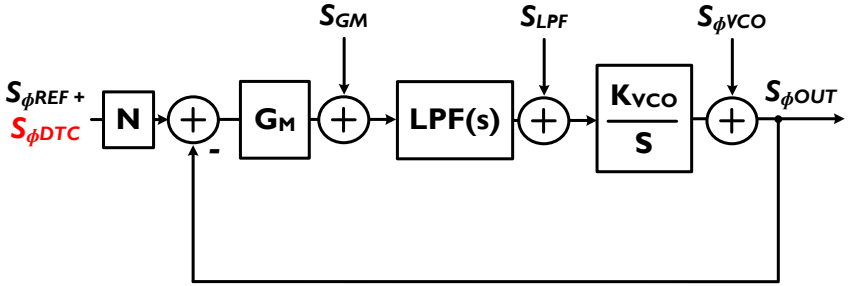


- ***N.f*** = 1.75
- ΔΣ modulator
  - 1st order (simplest)
  - toggles between 1 and 2
  - 1.75 on average
- ***Diff*** – cycle fractional residue
- ***Acc*** – accumulated fractional residue
  - Number of VCO periods to delay
  - Needs to be scaled to the available DTC input range

FRACTIONAL SYNTHESIS IN A SUBSAMPLING PLL

DTC induced errors





- With an ideal DTC
  - Integer = Fractional performance
  - In stark contract to classical loops
  - DTC → a true fractional divider?
- Real DTC is a data converter
  - Limited resolution
  - Random noise
  - Gain imbalance
  - Nonlinearities
- At the input of the system
  - Noise multiplied by N^2 in transfer to the output
  - Low pass filtered

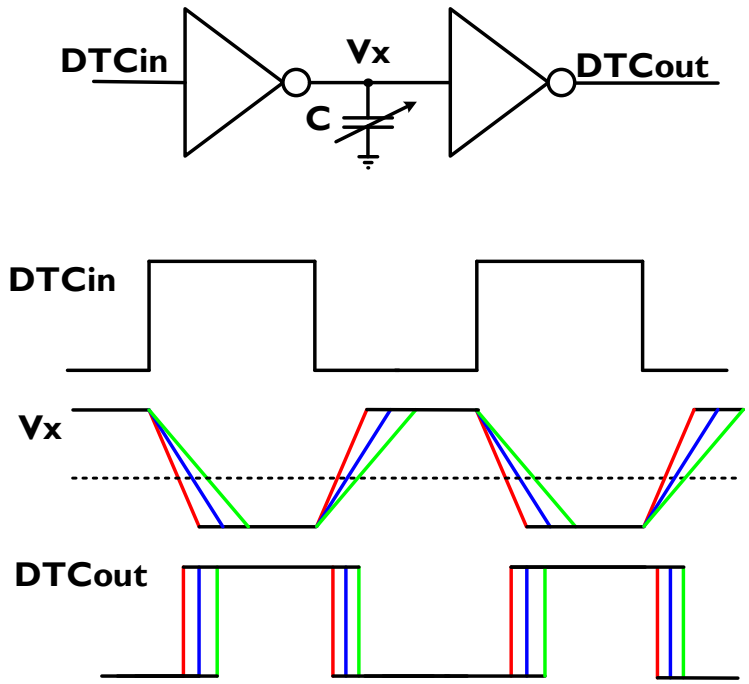
OUTLINE

From a Subsampling PLL to a Background-Calibrated Subsampling Phase Modulator

- Subsampling PLL basics
- Fractional Synthesis in a Subsampling PLL
- **Digital-to-Time Converter**
- Randomization & Background calibration techniques
- Subsampling PLL-based Phase Modulator

DIGITAL-TO-TIME CONVERTER

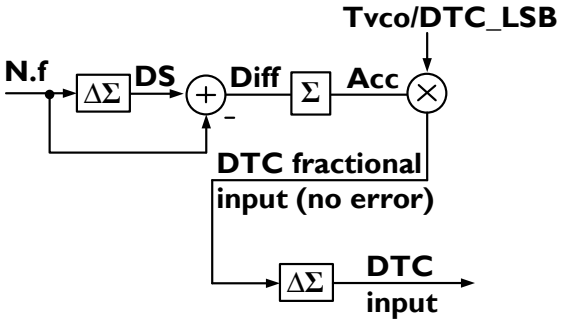
Basic idea: variable slope DTC

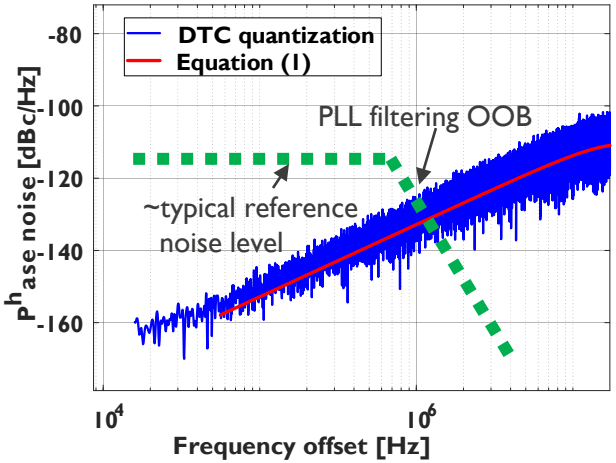


- Inverter with tunable load
- Delay step easily in range of ~100 fs
  - $\Delta\tau = I/\Delta C$
    - I – inverter driving strength
    - ΔC – capacitance change
- Quantization noise < Random noise floor

DIGITAL-TO-TIME CONVERTER

DTC quantization noise





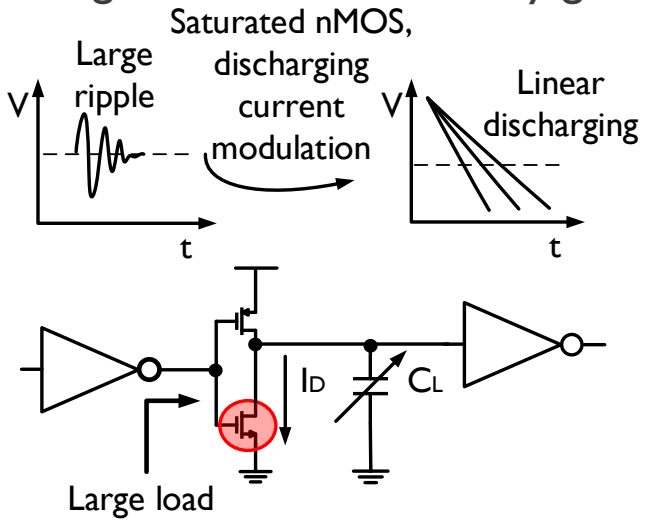
- DTC quantization noise can be low enough
  - ~100 fs LSB with 11 bits to cover a single VCO period at 5GHz
- Can be shaped by a ΔΣ modulator
  - Low pass filtered by the PLL
  - With ΔΣ shaping, in-band-noise guarded by:

$$\mathcal{L} = 10\log_{10}\left\{\left(\frac{2\pi}{\sqrt{12}}\cdot\frac{LSB}{T_{VCO}}\right)^2\frac{1}{F_{REF}}\left[2\sin(\pi f T_{REF})\right]^2\right\}\left[\frac{\text{dBc}}{\text{Hz}}\right]$$

  - With 0.5ps LSB (left) no impact in-band
    - Carrier at 10 GHz

DIGITAL-TO-TIME CONVERTER

DTC design considerations: delay generation (1/4)



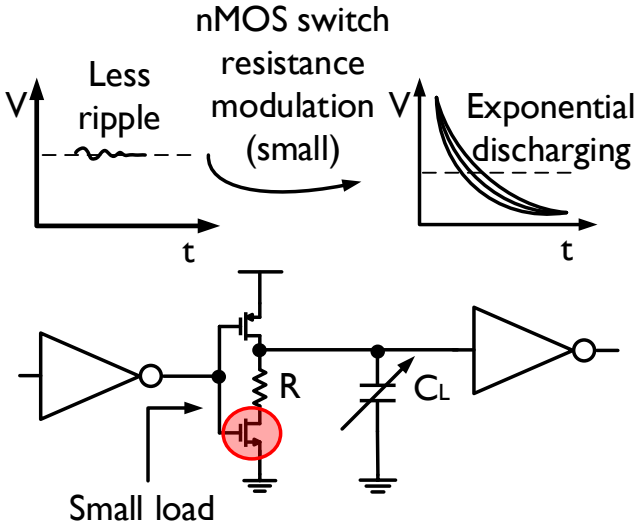
- nMOS as a current source
- Oversizing the nMOS for 1/f noise
- Heavy loading of the input buffer
- Input buffer induces supply ripple

$$\mathcal{L}_{fkr} \sim 10\log\left(f_{out}^2\left(\frac{C_L}{I_D}\right)^2\frac{2K}{WLf}\right) \sim 10\log\left(f_{out}^2\,\tau_{delay}^2\frac{2K}{WLf}\right)$$

$$\mathcal{L}_{white} \sim 10\log\left(f_{out}\frac{kTC_L}{I_D^2}\right) \sim 10\log\left(f_{out}\frac{kT\tau_{delay}}{I_D}\right)$$

DIGITAL-TO-TIME CONVERTER

DTC design considerations: Delay generation (2/4)

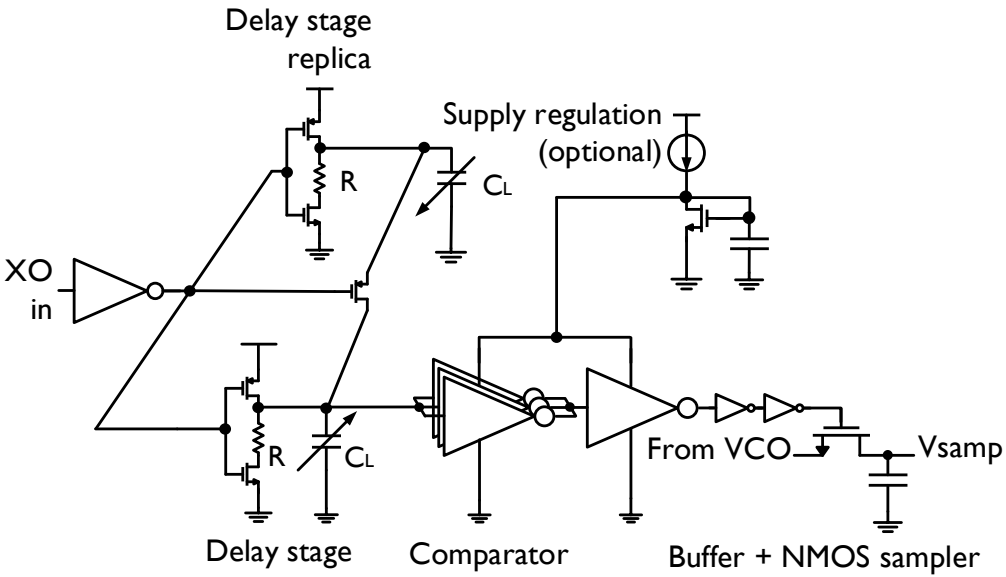


- nMOS as a switch
- No flicker noise
- Choose R for noise at largest τdelay
- Supply ripple problems reduced

$$\mathcal{L}_{white} \sim 10\log\left(f_{out}\frac{kT\,C_L\,R^2}{V_{DD}^2}\right) \sim 10\log\left(f_{out}\frac{kT\,\tau_{delay}\,R}{V_{DD}^2}\right)$$

DIGITAL-TO-TIME CONVERTER

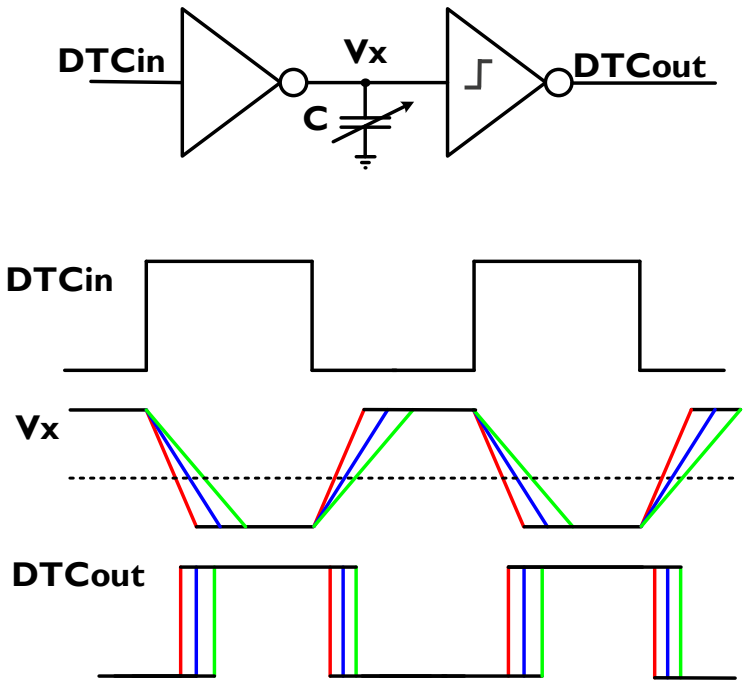
DTC design considerations (3/4)



- Avoiding code dependent power consumption
- Large comparator and buffer to restore steep slopes
  - Optional supply regulation to suppress time-dependent supply bounce

DIGITAL-TO-TIME CONVERTER

DTC design considerations (4/4)



- The comparator has limited bandwidth
  - Reacts to a variable slope
  - Does not ‘flip’ instantly at threshold
  - Slope dependent reaction
  - Induces nonlinearity
    - e.g. < 0.2% INL at 500 ps range
    - Solutions:
      - Constant slope DTC?
        - Not covered here
      - *Calibration*

