A 1V 15.6mW 1-2GHz -119dBc/Hz @ 200kHz clock multiplying DLL

Sander L.J. Gierkink
Conexant Systems, 100 Schulz Drive, Red Bank, NJ 07701

***Abstract*-A low-phase-noise 1-2GHz clock multiplier is configurable as PLL or DLL. Starting in PLL-mode, a lock-detect circuit switches the circuit to DLL and simultaneously reconfigures the loop filter. Upon loss of lock the circuit automatically falls back to PLL-mode. The number of stages in the VCO/delay line is programmable to 8-10-12-14-16, implementing band selection with modest tuning gain K_{VCO}. Multiplication is selectable from 2 to 64. The 0.11mm^2 90nm CMOS chip also includes charge pump and divider, both programmable. With 80MHz reference, phase noise is essentially flat and measures -125dBc/Hz and -119dBc/Hz @ 200kHz offset for 0.92 and 1.92GHz output respectively. Total maximum power consumption is 15.6mW from a 1V supply.**

I. Introduction

A PLL, with minimum loop order two, suppresses VCO phase noise over a bandwidth typically less than 1/20th of crystal reference frequency, due to loop stability requirements. A DLL, with one order less, suppresses delay line noise up to approximately half the crystal frequency, provided the periodic phase realignment is sufficiently strong. On the other hand, PLL static phase offset does not compromise frequency accuracy and a resultant reference spur can in principle be filtered by the loop filter. DLL static phase offset affects frequency accuracy and results in large reference spurs. Literature provides techniques to reduce DLL static phase offset [4,5] This paper focuses on minimizing DLL random phase noise, assuming a sufficiently stable low-frequency reference clock is available. A frequency domain model is explored [1], then circuitry is shown to maximize realignment strength, followed by measurements of a complete 1V 90nm 1-2GHz clock multiplier.

II. Phase Noise Transfer Functions

Figure 1 top left shows the DLL frequency-domain model. The transfer functions $H_{up}(j\omega)$ and $H_{rl}(j\omega)$ model reference noise up-conversion and phase realignment respectively [1]. The factor β, valued between <0-1>, models the strength of phase realignment by the reference. The model utilizes the VCO voltage–to-phase transfer $2\pi K_{VCO}/s$ with K_{VCO} in Hz/V. $H_{rl}(j\omega)$ transforms the VCO into a delay line by adding a first-order high-pass, resulting from the periodic realignment by reference *REF*. The table lists relevant transfer functions and approximations for $\omega<0.5\cdot\omega_{REF}$, with $H_{ref}(j\omega)=\phi_{out}/\phi_{ref}$ and $H_{vco}(j\omega)=\phi_{out}/\phi_{vco}$. Extensive time-domain simulations of a DLL in *Matlab* utilizing random number generators to model noise and zero crossing detection to track phase deviation, followed by Fourier transforms to plot phase PSD show good agreement with the expressions.

A simple, incorrect DLL model widely applied in literature uses $H_{up}(j\omega)=0$ and $H_{rl}(j\omega)=1$ and models the delay line's voltage-to-phase transfer as $2\pi K_{VCO}T_d$ with T_d the time of one delay-line circulation. However, it underestimates REF noise up-conversion and delay line noise suppression. It also fails to predict potential jitter-peaking in $H_{ref}(j\omega)$ [2].

Figure 1 top right shows $H_{up}(j\omega)$ and $H_{rl}(j\omega)$ while the bottom plot shows the closed-loop transfer functions $H_{ref}(j\omega)=\phi_{out}/\phi_{ref}$ and $H_{vco}(j\omega)=\phi_{out}/\phi_{vco}$ for β=0.25,0.5,1, all with f_{REF}=20MHz, N=100, I_{CP}=400uA and C=2pF. Larger β (better realignment) gives more VCO noise suppression, but also more HF reference noise upconversion. Small β results in peaking in both reference upconversion and VCO noise. Lowering loop gain by increasing C/lowering I_{CP} removes this peaking. Note that for β=1 $H_{vco}(j\omega)$ has +40dB/dec slope for $\omega<<\omega_{REF}$, changing to +20dB/dec for larger $\omega<\omega_{REF}$. The +40dB/dec region extends with larger loop gain, useful for suppressing delay line 1/f noise as shown in section IV.

