ECEN620: Network Theory
Broadband Circuit Design
Fall 2014

Lecture 15: Delay-Locked Loops (DLLs)

Sam Palermo
Analog & Mixed-Signal Center
Texas A&M University

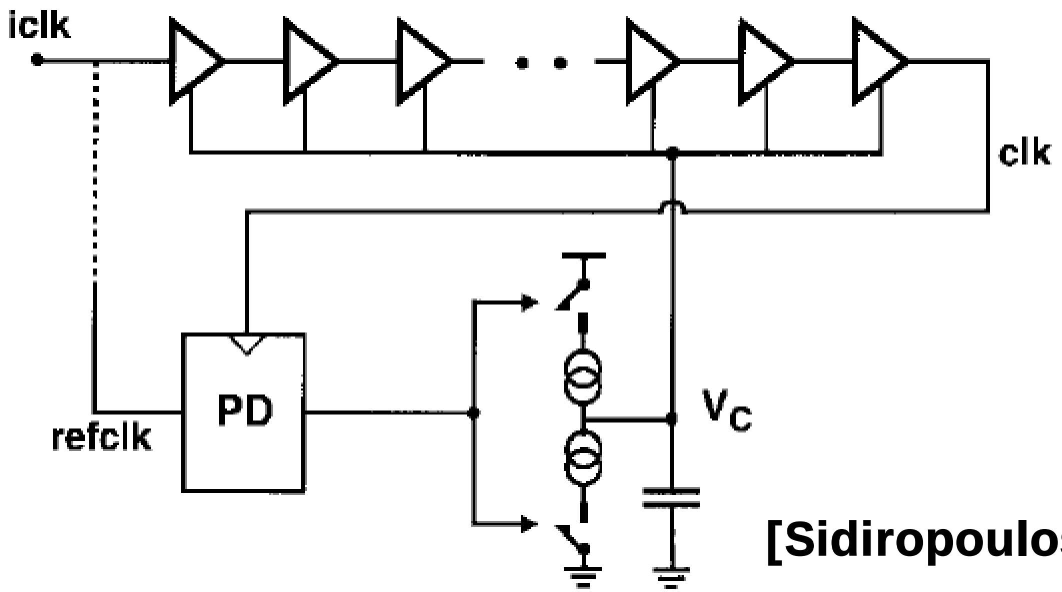
Announcements

- HW4 Due Wednesday Nov 5
- Exam 2 Friday Nov 7
  - One double-sided 8.5x11 notes page allowed
  - Bring your calculator
  - Covers through Lecture 14

Agenda

- DLL Basics
- DLL Delay Transfer Function
- DLL Applications

Delay-Locked Loop (DLL)



[Sidiropoulos JSSC 1997]

- DLLs lock delay of a voltage-controlled delay line (VCDL)
- Typically lock the delay to 1 or ½ input clock cycles
  - If locking to ½ clock cycle the DLL is sensitive to clock duty cycle
- DLL does not self-generate the output clock, only delays the input clock

DLL vs PLL Jitter Accumulation

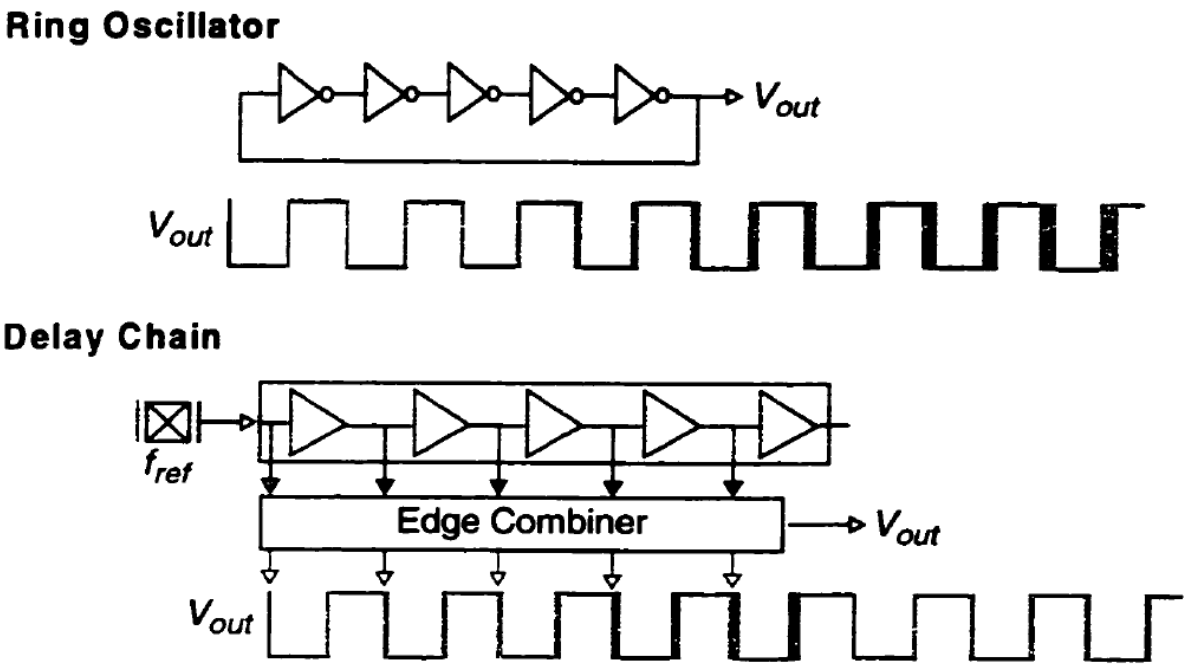


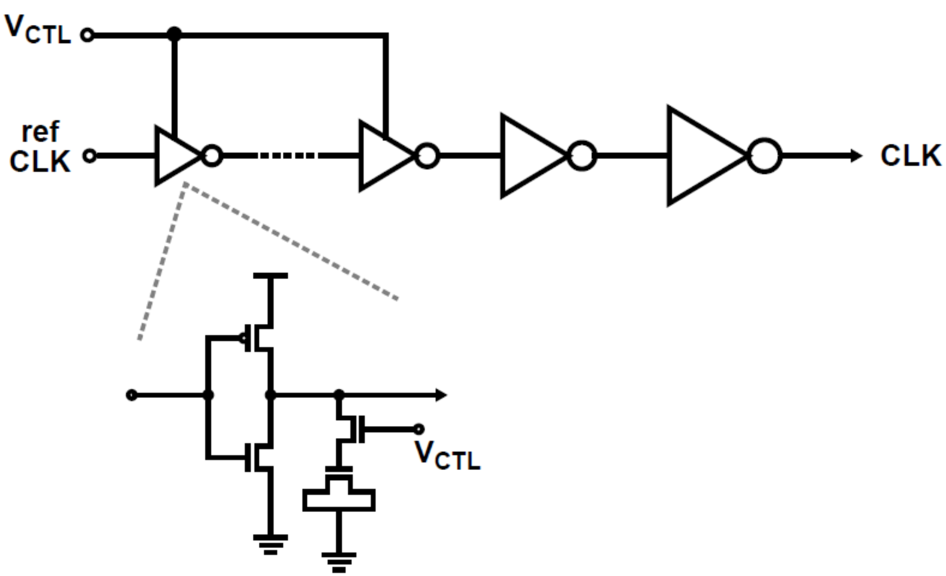
Fig. 4.3: Timing Jitter Accumulation for Ring Oscillator vs. Delay Chain.

- A VCO will accumulate jitter indefinitely
  - The rms jitter grows at a rate proportional to the sqrt(time)
- A delay line only accumulates jitter proportional to the total delay of the delay line

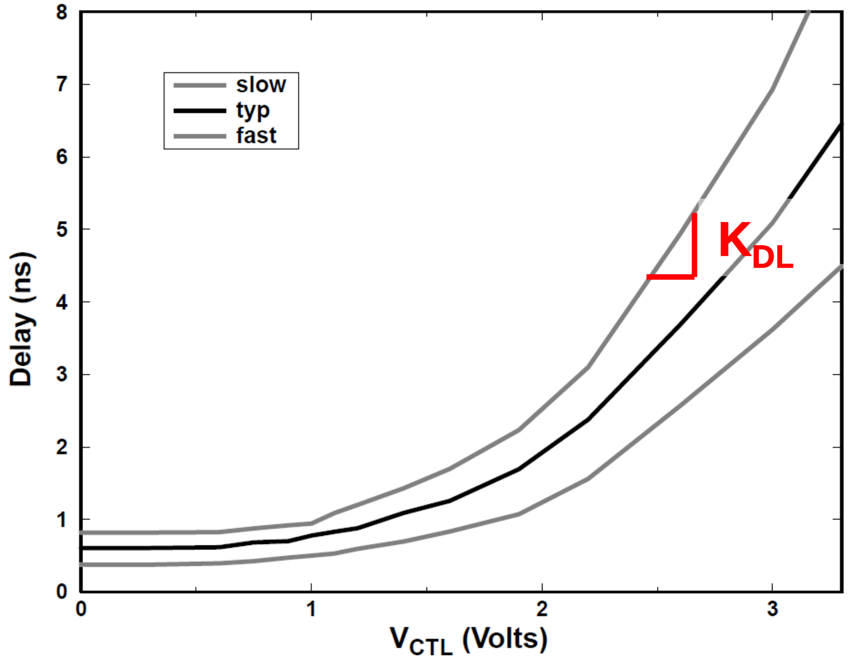
DLL vs PLL

- Jitter does not accumulate (as much) in a DLL delay line like in a PLL VCO
  - A jitter event simply gets transferred to the output of the delay line once and forgotten, unlike being re-circulated in a VCO
- The order of the DLL is generally equal to the loop-filter order, which is often one
  - DLL stability and settling issues are more relaxed relative to a PLL
- DLLs cannot easily generate different output frequencies, unlike a PLL where we can just change the divide ratio
- DLLs have the potential to delay lock to undesired multiples of the reference cycle, necessitating additional lock detect circuitry with a wide delay range delay line