DIGITAL TO TIME CONVERTER

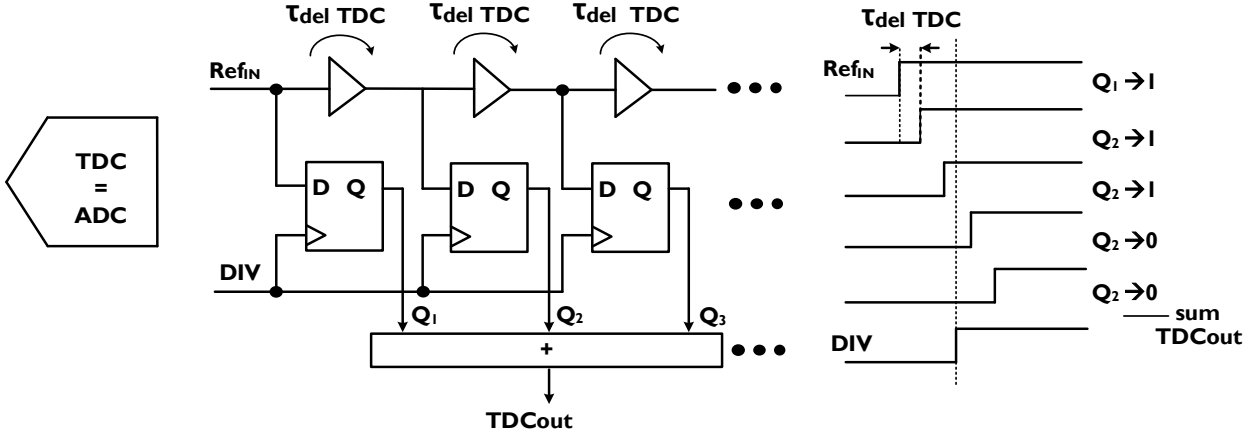
DTC-based subsampling PLL

- DTC
  - Can operate with quantization noise low enough
    - True fractional divider
  - Low random (analog) noise

- Keeps all the benefits of a subsampling PLL?
  - Integer-N = Fractional-N performance?
- If … gain error / nonlinearities are low enough
  - Can be compensated in digital domain

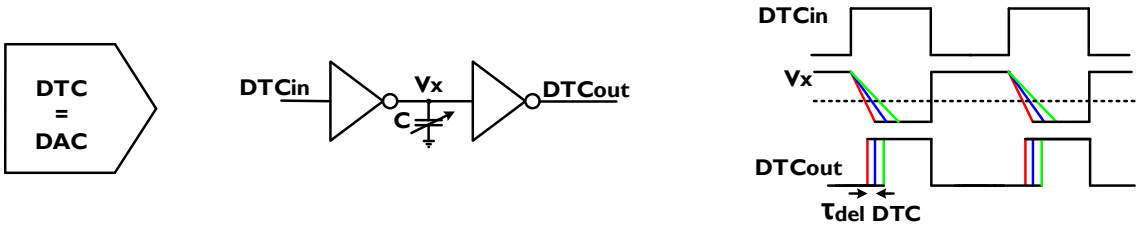
DIGITAL TO TIME CONVERTER

TDC vs a DTC



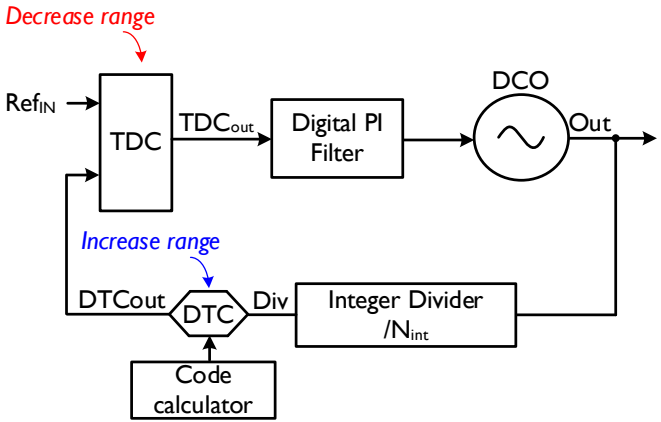
- 28nm CMOS:
- τdel TDC ~ 10 **pS**
- τdel DTC ~ 100 **fS**

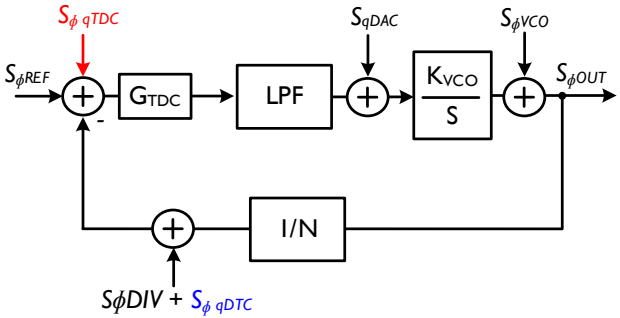
- τDTC << τTDC



DIGITAL TO TIME CONVERTER

DTC-based environments: digital PLL

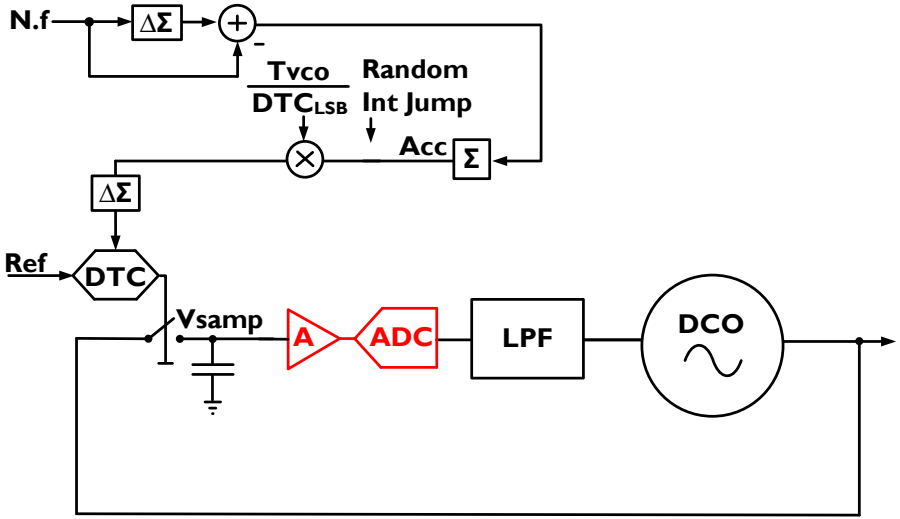




- DTC favoring trade-off
  - Popularity of Bang-Bang PLL as proof

DTC-BASED DIGITAL SUBAMPLING PLL

DTC-based environment: digital subsampling PLL



- Analog-to-digital conversion *after subsampling?*
  - Added ADC quantization noise can be comparably low
  - Single bit ADC (Bang-Bang) is an option
- Added DCO quantization noise

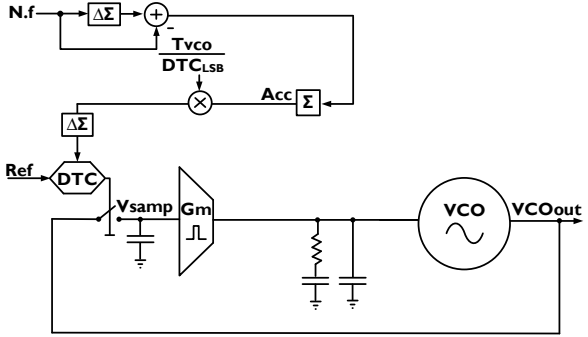
OUTLINE

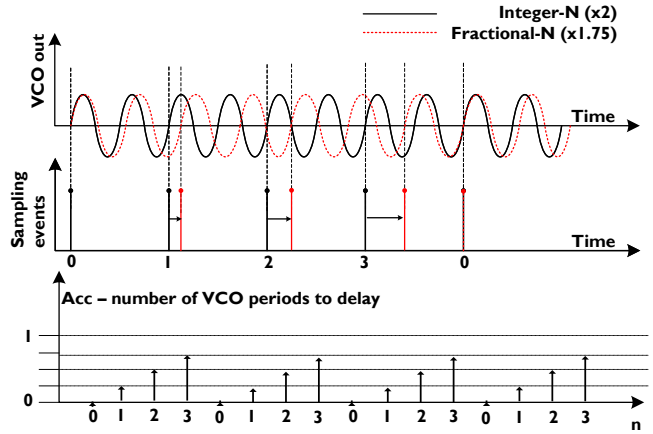
From a Subsampling PLL to a Background-Calibrated Subsampling Phase Modulator

- Subsampling PLL basics
- Fractional Synthesis in a Subsampling PLL
- Digital-to-Time Converter
- **Randomization & Background calibration techniques**
- Subsampling PLL-based Phase Modulator

RANDOMIZATION AND & BACKGROUND CALIBRATION

DTC non-idealities: gain error and nonlinearity

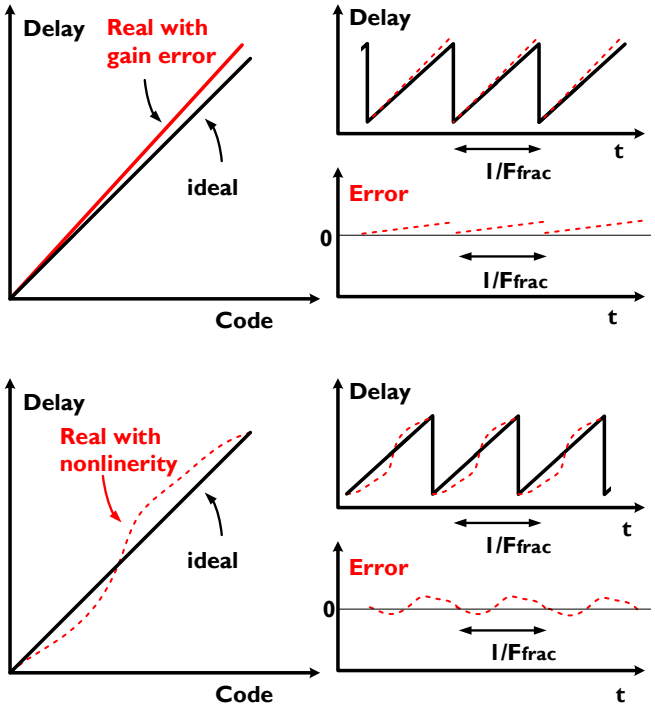


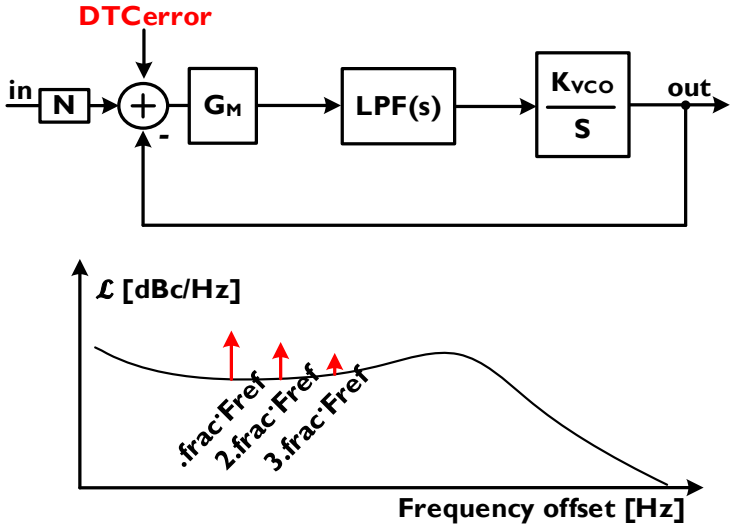


- Recall the basic operation principle
  - DTC cancels the fractional residue
  - DTC input is a *periodic* signal
  - DTC limitations
    - Gain error?
    - INL?

RANDOMIZATION AND & BACKGROUND CALIBRATION

DTC non-idealities: gain error and nonlinearity





- Gain error / nonlinearity
  - Periodically modulate PLL output phase
  - No filtering in-band

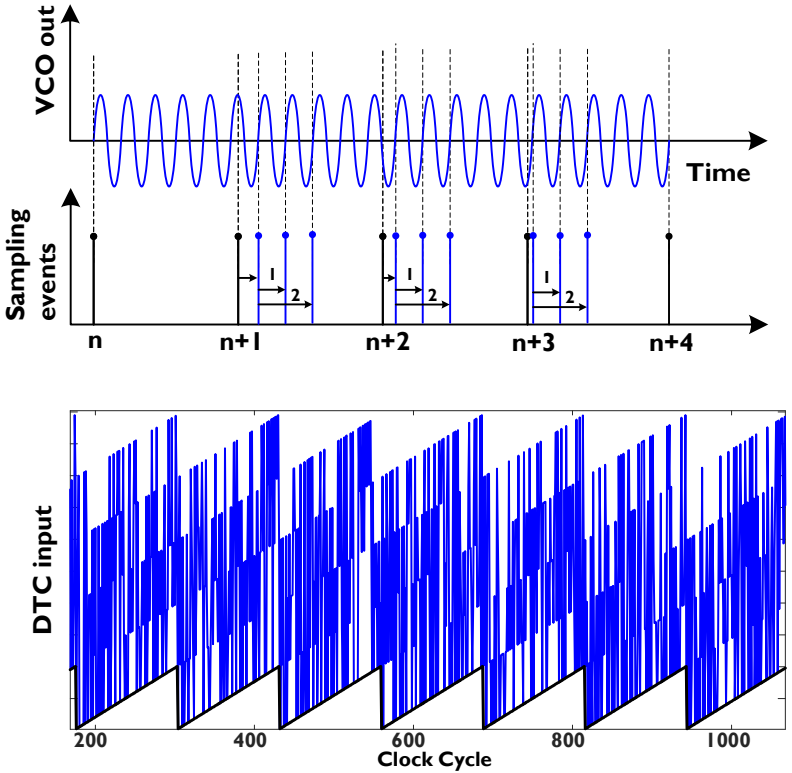
RANDOMIZATION AND & BACKGROUND CALIBRATION

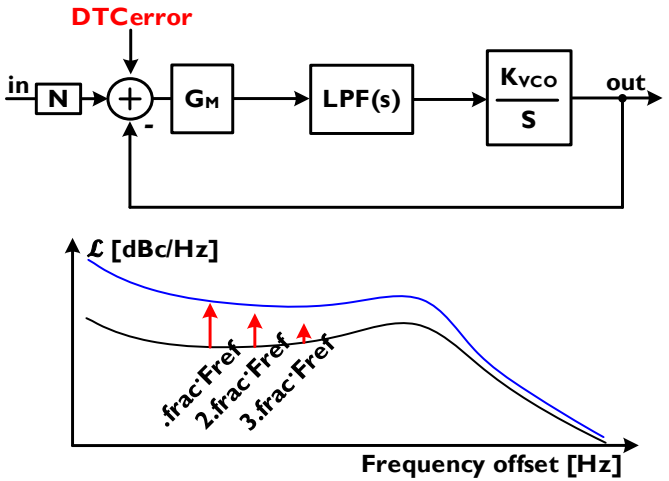
Similar issues in different PLL architectures

- Gain imbalance/Nonlinearity in the phase error detection exists in all PLLs
  - Charge pump up/down mismatch in analog PLLs
  - TDC gain/linearity issues in TDC-based PLLs
  - DTC gain/linearity issues in DTC-based PLLs