III. Circuitry

Figure 2 shows the delay stage. Inputs *A* and *B* each drive a tri-state inverter, (de)activated by *enB*. Both inverters drive the same input of a NAND structure. The other NAND input is driven by an inverter, connected to output P_n. This forms a latch that secures P_n when, after a short delay, it resets the active input *A* or *B*. As a result, waveform duty cycle is low, enabling easy pulse removal/reinsertion as shown later.

The PLL/DLL schematic in figure 4 is comprised of an 8/10/12/14/16-stage VCO/delay line of programmable length with single-stage "exit branch" plus an 8-stage reference delay line. All stages are identical and tuned by the PLL/DLL. Division factor N is programmable in the range 2-64. At startup, all stages are briefly reset and lock is low. This configures the loop filter as 2nd-order for the PLL and disables *eninj*, preventing pulse updates. In both PLL and DLL mode, rising *REF*-edges inject a pulse in the reference delay line. Pulses at stages *M4* and *P16* are phase locked by the PLL. Upon phase lock, signal *lock* goes high on a falling *REF*-edge, reconfiguring the loop filter for DLL-mode and enabling pulse removal/reinsertion by subsequent rising *REF*-edges.

Figure 3 shows the DLL pulse removal/reinsertion process. Pulses at *M4* and *P16* are locked. A rising *REF*-edge injects a pulse in the reference delay line *M1..8*. While this pulse travels down the delay line, *eninj* goes high. This enables *M4* to inject a pulse into stage *P1* and redirects the pulse arriving at stage *P16* to the exit branch, where it terminates. It can be shown that noise of the reference delay line is non-dominant in the final DLL output phase noise.

Charge sharing between injected- and removed pulse is minimized, maximizing β and thus suppression by $H_{rl}(j\omega)$ (see figure 1). The tri-state inverters in figure 2 and the exit branch in figure 4 isolate the injected/removed pulse. Separate low-output-impedance buffers control the tune node of oscillator- and reference delay line, suppressing spikes on the tune node that would otherwise arise during resets of a delay stage.

The loop filter in figure 4 is programmable independently for PLL- and DLL. In DLL the resistor can be shorted and the capacitance reduced in 6 steps of 10pF. The lock-detect circuit automatically switches between programmed configurations.

Figure 5 shows the lock-detect circuit. At startup, the multiplier acts a PLL. When rising edges at the detector's inputs $in_{1,2}$ line up, latch SR_1 is set. This condition is alternately retimed in latches $SR_{2,3}$. If lock is maintained over 32 consecutive *REF*-cycles, *lock* goes high on a falling *REF*-edge, switching the multiplier to DLL-mode. At the very first non-lock event, *lock* goes low, resetting the detector's counter and switching the multiplier back to PLL-mode, until lock is re-established. This prevents false-lock problems normally encountered in DLLs. Setting *enDLL* in figure 4 low forces the multiplier to work solely in PLL-mode.

IV. Experimental Results

The PLL/DLL is realized in 1V 90nm standard CMOS. Figure 6 shows a die photograph. Active area is $0.11mm^2$.

Figure 7 shows the output spectrum with f_{out}=1.92GHz, f_{ref}=80MHz over 300MHz span, using 8 stages in the ring. Trace 3 is the PLL, traces 1,2 are the DLL with HP8662A signal generator and Pletronics SQ33 crystal clock oscillator as reference respectively. Trace 1 clearly shows the sinc-shaped sideband spectrum due to upconversion of reference noise, as seen in the plot of $H_{ref}(j\omega)$ with β=1 in figure 1. Trace 2 has lower phase noise than trace 1 due to the lower phase noise of the crystal reference compared to the HP8662A. Even when using a crystal reference, the DLL phase noise is larger than that of the PLL at offsets larger than approx. 20MHz, showing that upconversion of reference noise is dominant at these offsets. Reference spur is -35.2dBc, comparable to [3].

Figure 8 shows the spectrum at f_{out}=1.92GHz over 20MHz span. Trace 1 is the PLL with f_{ref}=80MHz, traces 2,3 are the DLL with f_{ref}=40MHz and 80MHz respectively (N=48,24), all measured with a crystal clock reference. The DLL with 80MHz reference has 27dB better phase noise than the PLL at 1MHz offset. The difference between traces 2 and 3 is close to 6dB, accounting for a factor 2 difference in division factor N.