Voltage-Controlled Delay Line

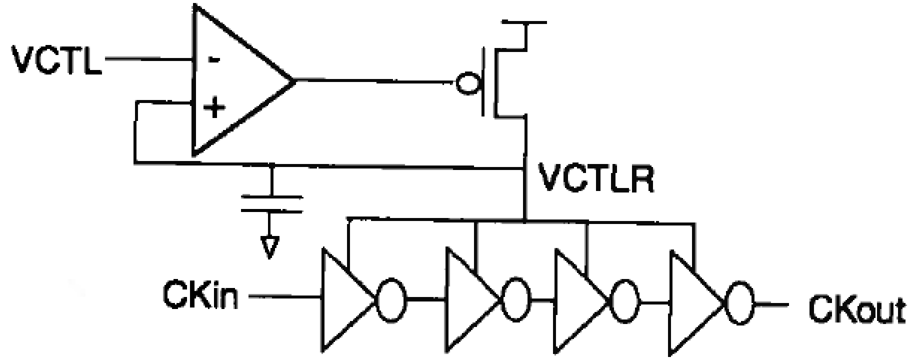


[Sidiropoulos]

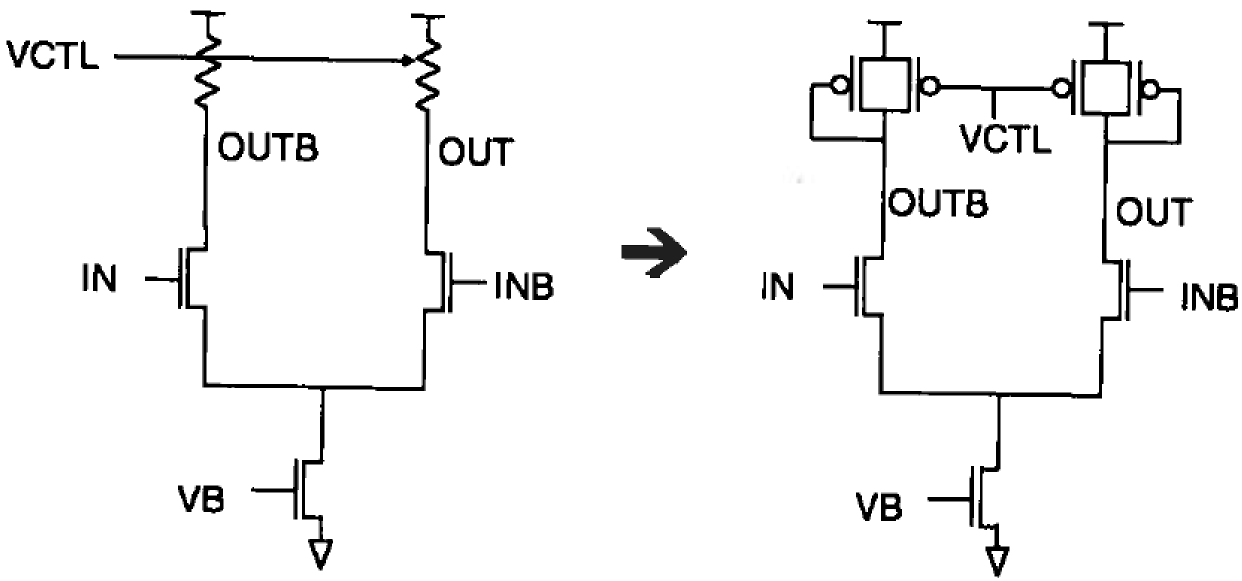


- The VCDL gain K_{DL} has units of s/V

Delay Cells



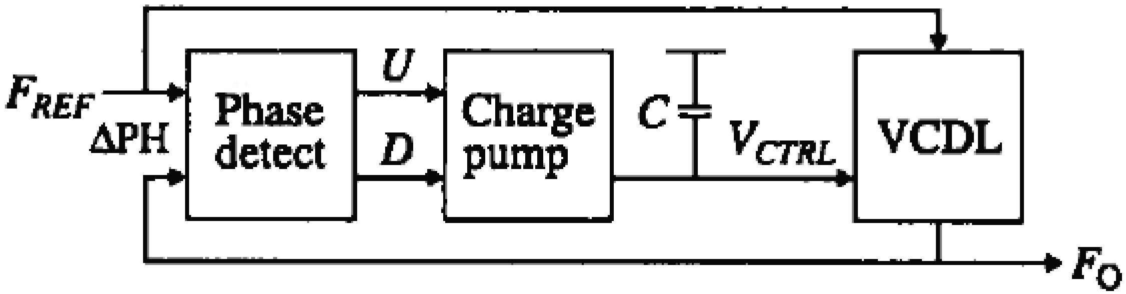
(a)



(b)

DLL Delay Transfer Function

[Maneatis]



$$D_O(s) = (D_I(s) - D_O(s)) \cdot F_{\text{REF}} \cdot \frac{I_{CH}}{sC_1} \cdot K_{DL}$$

$$\frac{D_O(s)}{D_I(s)} = \frac{1}{1 + s/\omega_N}$$

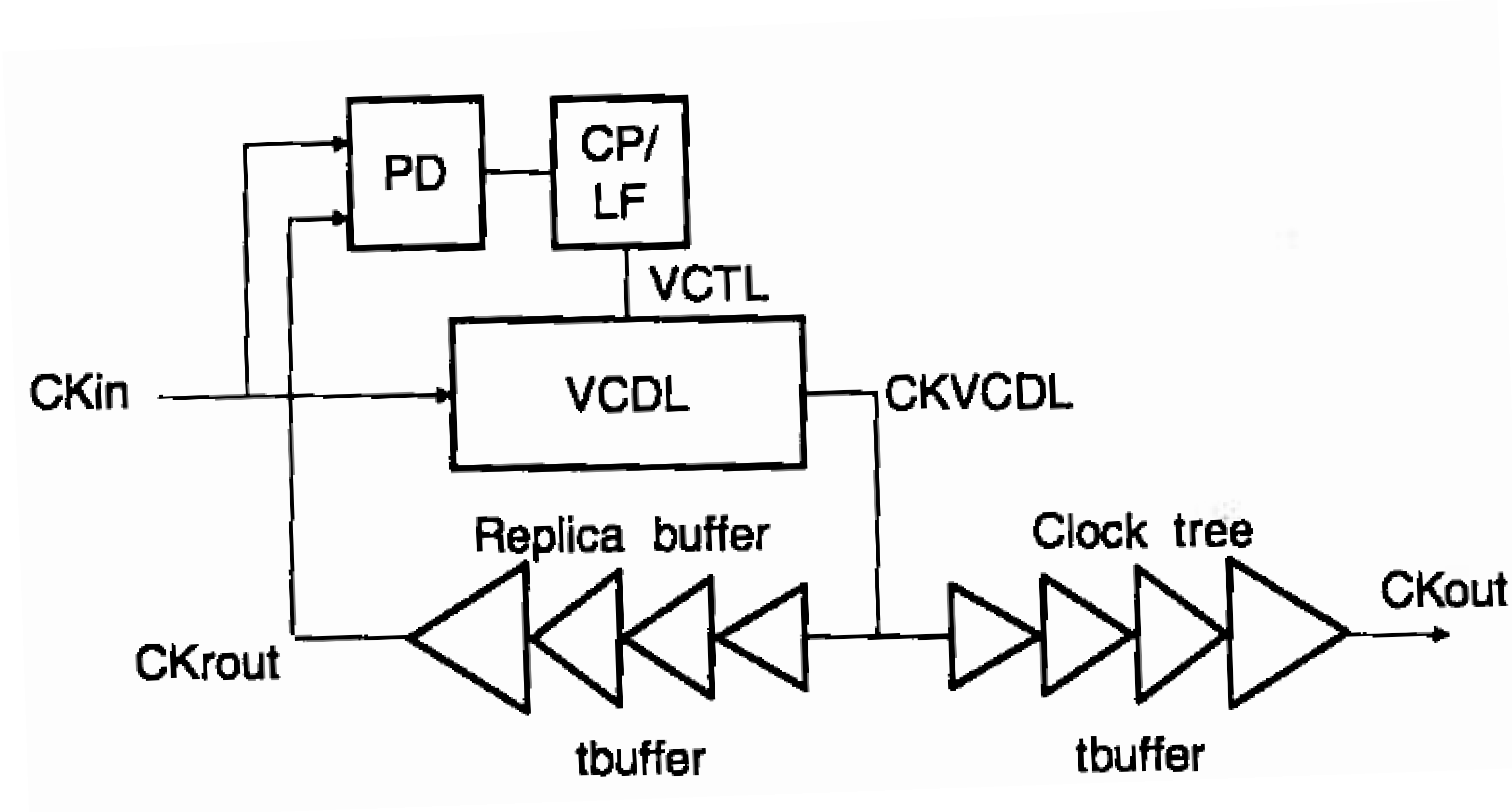
$$\omega_N = I_{CH} \cdot K_{DL} \cdot F_{\text{REF}} \cdot \frac{1}{C_1}$$