RANDOMIZATION AND & BACKGROUND CALIBRATION

Randomization principle

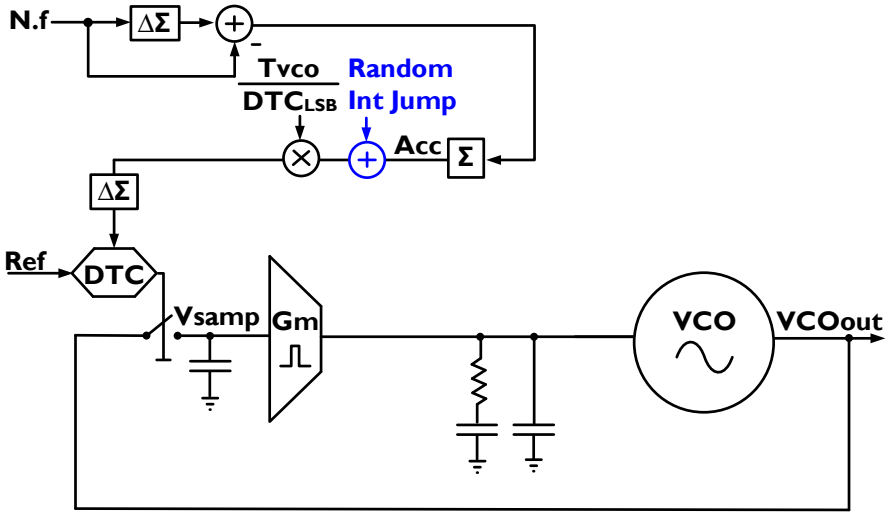




- Masks periodicity of the DTC input
  - Transforms spurs into noise
  - No error cancellation

RANDOMIZATION AND & BACKGROUND CALIBRATION

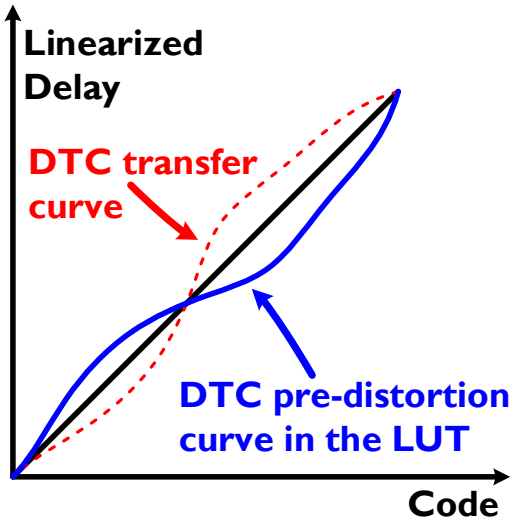
Randomization principle



- Simple modification in the digital domain
  - Pseudorandom number generation
- Extends DTC range requirement
  - Not critical at HF (TVCO is small)
  - DTC INL increases with larger range

RANDOMIZATION AND & BACKGROUND CALIBRATION

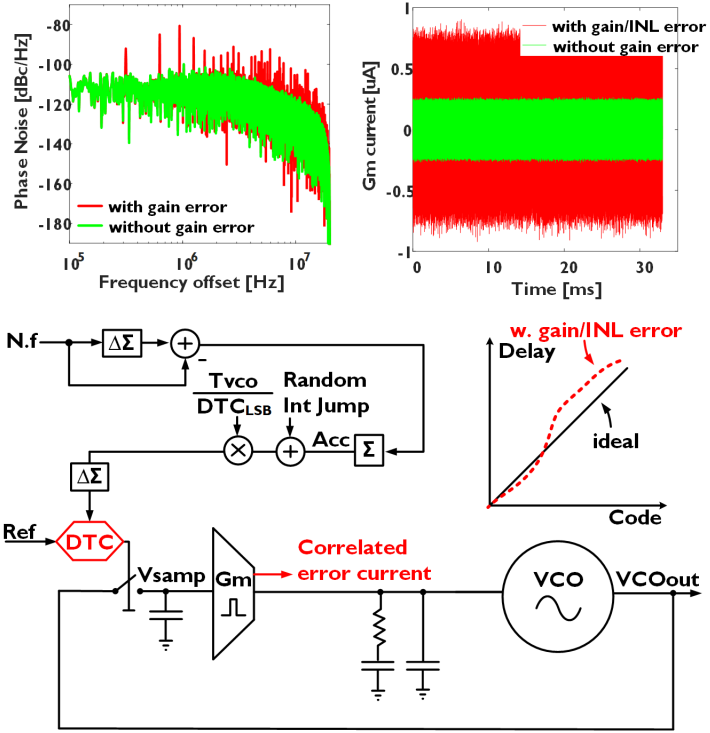
Predistortion for gain-error/nonlinearity cancellation



- Predistortion is a well-known technique
  - Inverse of the erroneous transfer function in a LUT
  - Simple scaling for gain error
- Measuring Gain-error/INL is not trivial
  - ~100 fs delay step
  - Susceptible to PVT
- Background calibration necessary

RANDOMIZATION AND & BACKGROUND CALIBRATION

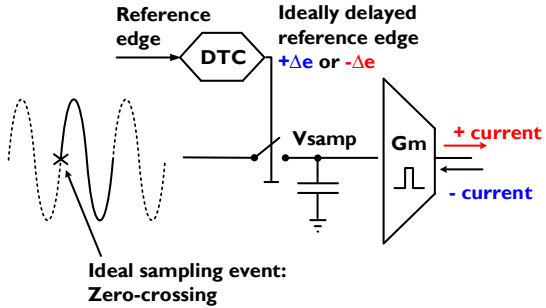
Core observation: "colored" errors = correlation

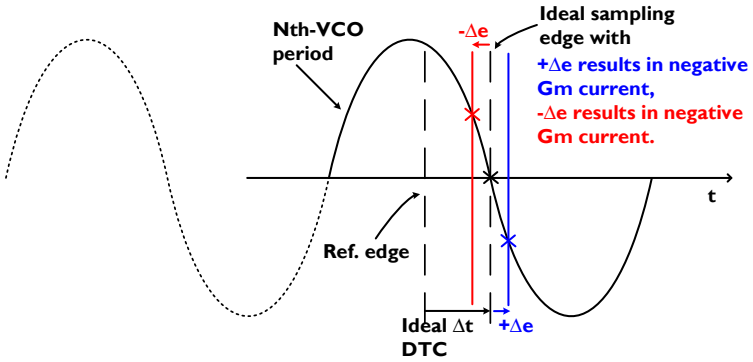


- Gm output current
  - is *random* without gain/INL errors in the DTC transfer function
  - is *colored* with gain/INL errors in the DTC transfer function
  - is *correlated* to the DTC input code in presence of gain/INL errors

RANDOMIZATION AND & BACKGROUND CALIBRATION

Core observation: "colored" errors

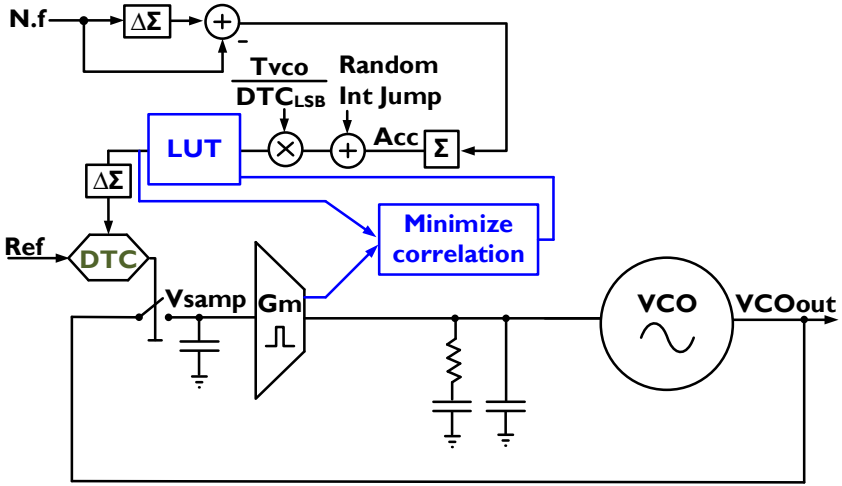




- DTC-induced error?
  - *Code*-dependent instantaneous errors
  - *Sign* of the current related to error direction

RANDOMIZATION AND & BACKGROUND CALIBRATION

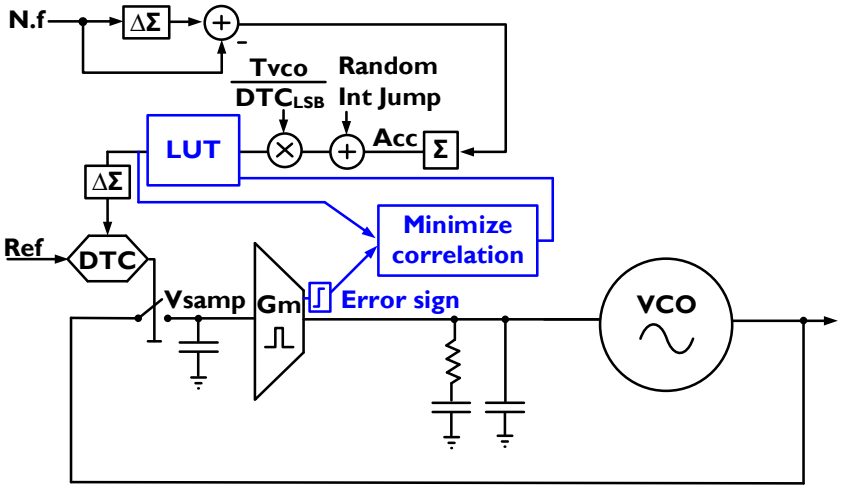
Gain-error/nonlinearity calibration principle in a subsampling PLL



- LUT to create correction "per code"
  - Single coefficient enough for gain calibration
- How to populate the look up table?

RANDOMIZATION AND & BACKGROUND CALIBRATION

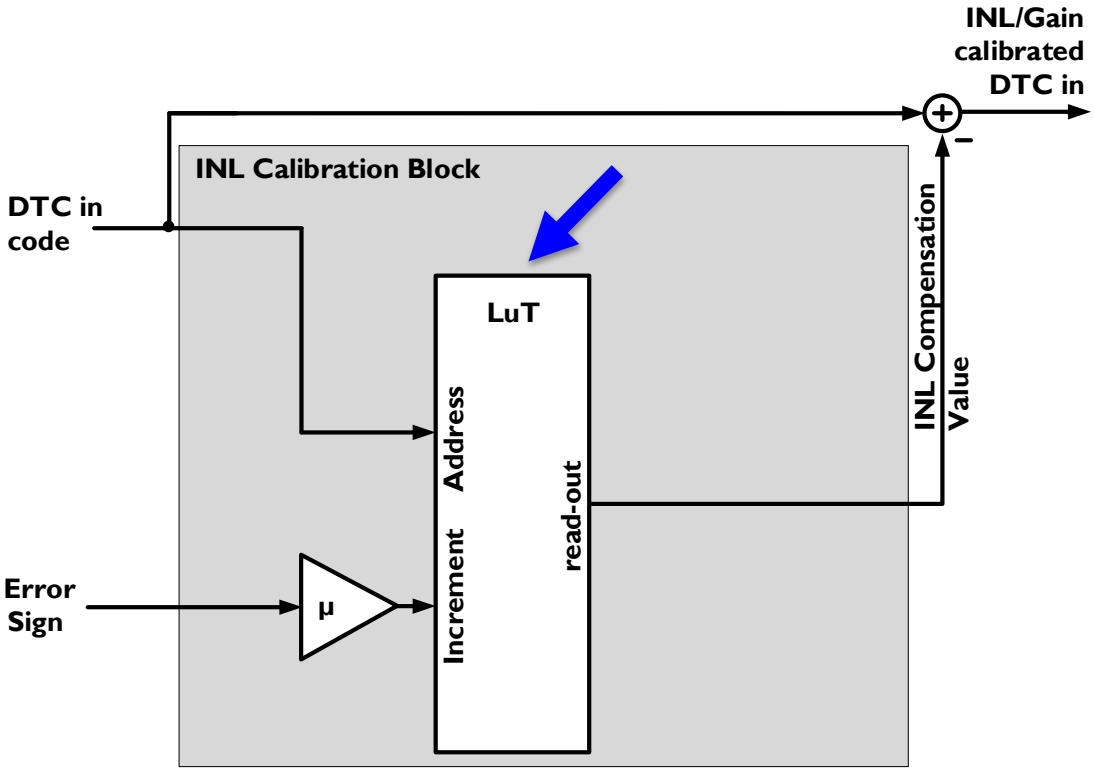
Gain-error/nonlinearity calibration principle in a subsampling PLL



- Analog error but digital information
- ADC needed
  - 1b ADC or Bang-Bang error detection
  - Error sign

RANDOMIZATION AND & BACKGROUND CALIBRATION

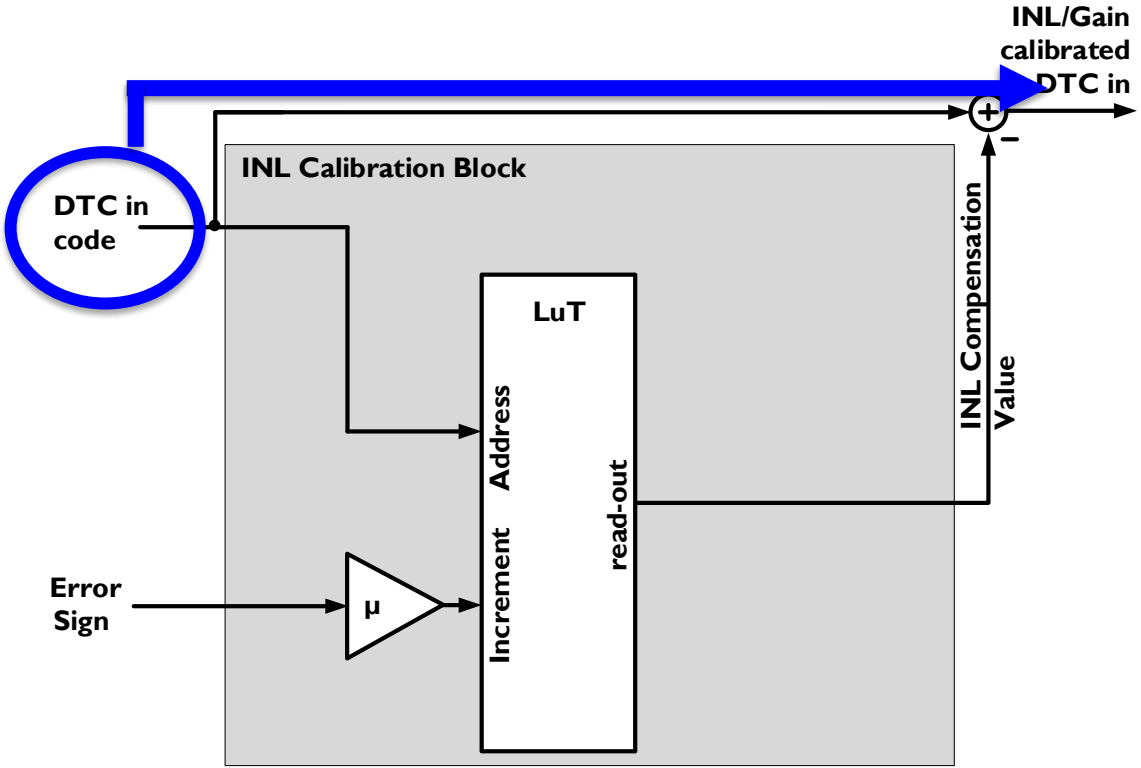
Gain-error/nonlinearity calibration principle in a subsampling PLL



