ECE 621
Signaling and Synchronization
Fall 2020

Topic 8
PLL & DLL Basics

Sameh A. Ibrahim

Ain Shams University

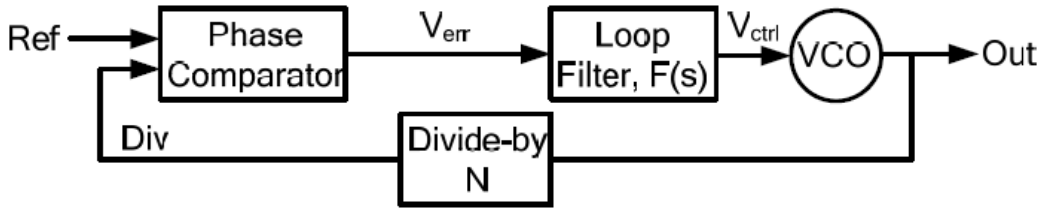
ICL

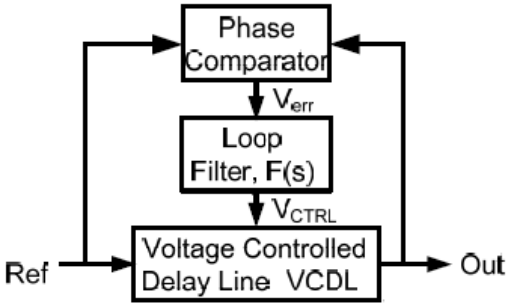
(Courtesy of S. Pamarti & B. Razavi – UCLA and S. Palermo – TAMU)

Outline

- **PLL**
  - Components
  - Models
  - Noise Transfer Functions
  - Circuits
- **DLL**
  - Types
  - Models
  - Noise Transfer Functions
- **PLL/DLL comparison**

Introduction





PLL

- **Negative Feedback Loop**
  - Generated clock
  - Same frequency and phase
- **PLLs applications**
  - Frequency synthesis
  - Skew cancellation
  - Clock recovery
  - Modulation/De-modulation

DLL

- **Negative Feedback Loop**
  - No clock generation
  - Defined phase relationship
- **DLLs applications**
  - Chip-to-chip communication
  - De-skew circuits

Loop Filter

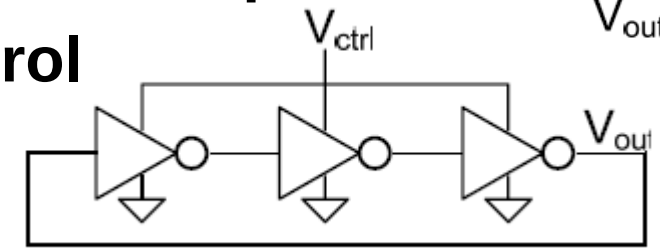
- **Many implementations**
  - Passive, active, sampled
- **Filter classification**
  - Based on number of integrators in F(s)
    - Type I: no integrators
    - Type II: one integrator
  - Based on number of poles in F(s)
    - 1st order, 2nd order, 3rd order, etc.

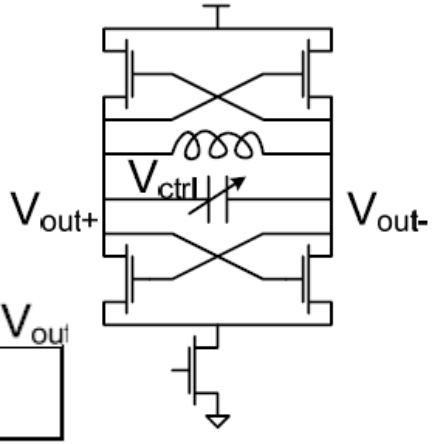
$$Type\ I\text{: } F(s)=1,\ F(s)=\frac{1+s\tau_2}{1+s\tau_1},$$

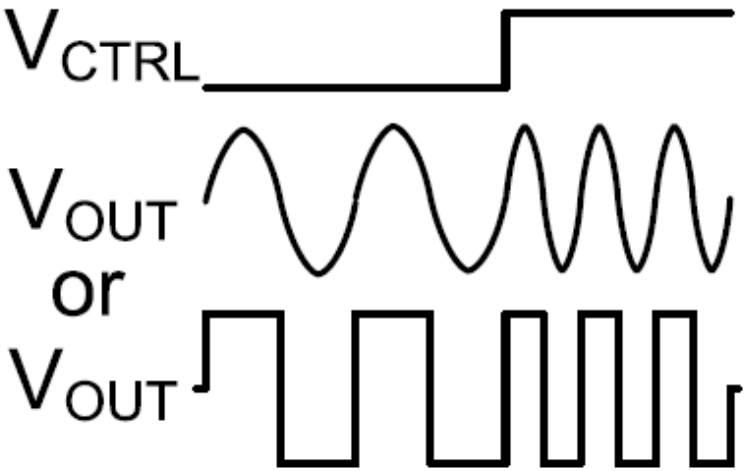
$$Type\ II\text{: } F(s)=\frac{1+s\tau_2}{s},\ F(s)=\frac{1+s\tau_2}{s(1+s\tau_p)}.$$

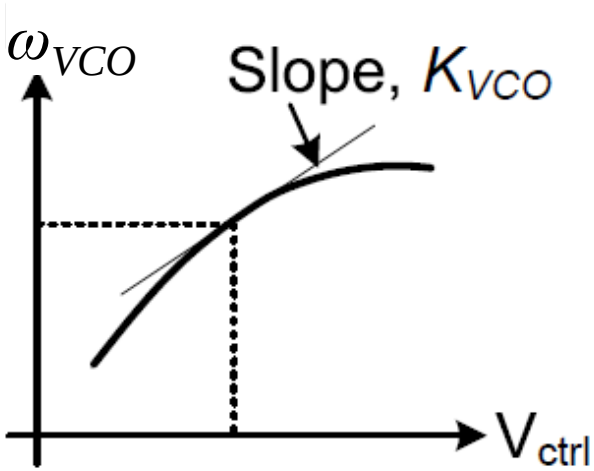
Voltage Control Oscillators

- **Ring or LC oscillators**
- **Differential or single-ended outputs**
- **Voltage or current control**





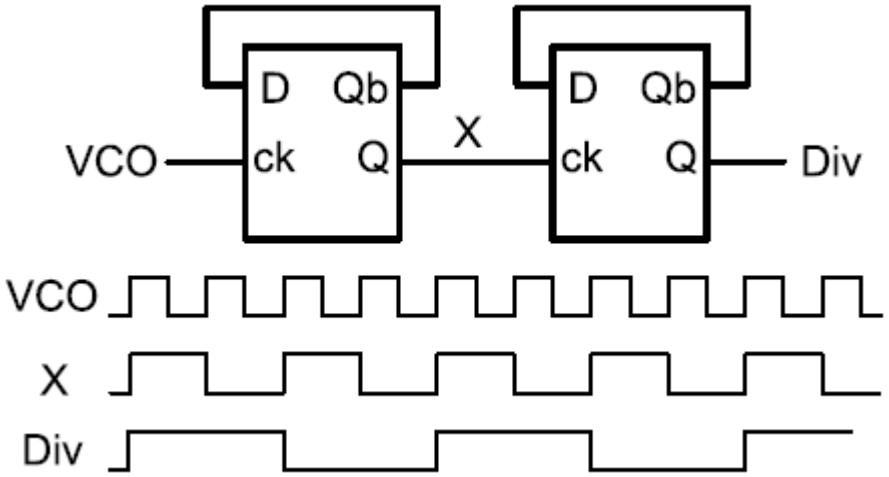




$$\omega_{VCO}(t) = K_{VCO} V_{ctrl}(t) \ rad/s,$$

$$\phi_{VCO}(t) = \int_{-\infty}^{t} K_{VCO} V_{ctrl}(\tau) d\tau \ radians$$

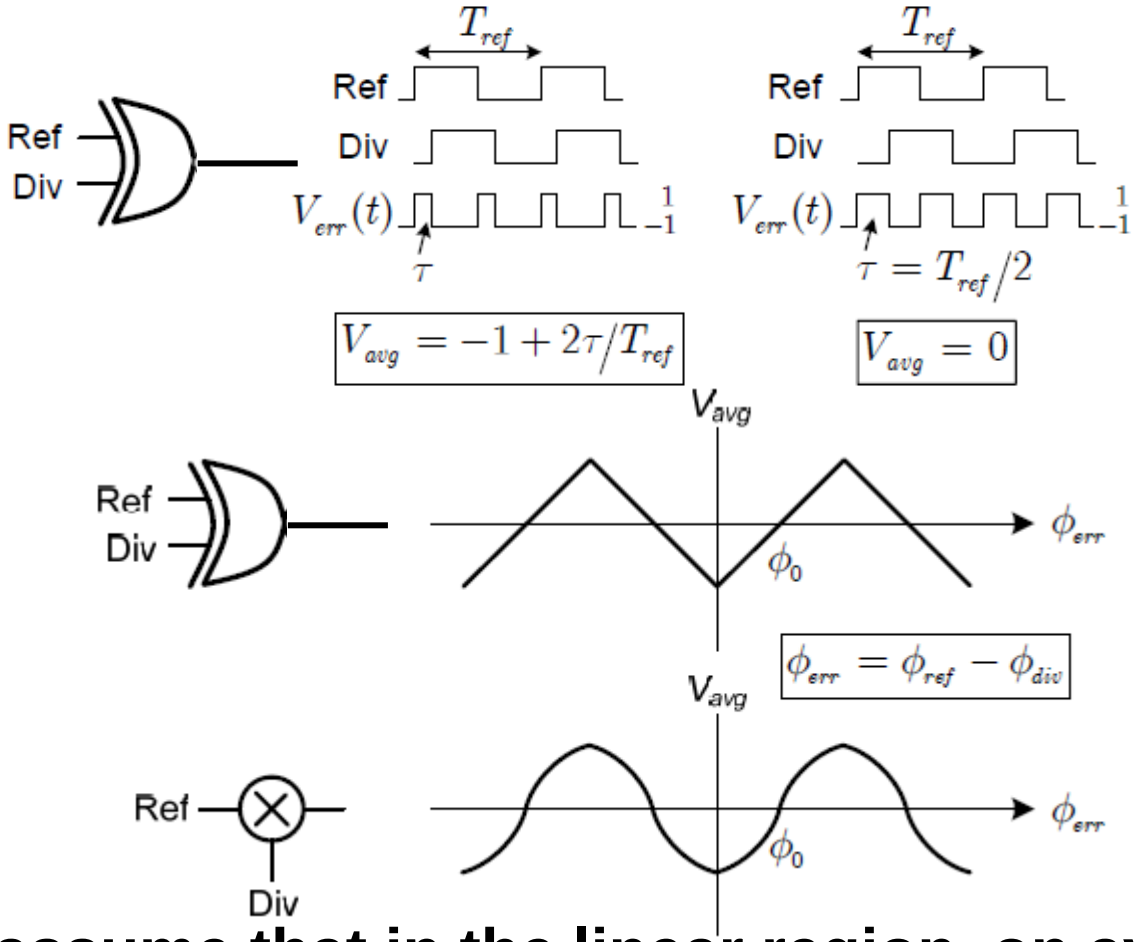
Frequency Divider



- **Counts integer number of VCO cycles**
  - Synchronous or asynchronous implementations
- **Many implementations**
  - Static CMOS, dynamic logic, current mode logic (CML)

$$\phi_{div}(t) = \frac{1}{N}\phi_{VCO}(t)$$

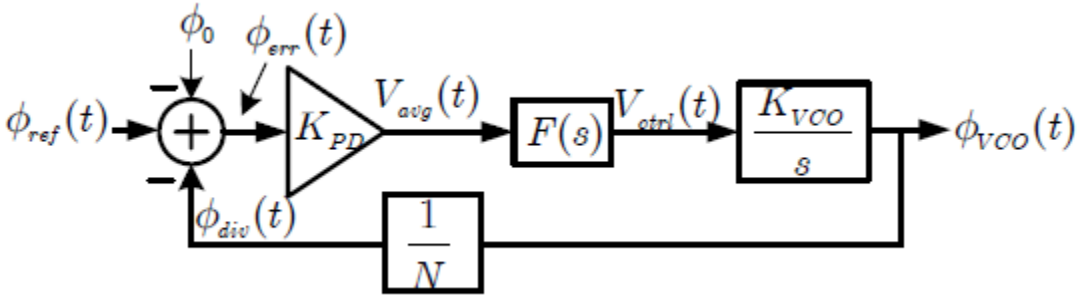
Phase Comparator / Detector



- **Can assume that in the linear region, on average,**

$$V_{avg}(t) \approx K_{PD}\left(\phi_{ref}(t) - \phi_{div}(t) - \phi_0\right)$$

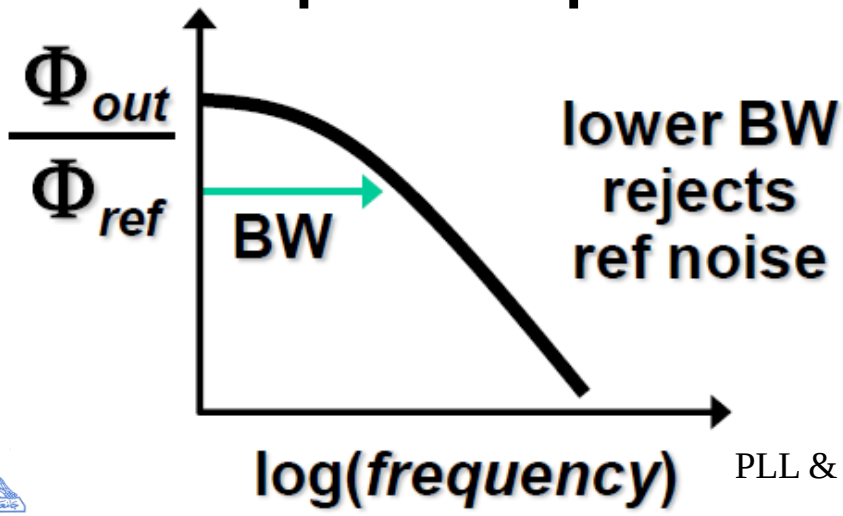
Intuitive PLL Model

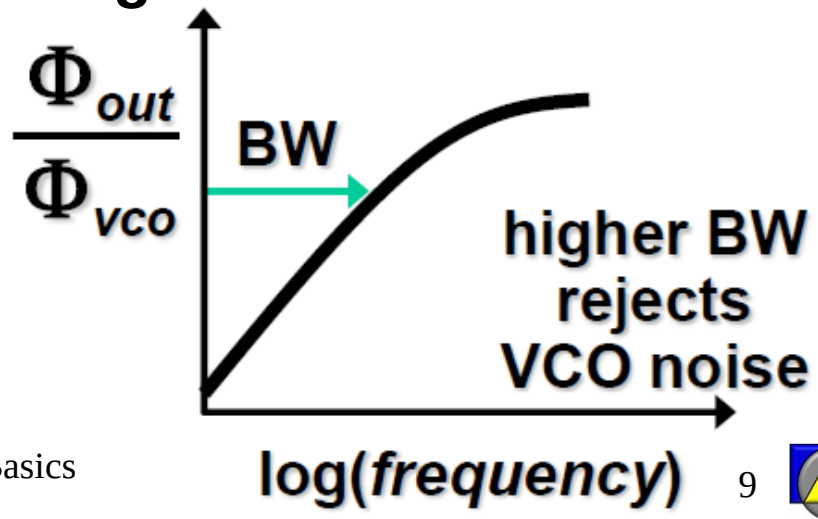


- **Negative feedback loop tries to force $\Phi_{err}(t) = \Phi_0$**
  - In steady state i.e. in "lock" $f_{VCO} = N * f_{ref}$
- **The continuous time, analog model is in general incorrect**
  - Many components have discrete time operation.
  - However, important conclusions can be drawn from it.

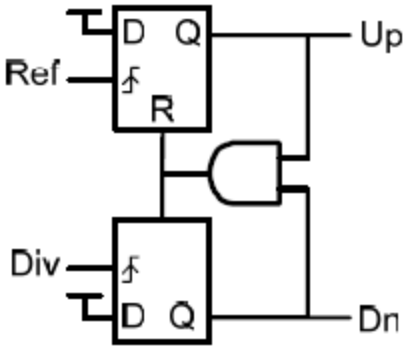
Understanding PLL Frequency Response

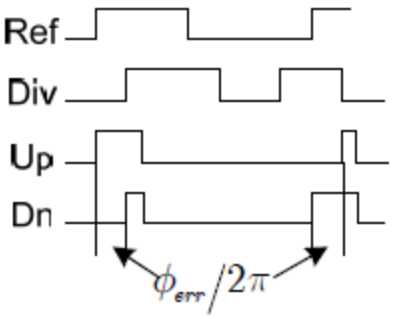
- **Linear "small-signal" analysis is useful for understanding PLL dynamics if**
  - PLL is locked (or near lock).
  - Input phase deviation amplitude is small enough to maintain operation in lock range.
- **Frequency domain analysis can tell us how well the PLL tracks the input phase as it changes at a certain frequency.**
- **PLL transfer function is different depending on which point in the loop the output is responding to.**

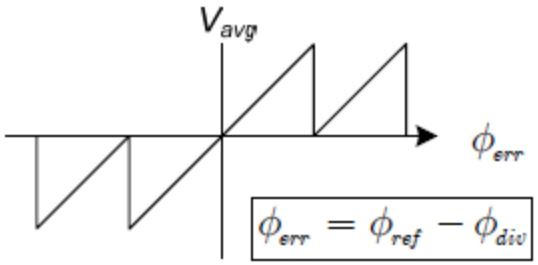




Phase Frequency Detector

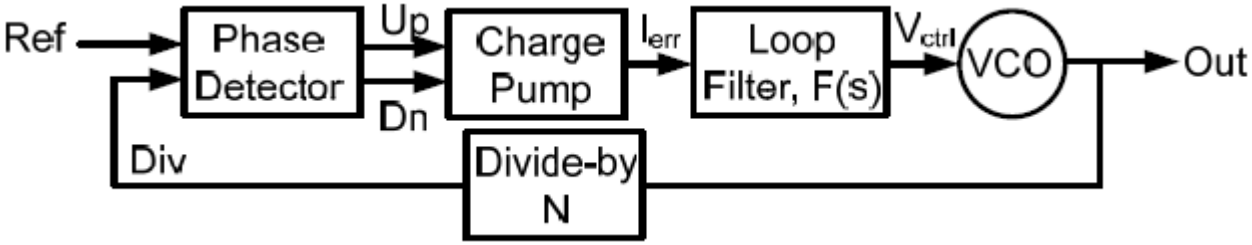


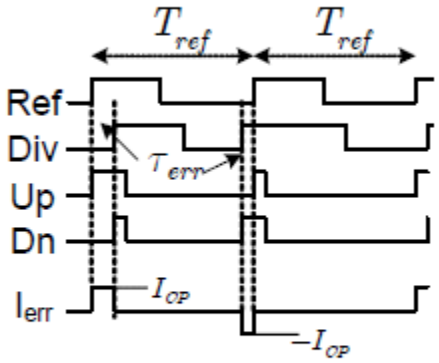


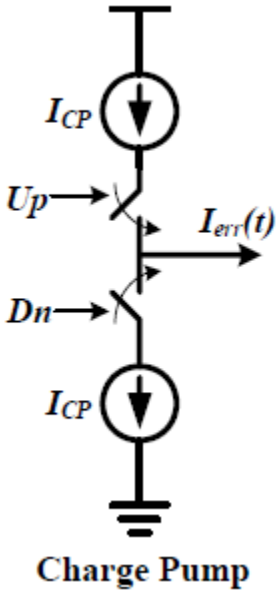


- **"Up" and "Dn" together contain the phase error information.**
  - Their time difference is proportional to the phase error.
- **A proportional voltage or current can be generated for F(s).**
  - A charge pump is best suited for this purpose.