- First-order loop as delay line doesn't introduce a (low-frequency) pole
- The delay between reference and feedback signal is low-pass filtered
- Unconditionally stable as long as continuous-time approximation holds, i.e. $\omega_n < \omega_{ref}/10$

DLL Applications

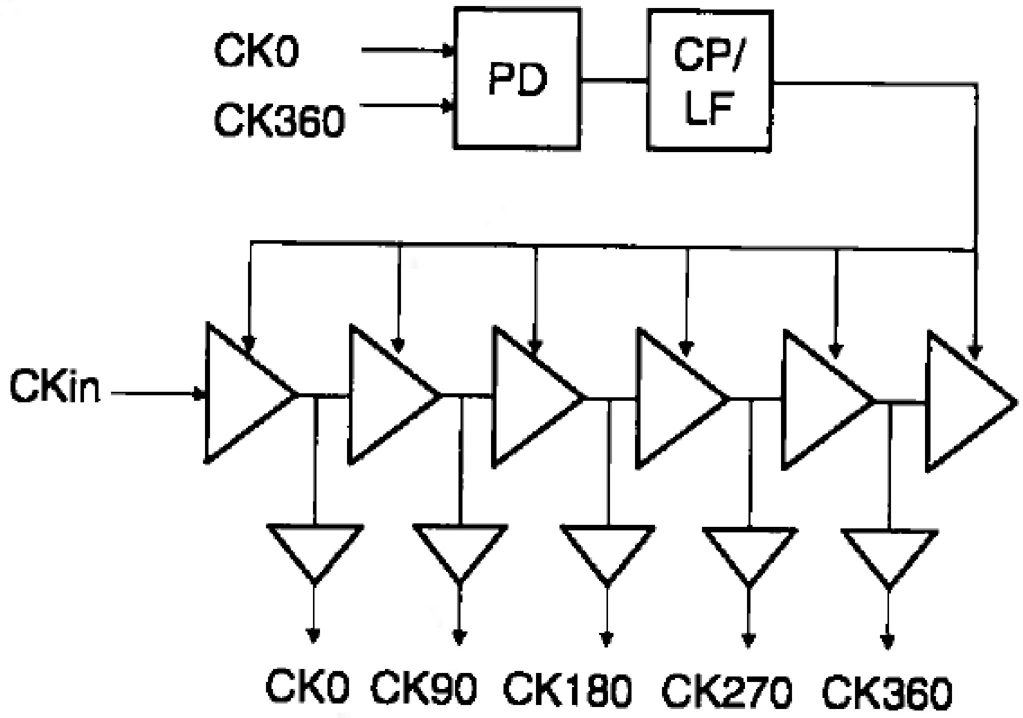
- Delay Compensation
- Multiphase Clock Generation
- Frequency Synthesis
- Clock & Data Recovery Systems

Delay Compensation



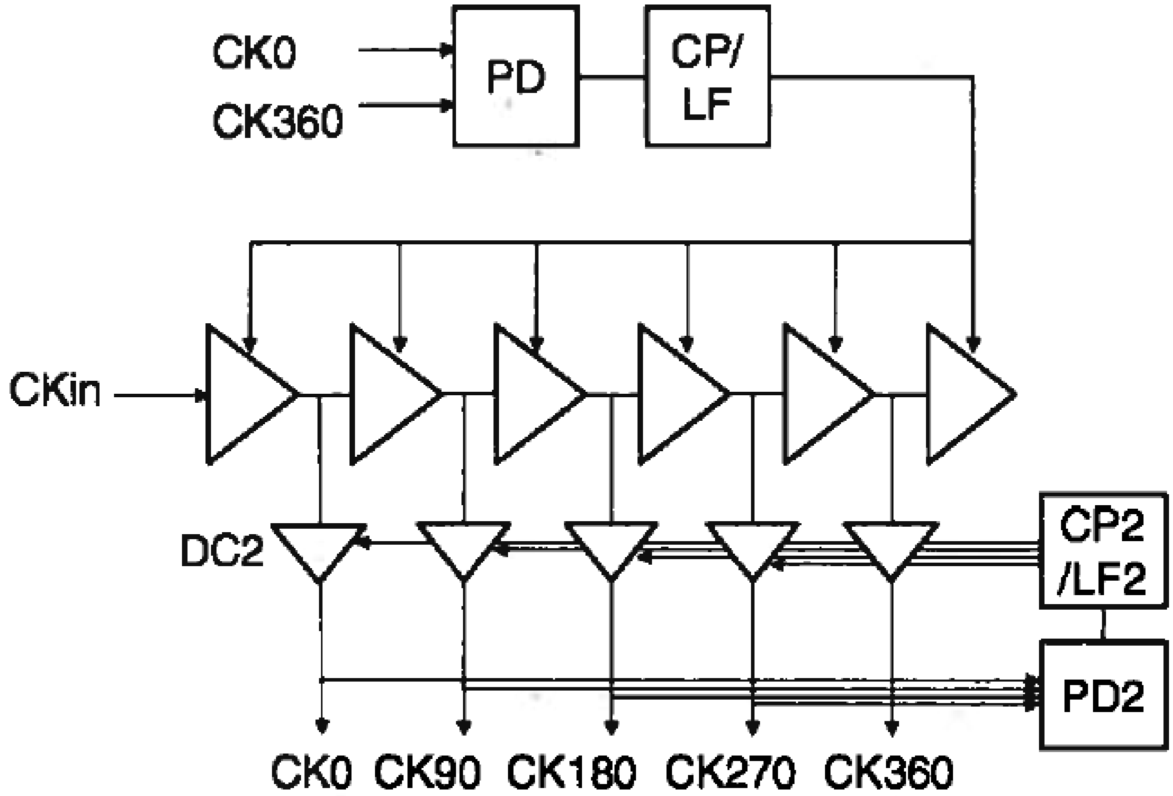
- A DLL with a replica buffer chain in the feedback path can be used to mask the delay of a clock buffer tree

Multiphase Clock Generation



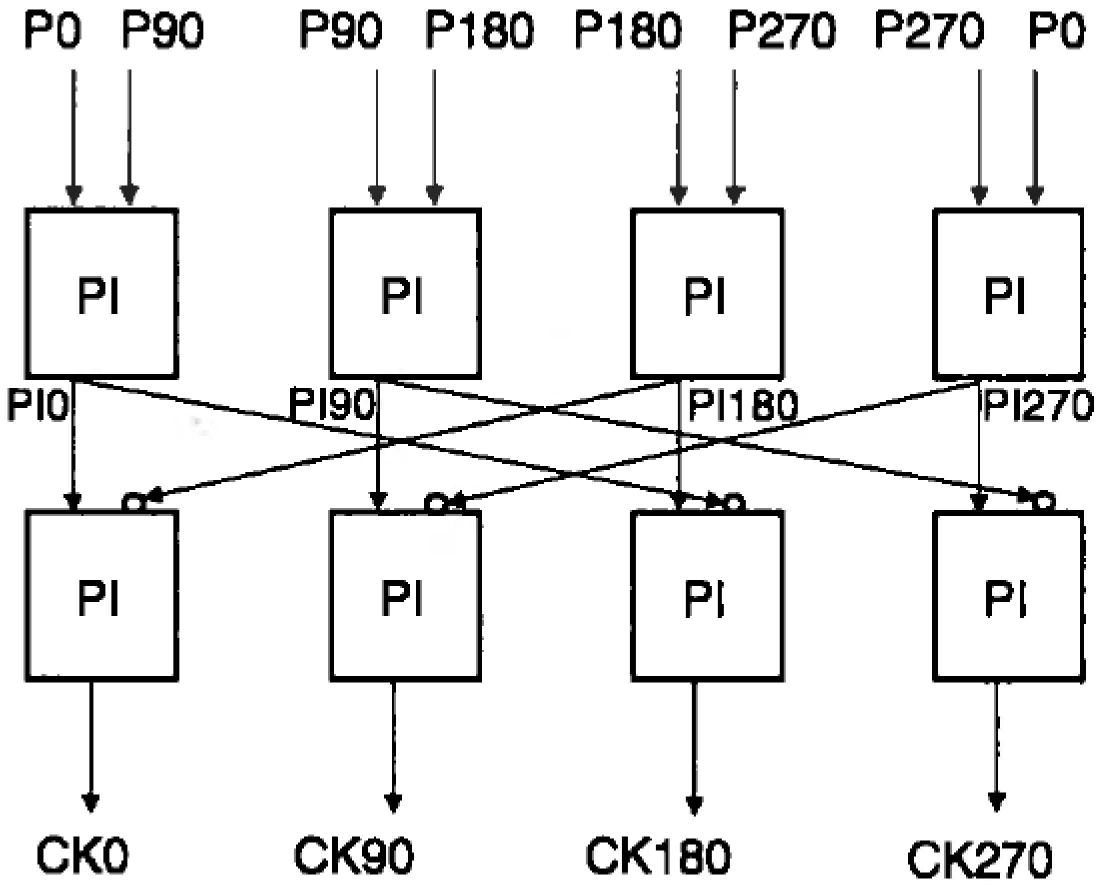
- A DLL can be used to generate multiple clock phases with precise phase spacing
- Useful in CDRs and RF modulation and up/down-conversion
- Phase errors are a function of the delay cell matching