- LUT stores information on DTC Gain/INL

RANDOMIZATION AND & BACKGROUND CALIBRATION

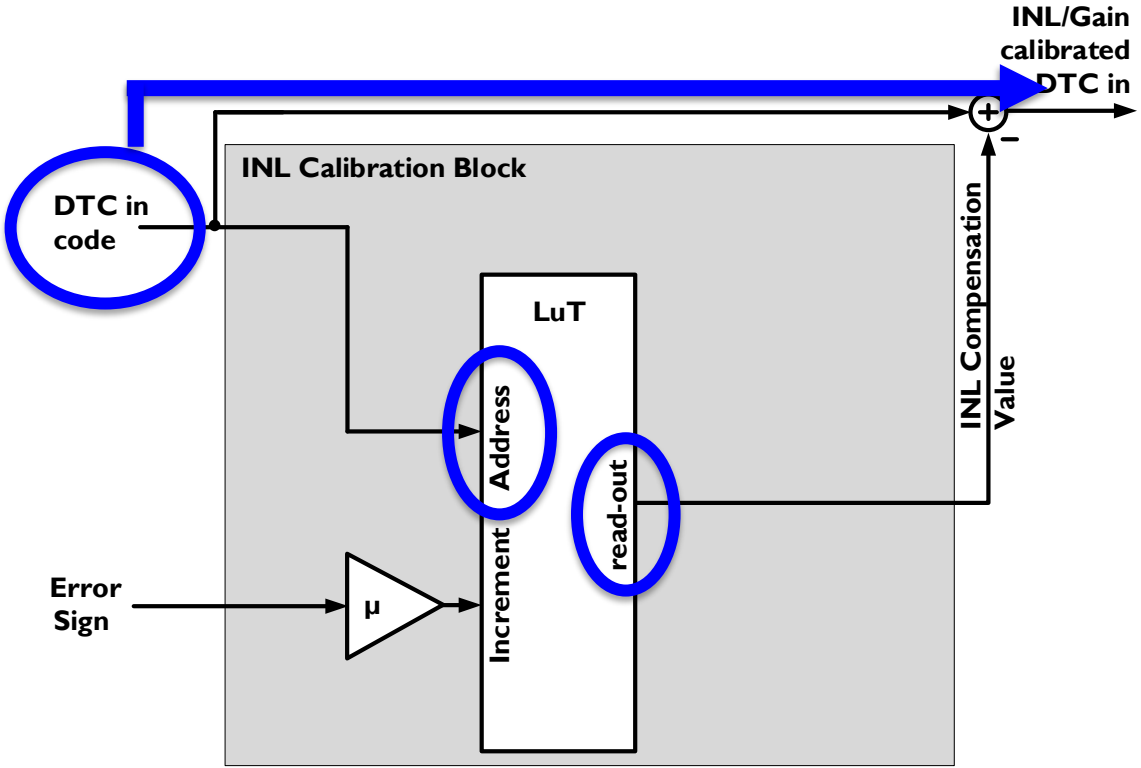
Gain-error/nonlinearity calibration principle in a subsampling PLL



- Code sent to the DTC
  - Nothing happens to it if LUT is clear
  - Impacted by data from LUT in presence of nonlinearity

RANDOMIZATION AND & BACKGROUND CALIBRATION

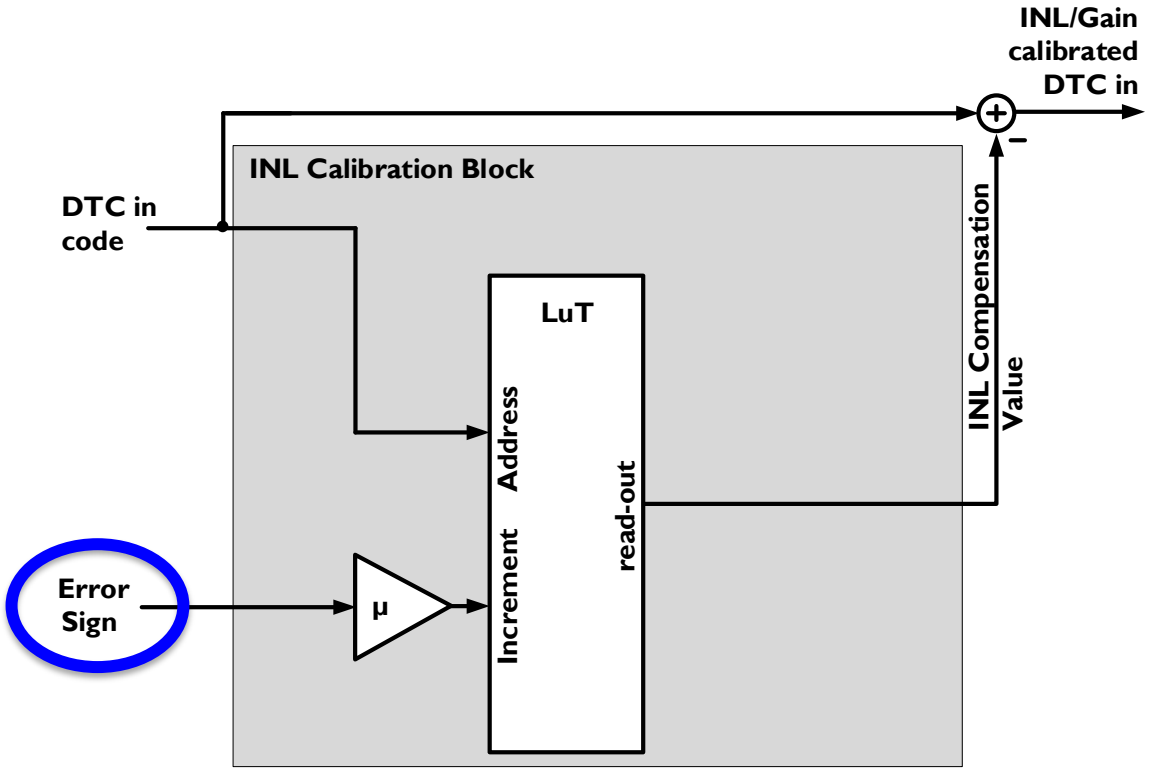
Gain-error/nonlinearity calibration principle in a subsampling PLL



- INL subtracted from original code

RANDOMIZATION AND & BACKGROUND CALIBRATION

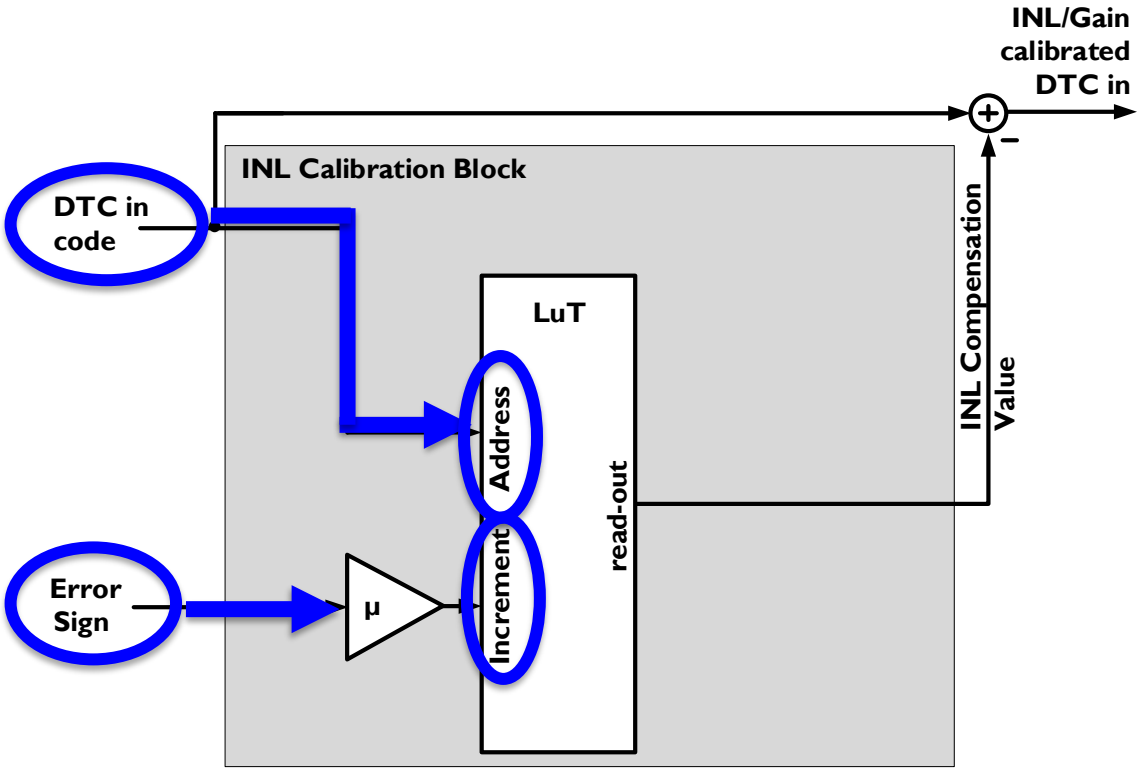
Gain-error/nonlinearity calibration principle in a subsampling PLL



- Error sign: DTC sampling "soon", "late" or correctly?

RANDOMIZATION AND & BACKGROUND CALIBRATION

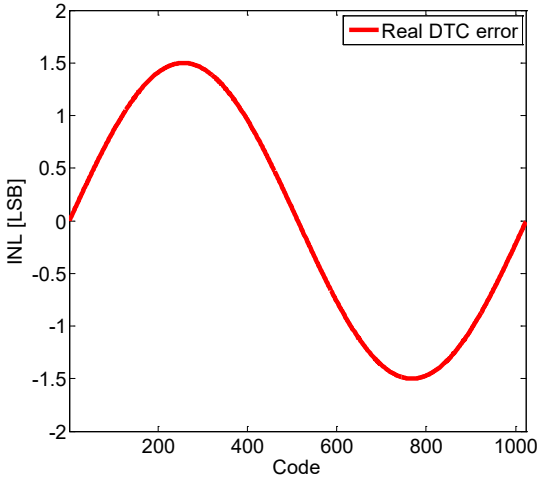
Gain-error/nonlinearity calibration principle in a subsampling PLL

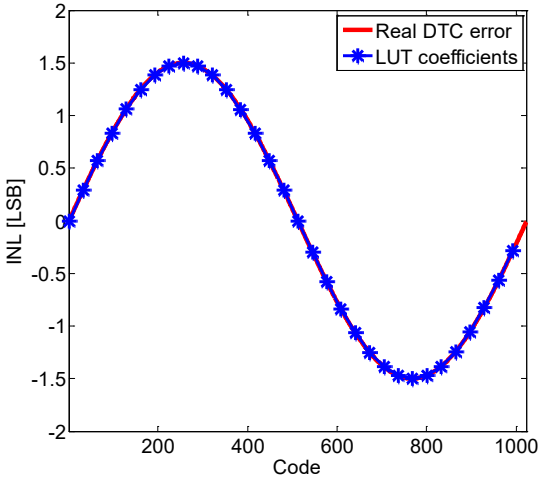


- Gradual adjustment of the correction coefficients
- Error sign on average 0 if correction is accurate

RANDOMIZATION AND & BACKGROUND CALIBRATION

Gain-error/nonlinearity calibration principle in a subsampling PLL

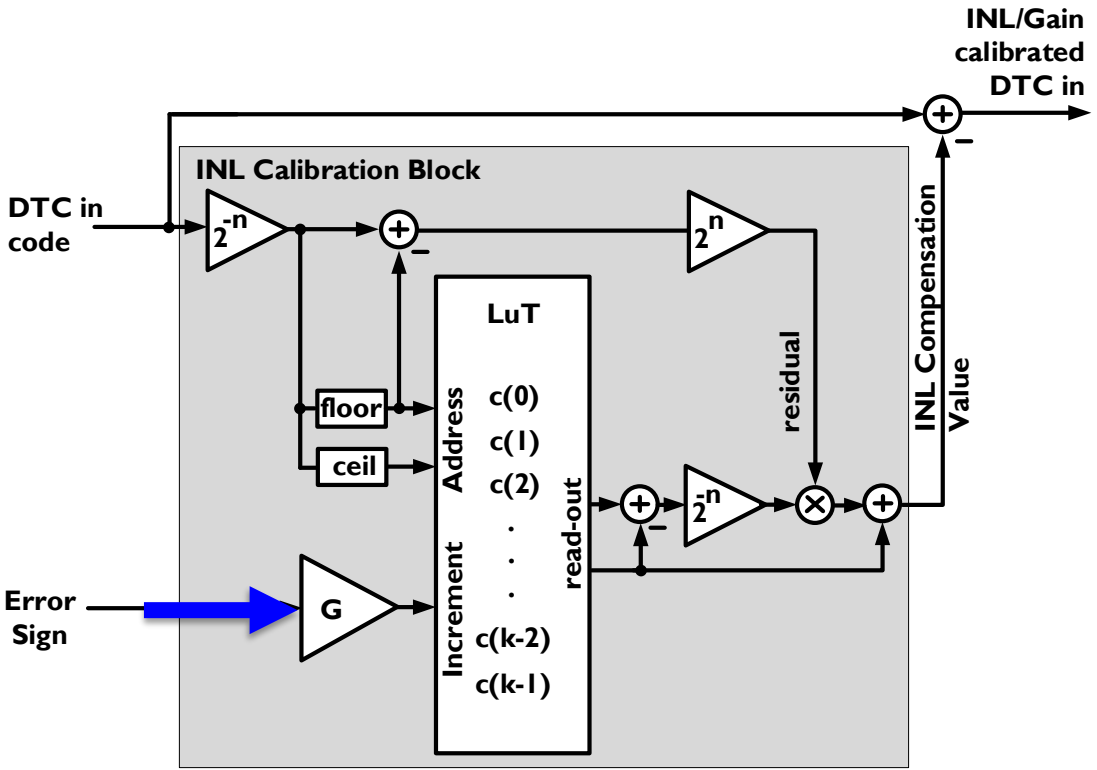




- Piecewise linear approximation of the error
  - No need to have 2^N correction coefficients