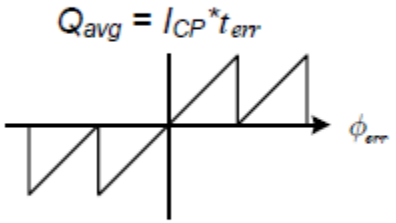
Charge Pump Integer-N PLLs





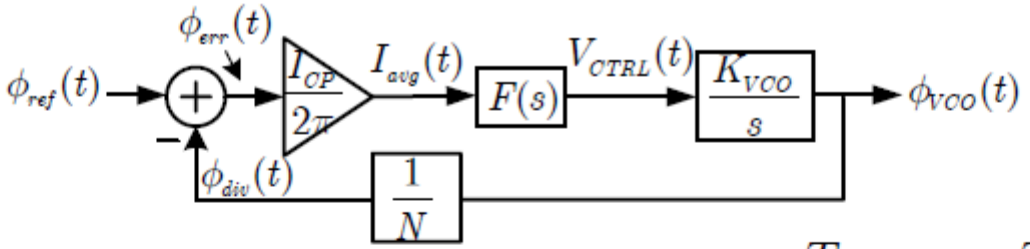


Charge Pump

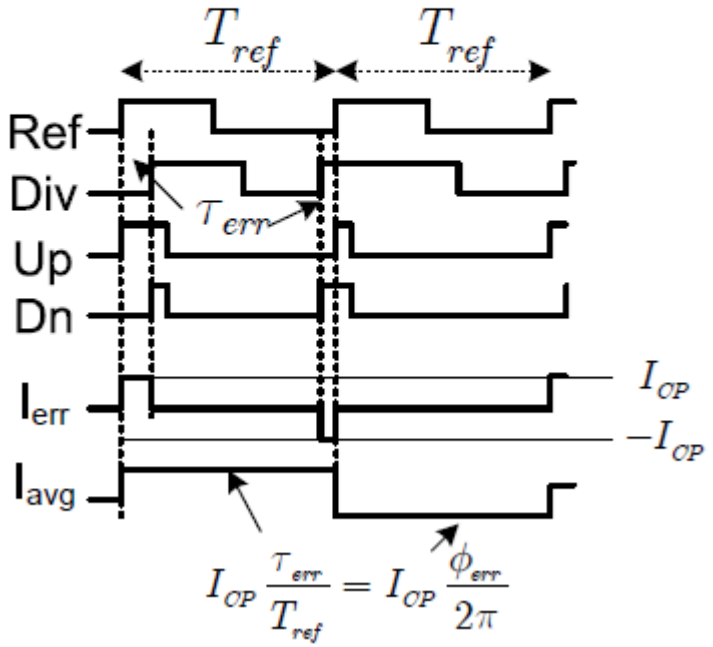


- **PFD + CP together dump charge, Q_{avg} proportional to Φ_{err} into the loop filter, during each reference period.**

Charge Pump Integer-N PLLs: LTI Model

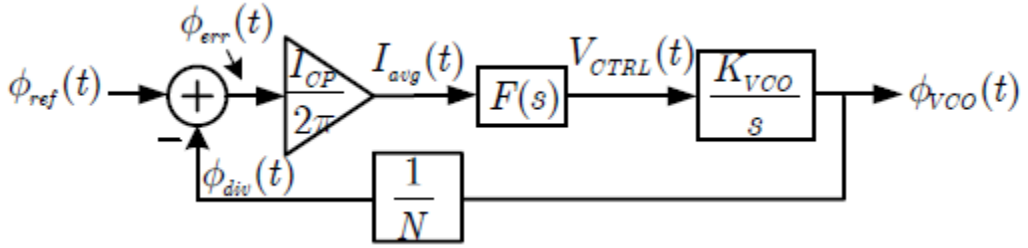


Assumption: CP produces an average current during each reference period.

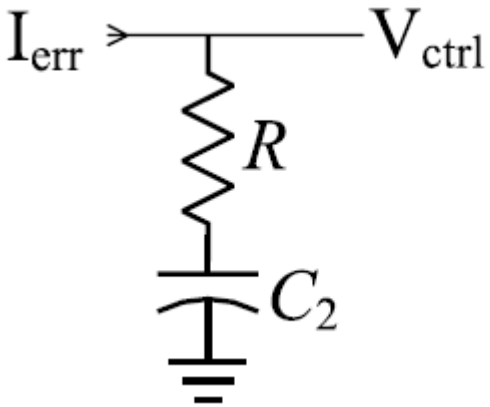


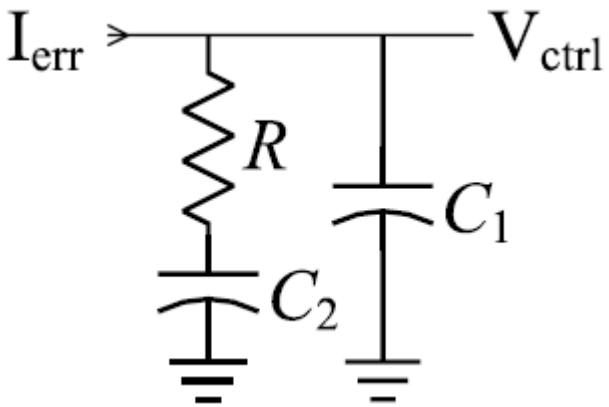
- **Linear, time-invariant model**
  - Ignores the non-linearity of the PFD
  - Ignores position and width of current pulses
  - Reasonable approximation for low bandwidth PLLs

Common Charge Pump Integer-N PLLs



- **Loop filter is usually of Type II.**





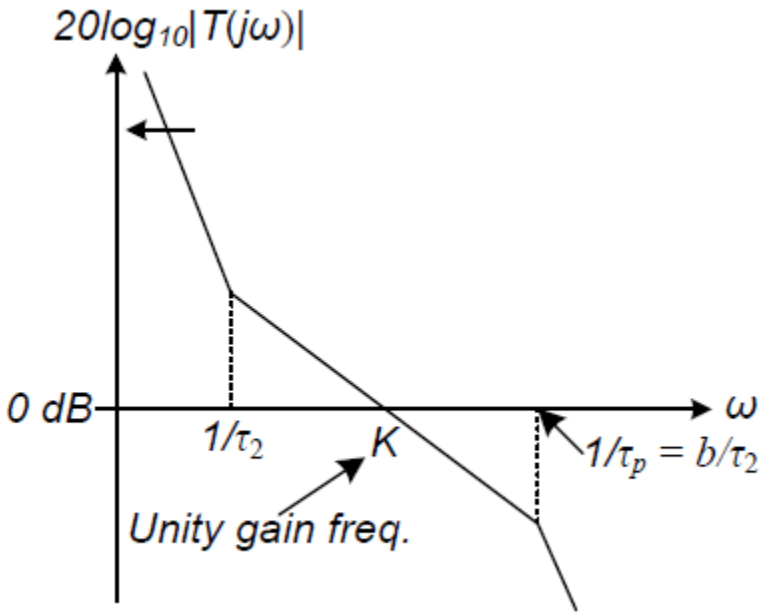
$$F(s) = R\left(\frac{1+s\tau_2}{s\tau_2}\right)$$

$$\tau_2 \triangleq RC_2,$$

$$F(s) = R\left(\frac{b-1}{b}\right)\left(\frac{1+s\tau_2}{s\tau_2\left(1+s\tau_p\right)}\right)$$

$$b \triangleq 1+\frac{C_2}{C_1},\ \tau_p \triangleq R\frac{C_1C_2}{C_1+C_2} = \frac{\tau_2}{b}$$

Stability (1)

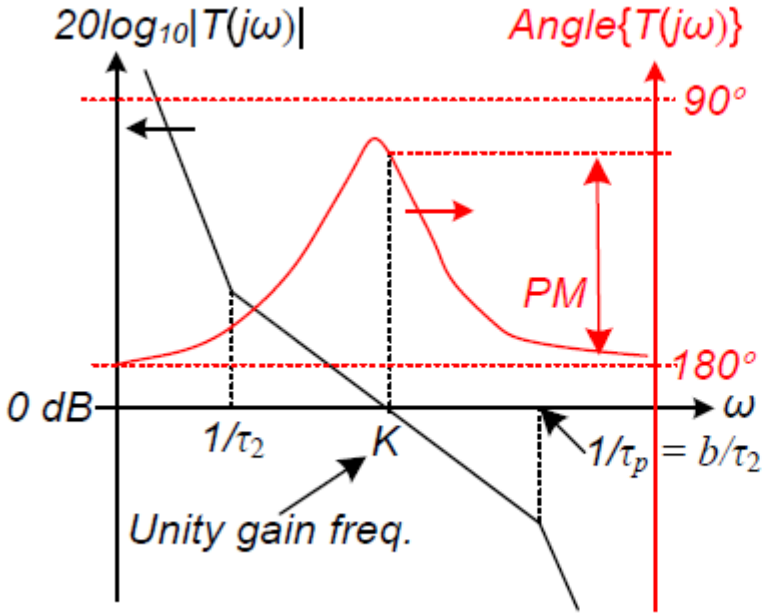


$$T(s) = \frac{K\left(1 + s\tau_2\right)}{s^2\tau_2\left(1 + s\tau_p\right)},$$

$$K = \left(\frac{b-1}{b}\right)\frac{I_{CP}K_{VCO}R}{2\pi N},$$

$$PM \approx \tan^{-1}\left(K\tau_2\right) - \tan^{-1}\left(K\tau_2/b\right)$$

Stability (2)



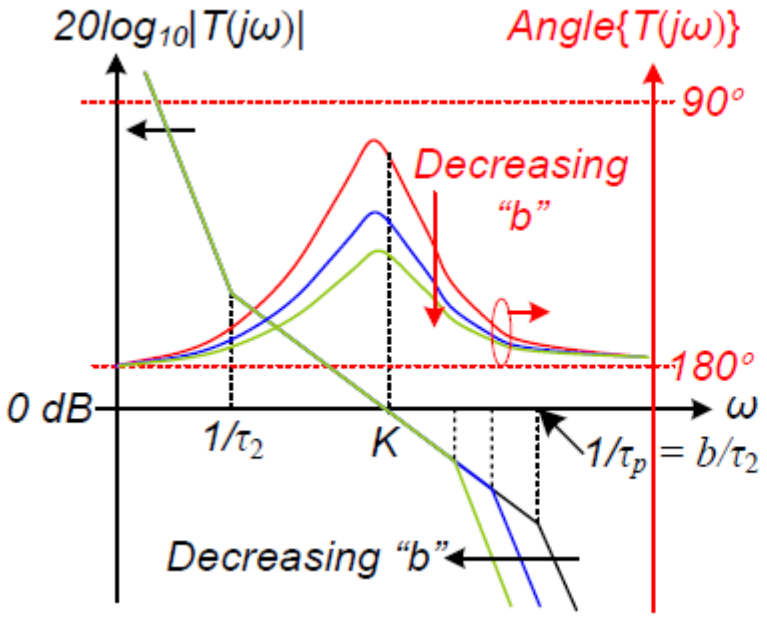
$$T(s) = \frac{K(1+s\tau_2)}{s^2\tau_2(1+s\tau_p)},$$

$$K = \left(\frac{b-1}{b}\right)\frac{I_{CP}K_{VCO}R}{2\pi N},$$

$$PM \approx \tan^{-1}(K\tau_2) - \tan^{-1}(K\tau_2/b)$$

- **Two poles at s= 0 cause instability**
  - The zero provides phase margin (PM).
  - Large $K\tau_2$ values give better PM.

Stability (3)



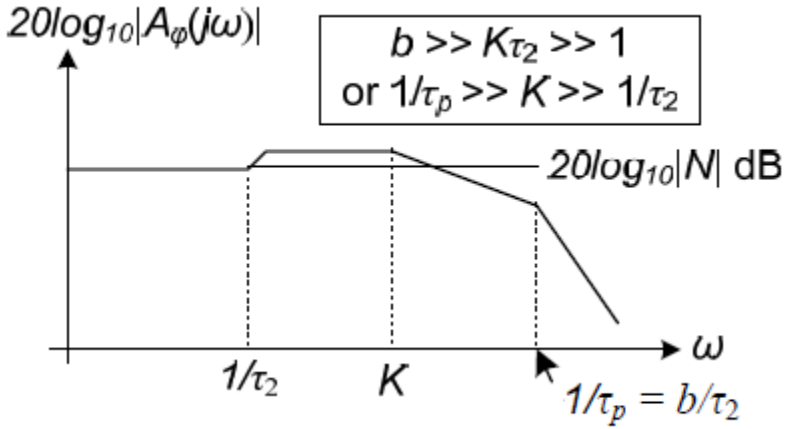
$$T(s) = \frac{K(1+s\tau_2)}{s^2\tau_2(1+s\tau_p)},$$

$$K = \left(\frac{b-1}{b}\right)\frac{I_{CP}K_{VCO}R}{2\pi N},$$

$$PM \approx \tan^{-1}(K\tau_2) - \tan^{-1}(K\tau_2/b)$$

- **Large "b" gives better PM: typically, b > 16 is used**

Closed Loop Response (1)

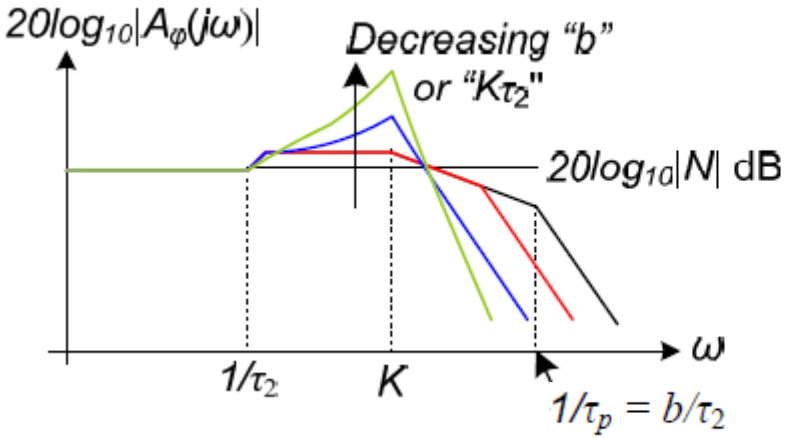


$$A_\phi(s) = \frac{N(1+s\tau_2)}{1+s\tau_2+s^2\tau_2/K+s^3\tau_2\tau_p/K},$$

$$\text{where } K = \left(\frac{b-1}{b}\right)\frac{I_{CP}K_{VCO}R}{2\pi N},$$

- **A pole-zero doublet exists at the zero location.**
  - Magnitude response of $A_\Phi(s)$ rises at the doublet.
  - Doublet slows down settling response.

Closed Loop Response (2)



$$A_\phi(s) = \frac{N(1+s\tau_2)}{1+s\tau_2+s^2\tau_2/K+s^3\tau_2\tau_p/K},$$

$$\text{where } K = \left(\frac{b-1}{b}\right)\frac{I_{CP}K_{VCO}R}{2\pi N},$$

- **A pole-zero doublet exists at the zero location.**
  - Magnitude response of $A_\Phi(s)$ rises at the doublet.
  - Doublet slows down settling response.
- **Decreasing "b" or "Kτ_2" increases peaking.**
  - Frequency where peaking occurs is not necessarily K.

Simple Design Procedure

- **Given quantities**
  - F_{ref}, N, K_{VCO}(Hz/V), minimum phase margin, $PM_{required}$
  - Required closed loop bandwidth, f_{BW}(Hz)
- **Calculate b**

$$PM_{required} \leq \tan^{-1}\left(\frac{\sqrt{b}-1/\sqrt{b}}{2}\right)$$

- **Calculate I_{CP}, R, C_2 such that**

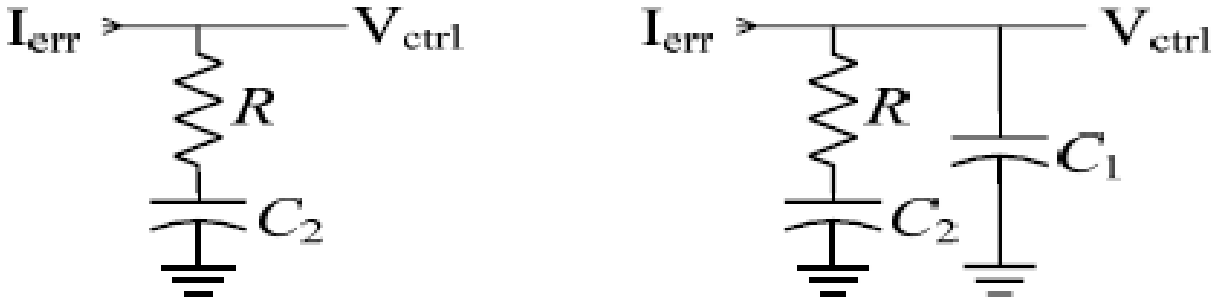
$$I_{CP}R = \left(\frac{b}{b-1}\right)\frac{2\pi N f_{BW}}{K_{VCO}} \text{ and } RC_2 = \frac{\sqrt{b}}{2\pi f_{BW}}$$

- **Calculate C_1 such that** $C_1 = \frac{C_2}{b-1}$
- **Note: Multiple choices for (I_{CP}, R, C_2) are possible**
  - Thermal noise, power considerations will help in making the choice

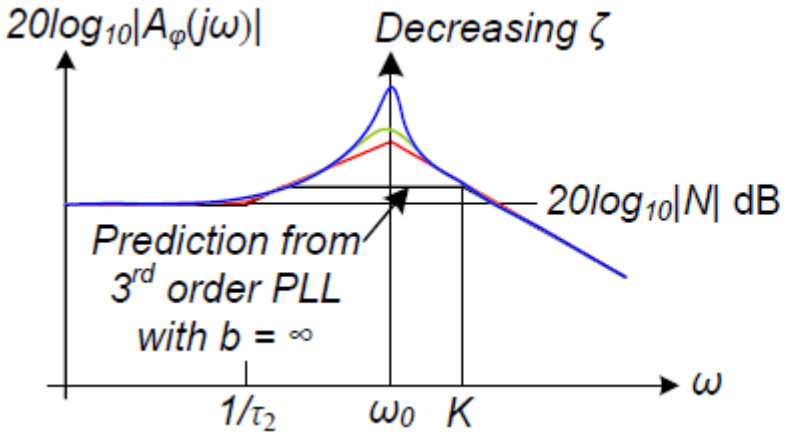
Design Issues

- **Sophisticated filter design is possible.**
  - Implementation concerns
  - Not much room to play with
- **Several tools available**
  - Mainly choose the components of a chosen filter structure,
  - Good example: Michael Perrott's PLL Design tool
- **The challenging problems in PLL design are usually elsewhere**
  - Dealing with component variability
  - How much phase margin is enough phase margin ?