Reducing Clock Phase Error - 1



- Additional delay cells can be added controlled by individual DLLs matching the 90 spacing
- Secondary DLLs can be classical analog or, for more efficiency, a digital implementation

Reducing Clock Phase Error - 2

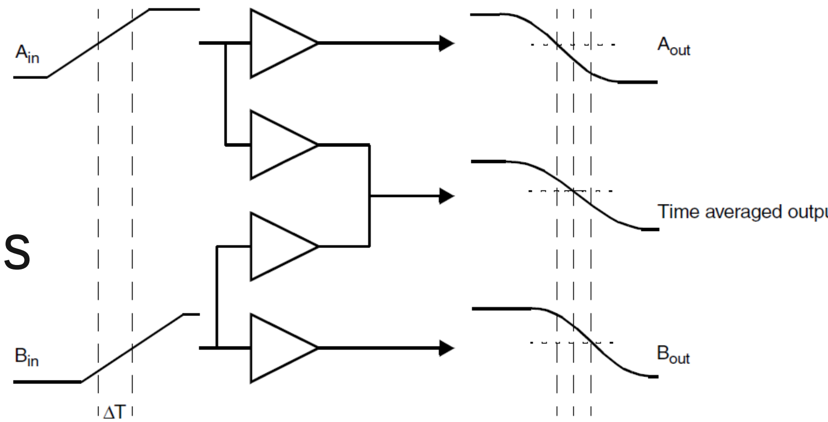


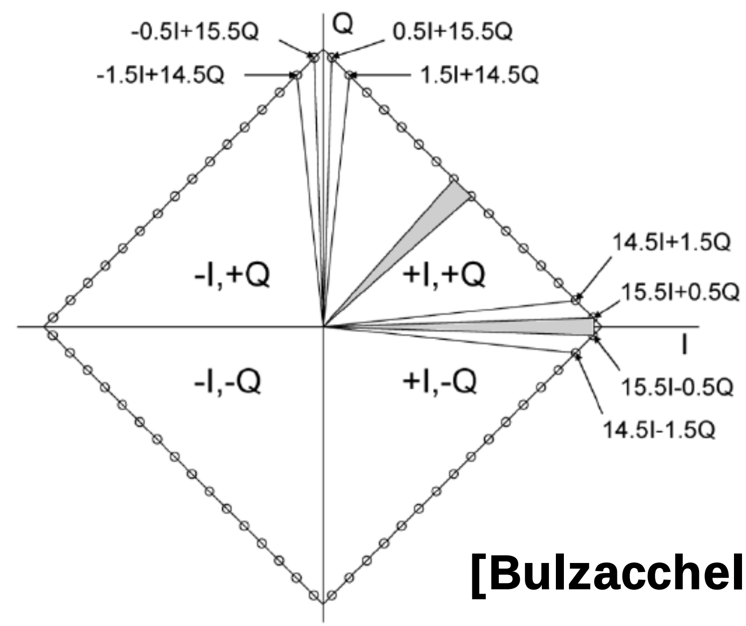
- Averaging with phase interpolator (PI) circuits can provide open-loop phase spacing compensation

Phase Interpolators

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

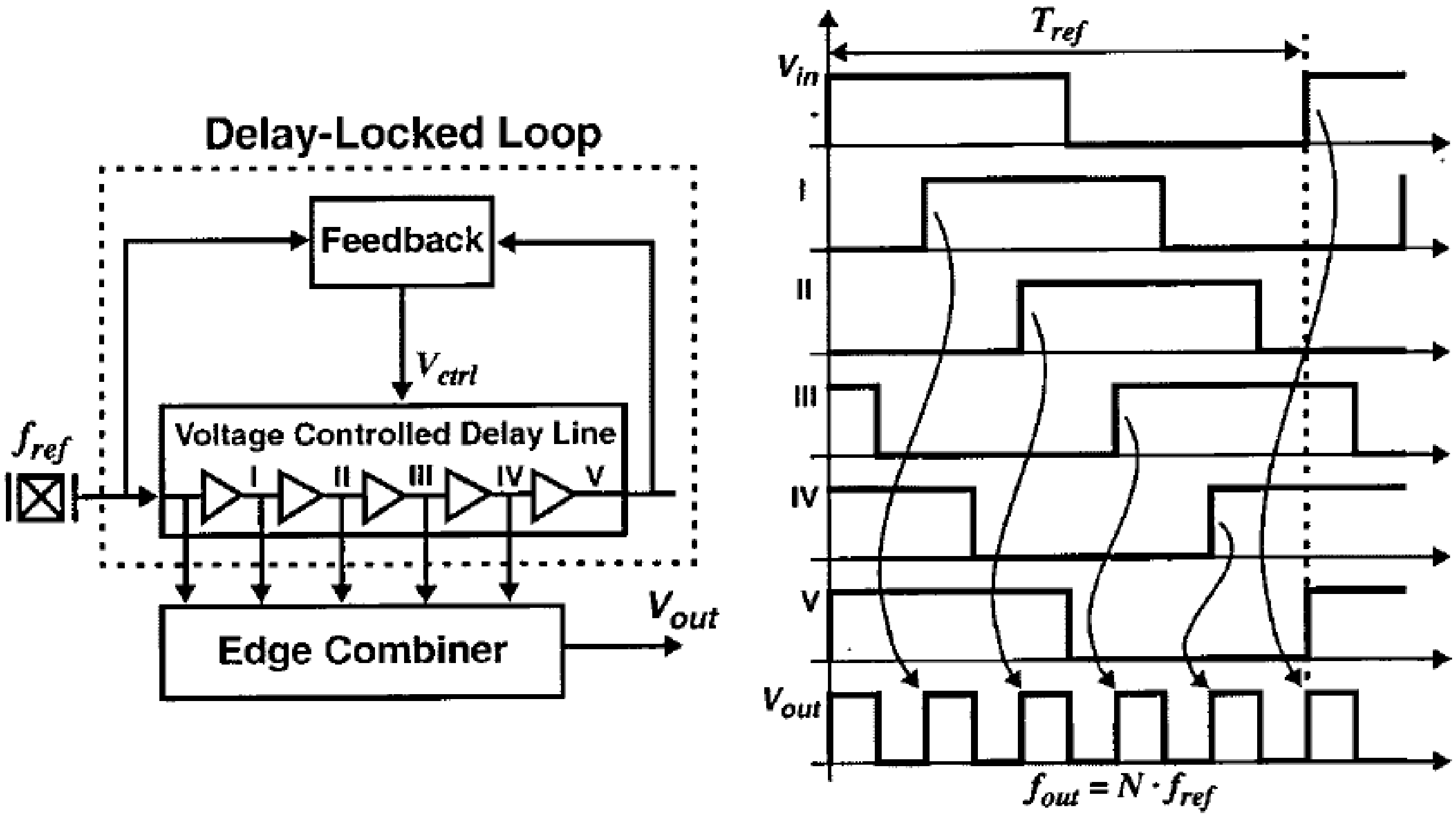
[Weinlader]





[Bulzacchelli]