RANDOMIZATION AND & BACKGROUND CALIBRATION

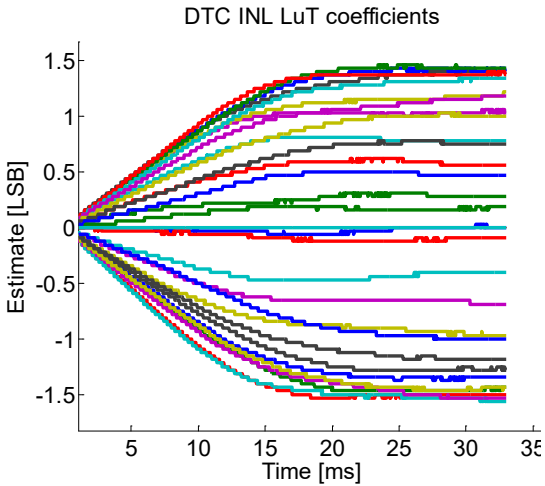
Nonlinearity calibration principle in a subsampling PLL

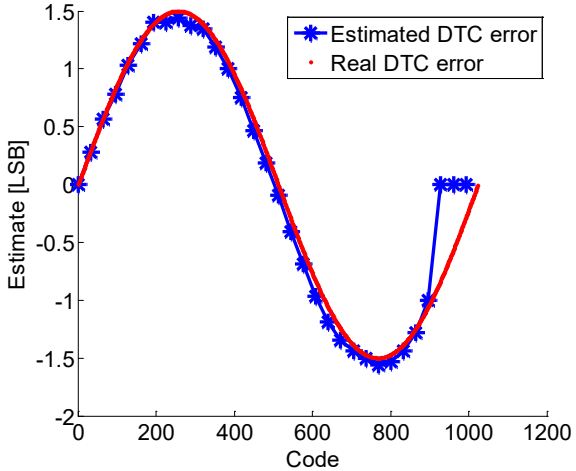


- Physical implementation example
- n = 5 to approximate 10b DTC input with 5b LUT

RANDOMIZATION AND & BACKGROUND CALIBRATION

Nonlinearity calibration principle in a subsampling PLL

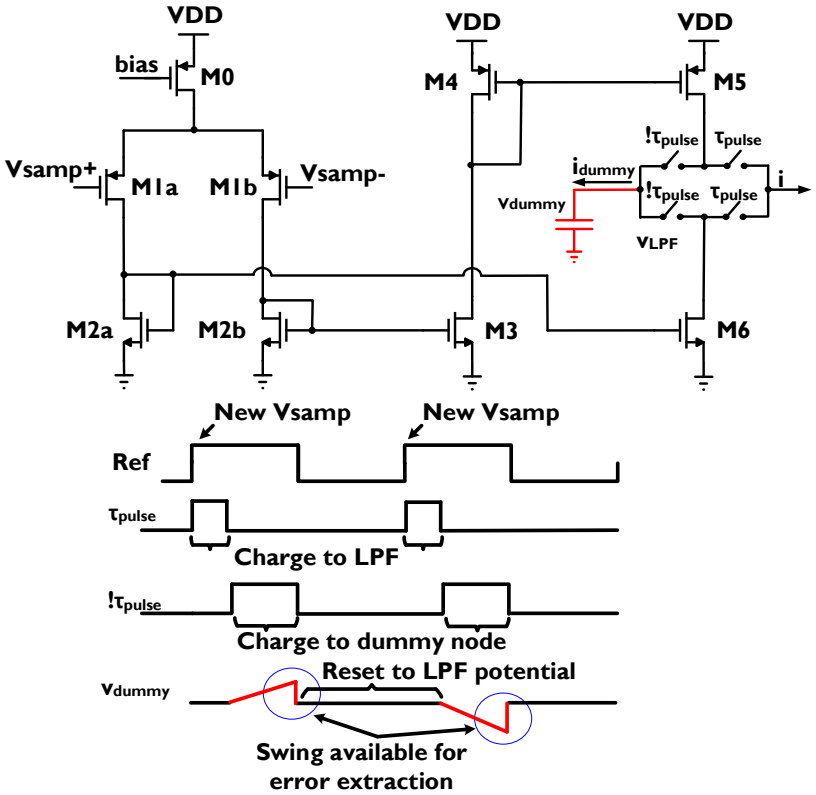




- Coefficient update over time
  - Initialize LUT to a nonzero value
  - Calibration time not that relevant
- Cannot compensate for the codes that are not excited

RANDOMIZATION AND & BACKGROUND CALIBRATION

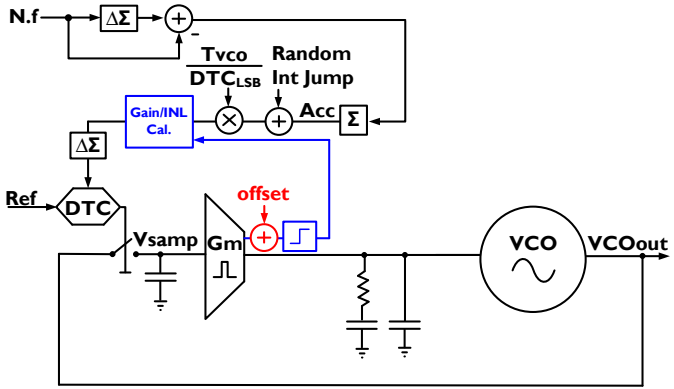
How to extract the current sign?

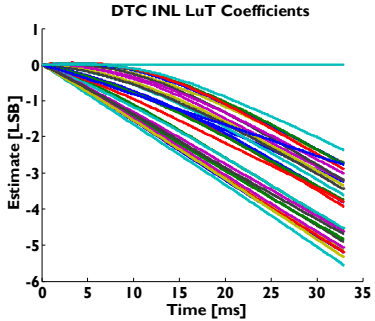


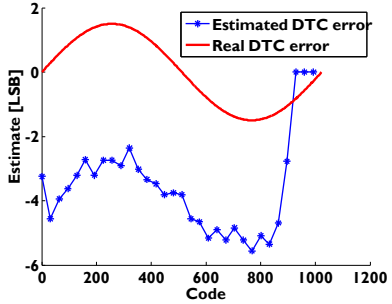
- Gm current is pulsed
  - Loop updated only a portion of the time
  - Steer current towards a dummy node for the rest of the time
  - Dummy node reset to LPF potential
    - Updated after every sampling event

RANDOMIZATION AND & BACKGROUND CALIBRATION

How to extract the current sign?



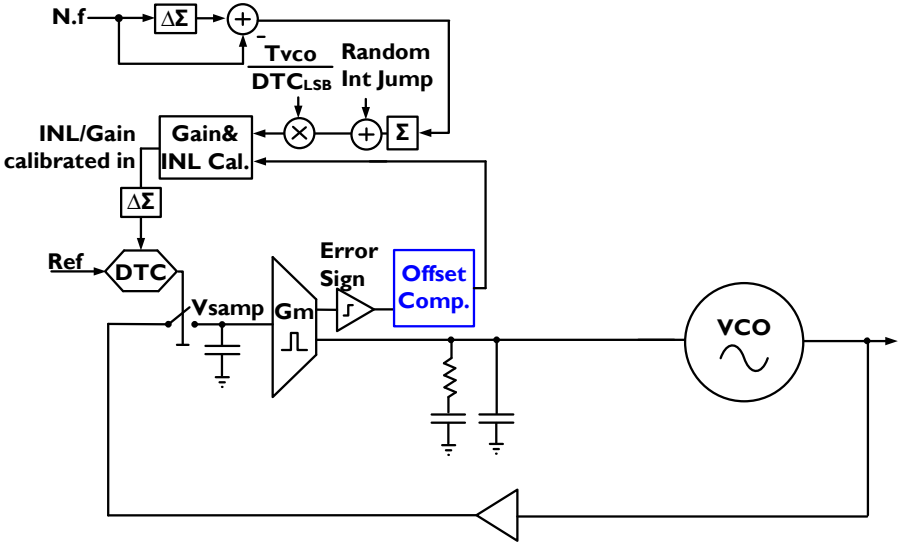




- Analog loop vs. digital calibration
- 1b comparator error *extraction offset*
  - Can cause drift of coefficients
    - Type-II PLL settles to a zero-phase error condition
    - Calibration algorithms detect fake errors
- Foreground calibration is not enough
  - Even a very small residue accumulates over time

RANDOMIZATION AND & BACKGROUND CALIBRATION

How to extract the current sign?



- Digital offset compensation
  - Idea: periodically over-rule the extracted sign to ensure zero-mean stream
    - Convergence for offsets < ~σ
    - Unharmful noise in the Gain/INL estimation
- Analog (mixed-mode) compensation possible, too

RANDOMIZATION AND & BACKGROUND CALIBRATION

Summary

- Fractional-N performance = Integer-N performance
  - If DTC LSB low enough → analog design
  - If DTC is low noise → analog design
  - If DTC is resilient to supply noise → analog design
  - If there is no Gain error → digital background calibration
  - If there is no INL error → digital background calibration
- Digitally enhanced analog
  - Exploit modern CMOS for power efficient linearization of the high-performance analog core
  - Presented techniques applicable in
    - Digital PLLs, Bang-Bang PLLs, Type-I subsampling PLL, Sampling PLLs, digital subsampling PLLs etc.

OUTLINE

From a Subsampling PLL to a Background-Calibrated Subsampling Phase Modulator

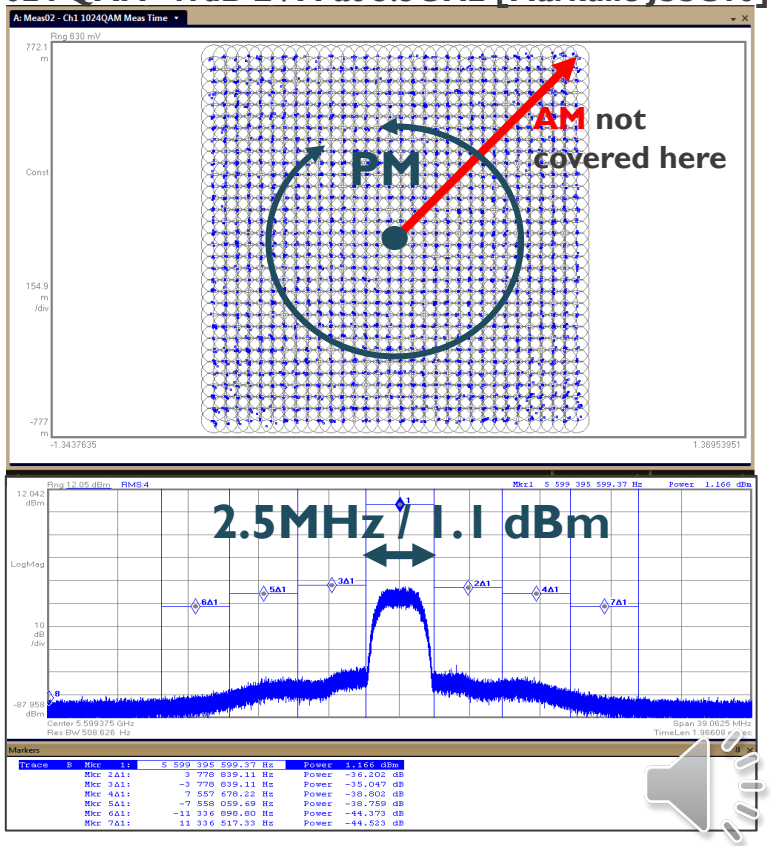
- Subsampling PLL basics
- Fractional Synthesis in a Subsampling PLL
- Digital-to-Time Converter
- Randomization & Background calibration techniques
- **Subsampling PLL-based Phase Modulator**

SUBSAMPLING PLL-BASED PHASE MODULATOR

Motivation

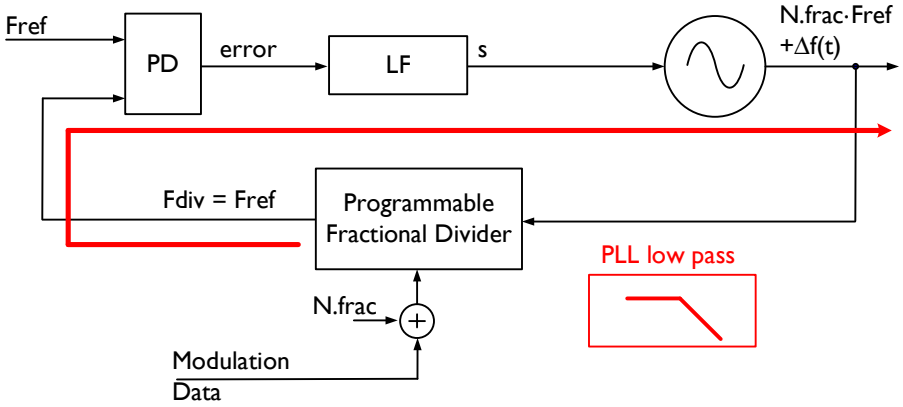
- DTC → phase modulating capabilities
- PLL as a phase modulator → Part of a polar transmitter (PM)
  - Power-efficiency
  - But integrated phase noise translates into EVM
- Low Phase-noise PLL → low EVM
  - Complex constellations schemes possible
  - Spectral efficiency or large number of b/s/Hz
  - Subsampling environment offers very low noise
- Wide-band modulation challenges