2nd Odrer vs. 3rd Order Type II CP PLL



- **2nd order PLL is a special case of the 3rd order PLL.**
  - b = infinity
- **However, analysis has been application specific**
  - Digital, wired communication analyses focus on 2nd order PLL

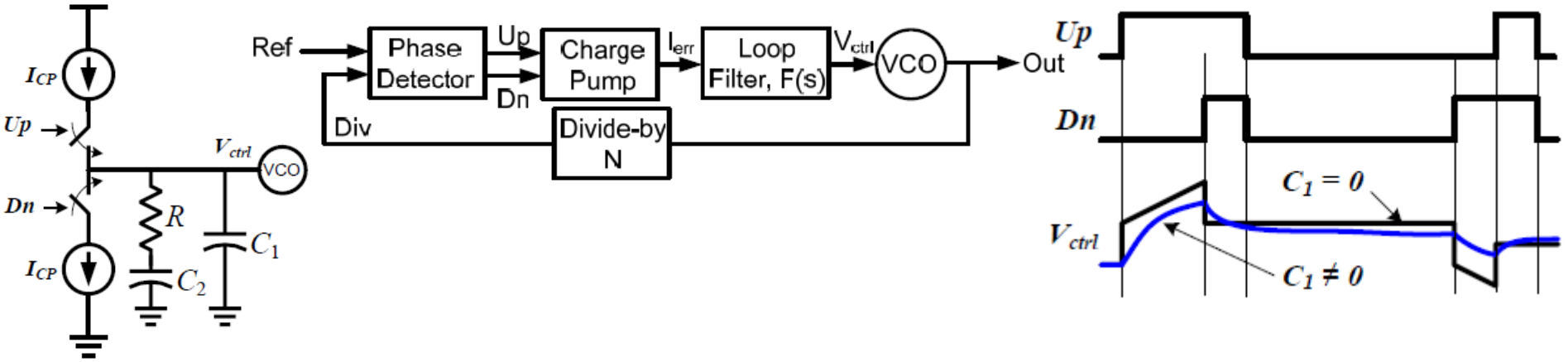


$$PM = \tan^{-1}\left(K\tau_2\right)$$

$$A_\phi(s) = \frac{N\left(1+2\zeta s/\omega_0\right)}{1+2\zeta s/\omega_0+s^2/\omega_0^2}$$

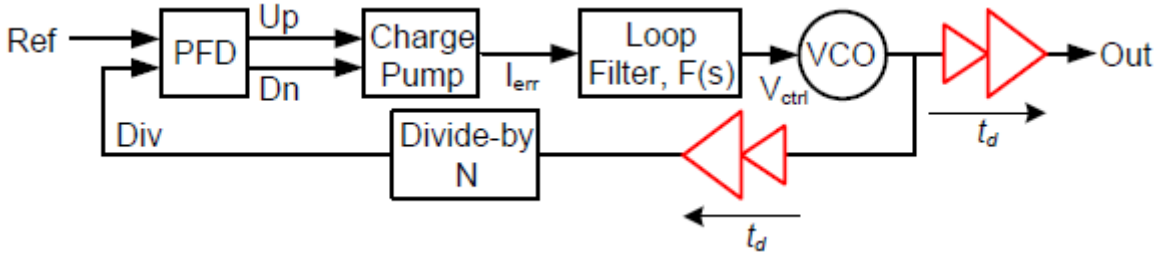
where $\omega_0 = \sqrt{\frac{1}{2\pi N}\frac{I_{CP}K_{VCO}}{C_2}}$, $\zeta = \frac{1}{2}\omega_0 R C_2$

Control Voltage Ripple



- **The "ripple" on V_{ctrl} causes periodic jumps in VCO phase**
  - Causes spikes in power spectral density of VCO output
  - Bad for both wireless and wire-line communications
- **3rd pole attenuates the ripple and its effects**
  - Small "b" is desired for more attenuation
  - Recall: small "b" reduces phase margin
- **Often choose K to be the geometric mean of $1/\tau_2$, and $1/\tau_p$**
  - Used this in the design procedure mentioned earlier

Effect of Feedback Delay in PLLs

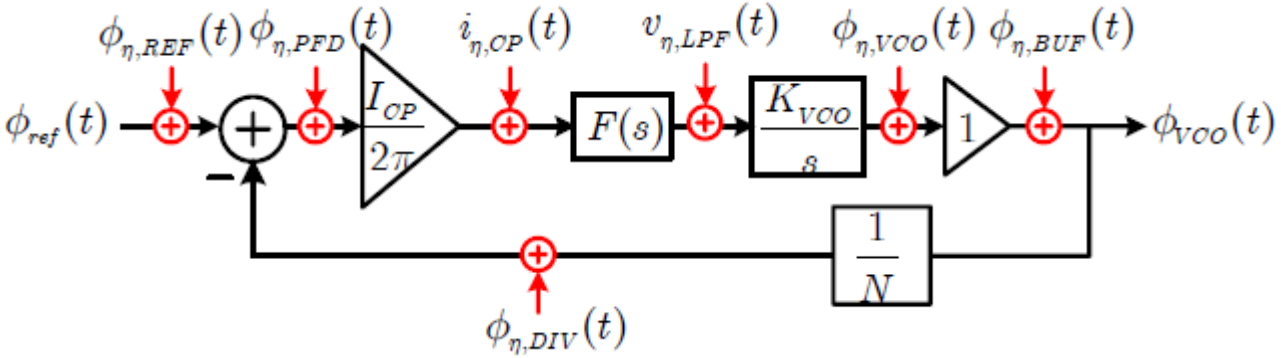


- **Non-zero delay in the feedback is common in PLLs.**
  - E.g., clock distribution in chip-to-chip I/O
- **Feedback delay degrades the phase margin of the PLL.**
- **Can be shown that**

$$PM \approx PM_{t_d=0} - f_{BW} t_d \times 360^\circ$$

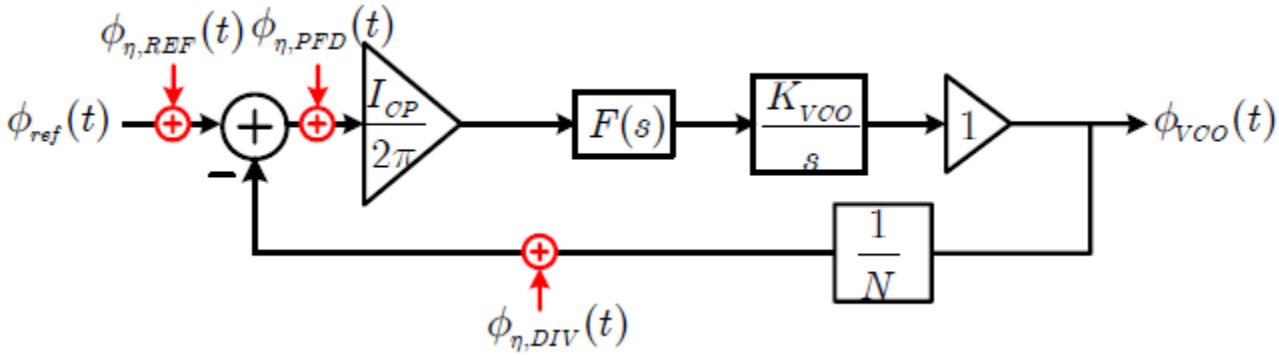
  - Assuming $K \approx f_{BW}$

Noise in the PLL



- **Many sources of noise**
  - Charge pump, loop filter, reference, divider, VCO, buffers, PFD
- **Both random and deterministic sources of "noise"**
- **Linear model can be used because noise terms are small.**

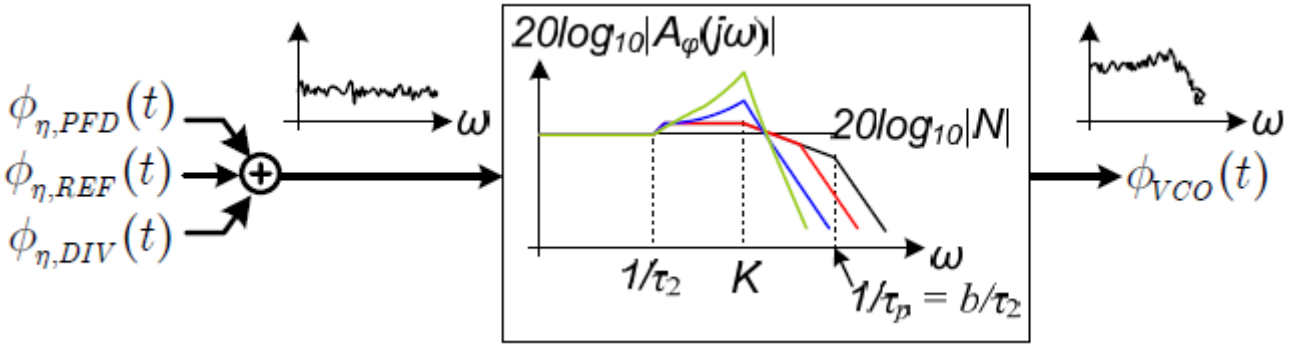
Ref./Divider/PFD Noise Contribution (1)



- **Reference, divider, and PFD noise are combined into a single equivalent phase error term, $\Phi_{\eta,ref}(t)$.**
- **This noise is effectively low-pass filtered by the PLL.**

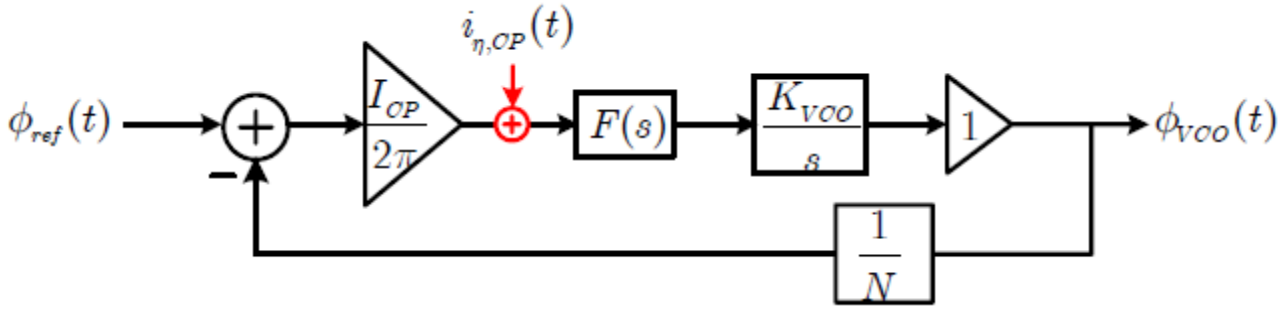
$$\frac{\phi_{VCO}(s)}{\phi_{\eta,ref}(s)} = \frac{NT(s)}{1+T(s)}, \quad T(s) \text{ is open-loop transfer}$$
$$= A_{\phi}(s), \text{ "PLL closed loop transfer function".}$$

Ref./Divider/PFD Noise Contribution (2)



- **Circuit noise**
  - Usually white, possibly some flicker noise
  - A small “N”, small “K”, and a small “b” are advisable.
- **Noise in the reference itself**
  - Considerable particularly in clock recovery loops
  - “Peaking” in $|A_\Phi(j\omega)|$ is bad.
    - A small “Kτ_2” and a large “b” are advisable.

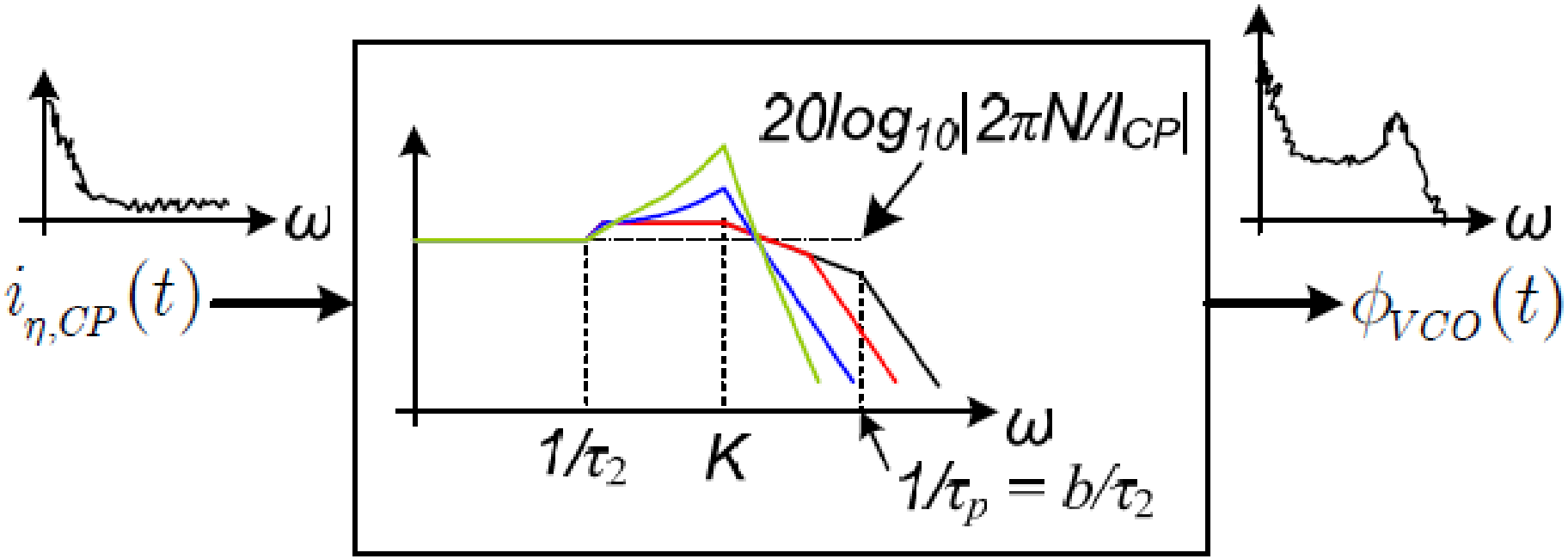
Charge Pump Noise Contribution (1)



- **Charge pump dumps a net noise charge into the loop filter**
  - Both white noise and flicker noise components are present
- **Charge pump noise charge can be translated into an equivalent phase error at the reference**
  - Charge pump noise is low-pass filtered by the PLL.

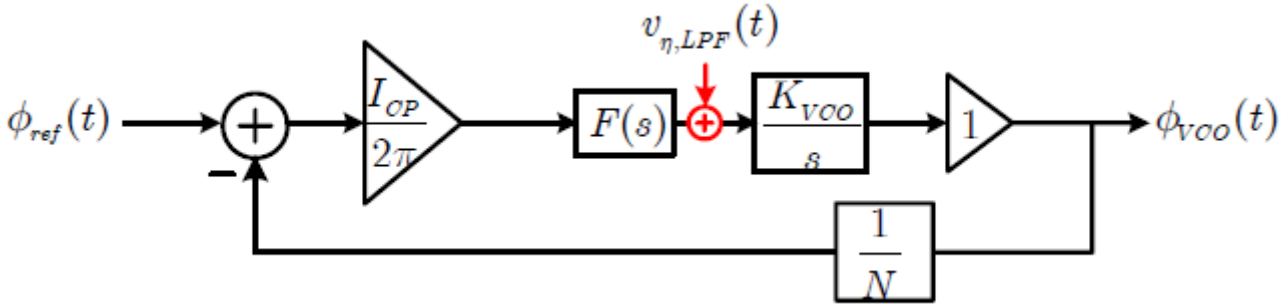
$$\frac{\phi_{VCO}(s)}{i_{\eta,CP}(s)} = \left(\frac{2\pi}{I_{CP}}\right)\frac{NT(s)}{1+T(s)} = \frac{2\pi}{I_{CP}}A_{\phi}(s)$$

Charge Pump Noise Contribution (2)



- **Charge pump noise is low-pass filtered by the PLL.**
  - Again, a small “N”, small “K”, and a small “b” are advisable.

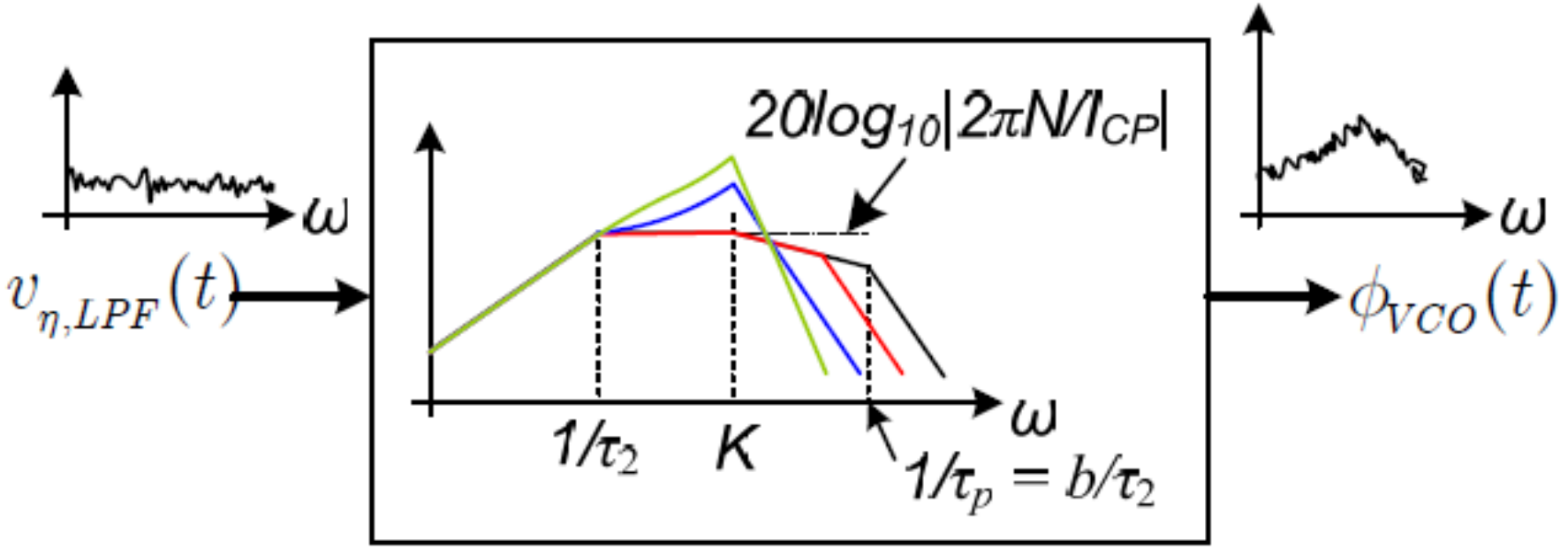
Loop Filter Noise Contribution (1)



- **Noise comes from resistors and transistors in active filters.**
- **Noise can be referred to the output of the filter.**
  - E.g., $v_{\eta,LPF} = v_{\eta}$ for the 2nd order loop filter
  - E.g., for the 3rd order loop filter, $$v_{\eta,LPF}(s) = \left(\frac{b-1}{b}\right)\left(\frac{1}{1+s\tau_p}\right)v_n(s)$$
- **Refer the voltage noise to the filter output**
  - It is band pass filtered by the PLL.