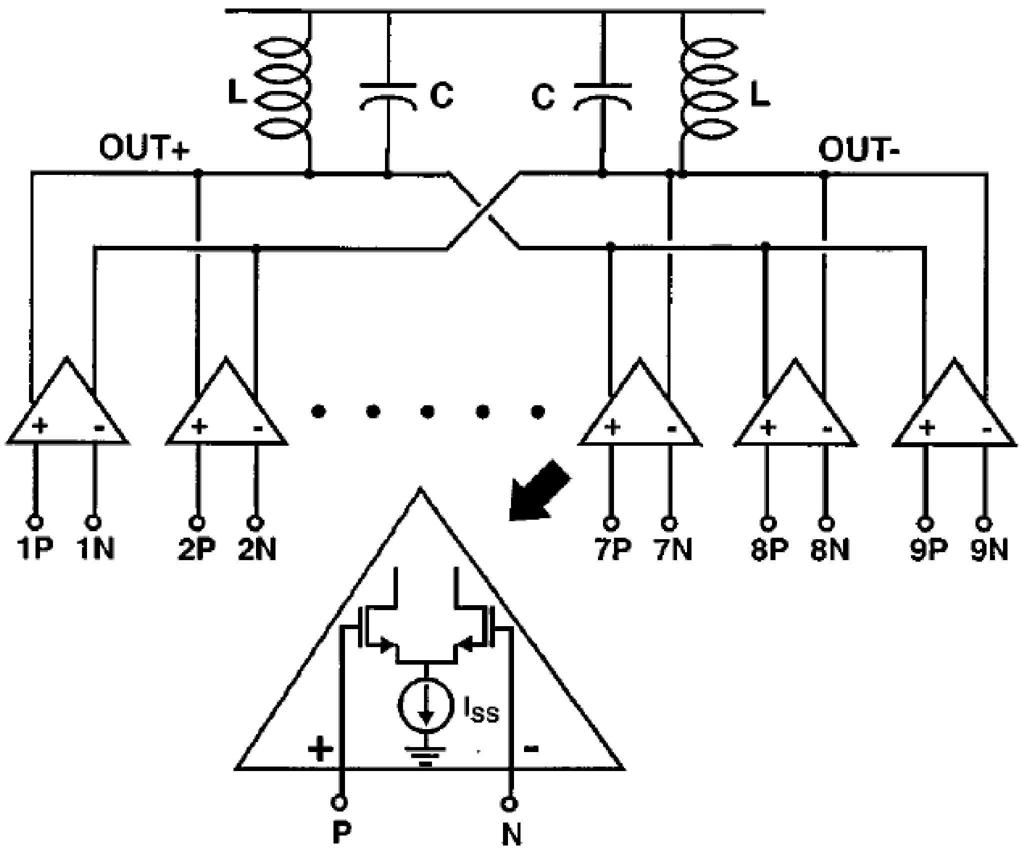
DLL Frequency Multiplier



- By adding an **ODD** number of clock phases generated by a DLL, an output frequency component results which is the input reference signal multiplied by the number of phases that are combined

A 900-MHz Local Oscillator Using a DLL-Based Frequency Multiplier Technique for PCS Applications, George Chien and Paul R. Gray

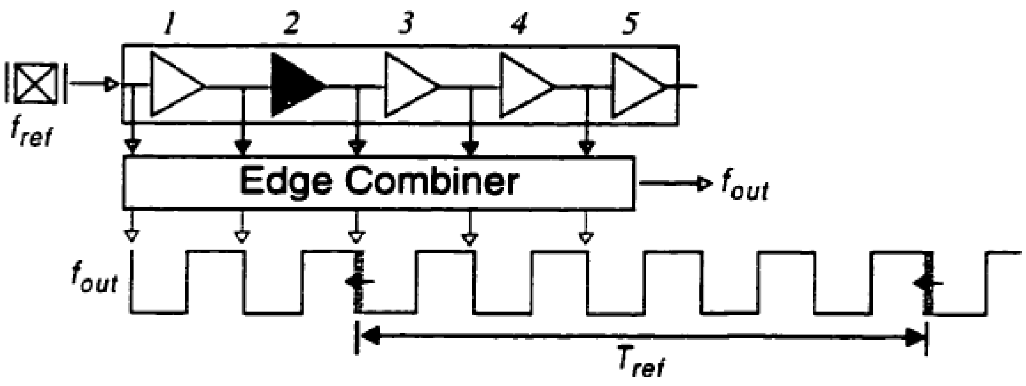
DLL Frequency Multiplier

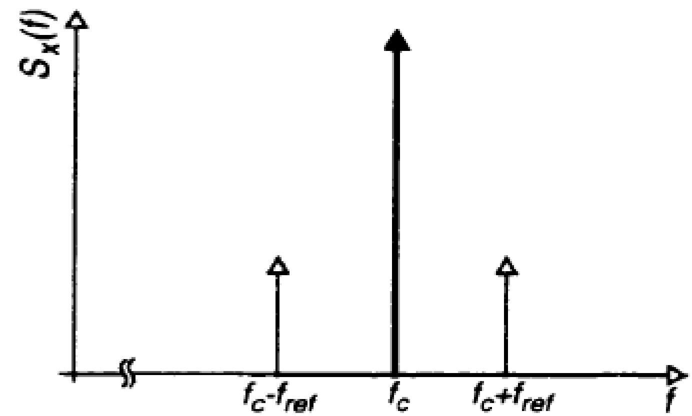


- A resonance LC tank load can be used to enhance the desired harmonic and provide filtering of unwanted reference harmonics/spurs due to DLL static phase error and delay element mismatches

A 900-MHz Local Oscillator Using a DLL-Based Frequency Multiplier Technique for PCS Applications, George Chien and Paul R. Gray

Delay Cell Mismatch Impact

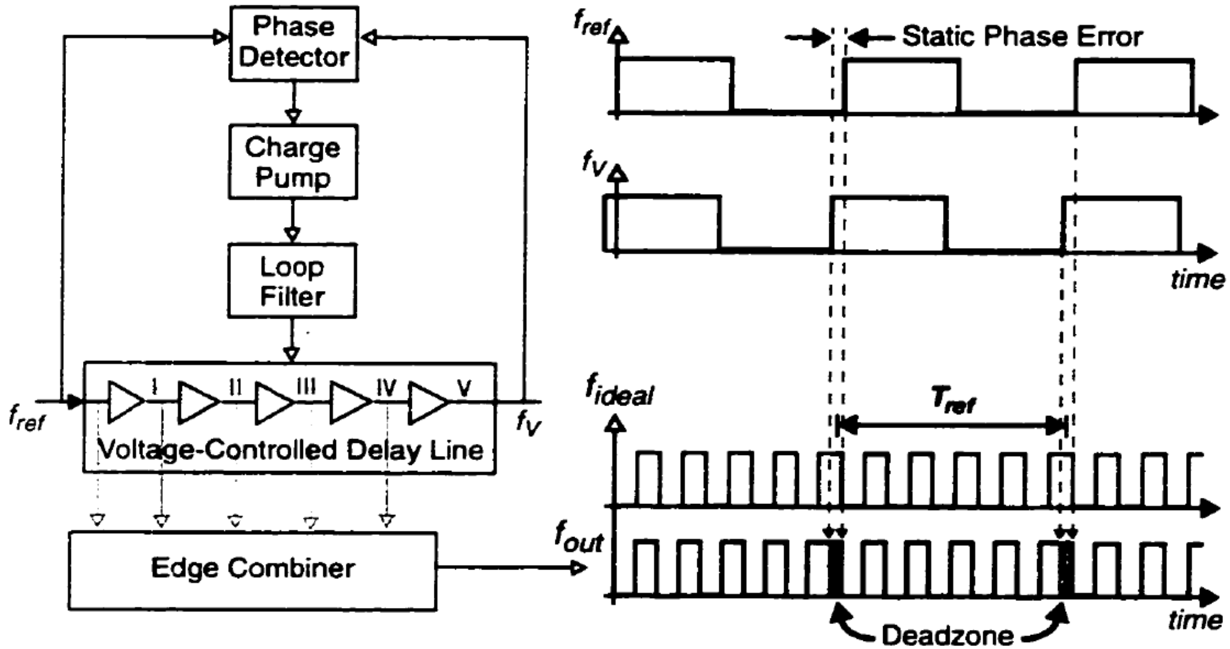




- DLL delay cell mismatch, due to process variation or deterministic layout mismatches, causes the delay of the cells to deviate from the ideal value
- This results in phases that are not in the ideal position
- An offset in one delay cell will show up at the output every f_{ref} period, resulting in spurious tones at f_{ref} and harmonics of f_{ref}

A 900-MHz Local Oscillator Using a DLL-Based Frequency Multiplier Technique for PCS Applications, George Chien and Paul R. Gray