Figure 9 shows the DLL spectrum at f_{out}=1.92GHz, f_{ref}=80MHz, N=24 at various capacitor C settings of the DLL loop filter and charge pump current I_{CP}. With increased loop gain (larger I_{CP} and lower C) the close-carrier phase noise decreases, as it increases the transition frequency between the +40/+20dB/dec region of $H_{VCO}(j\omega)$ with β=1 in figure 1.

The table in figure 10 summarizes DLL phase noise at offsets of 200kHz and 1MHz for various division factors N, measured with 40,60,80MHz crystal clock reference oscillators. The number of stages used in the ring is 16,12,8 for f_{out}=0.96,1.44,1.92GHz respectively. Taking into account the DLL "flat" phase noise and normalizing for total power P [mW], division factor N and output frequency f_{out} a DLL figure-of-merit can be defined as:

$$FOM_{DLL} = 10 \cdot \log\left(\frac{L \cdot P}{f_{out}^2 \cdot N^2}\right)$$

with phase noise L taken here at 200kHz offset. This work shows a minimum 9dB improvement over state-of-the-art.

Figure 11 shows accumulated random RMS jitter vs. time interval τ for PLL (loop bandwidth BW=0.2MHz) and DLL at f_{out}=1.92GHz, measured with a sampling scope. DLL jitter is τ-independent, showing that β=1. It is about 1.26ps and 1.70ps for N=24,48 respectively. The jitter ratio for N=48/N=24 is approximately $\sqrt{2}$. PLL jitter is proportional to $\sqrt{\tau}$ for small τ, and flat for τ>1/2/π/BW, as expected.

V. Conclusions

A reconfigurable 1-2GHz PLL/DLL clock multiplier is presented with maximized realignment strength β=1 for maximum delay line noise suppression. Ring length, loop filter, charge-pump current and division factor are all programmable. A lock-detect circuit continuously monitors lock and automatically switches to PLL mode to prevent false lock in DLL mode. The effects of loop gain and reference noise upconversion on DLL phase noise are discussed and measured. Compared to DLLs in literature this work presents a minimum 9dB improvement in random phase noise, normalized for multiplication factor, power and frequency.

References

[1] S. Ye and I. Galton, "Techniques for Phase Noise Suppression in Recirculating DLLs", IEEE J. Solid-State Circuits, vol. 39, no. 8, pp. 1222-1230, Aug 2004.

[2] M.-J. Lee, W.J. Dally et. al., "Jitter Transfer Characteristics of Delay-Locked Loops – Theories and Design Techniques", IEEE JSSC, vol. 38, no. 4, pp. 614-621, Apr 2003.

[3] R. Farjad-Rad et. al. "A Low-Power Multiplying DLL for Low-Jitter Multigigahertz Clock Generation in Highly Integrated Digital Chips", IEEE JSSC, vol. 37, no. 12, pp. 1804-1812, Dec 2002.

[4] P.C. Maulik and D.A. Mercer, "A DLL-Based Programmable Clock Multiplier in 0.18-µm CMOS With -70 dBc Reference Spur", IEEE JSSC, vol. 42, no. 8, pp. 1642-1648, Aug 2007.

[5] S. Gierkink, "An 800MHz -122dBc/Hz-at-200kHz Clock Multiplier based on a Combination of PLL and Recirculating DLL", Proc. ISSCC 2008, vol. 51, pp. 454-455, Feb. 2008.