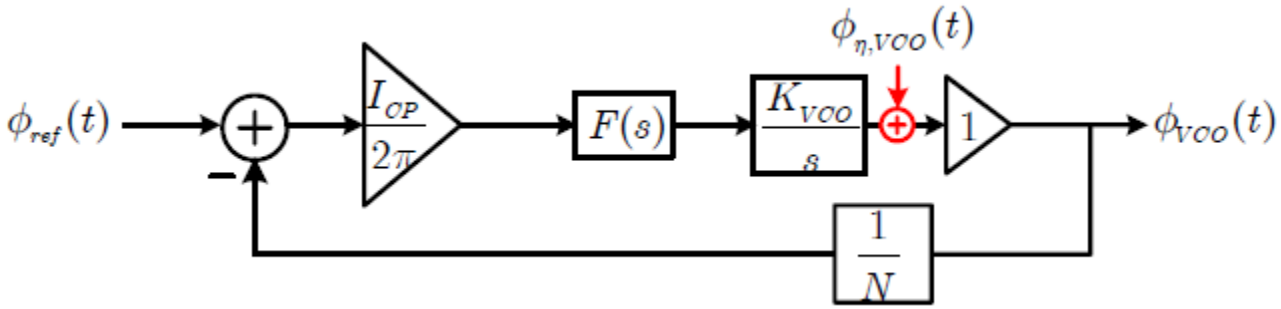
$$\frac{\phi_{VCO}(s)}{v_{\eta,LPF}(s)} = \frac{2\pi}{I_{CP}}\left(\frac{1}{F(s)}\right)A_{\phi}(s)$$

Loop Filter Noise Contribution (2)

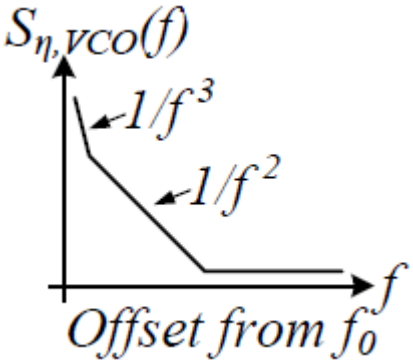


- **For the 3rd order LPF,**
  - Again, a small “N”, a small “K”, and a small “b” are advisable

Oscillator Noise Contribution (1)



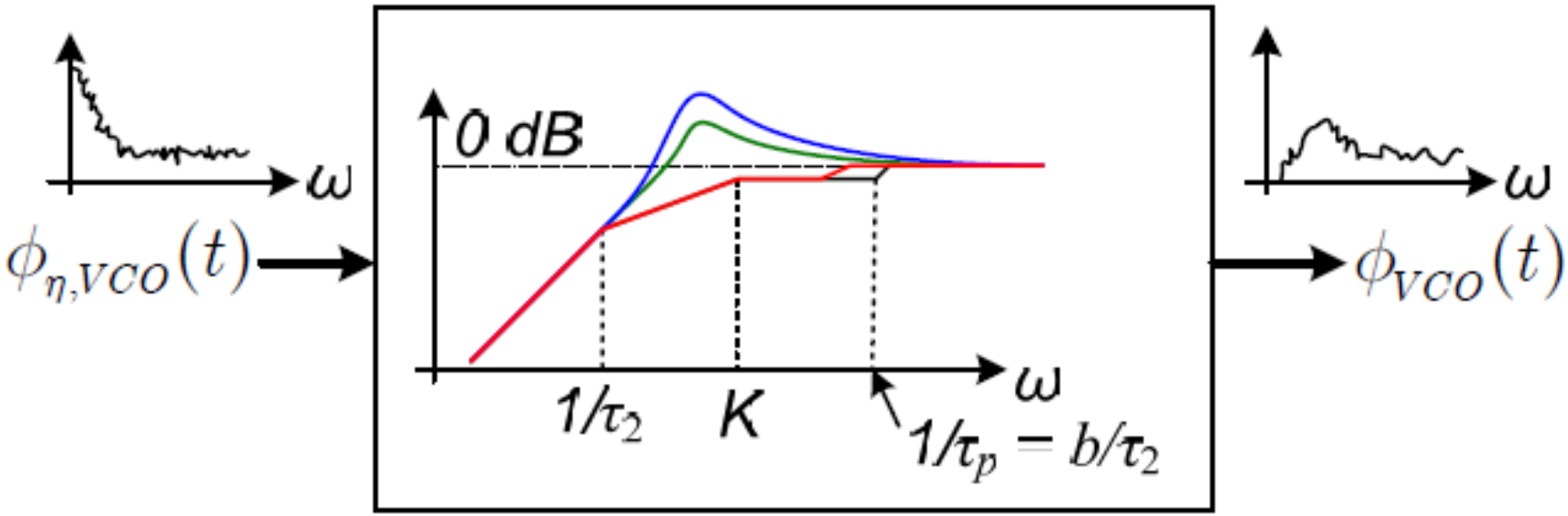
- **Free running oscillators usually accumulate phase noise.**
  - Has a complicated power spectral density.
  - Usually, the $1/f^2$ terms dominates.



- **Noise on the VCO supply also causes phase noise.**
- **VCO noise gets high-pass filtered by the PLL.**

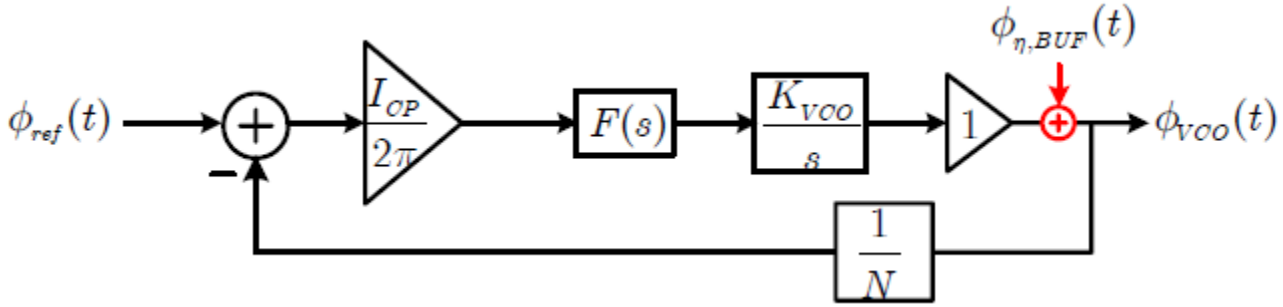
$$\frac{\phi_{VCO}(s)}{\phi_{\eta,VCO}(s)} = \frac{1}{1+T(s)}$$

Oscillator Noise Contribution (2)



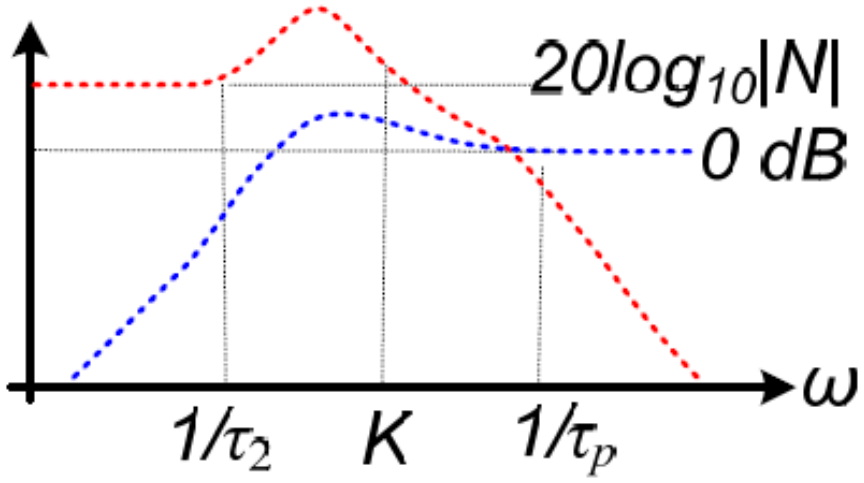
- **VCO noise is suppressed within the PLL bandwidth.**
- **A wide PLL bandwidth is desired to suppress VCO noise.**
  - Typical target for PLLs in chip-to-chip communications

Buffer Noise Contribution



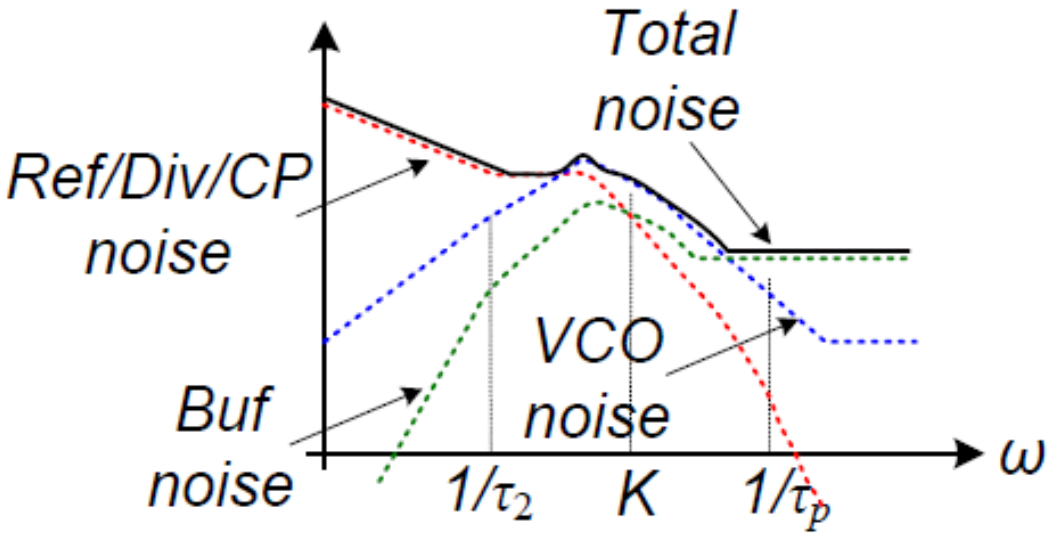
- Phase noise induced by both thermal noise and supply noise.
- Thermal noise
  - Induces timing jitter.
- Supply noise
  - Changes the buffer delay.
- Buffer noise sees the same PLL response as VCO noise.

Overall PLL Noise



Noise Transfer Functions

PLL Noise Contributions

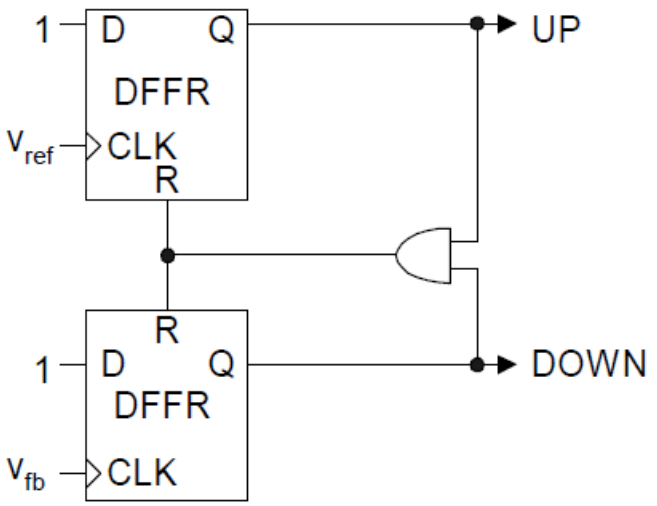


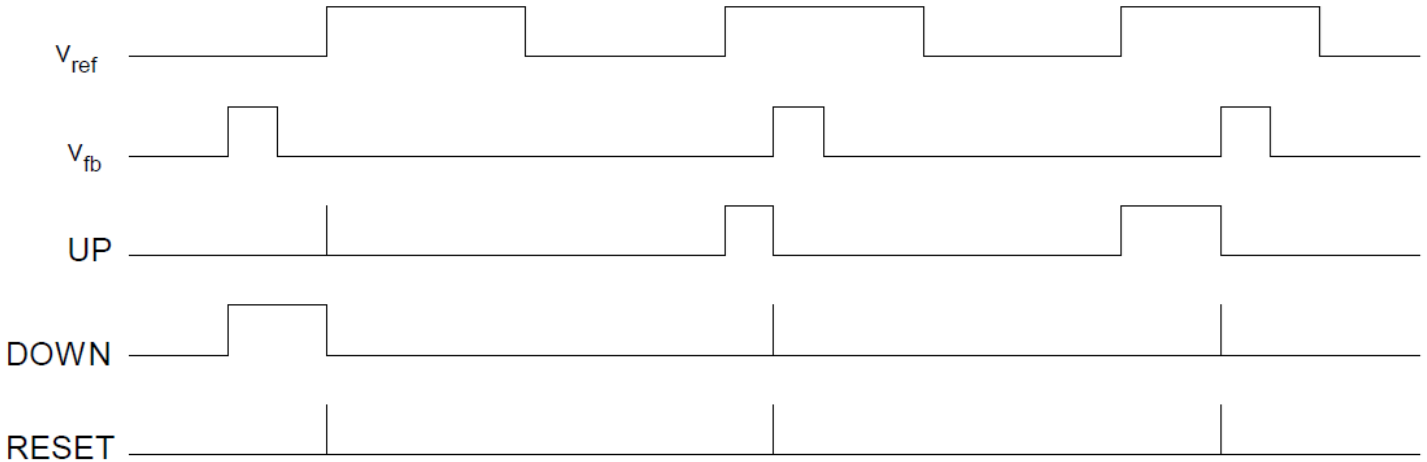
Outline

- PLL
  - Components
  - Models
  - Noise Transfer Functions
  - Circuits
- **DLL**
  - Types
  - Models
  - Noise Transfer Functions
- **PLL/DLL comparison**

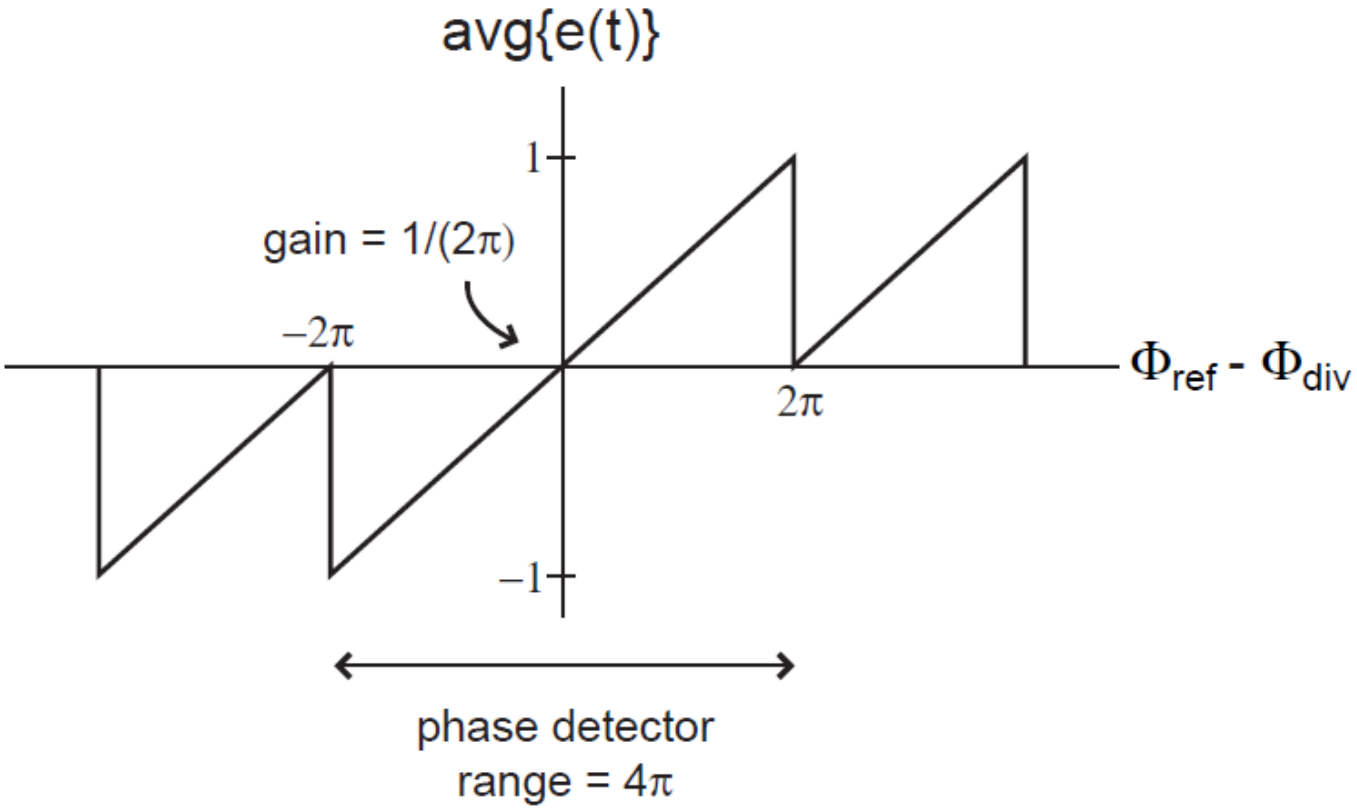
Phase Frequency Detector (PFD)

- **Phase Frequency Detector allows for wide frequency locking range, potentially entire VCO tuning range.**
- **3-stage operation with UP and DOWN outputs**
- **Edge-triggered results in duty cycle insensitivity**