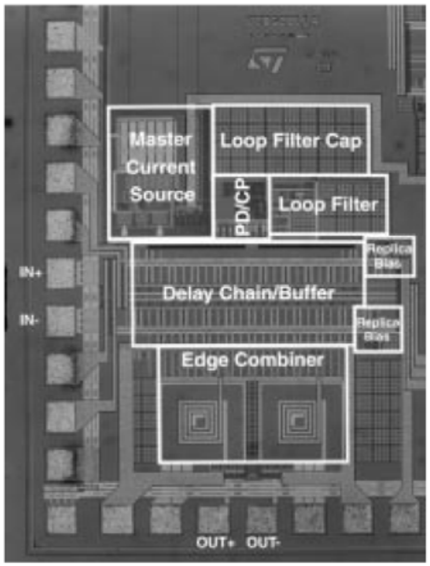
Static Phase Error Impact

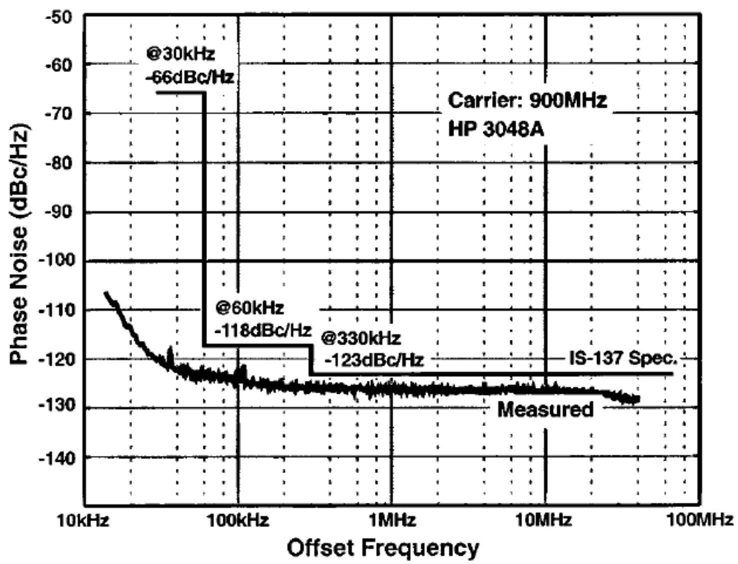


- If the DLL locks with a static phase error, then the output will have an output cycle with an exaggerated duty cycle error at the end of the delay chain cycle
- This occurs every reference clock period and produces frequency-domain spurs at f_{ref} multiples away from the multiplied output frequency

A 900-MHz Local Oscillator Using a DLL-Based Frequency Multiplier Technique for PCS Applications, George Chien and Paul R. Gray

Experimental Results



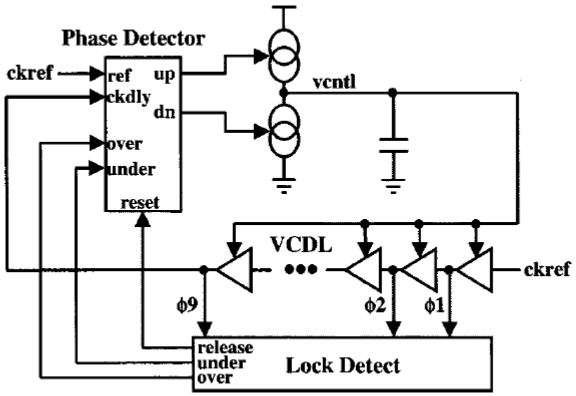


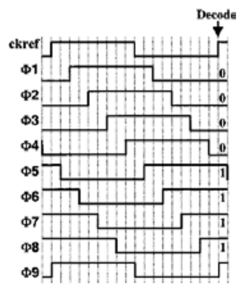
V_{DD}	3.3V
Technology	0.35µm CMOS
Output Frequency	900 MHz
Ref. Frequency	100 MHz
Phase Noise (dBc/Hz)	-120 @ 30kHz -123 @ 60kHz -127 @ 330kHz
Spurious Tones (dBc)	-30 @ 100MHz -37 @ 200MHz
Power	130 mW
Active Area	1.2 x 1.0 mm^2

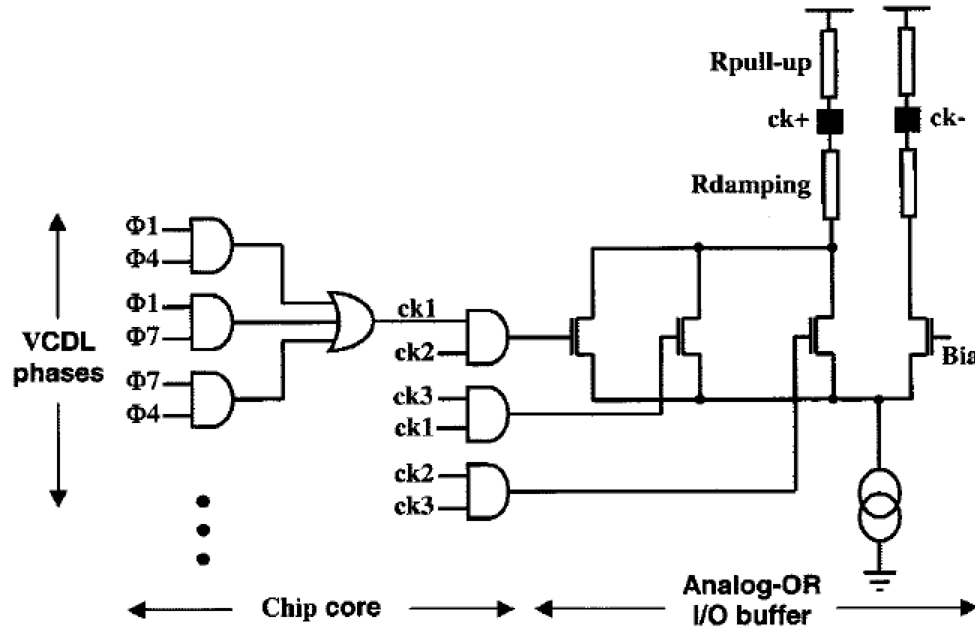
- While excellent phase noise performance is achieved, the relatively high spurious tones may be an issue in some applications

A 900-MHz Local Oscillator Using a DLL-Based Frequency Multiplier Technique for PCS Applications, George Chien and Paul R. Gray

DLL Frequency Synthesis w/ Digital Edge Combining



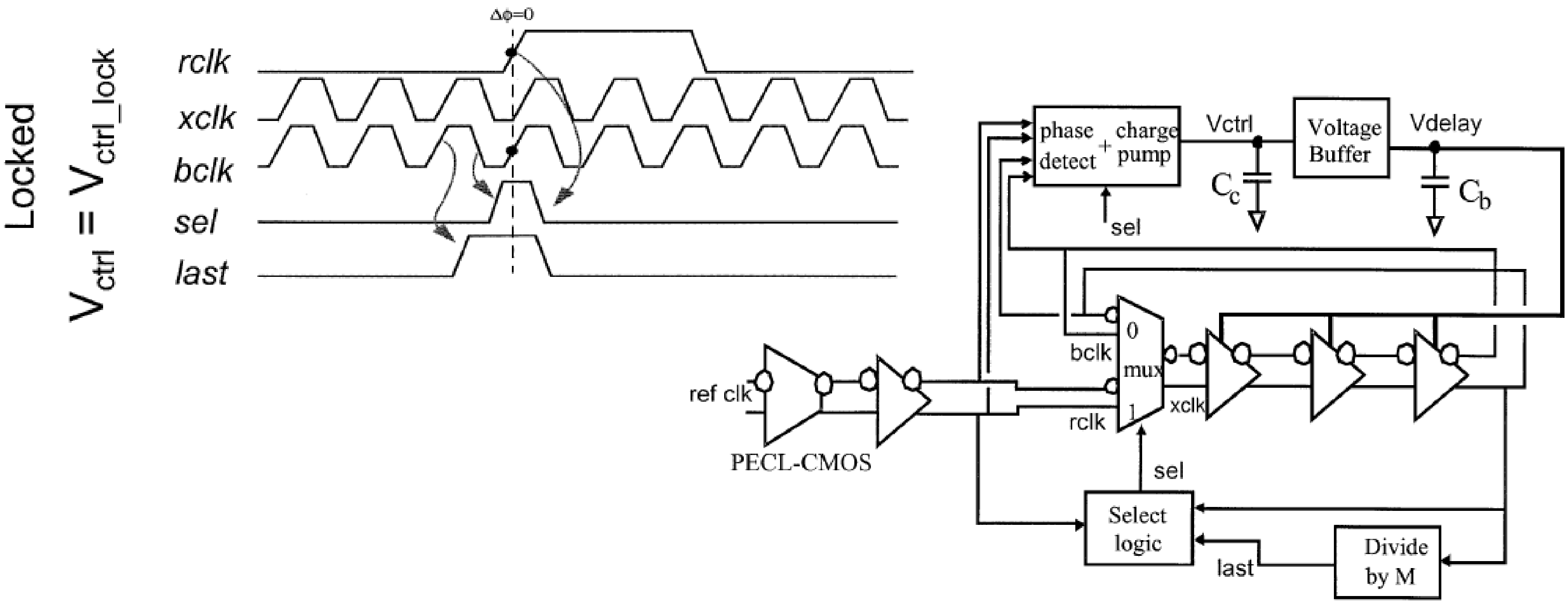




- The DLL edges can also be combined with digital logic, allowing for area savings relative to LC-tank filtering