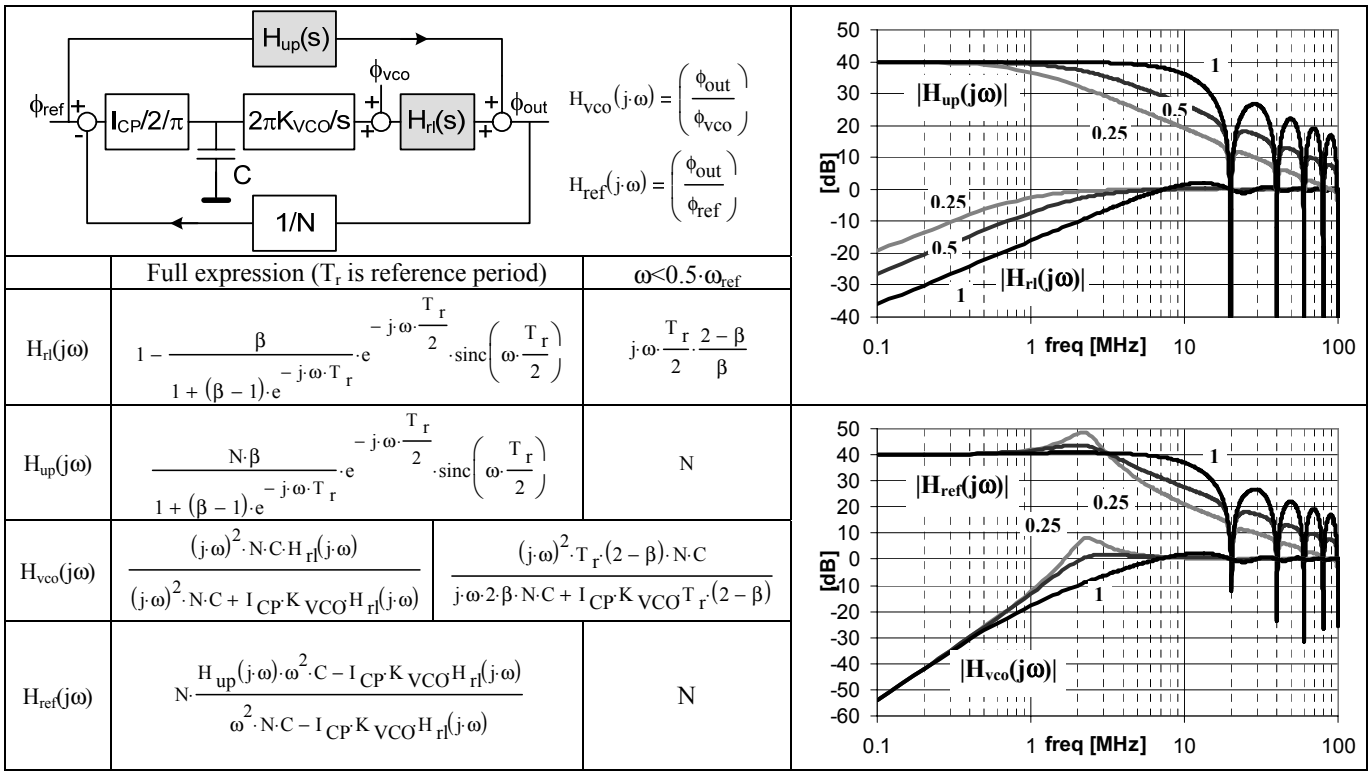


Fig. 1. Freq.-domain DLL model [1] (top left); transfer functions and low-freq. approximations (left); plot of transfer functions for β=0.25,0.5,1 (right).

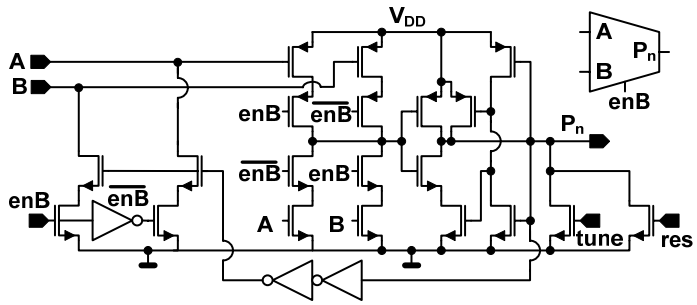


Fig. 2. Schematic of delay stage.

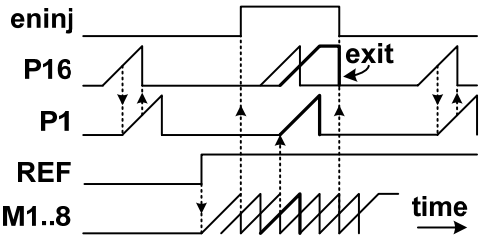


Fig. 3. Waveforms during pulse removal/insertion in DLL mode.