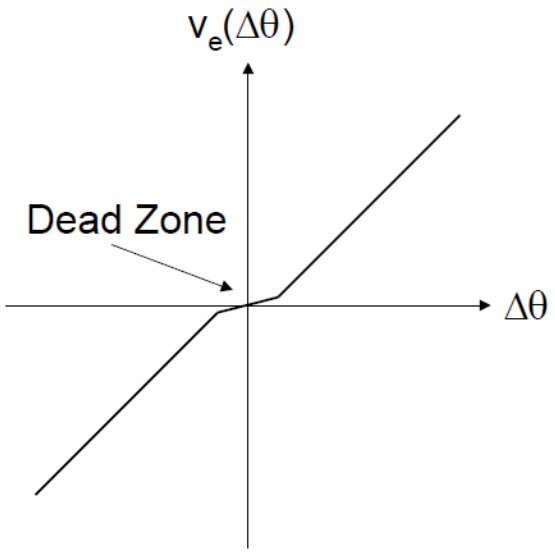
PFD Transfer Characteristic

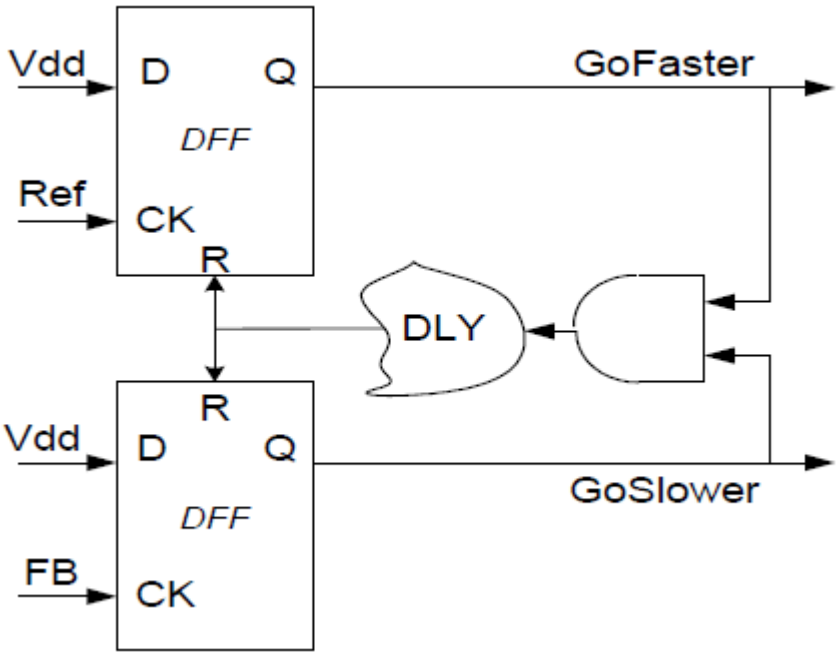


- **Constant slope and polarity asymmetry about zero phase allows for wide frequency range operation.**

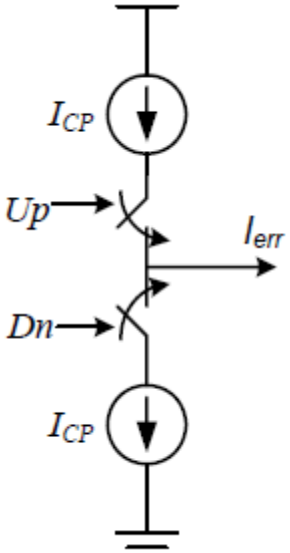
PFD Deadzone

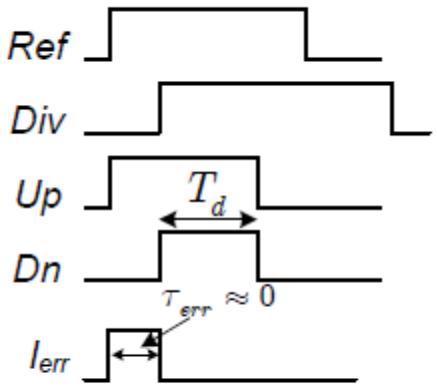
- If phase error is small, then short output pulses are produced by PFD.
- Cannot effectively propagate these pulses to switch charge pump.
- Results in phase detector “dead zone” which causes low loop gain and increased jitter.
- Solution is to add delay in PFD reset path to force a minimum UP and DOWN pulse length.

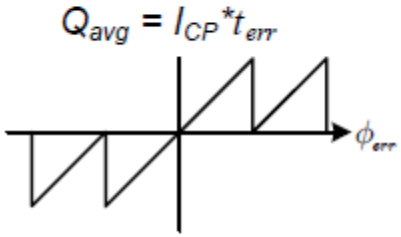




Charge Pump



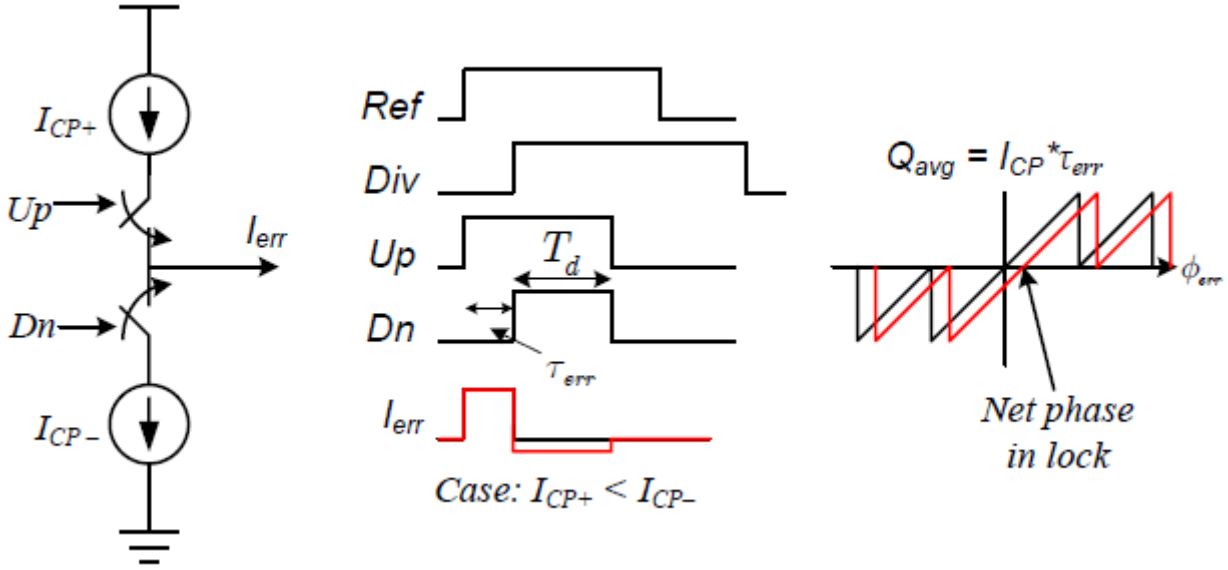




PFD-CP Gain: $\left(\frac{1}{2\pi}\right) I_{CP}$

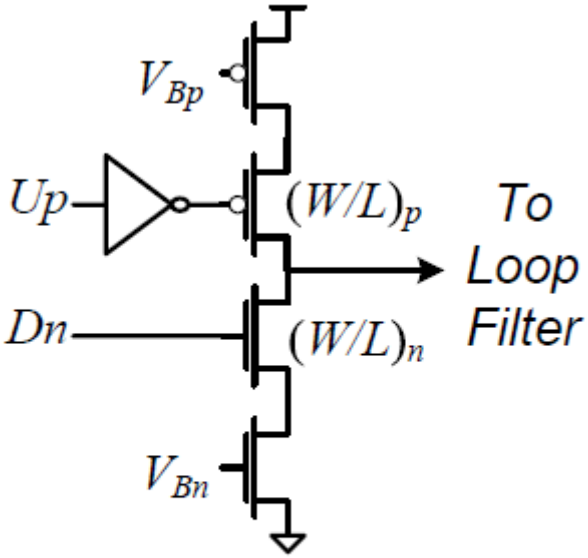
- **Design Goals**
  - Low noise
  - Low power consumption
  - Low mismatch between Up and Down currents
  - Low switch charge injection
  - High speed of switching for wide bandwidth PLLs

Mismatch and Current Injection Problem



- **Mismatch and charge injection add a net charge (+ve or –ve) into the loop every period.**
- **The PLL locks with a non-zero phase offset between the reference and the divider output.**
  - Undesirable, mainly in chip-to-chip communications

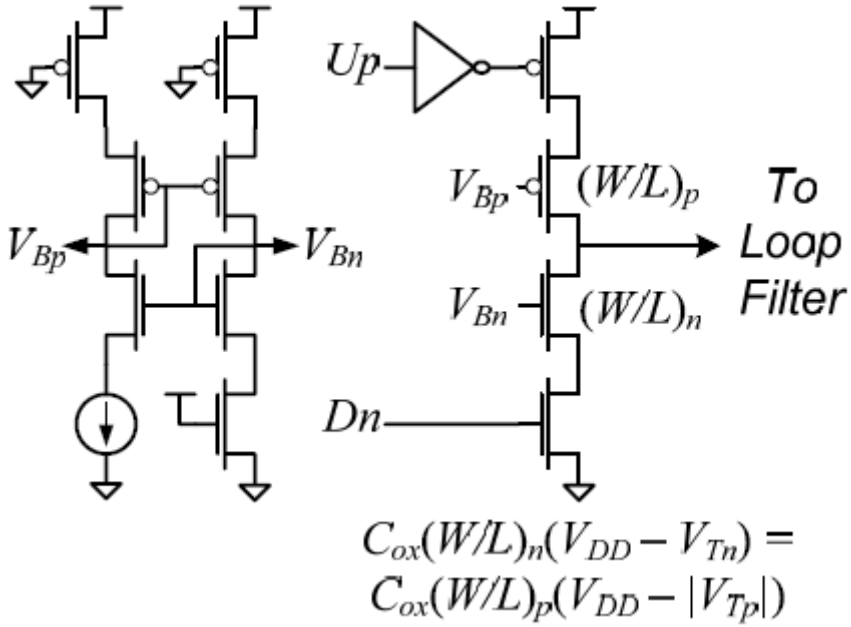
Reducing Charge Injection (1)



$$C_{ox}(W/L)_n(V_{DD} - V_{Tn}) = C_{ox}(W/L)_p(V_{DD} - |V_{Tp}|)$$

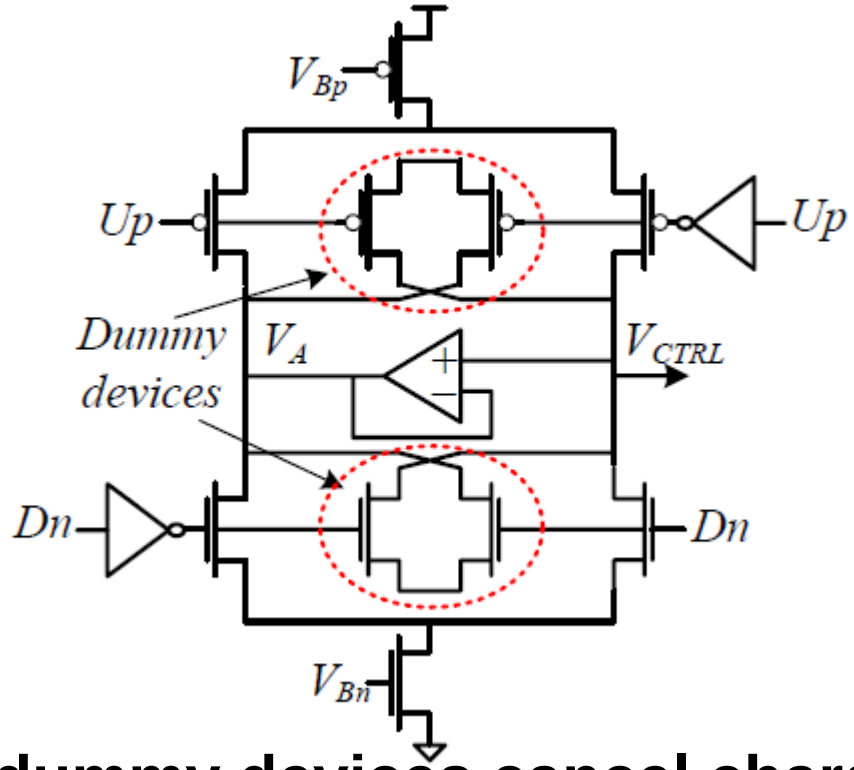
- **Choose the switch sizes to match charge injection from nMOS and pMOS switches.**
  - Not reliable, just an estimate

Reducing Charge Injection (2)



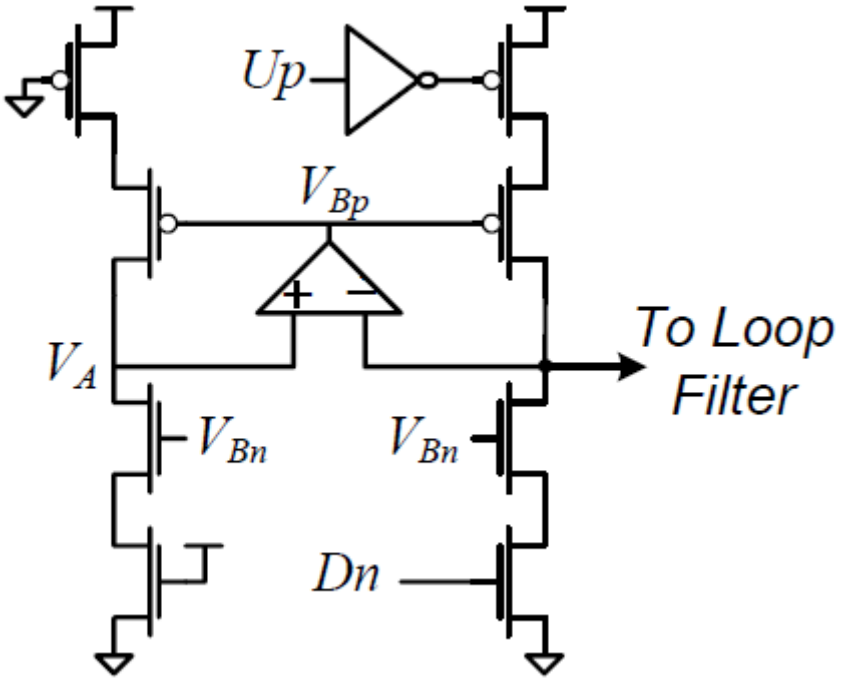
- **Move the switches close to the supply and ground rails.**
  - Channel charge goes into the parasitic cap instead of the loop filter
- **Bias generation is somewhat complicated.**

Reducing Charge Injection (3)



- **Equal sized dummy devices cancel charge injection.**
  - They also cancel clock feed through.
- **The amplifier keeps V_A at the same voltage as V_{CTRL} to cancel charge injection from dummy devices.**

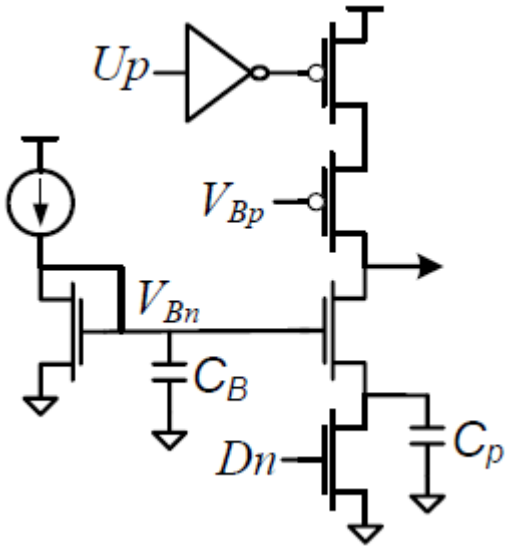
Reducing Charge Pump Mismatch

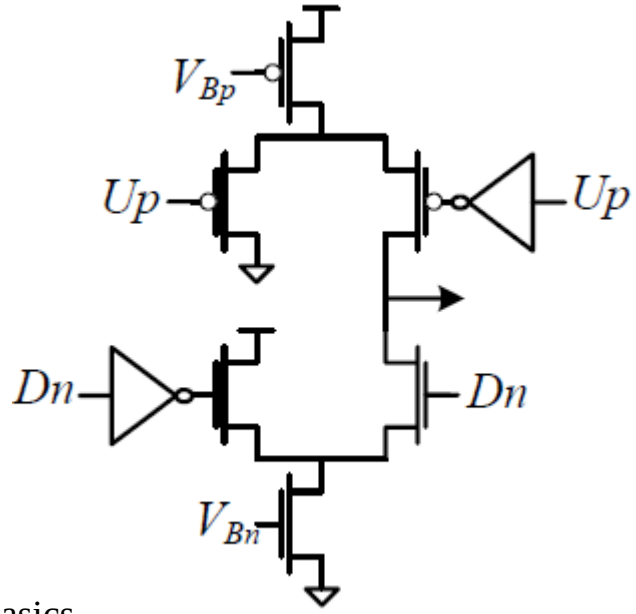


- **Very slow feedback loop to match Up current to Down current**
  - Mitigates current mismatches due to finite R_{out} of current sources.
  - Loop picks appropriate V_{BP} to make V_A match the loop filter voltage.