CMOS DLL-Based 2-V 3.2-ps Jitter 1-GHz Clock Synthesizer and Temperature-Compensated Tunable Oscillator David J. Foley*, and Michael P. Flynn*

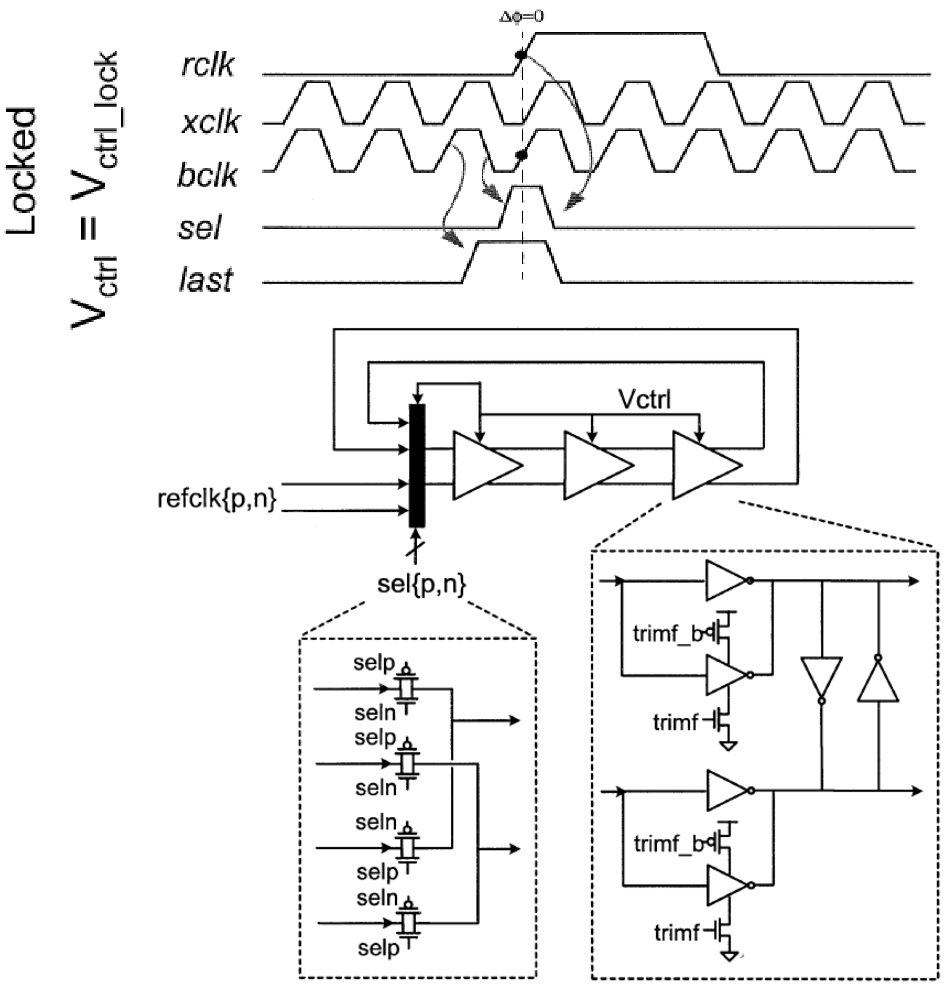
Multiplying DLL



- The delay line is configured as an oscillator for N-1 cycles
- Then is “reset” by the input clock for one cycle

A Low-Power Multiplying DLL for Low-Jitter Multigigahertz Clock Generation in Highly Integrated Digital Chips, Ramin Farjad-Rad, *William Dally, Hiok-Tiaq Ng,* Ramesh Senthinathan, M.-J. Edward Lee, Rohit Rathi, and John Poulton

Multiplying DLL Delay Line

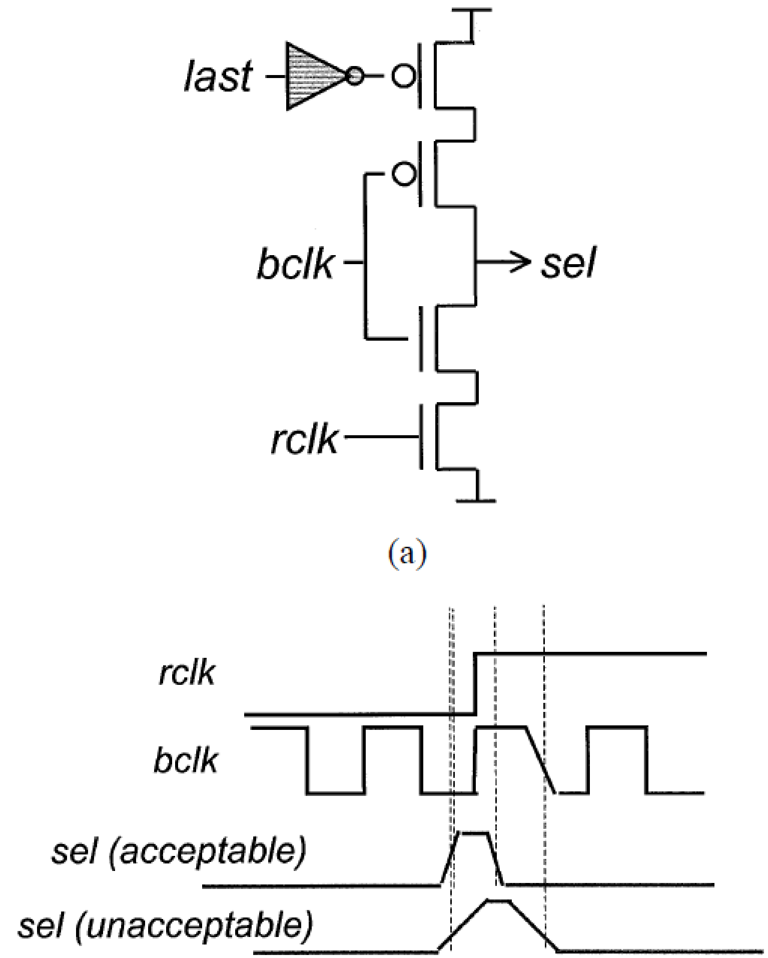


- Clock selection mux is implemented with CMOS transmission gates
- Delay elements are supply-regulated CMOS inverters
- Additional "trim" inverters are added to prevent Vctrl from falling too low and KDL from increasing dramatically

A Low-Power Multiplying DLL for Low-Jitter Multigigahertz Clock Generation in Highly Integrated Digital Chips, Ramin Farjad-Rad*, William Dally, Hiok-Tiaq Ng,* Ramesh Senthinathan, M.-J. Edward Lee, Rohit Rathi, and John Poulton

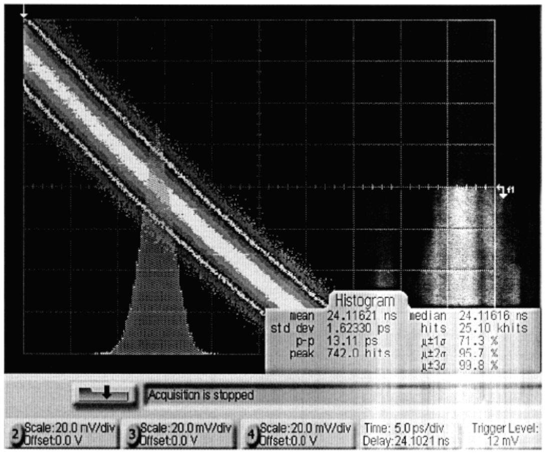
Critical Mux Select Signal

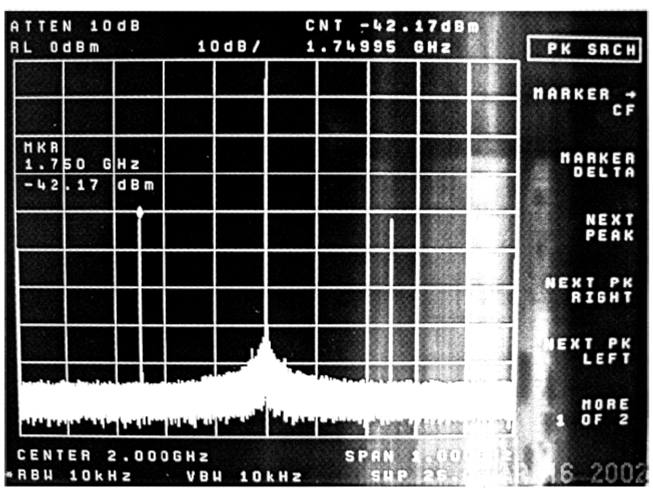
- When sel is high, rclk should get transferred to the delay line
- However, the next falling edge should be defined by bclk
- This results in very tight timing constraints in the sel generation circuitry
- If the sel signal is too slow, then an output cycle will have significant duty cycle distortion, resulting in bad spur performance
- A dynamic gate is used to increase the sel signal speed