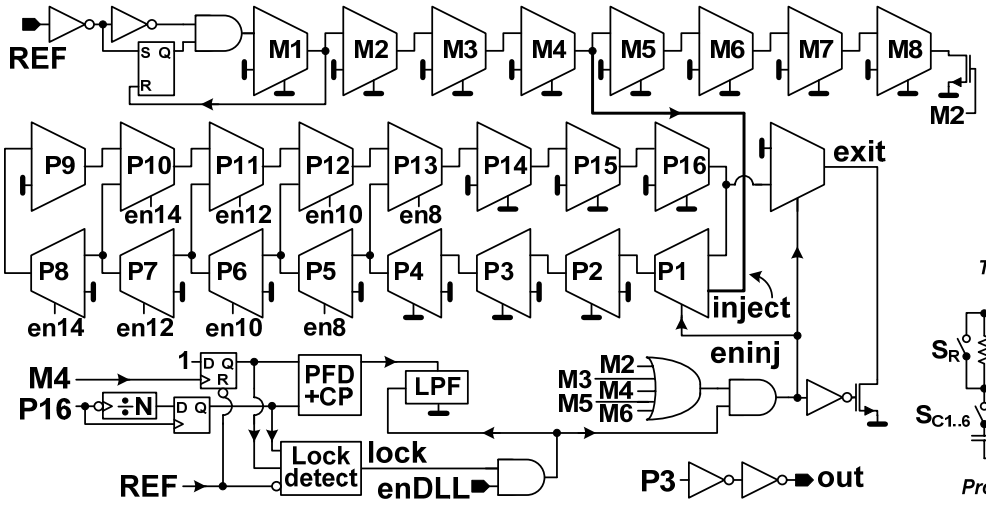
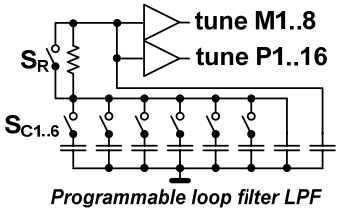


Fig. 4. Schematic of clock multiplier.

stages →	8	10	12	14	16
en8	1	0	0	0	0
en10	D	1	0	0	0
en12	D	D	1	0	0
en14	D	D	D	1	0

Truth table number of stages



Programmable loop filter LPF

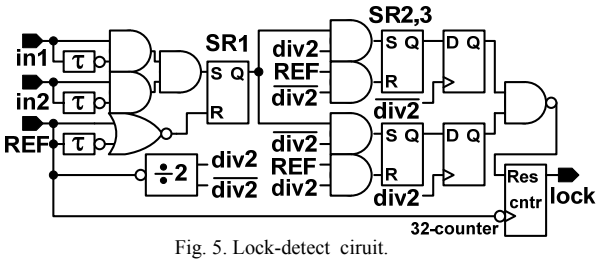


Fig. 5. Lock-detect ciruit.

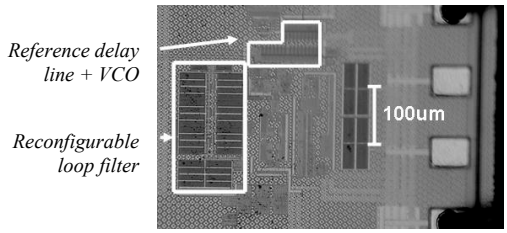


Fig. 6. Die photograph

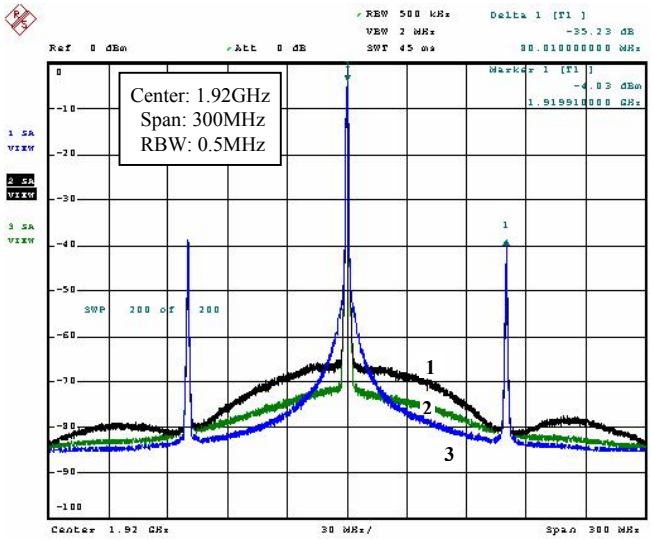


Fig. 7. PLL/DLL output spectrum; the latter with two different REF sources.