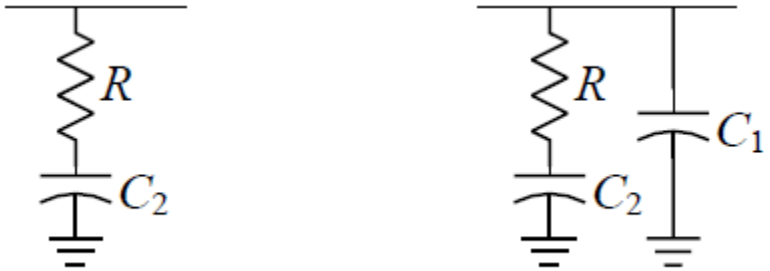
Design of Fast Switching Charge Pumps

- **Turning the current sources off makes switching them back on slow.**
  - This is particularly so when the switches are close to the rails.
  - Use a large capacitor on the bias lines
- **Redirecting the current (instead of switching it off) is much faster.**
  - E.g., differential structure shown below
  - Too much power consumption





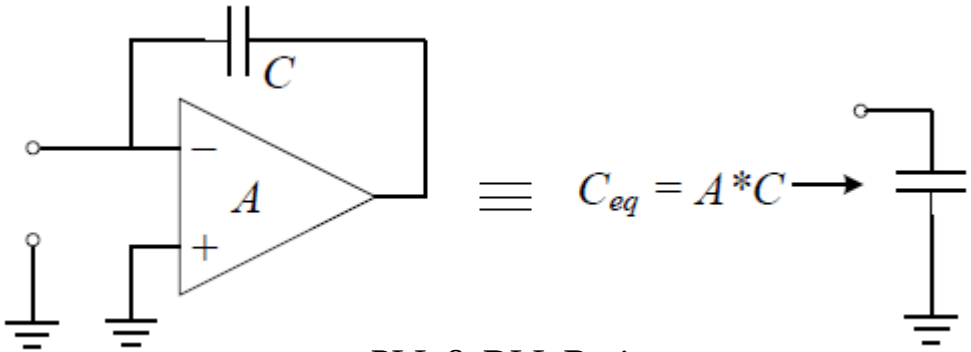
Loop Filter Design



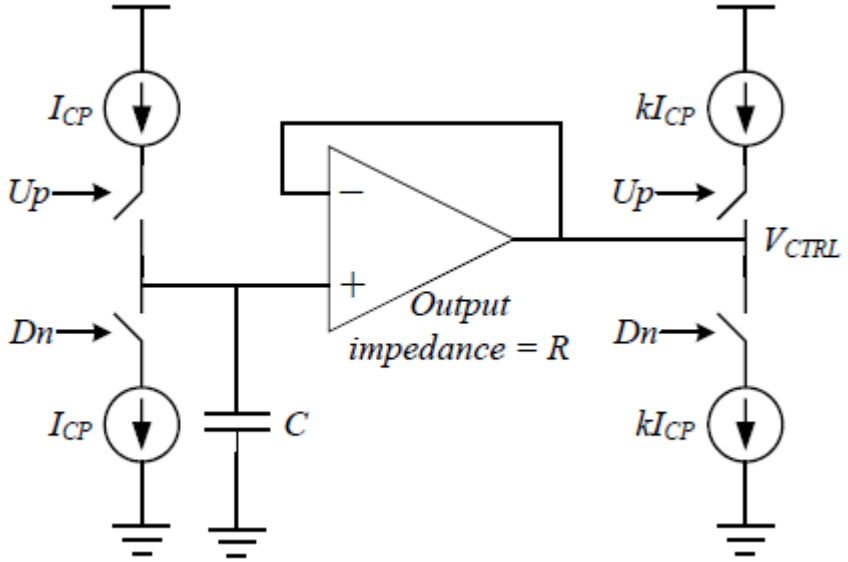
- **Design Goals**
  - Low ripple on the control voltage
  - Low die area
    - Important concern in chip-to-chip communications
  - Adaptability and re-configurability
    - Component variability over process, voltage, and temperature (PVT)

Loop Filter Passive Components

- **Resistors**
  - Poly, n-well, non-silicided diffusion
  - MOS triode resistors
    - Limited operating range
    - Very sensitive to PVT variations
- **Capacitors**
  - Typically MOS capacitors, 5 – 15 fF/μm^2
  - Occupy a large die area
  - Miller multiplication can decrease die area



Active Zero for Reduced Cap Area



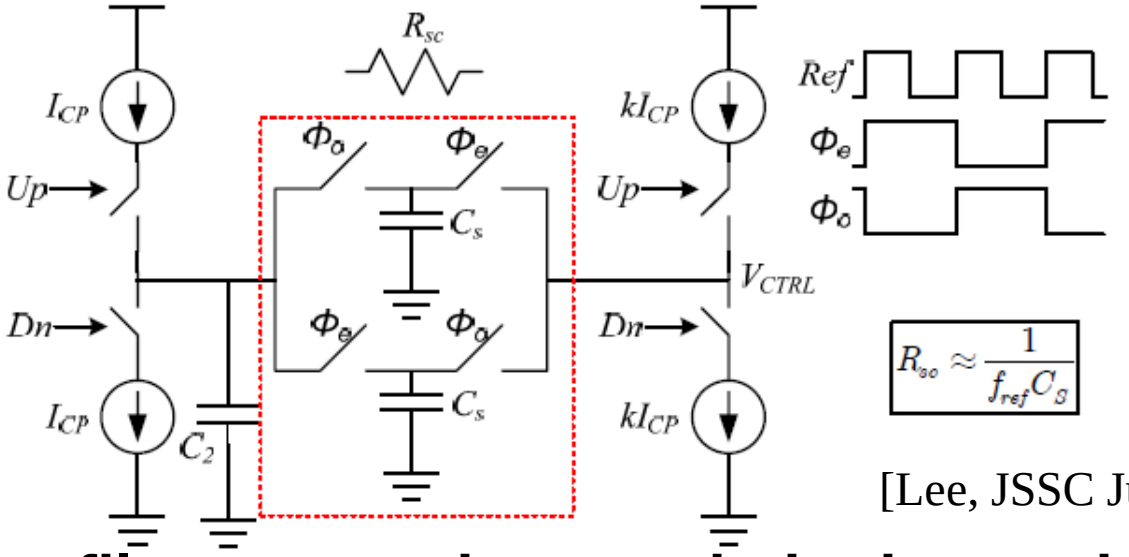
$$V_{CTRL}(s) = \frac{I_{CP}}{sC} + kI_{CP}R$$

$$\Rightarrow F(s) = R_{eq}\frac{\left(1 + sCR_{eq}\right)}{sCR_{eq}},$$

$$R_{eq} = kR$$

- A large cap is required for a low zero frequency.
- A second, scaled, charge pump places the zero at 1/kCR rad/s.
- The resistance, R, is the output impedance of an amplifier.
- Helps in loop BW adaptability.

Switched-Capacitor Loop Filters



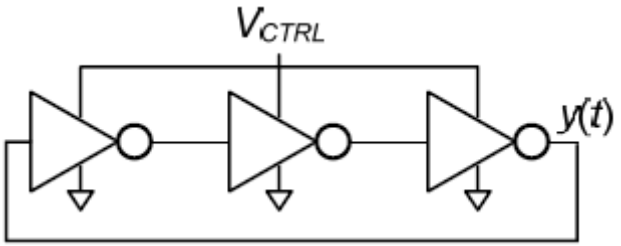
[Lee, JSSC Jun. 2003]

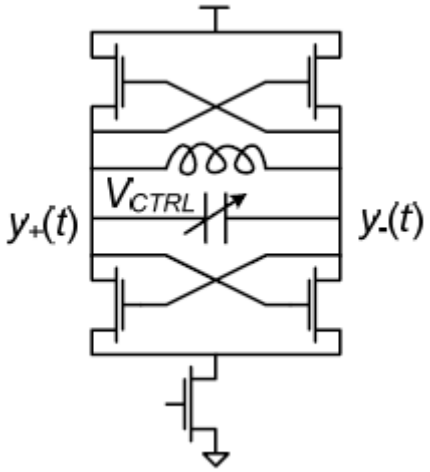
- **Realizes the filter zero using a switched capacitor circuit.**
- **V_{CTRL} is changed only as much as needed**
  - Can reduce ripple significantly.
- **Charge injection and clock feed-through are serious problems.**

$$V_{CTRL}(s) = \frac{I_{CP}}{sC_2} + kI_{CP}R_{sc} \Rightarrow F(s) = R_{eq}\frac{(1+sCR_{eq})}{sCR_{eq}}, R_{eq} = kR_{sc}$$

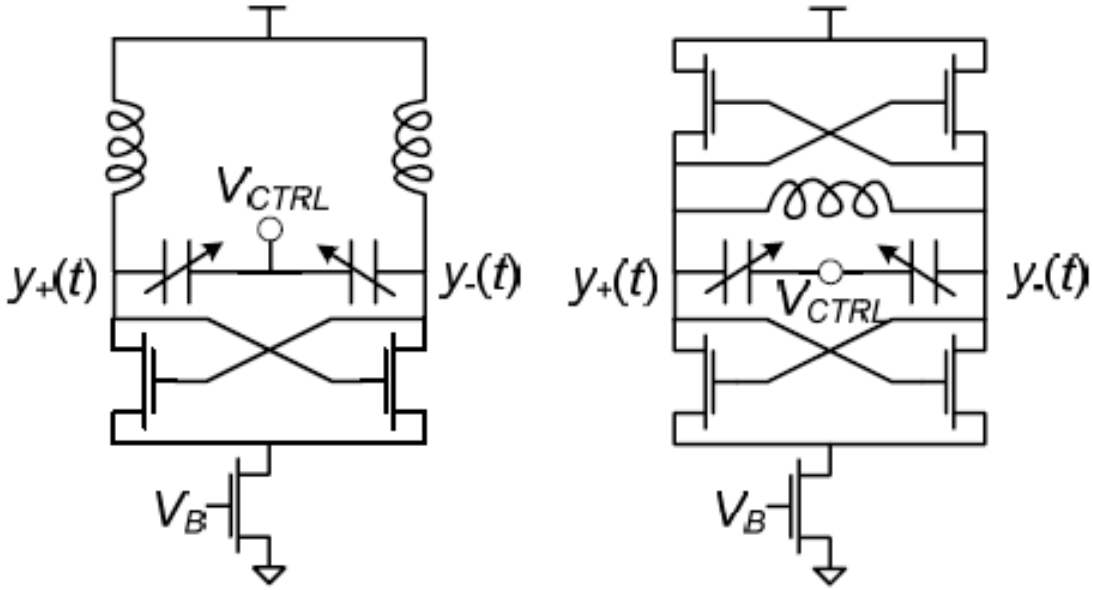
VCO

- **Ring Oscillator**
  - Easy to integrate
  - Wide tuning range (5x)
  - Higher phase noise
- **LC Oscillator**
  - Large area
  - Narrow tuning range (20-30%)
  - Lower phase noise
- **Matters of Interest**
  - Frequency range, K_{VCO}
  - Phase noise and/or jitter
  - Thermal noise and power supply noise
  - Power consumption and die area



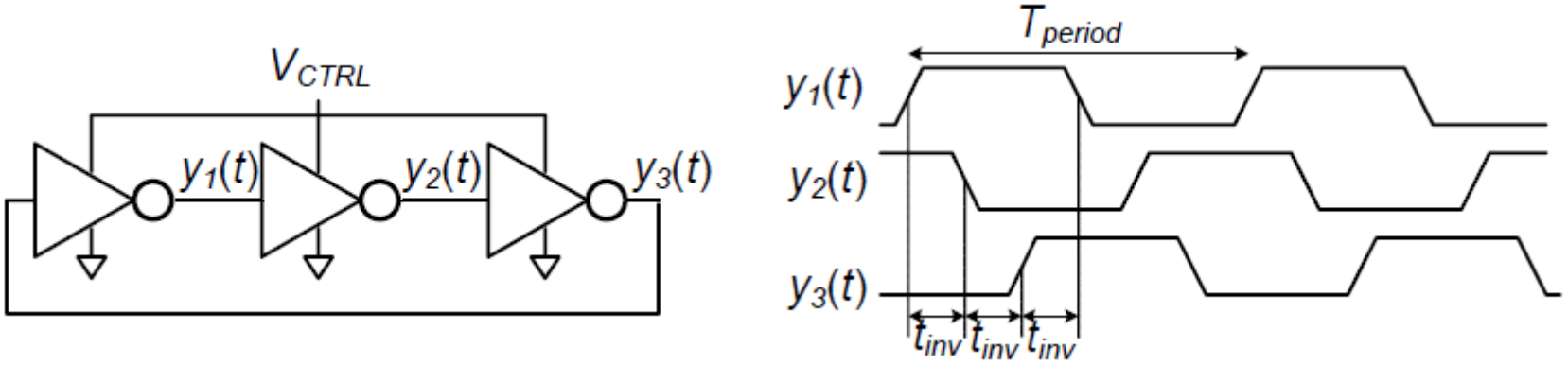


LC VCOs



- **Multiple oscillator structures**
  - Negative-gm, Colpitts, Clapp etc.
- **Frequency control is achieved using varactors.**

Ring Oscillators

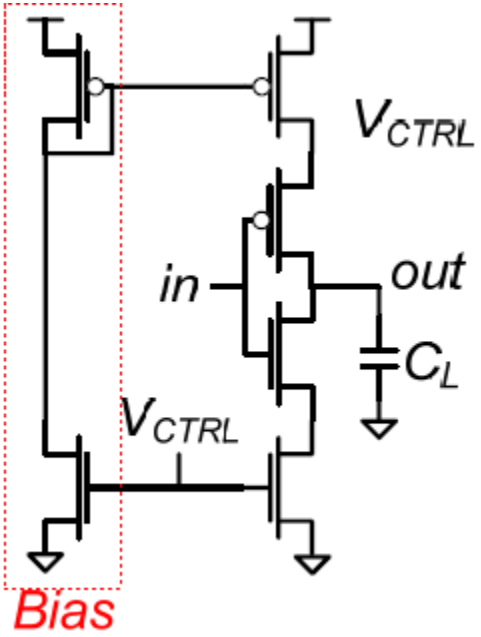


- **N-stage ring oscillator will have a period:**

$$T_{period} = 2N \times t_{inv}$$

- **N should be odd.**
  - Differential inverter stages enable even number of stages.
- **The delay of each inverter stage depends on V_{CTRL}.**

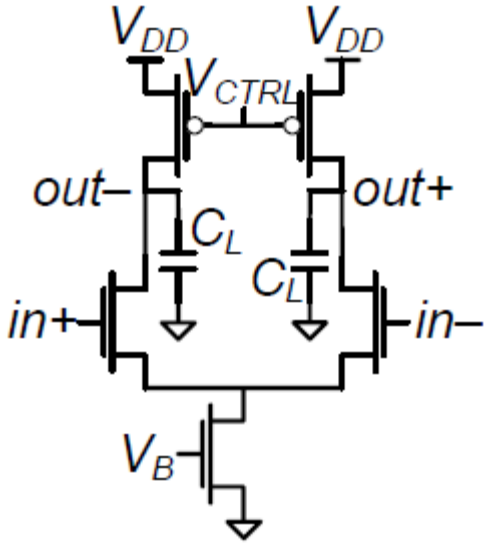
Current Starved Inverter Stage



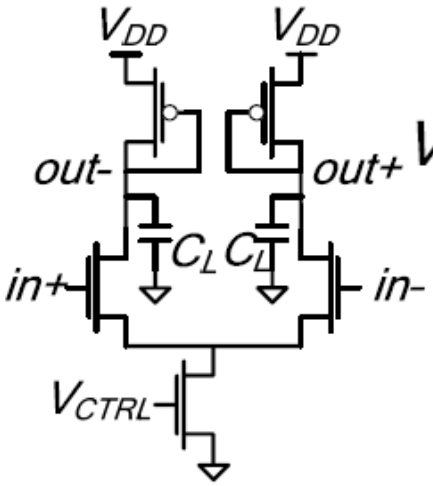
$$t_{inv} \ \alpha \ \frac{V_{DD}}{(V_{CTRL} - V_T)} \ \text{or} \ t_{inv} \ \alpha \ \frac{V_{DD}}{(V_{CTRL} - V_T)^2}$$

- **Very wide tuning range**
- **Terrible supply sensitivity**
  - 1% change in V_{DD} causes a 1% change in t_{inv}.

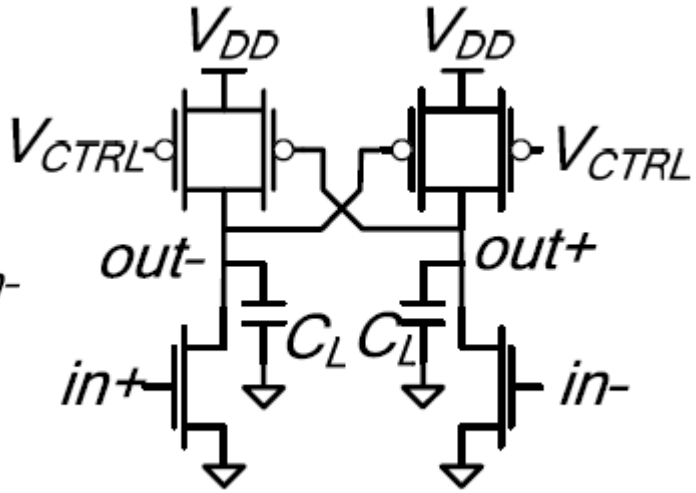
Differential Inverter Stages



$$t_{inv} \alpha \frac{C_L}{(V_{DD} - V_{CTRL} - |V_T|)}$$

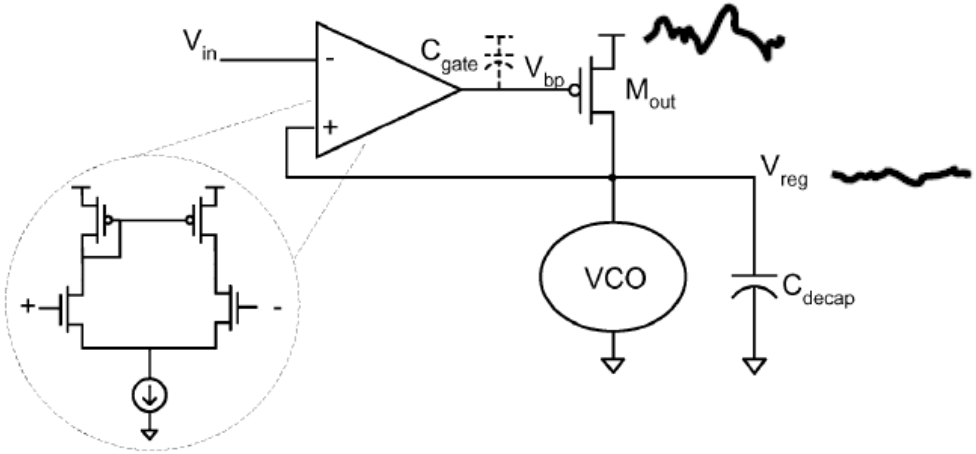


$$t_{inv} \alpha \frac{C_L}{g_{m,p}}$$



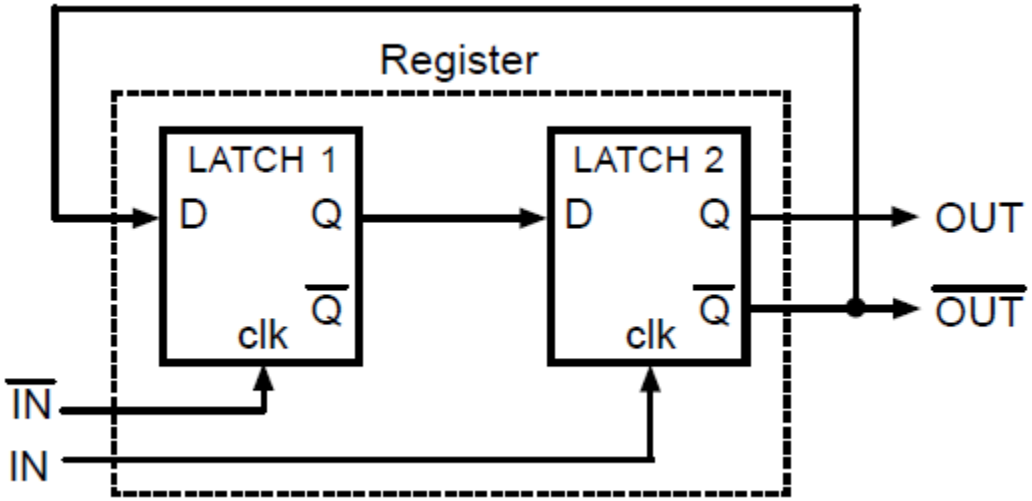
- **Differential nature reduces supply noise only slightly.**
- **Bias current has to be adjusted to maintain swing.**
- **Third circuit is a full swing VCO.**

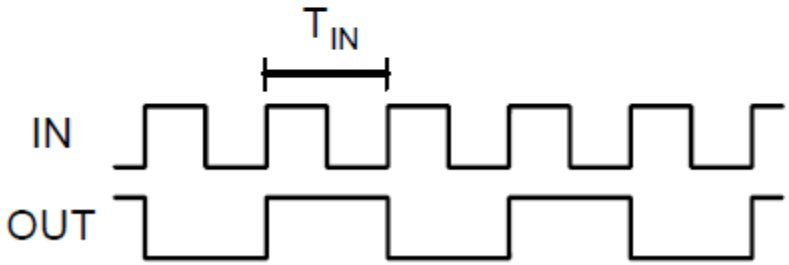
Supply-Regulated VCOs



- **The amplifier adjusts V_{bp}**
  - So, V_{reg} is relatively free of V_{DD} variations
- **Desirable features**
  - Large regulation range i.e. low dropout regulation
  - Low amplifier power consumption
  - Small area
  - Wide bandwidth suppression of V_{DD} variations

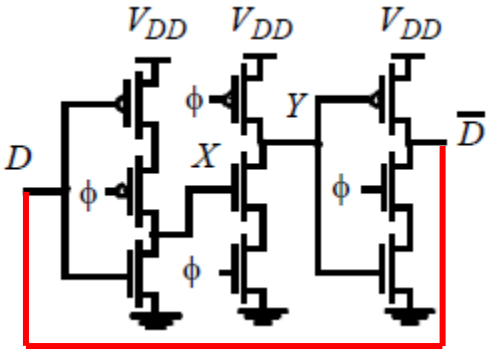
Basic Divide-by-2





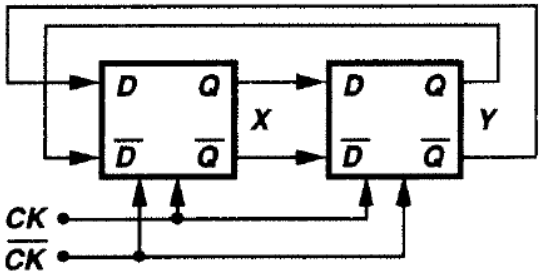
- **Divide-by-2 can be realized by a flip-flip in “negative feedback”.**
- **Divider should operate correctly up to the maximum output clock frequency of interest PLUS some margin.**

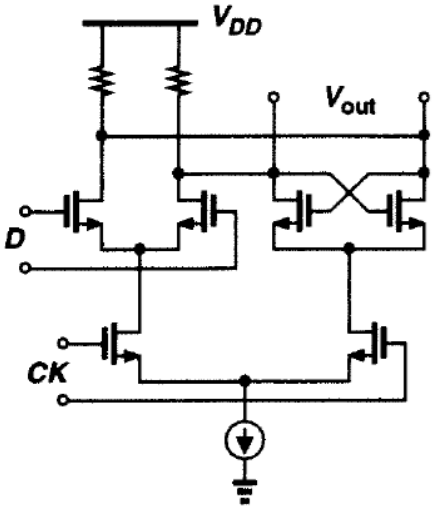
Divide-by-2 with TSPC FF



- **Advantages**
  - Reasonably fast, compact size, and no static power
  - Requires only one phase of the clock.
- **Disadvantages**
  - Signal needs to propagate through three gates per input cycle.
  - Need full swing CMOS inputs.
  - Dynamic flip-flop may have issues at very low frequency operation (test mode) depending on process leakage.