1024 QAM -41dB EVM at 5.5GHz [Markulic JSSC16]



SUBSAMPLING PLL-BASED PHASE MODULATOR

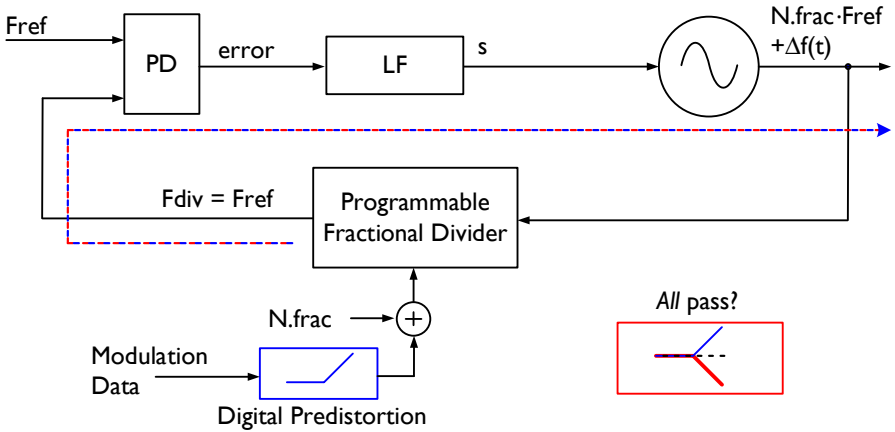
PLL-based modulation basics: classical PLLs



- Modulation using a programmable divider
- Recall: PLL is a low pass filter
  - Bandwidth of modulation severely limited

SUBSAMPLING PLL-BASED PHASE MODULATOR

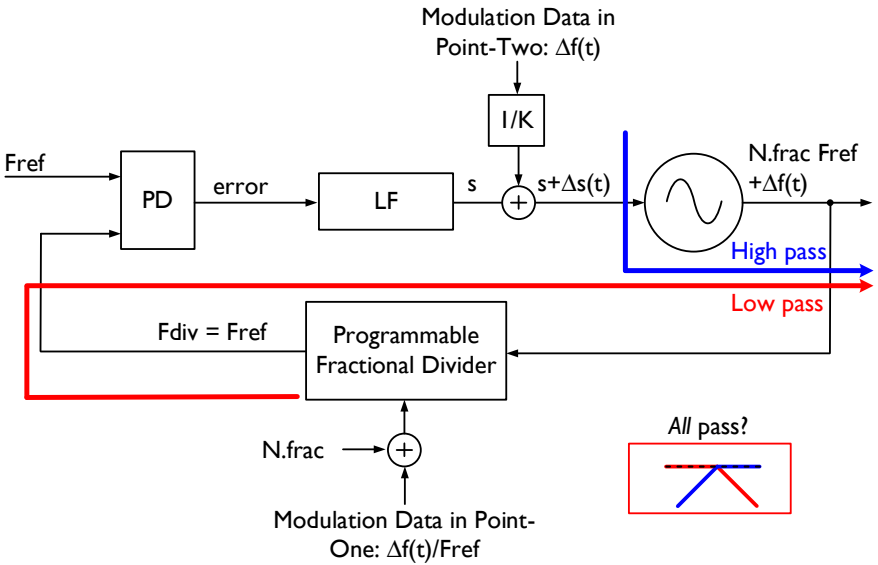
PLL-based modulation basics: predistortion



- Predistortion in digital domain to overcome the loop filtering
- Bandwidth extension rather than data all-pass
  - Limited accuracy

SUBSAMPLING PLL-BASED PHASE MODULATOR

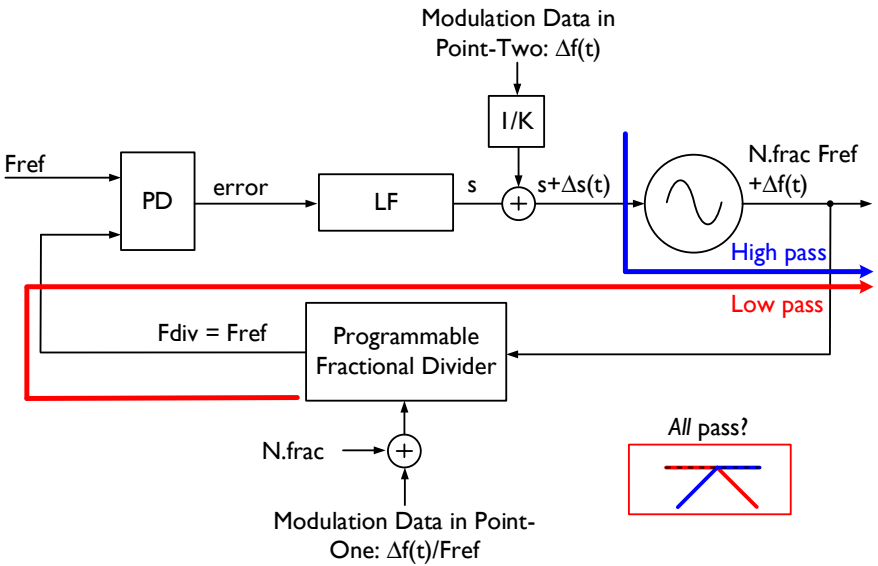
PLL-based modulation basics: two-point modulation



- Injecting modulation signal in the divider
  - Point 1
  - Low-pass characteristic for the signal
- Injecting modulation signal in-front of the controlled oscillator
  - Point-2
  - High-pass characteristic for the signal
- All pass transfer to the output
- Phase modulator = Digital-to-phase converter = DAC
  - BW limed only by Nyquist
  - Aliases

SUBSAMPLING PLL-BASED PHASE MODULATOR

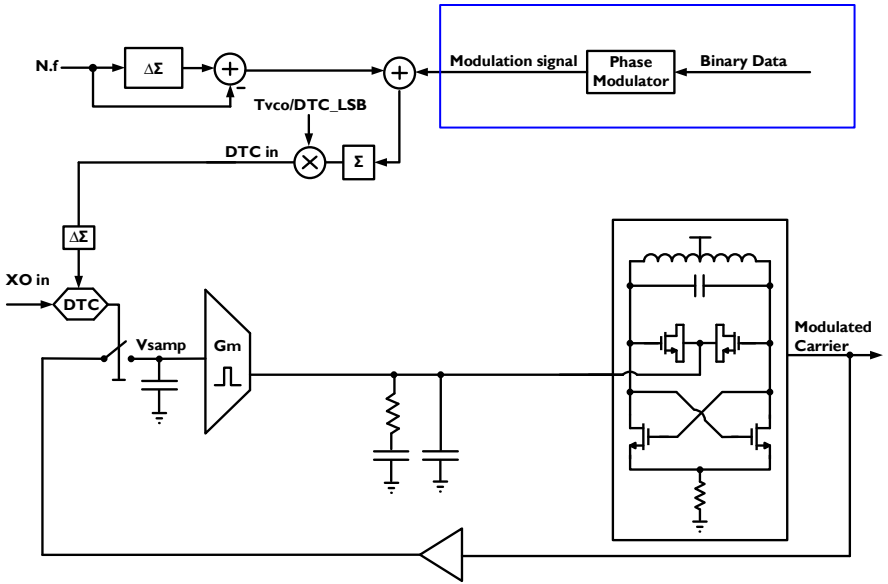
PLL-based modulation basics: two-point modulation



- All pass transfer to the output = No pass to the loop
  - Cancellation in PD
  - Loop remains undisturbed
  - Loop filter at DC
  - Small signal models valid
- Sensitive to injection inaccuracies
  - Point-1 gain/linearity imbalance
  - Point-2 gain/linearity imbalance
  - Timing inaccuracies

SUBSAMPLING PLL-BASED PHASE MODULATOR

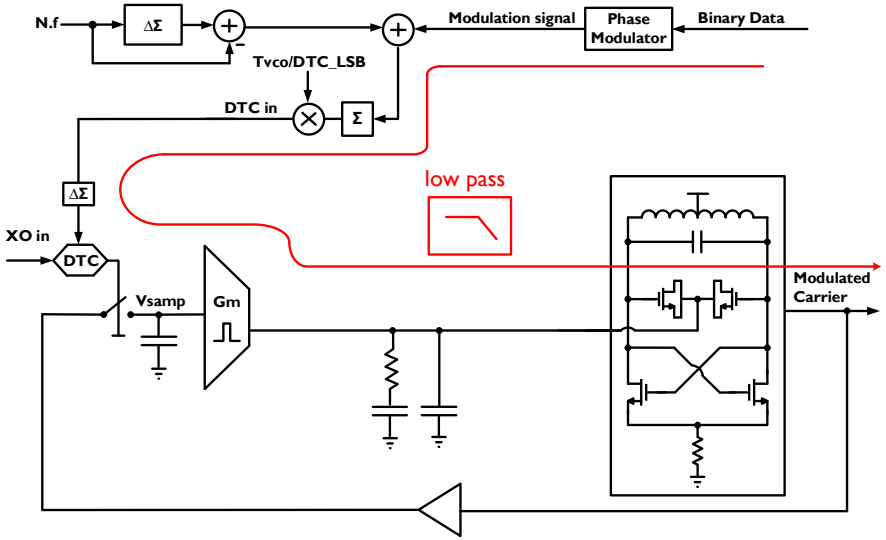
DTC-based modulation in a subsampling PLL



- DTC for phase modulation
  - In a divider-less loop
- Trivial computation path upgrade

SUBSAMPLING PLL-BASED PHASE MODULATOR

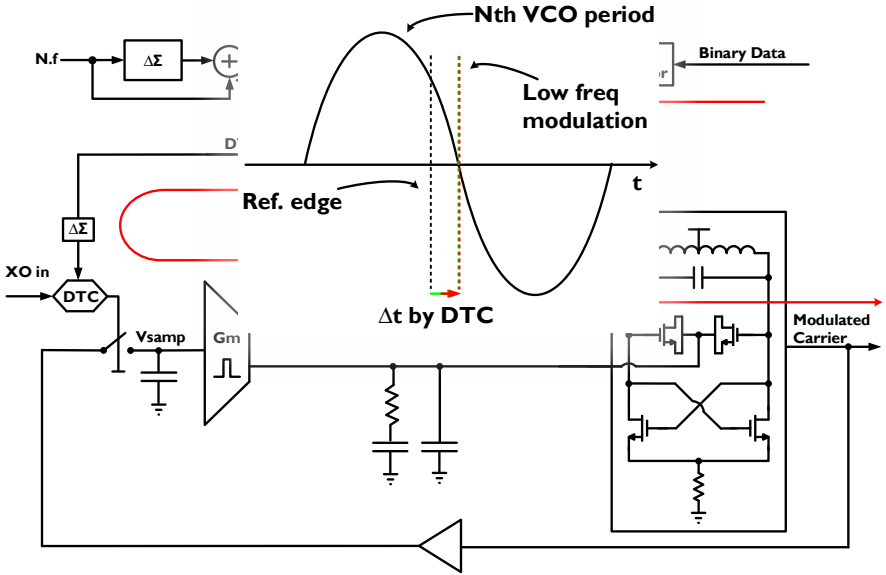
DTC-based modulation in a subsampling PLL: frequency domain analysis



- Low pass characteristic as in classical PLL
- Modulation signal passes through the loop up to PLL cut-off
  - Subsampling PLL → wide bandwidth PLL

SUBSAMPLING PLL-BASED PHASE MODULATOR

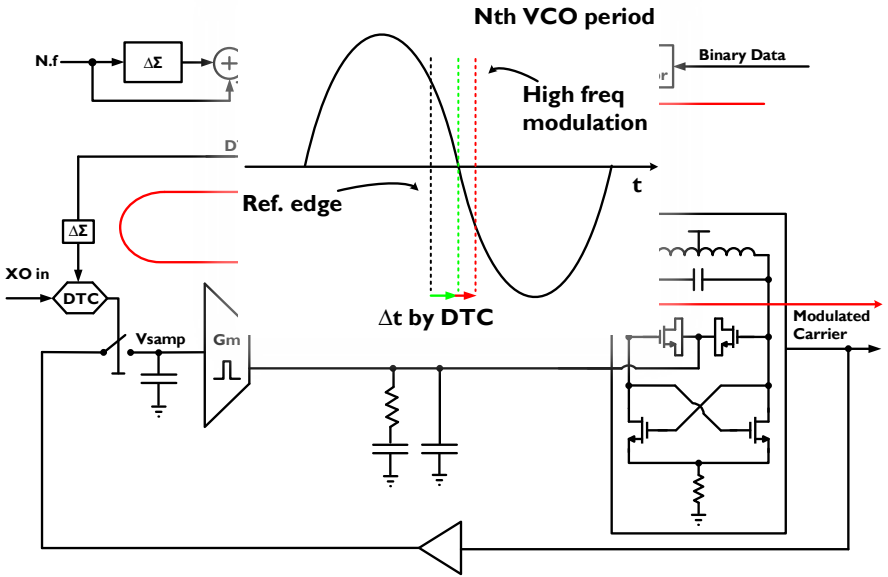
DTC-based modulation in a subsampling PLL: time domain analysis



- Low frequency modulation signal = small instantaneous phase variation at the PLL input/output
- Controlled error injected into the loop
  - Subsampling away from the zero crossing
    - *purposeful* error
  - Loop can track *slow* variation

SUBSAMPLING PLL-BASED PHASE MODULATOR

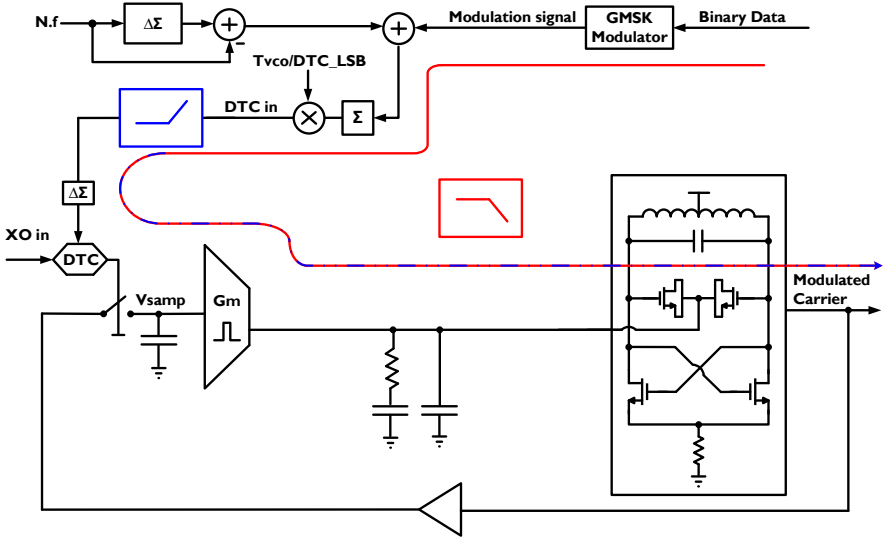
DTC-based modulation in a subsampling PLL: time domain analysis