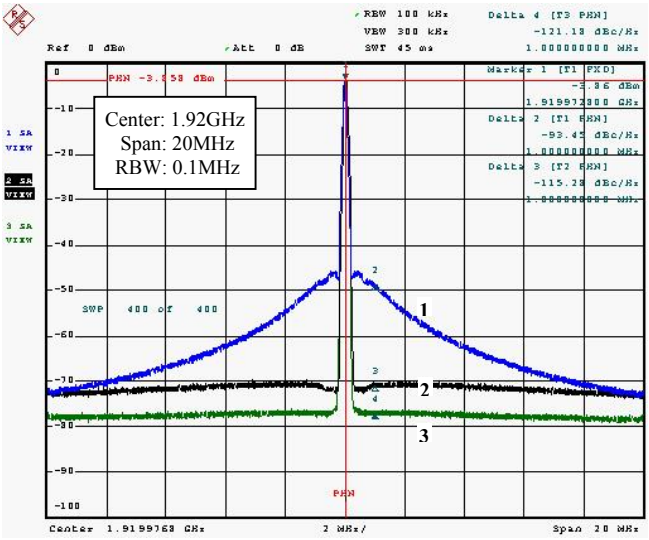


Fig. 8. PLL/DLL output spectrum; the latter with 40/80MHz reference.

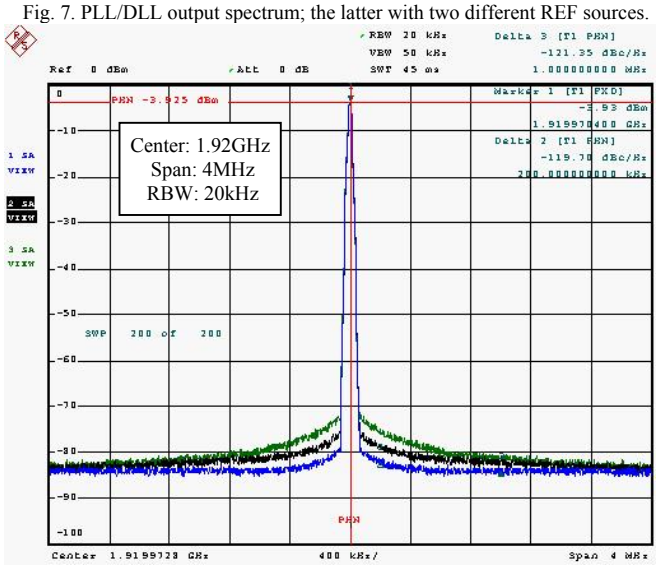


Fig. 9. DLL output spectrum at different loop gains.

f_{REF} MHz	N	f_{OUT} GHz	P_{tot} mW	L(200kHz) dBc/Hz	L(1MHz) dBc/Hz	FOM dB
40	24	0.96	11.6	-122.5	-124.4	-199.1
60	16	0.96	12.4	-122.6	-123.9	-195.6
80	12	0.96	11.9	-125.4	-128.9	-195.9
40	36	1.44	15.3	-118.9	-119.3	-201.3
60	24	1.44	15.4	-121.3	-122.1	-200.2
80	18	1.44	15.6	-122.7	-124.7	-199.0
40	48	1.92	14.4	-115.5	-115.7	-203.2
60	32	1.92	14.6	-118.2	-118.5	-202.3
80	24	1.92	14.9	-119.7	-121.2	-201.2
4	25	0.1	8.6	-117	[1]	-175.6
250	8	2.0	12	-110	[3]	-183.3
30	8	0.24	16	-116	[4]	-169.6
100	8	0.8	15	-122	[5]	-186.4

Fig. 10. Performance summary and comparison.

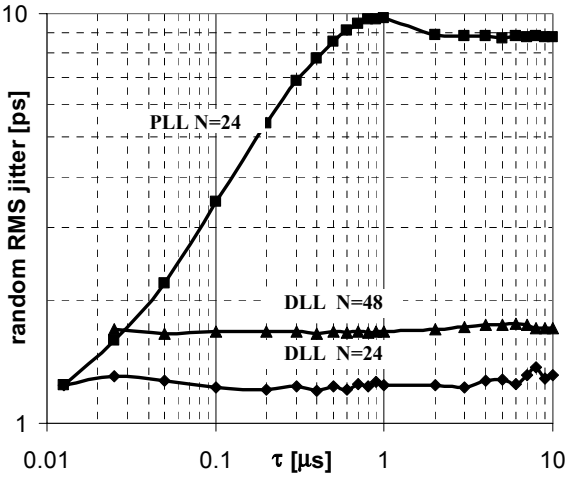


Fig. 11. Random RMS jitter vs. time interval τ; f_{out}=1.92GHz.