A Low-Power Multiplying DLL for Low-Jitter Multigigahertz Clock Generation in Highly Integrated Digital Chips, Ramin Farjad-Rad*, William Dally, Hiok-Tiaq Ng,* Ramesh Senthinathan, M.-J. Edward Lee, Rohit Rathi, and John Poulton

Experimental Results



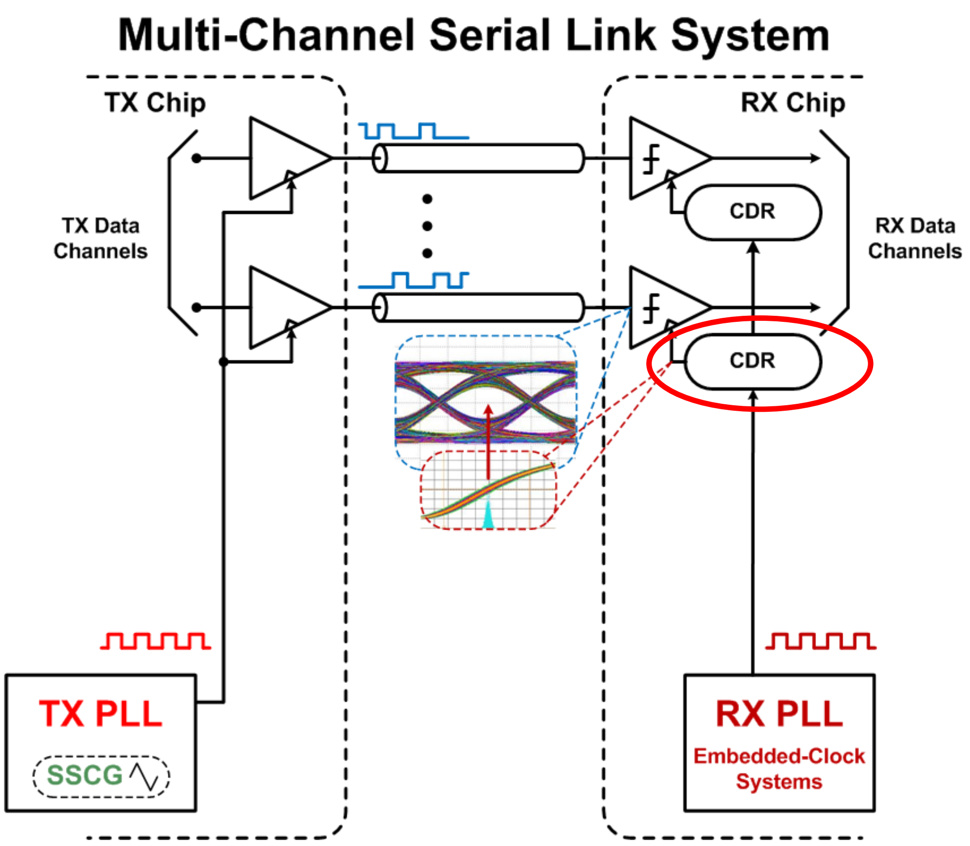


f_{ref}=250MHz, M=8, f_{out}=2GHz

Process Technology	0.18μm standard CMOS
Supply Voltage	1.8V
Active Area	0.05mm^2
Power @ 2.0GHz	12mW
Frequency Range	200MHz – 2.0GHz
Random jitter @ 2.0GHz (1σ)	1.64ps (250MHz X 8) 1.73ps (250MHz X 8, 72 X 72 grooming switch)
Random jitter @2.0GHz (peak-to-peak)	13.11ps (250MHz X 8)
Deterministic jitter @2.0GHz	12.00ps (250MHz X 8)
Random jitter @2.0GHz, 1.65V (1σ)	1.89ps (250MHz X 8)
Random jitter @2.5GHz, 2.00V (1σ)	1.94ps (250MHz X 10)
Random jitter @1.25GHz (1σ)	1.74ps (250MHz X 5)
Random jitter @1.25GHz (1σ)	1.99ps (125MHz X 10)

- The deterministic jitter is most likely dominated by the mux select operation
- Random jitter increases with N factor
  - The delay line is in oscillator mode for more cycles

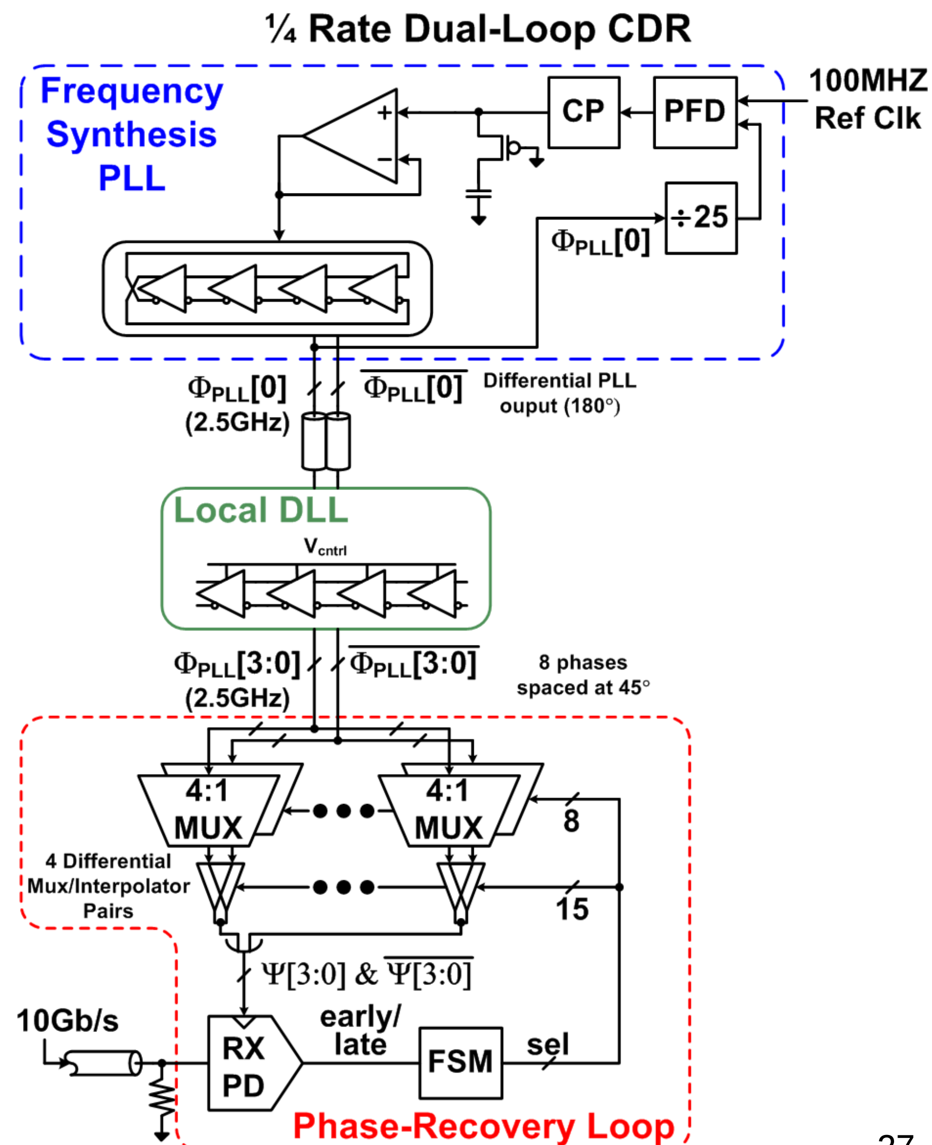
Embedded Clock I/O Circuits



- TX PLL
- TX Clock Distribution
- CDR
  - Per-channel PLL-based
  - Dual-loop w/ Global PLL &
    - Local DLL/PI
    - Local Phase-Rotator PLLs
    - Global PLL requires RX clock distribution to individual channels

DLL Local Phase Generation

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



Some Additional DLL References

- "A Low-Power Small-Area 7.28-ps-Jitter 1-GHz DLL-Based Clock Generator" Chulwoo Kim, et al.. JSSC 2002
- "Jitter Transfer Characteristics of Delay-Locked Loops Theories and Design Techniques", M.-J. Edward Lee, et all.. JSSC 2003
- "The Design and Analysis of a DLL-Based Frequency Synthesizer for UWB Application", Tai-Cheng Lee *and Keng-Jan Hsiao. JSSC 2006*
- "A 120-MHz–1.8-GHz CMOS DLL-Based Clock Generator for Dynamic Frequency Scaling", Jin-Han Kim, JSSC 2006

Next Time

- Clock-and-Data Recovery Systems