- High frequency modulation signal = large instantaneous phase variation at the PLL input/output
- PLL attenuates high frequency modulation
  - No time for loop to react
  - Limited linearity

SUBSAMPLING PLL-BASED PHASE MODULATOR

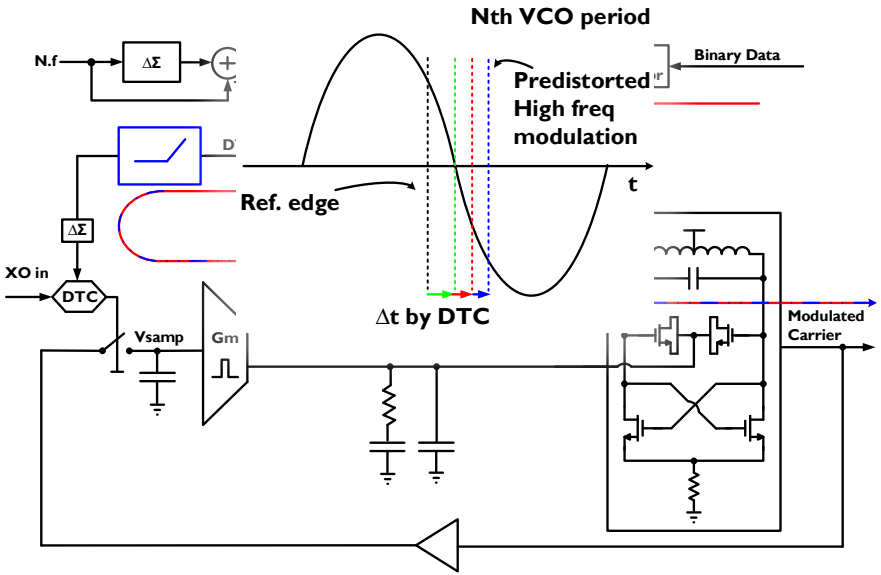
DTC-based modulation in a subsampling PLL: predistortion in frequency domain



- Predistortion in digital domain
  - BW extension rather than a true all-pass
- Filter matching issues
  - Calibration is problematic

SUBSAMPLING PLL-BASED PHASE MODULATOR

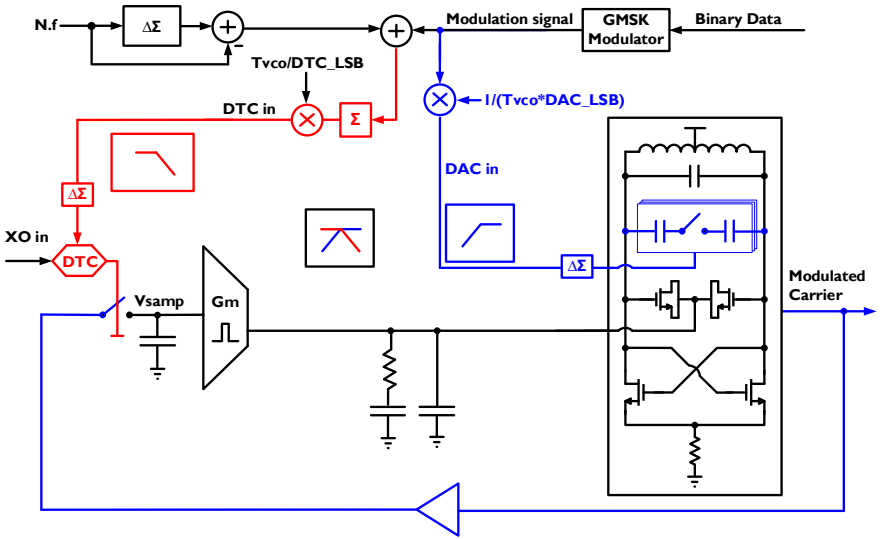
DTC-based modulation in a subsampling PLL: predistortion



- Predistortion means purposeful error become even larger
  - Linearity concern
  - Clipping in extreme cases
  - Small signal model?

SUBSAMPLING PLL-BASED PHASE MODULATOR

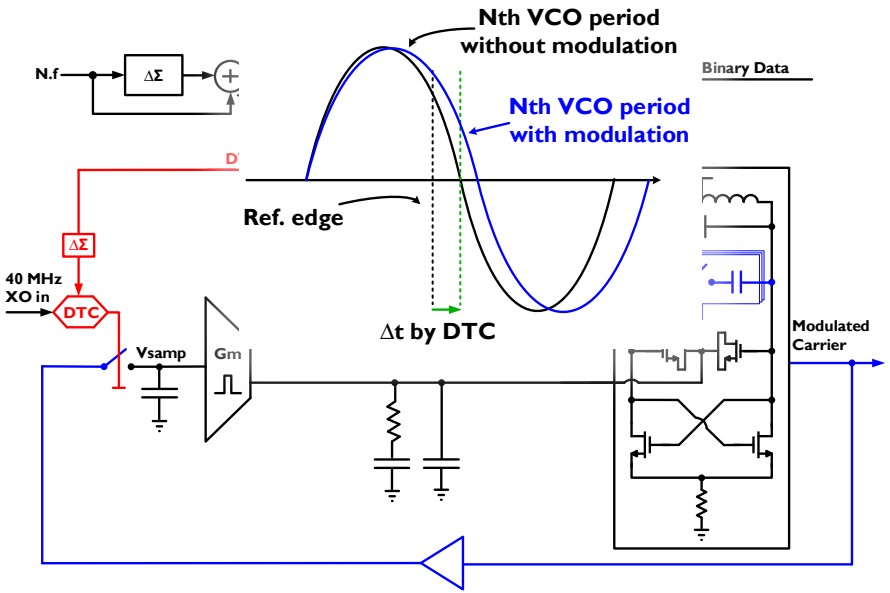
DTC-based modulation in a subsampling PLL: two-point modulation in frequency domain



- Hybrid loop
  - Digital modulation in point-2 (DAC)
  - Digital modulation in point-1 (DTC)
  - Subsampling loop analog
- High pass + Low pass = All pass
  - No pass within the loop
    - Cancellation at the sampler
  - Quite/stable loop filter voltage

SUBSAMPLING PLL-BASED PHASE MODULATOR

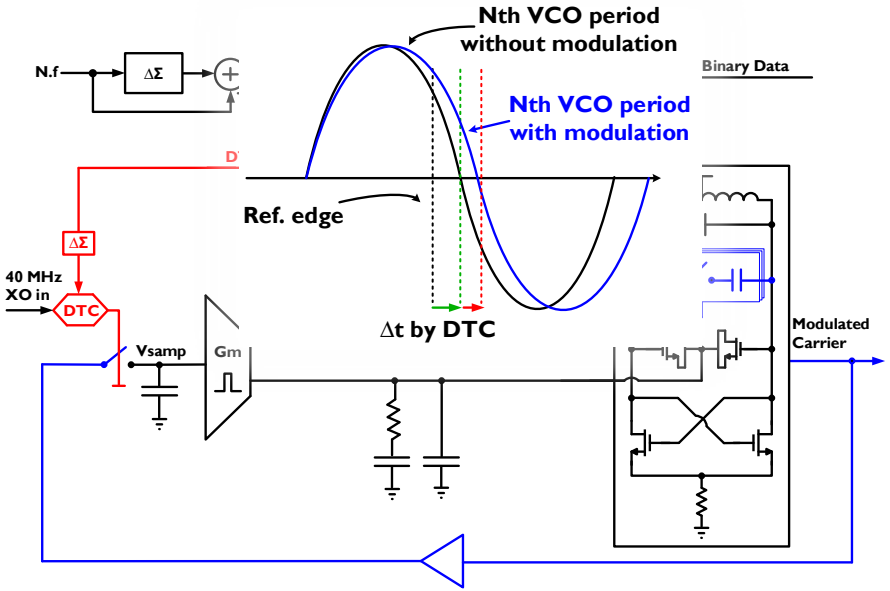
DTC-based modulation in a subsampling PLL: two-point modulation in time domain



- Point-2 initiates modulation
  - Modulated period ≠ fractional-N period

SUBSAMPLING PLL-BASED PHASE MODULATOR

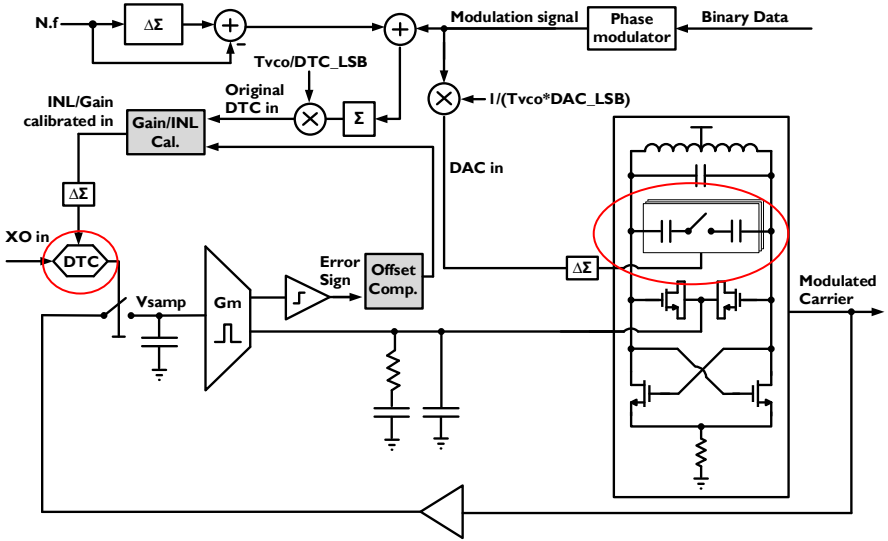
DTC-based modulation in a subsampling PLL: two-point modulation in time domain



- Point-2 receives information about the phase shift
- Sampling the zero crossing
  - Linear small signal operation valid

SUBSAMPLING PLL-BASED PHASE MODULATOR

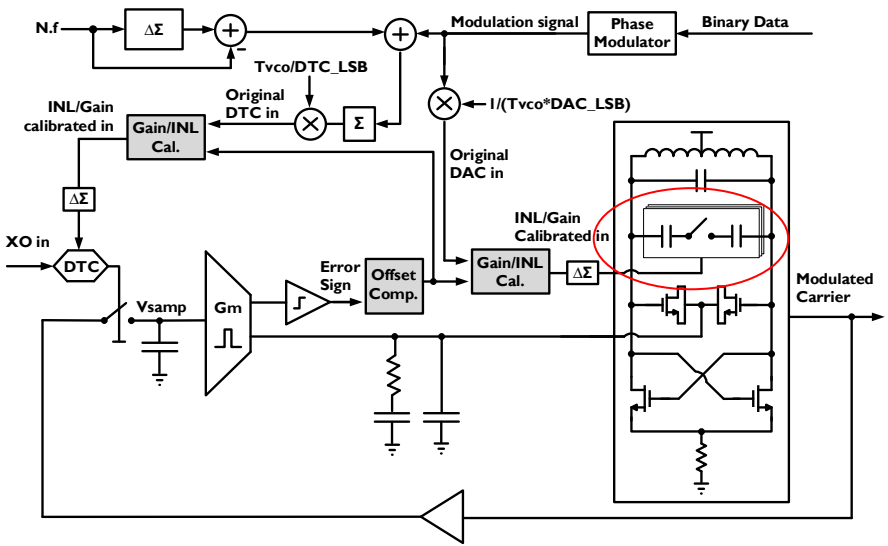
DTC-based modulation in a subsampling PLL: background calibration



- Point-1 inaccuracies calibrated as in fractional-N mode
  - DTC gain/INL algorithm
- Point-2 inaccuracies
  - Nonlinearity likely pronounced at wide-band modulation
    - Nonlinear capacitance-to-frequency conversion
    - DAC INL

SUBSAMPLING PLL-BASED PHASE MODULATOR

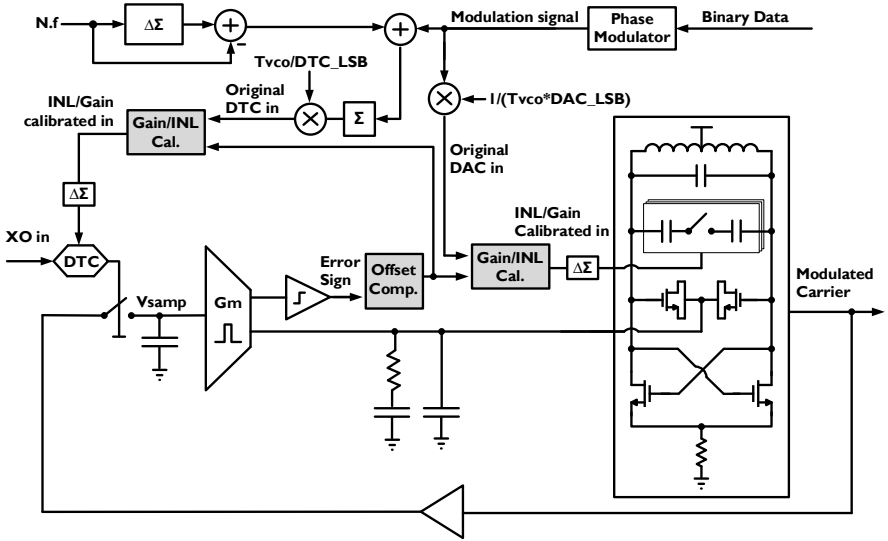
DTC-based modulation in a subsampling PLL: background calibration



- Re-use of DTC calibration techniques
  - Zero crossing subsampling valid
  - Same principles
- Phase-error colored by DAC or DTC
  - Correlation → correct error mapping
  - Can operate simultaneously/continuously in the background