Divide-by-2 with CML FF

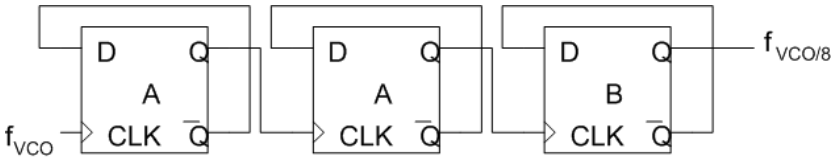




- **Advantages**
  - Signal only propagates through two CML gates per input cycle.
  - Accepts CML input levels.
- **Disadvantages**
  - Larger size and dissipates static power.
  - Requires differential input.
  - Additional speedup (>50%) can be achieved with shunt peaking inductors.

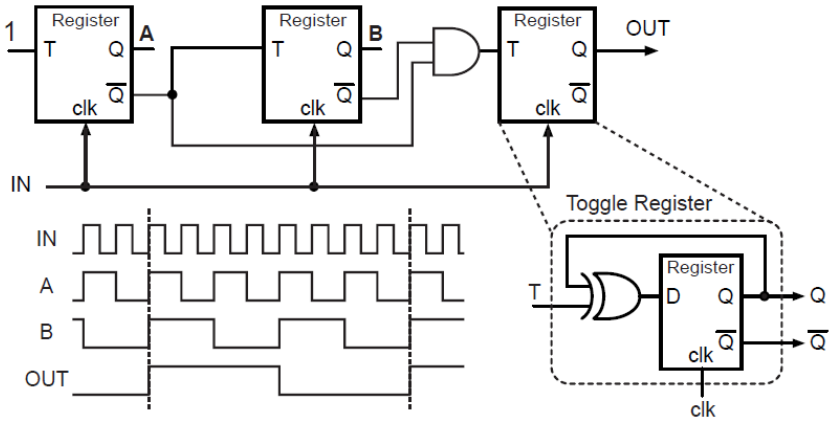
Binary Dividers

Asynchronous Divider



- **Advantages**
  - Each stage runs at lower frequency, resulting in reduced power
  - Reduced high frequency clock loading
- **Disadvantage**
  - Jitter accumulation

Synchronous Divider



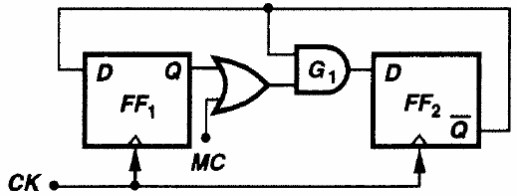
- **Advantage**
  - Reduced jitter

Disadvantage

- All flip-flops work at maximum frequency, resulting in high power
- Large loading on high frequency clock

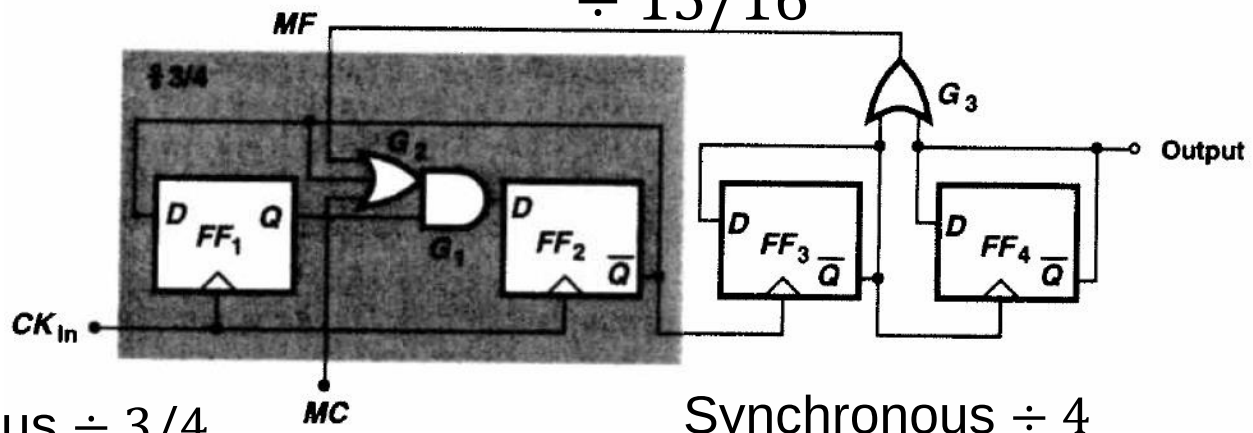
Dual Modulus Prescalers

$\div 2/3$



MC = 0 → ÷ 3
MC = 1 → ÷ 2

$\div 15/16$



Synchronous ÷ 3/4

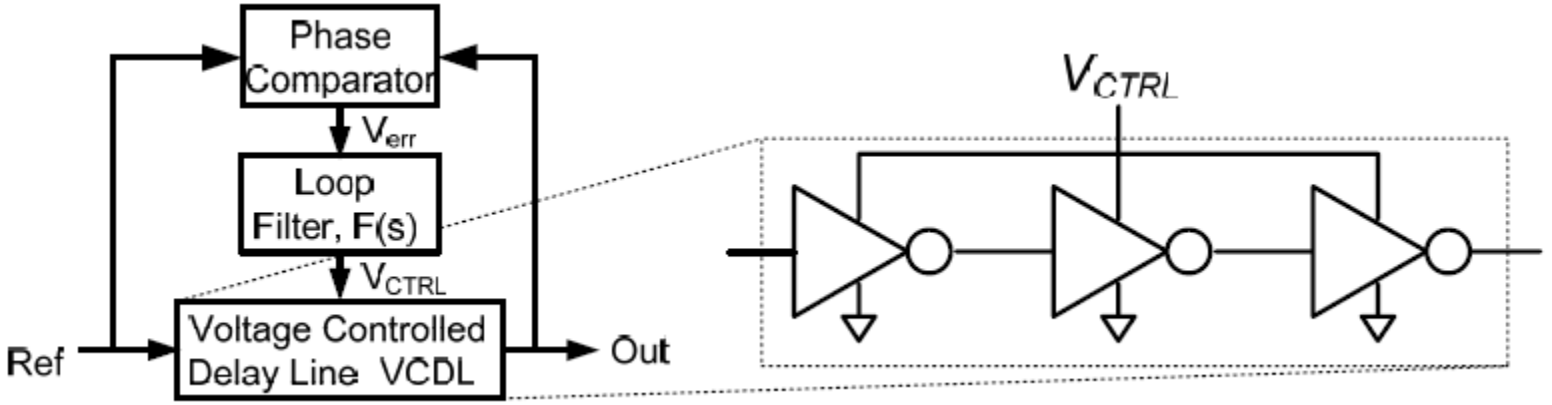
Synchronous ÷ 4

- **For /15, first prescaler circuit divides by 3 once and 4 three times during the 15 cycles.**

Outline

- PLL
  - Components
  - Models
  - Noise Transfer Functions
  - Circuits
- **DLL**
  - Types
  - Models
  - Noise Transfer Functions
- **PLL/DLL comparison**

DLL VCDL



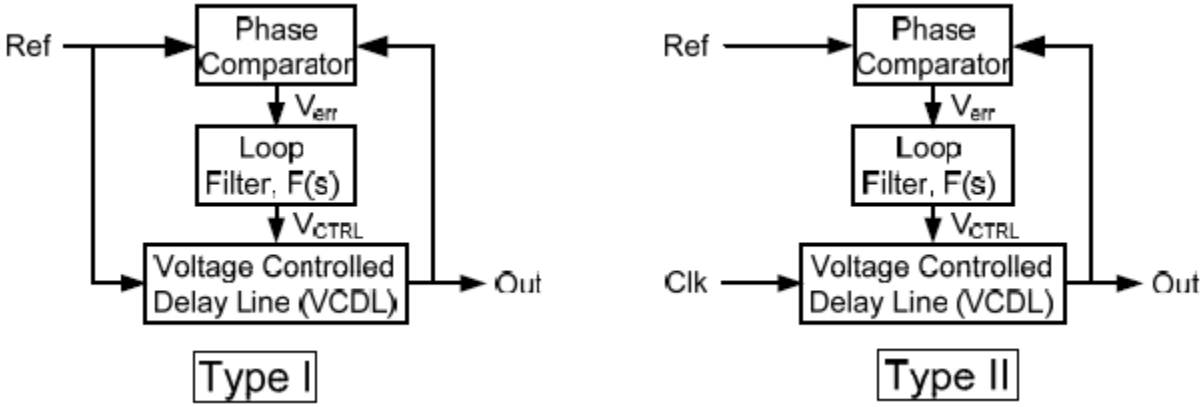
- **Just a chain of inverter stages (similar to ring oscillators)**
- **Delay is controlled by V_{CTRL}.**

$$\Delta T(t) = K_{VCDL} V_{CTRL}(t) \text{ or } \phi(t) = K_{DL} V_{CTRL}(t)$$

seconds/volts

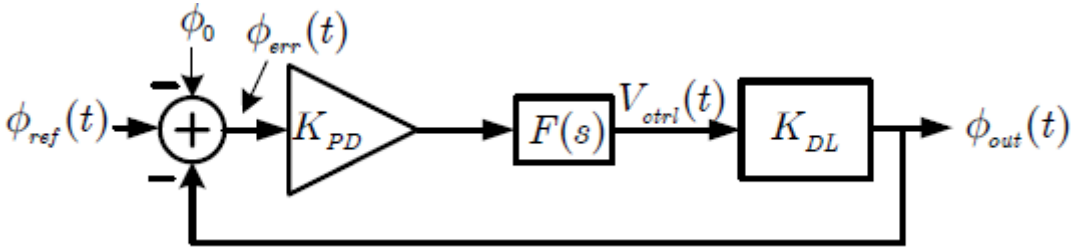
radians/volts, $K_{DL} = \frac{2\pi}{T_{ref}} K_{VCDL}$

Delay Locked Loop –Types I and II



- **Closed loop response is different between Type I and II DLLs**
- **Type I**
  - More popular type of DLL
  - Used in multi-phase clock generation
  - High frequency reference noise shows up in the output
- **Type II**
  - Used in clock and data recovery (CDR) loops

Type II Continuous-Time Model



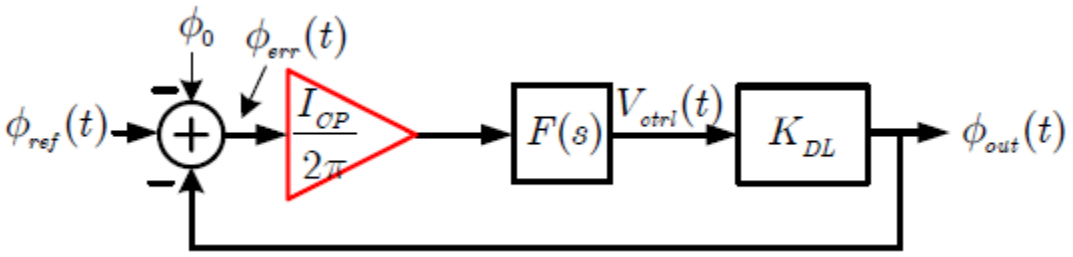
- **Just as in the PLL, the continuous time model is accurate for bandwidths << $f_{ref}/10$.**
- **Negative feedback loop tries to force $\Phi_{err}(t) = \Phi_0$**
- **Loop transfer function is**

$$T(s) = K_{PD} K_{DL} F(s)$$

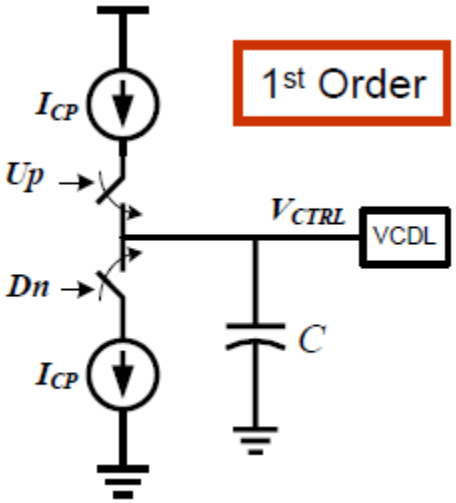
- **Closed loop response is**

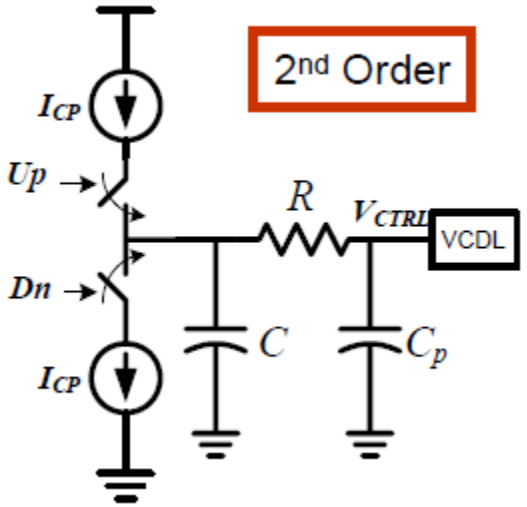
$$A_\phi(s) = \frac{K_{PD} K_{DL} F(s)}{1 + K_{PD} K_{DL} F(s)}$$

Type II Charge Pump DLL

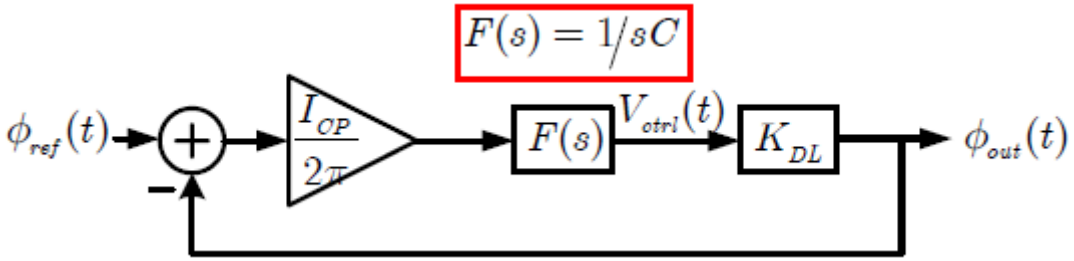


- **F(s) is typically just an integrator.**
- **Sometimes, an additional pole is added**





1st Order Type II CP DLL



- **Loop transfer function is**

$$T(s) = \frac{K}{s}, \quad K = \frac{I_{CP} K_{DL}}{2\pi C}$$

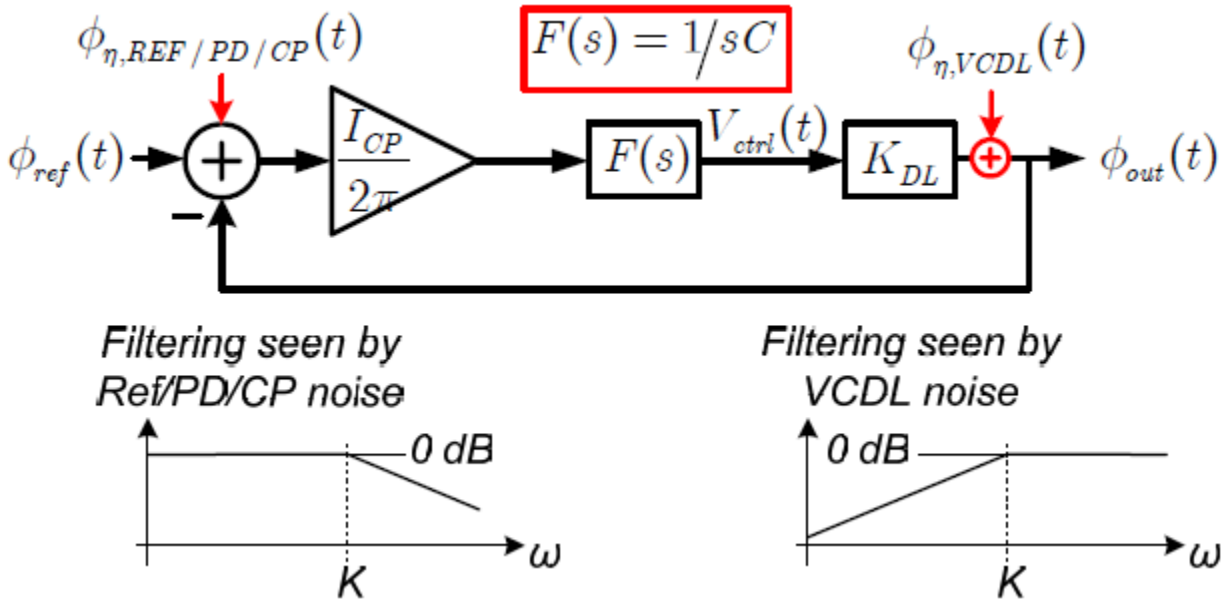
Unconditionally stable !