SUBSAMPLING PLL-BASED PHASE MODULATOR

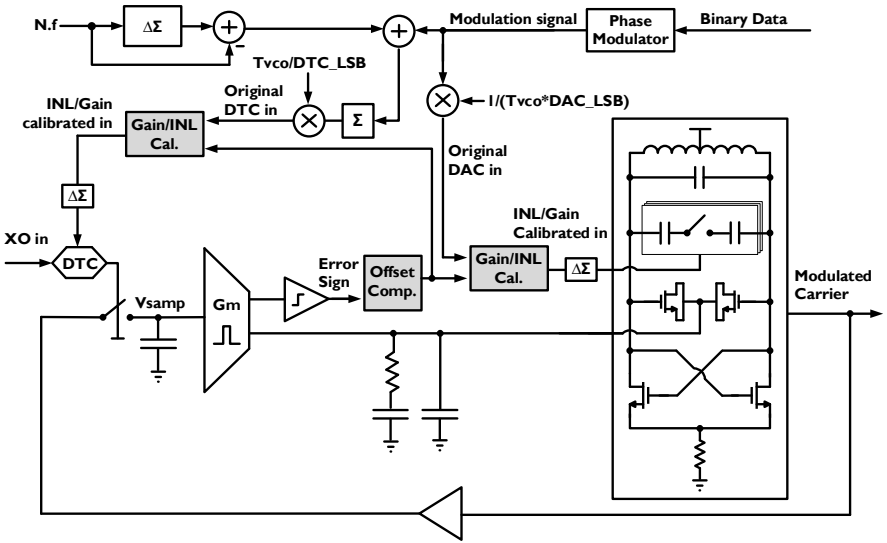
DTC-based modulation in a subsampling PLL: background calibration



- Two-point injection gain imbalance/nonlinearity resolved
  - DAC induces desired frequency shift
  - DTC induces desired time delay

SUBSAMPLING PLL-BASED PHASE MODULATOR

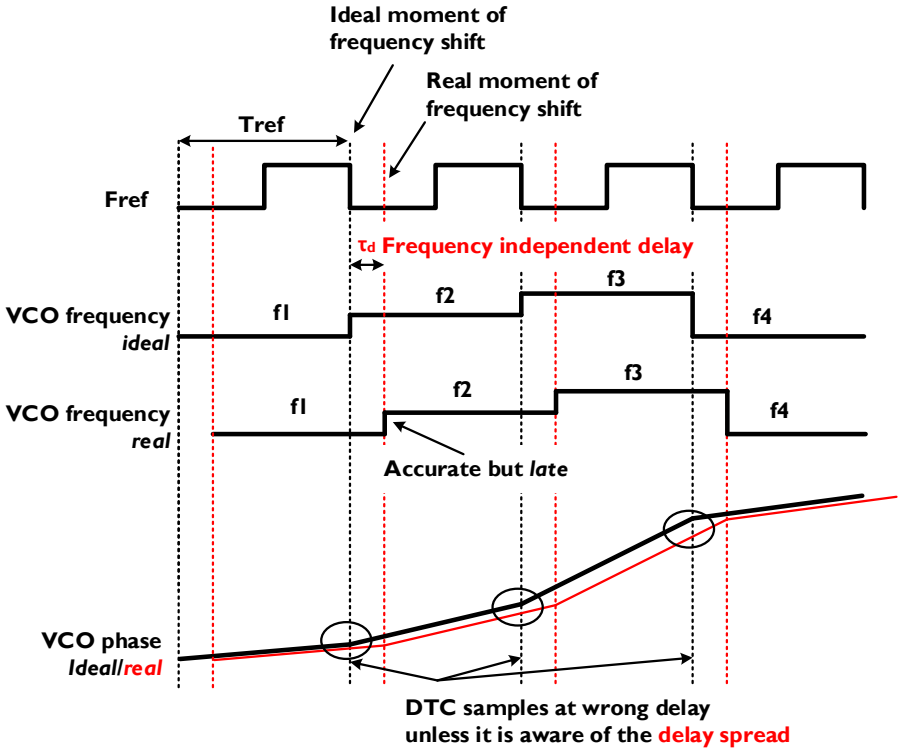
DTC-based modulation in a subsampling PLL: background calibration



- What about timing inaccuracies?
  - DAC induces *correct* frequency shift at a *wrong moment* in time
    - Non-instant frequency shift frequency shift
    - Clock skew
  - Important at higher modulation bandwidths
    - >10 MHz

SUBSAMPLING PLL-BASED PHASE MODULATOR

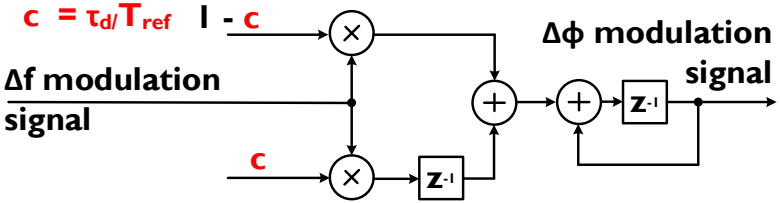
DTC-based modulation in a subsampling PLL: background calibration



- "Delay spread"
- DAC induces *correct* frequency shift at a *wrong moment* in time
  - Typically code/frequency independent delay
- DTC *should be* delay spread (τd) aware

SUBSAMPLING PLL-BASED PHASE MODULATOR

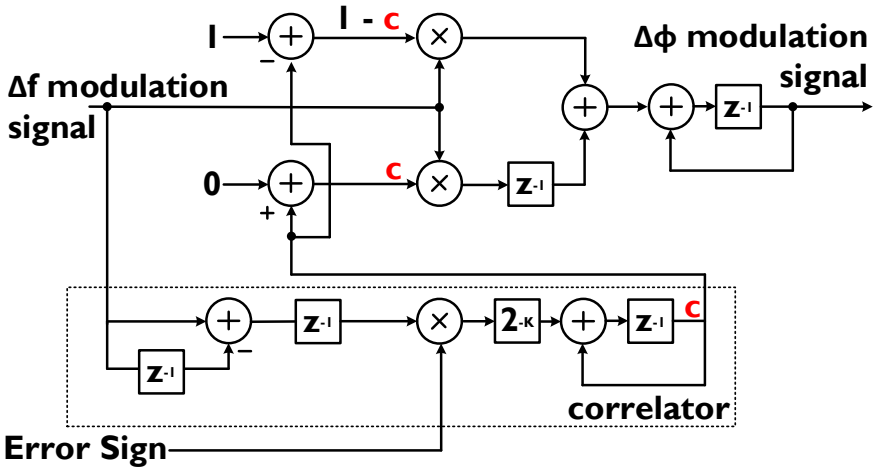
DTC-based modulation in a subsampling PLL: background calibration



- Simple predistortion technique for delay spread
  - DTC "becomes aware" of phase accumulation at the wrong frequency

SUBSAMPLING PLL-BASED PHASE MODULATOR

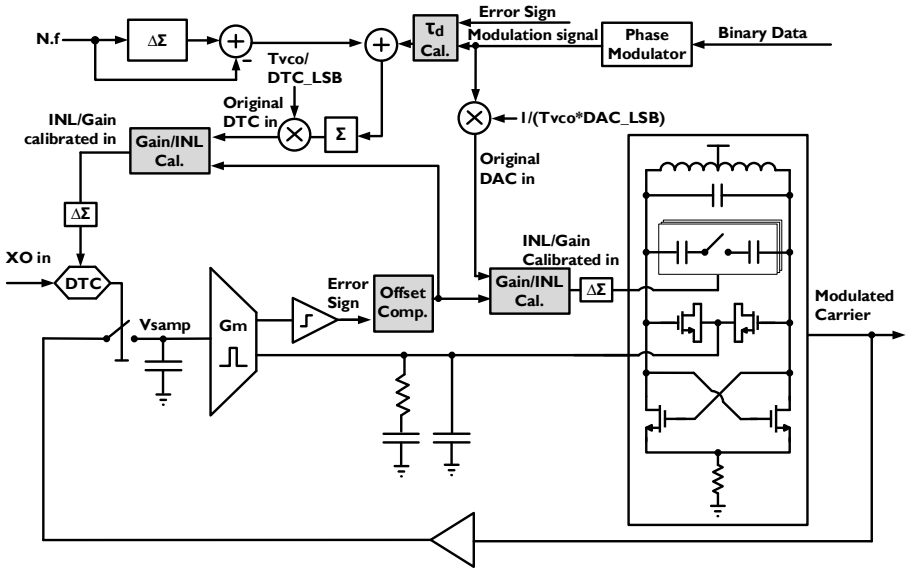
DTC-based modulation in a subsampling PLL: background calibration



- Error can be tracked in the background

SUBSAMPLING PLL-BASED PHASE MODULATOR

DTC-based modulation in a subsampling PLL: summary



- Subsampling for low phase noise
  - Very low fundamental EVM limitation
- Fully calibrated environment
  - No injection gain/nonlinearity imbalance
  - No timing mismatch
  - Wideband possible

CONCLUSION

From a Subsampling PLL to a Background-Calibrated Subsampling Phase Modulator

- Analog subsampling core for extreme low noise operation
- DTC-based environment for superior modulation capabilities
- Digital background calibration engines for mitigation of analog inaccuracies

REFERENCES

- Integer-N Subsampling PLL
  - Gao, Xiang, et al. "A low noise sub-sampling PLL in which divider noise is eliminated and PD/CP noise is not multiplied by N^2 " *IEEE Journal of solid-state circuits* 44.12 (2009): 3253-3263.
  - Gao, Xiang, et al. "Spur reduction techniques for phase-locked loops exploiting a sub-sampling phase detector." *IEEE journal of solid-state circuits* 45.9 (2010): 1809-1821.
  - Gao, Xiang, Eric Klumperink, and Bram Nauta. "Sub-sampling PLL techniques." *2015 IEEE Custom Integrated Circuits Conference (CICC)*. IEEE, 2015.
  - Kim, Juyeop, et al. "16.2 A 76fs rms Jitter and–40dBc Integrated-Phase-Noise 28-to-31GHz Frequency Synthesizer Based on Digital Sub-Sampling PLL Using Optimally Spaced Voltage Comparators and Background Loop-Gain Optimization." *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 2019.
  - Sharkia, Ahmad, Shahriar Mirabbasi, and Sudip Shekhar. "A 0.01 mm 2 4.6-to-5.6 GHz sub-sampling type-I frequency synthesizer with− 254dB FOM." *2018 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 2018.
  - Yang, Zunsong, et al. "16.8 A 25.4-to-29.5 GHz 10.2 mW Isolated Sub-Sampling PLL Achieving-252.9 dB Jitter-Power FoM and-63dBc Reference Spur." *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 2019.
- Digital-to-Time Conversion
  - Ru, Jiayoon Zhiyu, et al. "A high-linearity digital-to-time converter technique: Constant-slope charging." *IEEE journal of solid-state circuits* 50.6 (2015): 1412-1423.
  - Markulic, Nereo, et al. "A 10-bit, 550-fs step Digital-to-Time Converter in 28nm CMOS." *ESSCIRC 2014-40th European Solid State Circuits Conference (ESSCIRC)*. IEEE, 2014.
  - Chen, Peng, et al. "A 15-μW, 103-fs step, 5-bit capacitor-DAC-based constant-slope digital-to-time converter in 28nm CMOS." *2017 IEEE Asian Solid-State Circuits Conference (A-SSCC)*. IEEE, 2017.
  - Liu, Hanli, et al. "A 0.98 mW fractional-N ADPLL using 10b isolated constant-slope DTC with FOM of− 246dB for IoT applications in 65nm CMOS." *2018 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 2018.

REFERENCES

- Fractional-N subsampling PLLs
  - Raczkowski, Kuba, et al. "A 9.2–12.7 GHz wideband fractional-N subsampling PLL in 28 nm CMOS with 280 fs RMS jitter." *IEEE Journal of Solid-State Circuits* 50.5 (2015): 1203-1213.
  - Markulic, Nereo, et al. "A DTC-based subsampling PLL capable of self-calibrated fractional synthesis and two-point modulation." *IEEE Journal of Solid-State Circuits* 51.12 (2016): 3078-3092.
  - Gao, Xiang, et al. "9.6 A 2.7-to-4.3 GHz, 0.16 psrms-jitter,− 246.8 dB-FOM, digital fractional-N sampling PLL in 28nm CMOS." *2016 IEEE International Solid-State Circuits Conference (ISSCC)*. IEEE, 2016.
  - Chang, Wei-Sung, Po-Chun Huang, and Tai-Cheng Lee. "A fractional-N divider-less phase-locked loop with a subsampling phase detector." *IEEE Journal of Solid-State Circuits* 49.12 (2014): 2964-2975.
  - Liu, Hanli, et al. "16.1 A 265μW Fractional-N Digital PLL with Seamless Automatic Switching Subsampling/Sampling Feedback Path and Duty-Cycled Frequency-Locked Loop in 65nm CMOS." *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 2019.
  - Chen, Zuow-Zun, et al. "14.9 sub-sampling all-digital fractional-N frequency synthesizer with− 111dBc/Hz in-band phase noise and an FOM of− 242dB." *2015 IEEE International Solid-State Circuits Conference-(ISSCC) Digest of Technical Papers*. IEEE, 2015.
- Randomization and Background calibration techniques for DTC-based PLLs
  - He, Yuming, et al. "24.7 A 673μW 1.8-to-2.5 GHz dividerless fractional-N digital PLL with an inherent frequency-capture capability and a phase-dithering spur mitigation for IoT applications." *2017 IEEE International Solid-State Circuits Conference (ISSCC)*. IEEE, 2017.
  - Levantino, Salvatore, Giovanni Marzin, and Carlo Samori. "An adaptive pre-distortion technique to mitigate the DTC nonlinearity in digital PLLs." *IEEE Journal of Solid-State Circuits* 49.8 (2014): 1762-1772.
  - Marzin, Giovanni, et al. "2.9 A background calibration technique to control bandwidth in digital PLLs." *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*. IEEE, 2014.

REFERENCES

- Phase/Frequency Modulation in DTC-based PLLs
  - Marzin, Giovanni, et al. "A 20 Mb/s phase modulator based on a 3.6 GHz digital PLL with− 36 dB EVM at 5 mW power." *IEEE journal of solid-state circuits* 47.12 (2012): 2974-2988.
  - Markulic, Nereo, et al. "9.7 a self-calibrated 10mb/s phase modulator with-37.4 db evm based on a 10.1-to-12.4 ghz,-246.6 db-fom, fractional-n subsampling pll." *2016 IEEE International Solid-State Circuits Conference (ISSCC)*. IEEE, 2016.
  - Markulic, Nereo, et al. "A 5.5-GHz Background-Calibrated Subsampling Polar Transmitter With− 41.3-dB EVM at 1024 QAM in 28-nm CMOS." *IEEE Journal of Solid-State Circuits* 54.4 (2019): 1059-1073.
  - Shi, Qixian, et al. "A Self-Calibrated 16-GHz Subsampling-PLL-Based Fast-Chirp FMCW Modulator With 1.5-GHz Bandwidth." *IEEE Journal of Solid-State Circuits* 54.12 (2019): 3503-3512.

THANK YOU!

imec

embracing a better life