- **Closed loop response is**

$$A_{\phi}(s) = \frac{1}{1 + s/K}$$

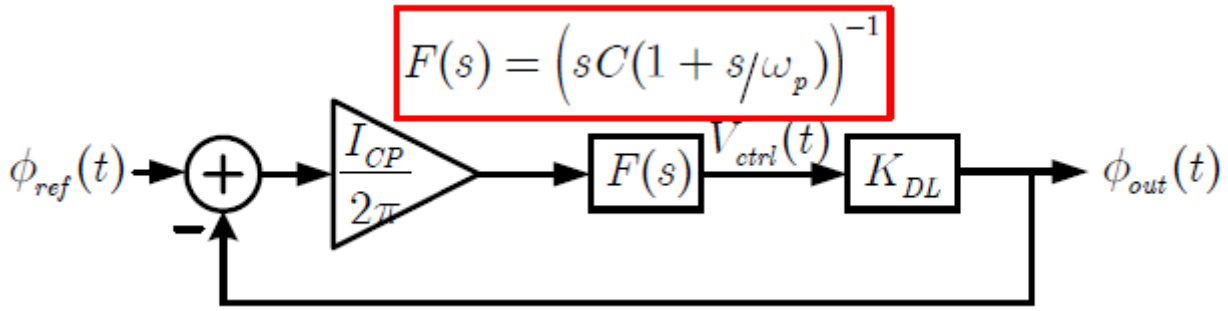
1st order response, no peaking !

Noise Transfer Functions



- **VCDL noise is suppressed by a high-pass filter.**
  - Wide bandwidth can be easily achieved because stability is not a problem.
- **Ref/PD/CP noise is suppressed by a low-pass filter.**
  - There is no jitter peaking.

2nd Order Type II CP DLL



- **Loop transfer function is**

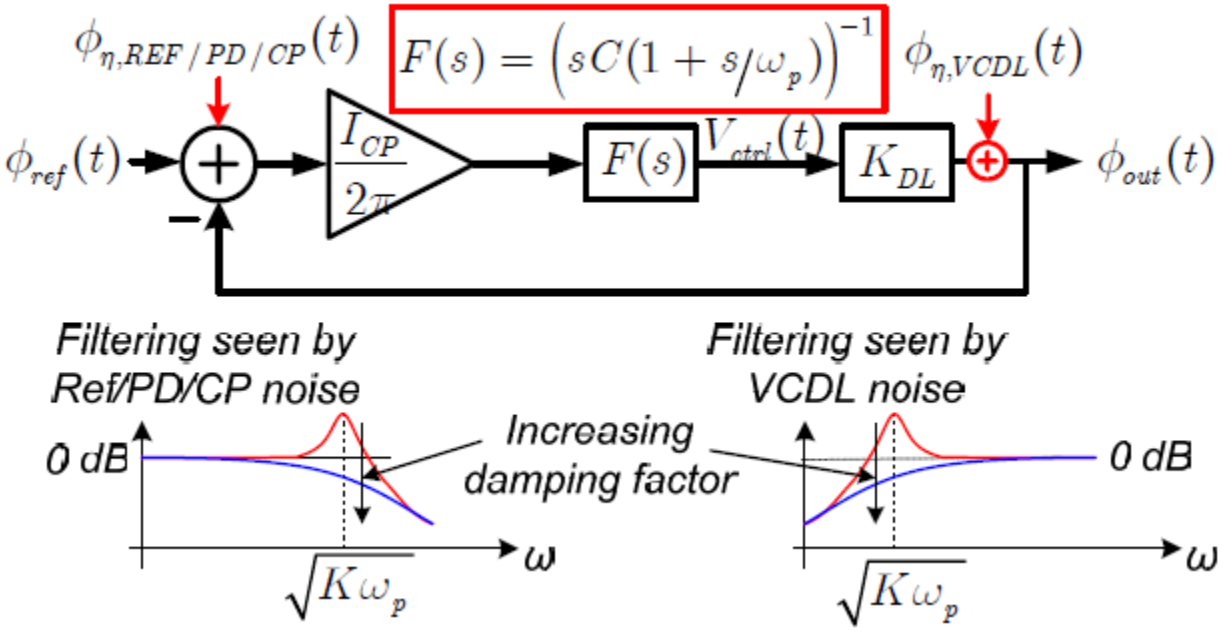
$$T(s) = \frac{K}{s(1 + s/\omega_p)}, \quad K = \frac{I_{CP} K_{DL}}{2\pi C}$$

Not unconditionally stable, anymore !

- **Closed loop response is**

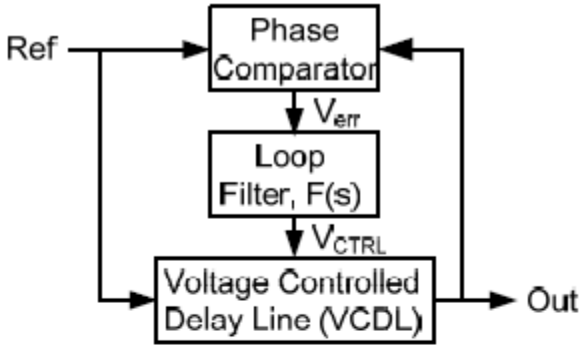
$$A_\phi(s) = \frac{1}{1 + s/K + s^2/K\omega_p}$$

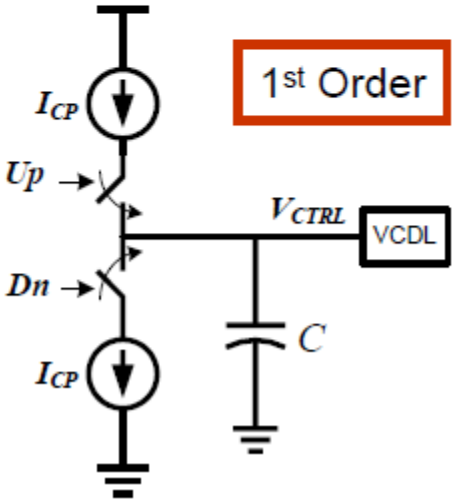
Noise Transfer Functions

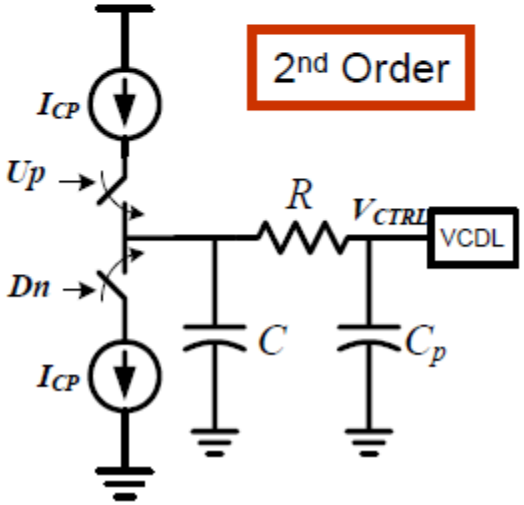


- **Additional pole has multiple effects.**
  - Increases jitter peaking.
  - Attenuates high frequency Ref/PD/CP noise a lot more.
  - Could increase VCDL noise contribution.

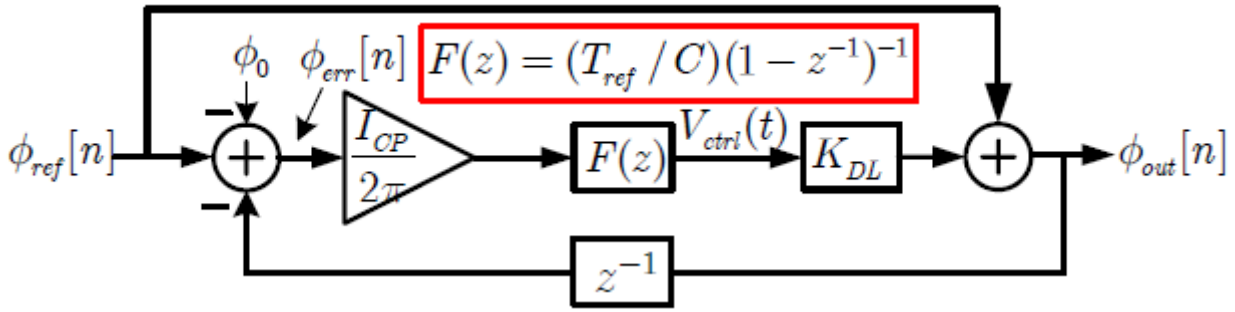
Type I CP DLL


Ref
Phase Comparator
V_{err}
Loop Filter, F(s)
V_{CTRL}
Voltage Controlled Delay Line (VCDL)
Out


1st Order
I_{CP}
Up
Dn
I_{CP}
V_{CTRL}
VCDL
C


2nd Order
I_{CP}
Up
Dn
I_{CP}
R
V_{CTRL}
VCDL
C
C_p

Type I DLL Discrete-Time Model



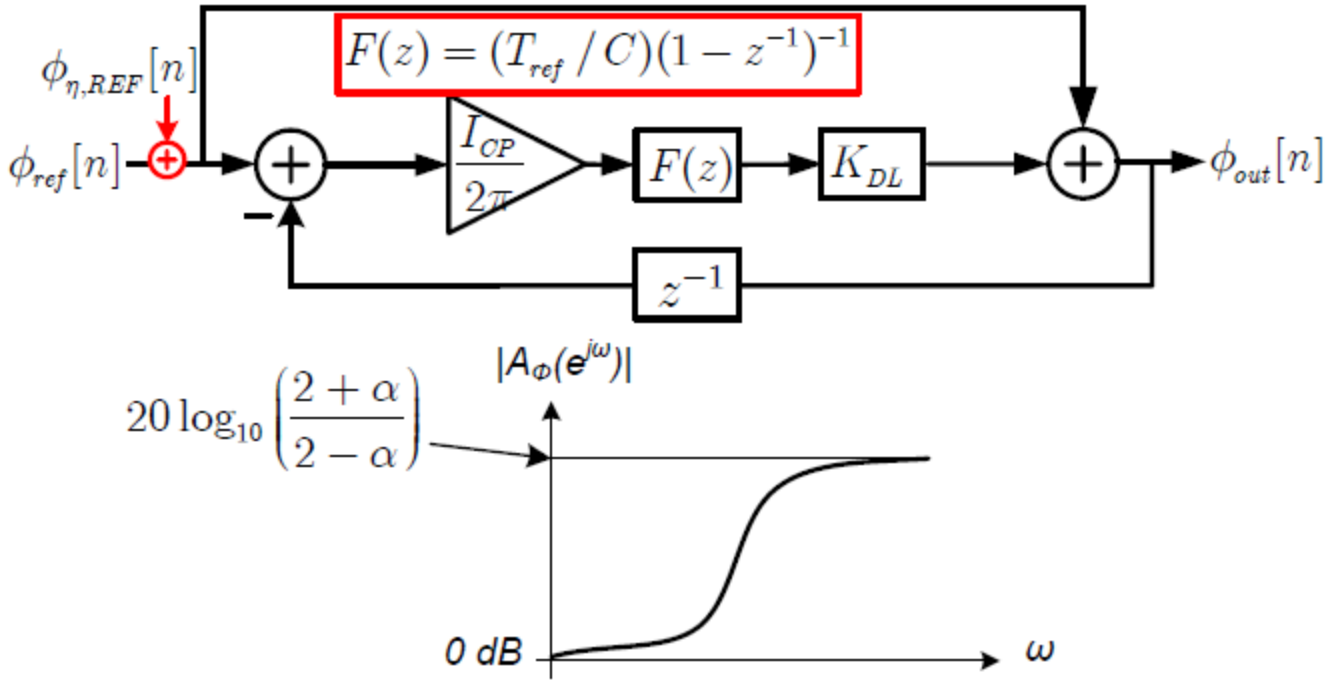
- **A continuous time mode can be derived, but a discrete time model is simpler and more appropriate.**
- **Loop transfer function is**

$$T(z) = z^{-1}\frac{I_{CP}K_{DL}F(z)}{2\pi} \qquad T(z) = \frac{KT_{ref}z^{-1}}{1 - z^{-1}},\ K = \frac{I_{CP}K_{DL}}{2\pi C}$$

- **Closed loop response is**

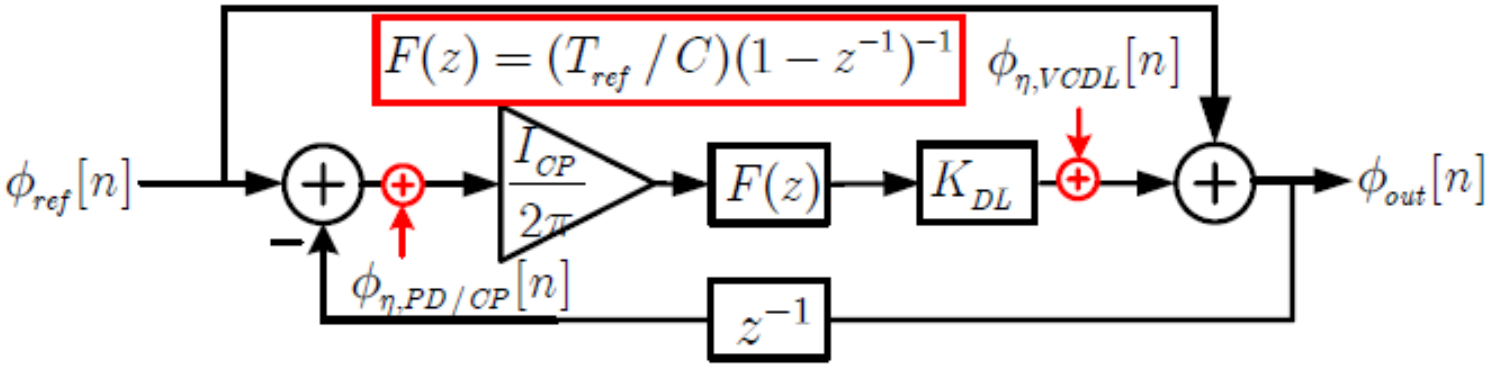
$$A_\phi(z) = \frac{1 + zT(z)}{1 + T(z)} \qquad A_\phi(z) = \frac{1 - z^{-1} + KT_{ref}}{1 - z^{-1} + KT_{ref}z^{-1}} = \frac{(1+\alpha) - z^{-1}}{1 - (1-\alpha)z^{-1}}$$

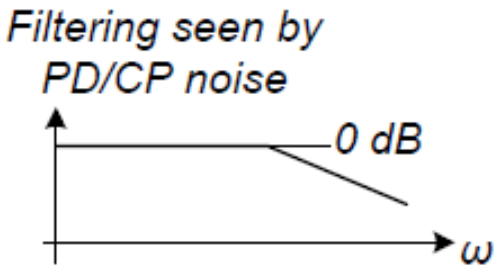
Jitter Peaking

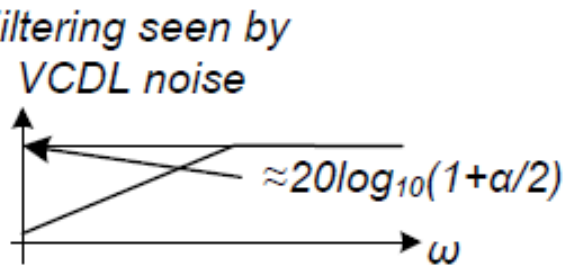


- **Noise on the reference does NOT see a low-pass filter.**
- **The reference noise is actually, significantly, amplified**
  - The Type I DLL thinks a sudden +ve jump in the Reference phase is actually a sudden –ve jump on the VCDL output phase
  - So, it increases the VCDL delay → jitter amplification

Other Noise Transfer Functions





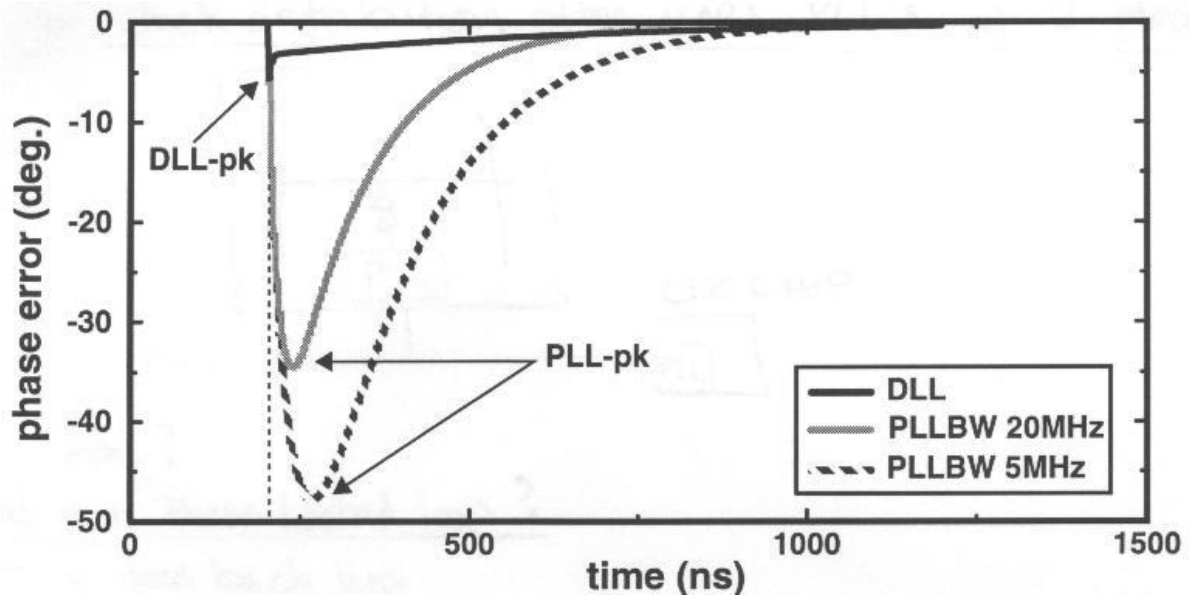


$$\frac{\phi_{out}(z)}{\phi_{\eta,PD/CP}(z)} = \frac{\alpha z^{-1}}{1-(1-\alpha)z^{-1}}$$

$$\frac{\phi_{out}(z)}{\phi_{\eta,VCDL}(z)} = \frac{1-z^{-1}}{1-(1-\alpha)z^{-1}}$$

PLL vs. DLL (1)

- **Stability**
  - DLLs are more stable, and hence easier to design.
- **Supply noise**
  - DLLs can be designed with higher bandwidths.
  - So, supply noise is better attenuated.



PLL vs. DLL (2)

- **Circuit noise**
  - The VCDL does not accumulate jitter unlike the VCO.
    - The VCDL actually accumulates noise within a period.
    - But, at the end of the period, the noise is cleaned up.
  - So, DLLs have lower phase noise and/or jitter due to circuit noise.

- **Reference noise**
  - DLLs (particularly Type I) have higher jitter peaking.