Appendix A

PHASE-FREQUENCY DETECTORS AND CHARGE PUMPS

1 PHASE-FREQUENCY DETECTORS

The typical PFD employed in frequency synthesizers is shown in Figure A-1. IN1 and IN2 are the input signals to the PFD and they conventionally represent the reference and feedback signals in the PLL, respectively. The state diagram of the PFD of Figure A-1 is shown in Figure A-2. Three states can be distinguished. Those are denoted state 1, state 0, and state +1. The state diagram can be further illustrated with the aid of a timing diagram.

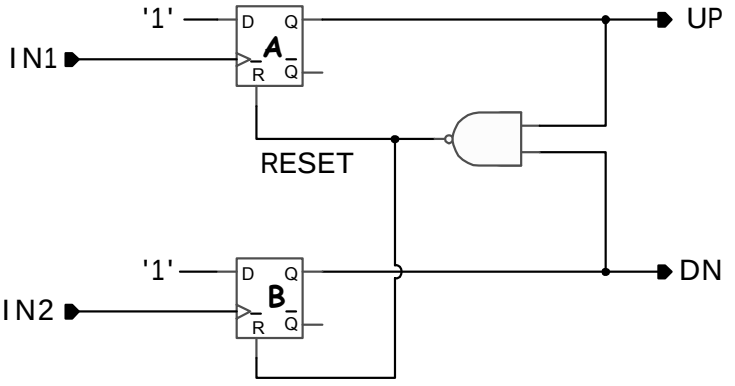


Figure A-1. Phase-Frequency Detector Block Diagram

Three cases can be distinguished. The first case is when IN2 is leading IN1. In this case, the UP goes high with the rising edge of IN1 and is reset to low with the rising edge of IN2. The DN goes high with the rising edge of IN2 and resets instantaneously. The second case IN2 is lagging IN1. In this case, the DN goes high with the rising edge of IN2 and is reset to low with the rising edge of IN1. The UP goes high with the rising edge of IN1 and resets instantaneously. The third case is when IN2 tracks IN1, while the phase-locked loop is locked. All the three cases are illustrated in Figures A-3–A-5.

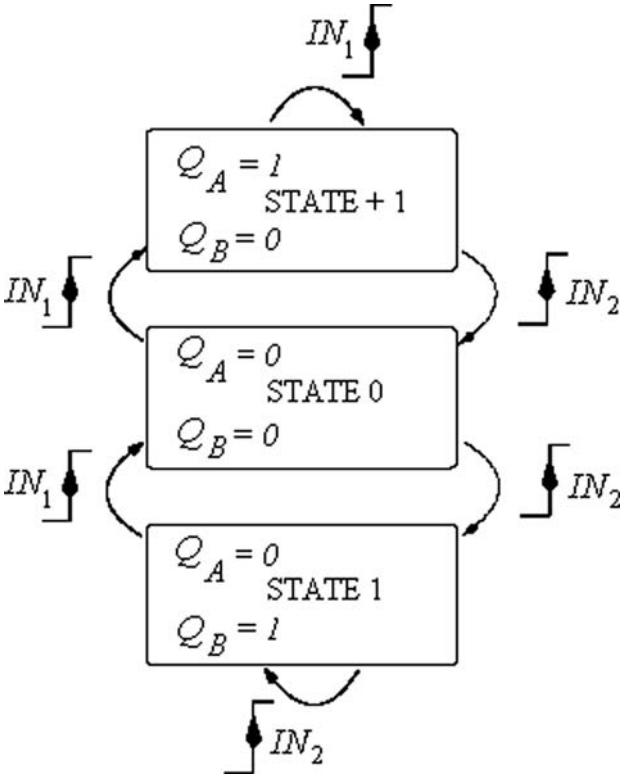


Figure A-2. State Diagram of the PFD in Figure A-1

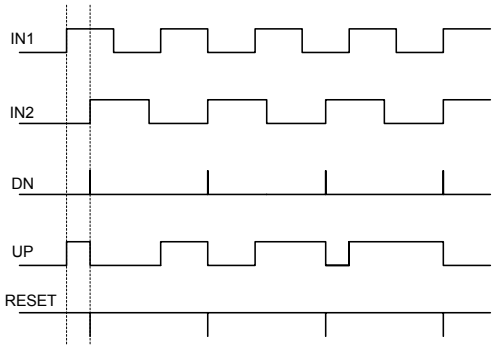


Figure A-3. Timing Diagram of the PFD, Case IN2 Leads IN1

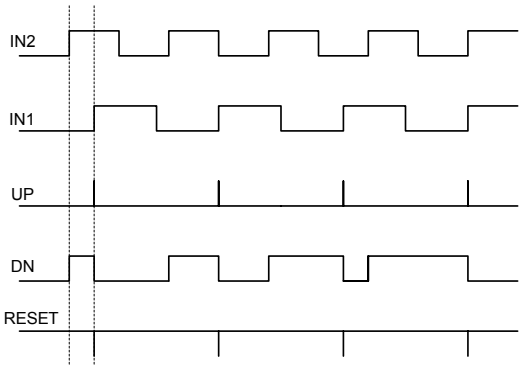


Figure A-4. Case IN2 Lags IN1

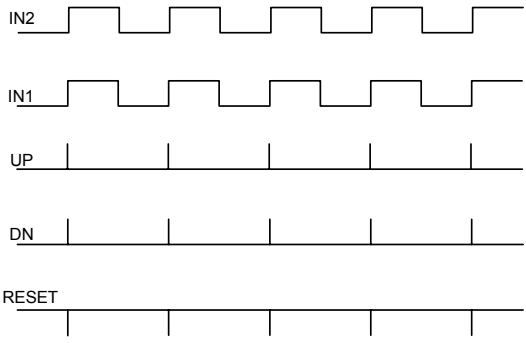


Figure A-5. Case IN2 Tracks IN1

In practice, the UP and DN signals do not reset to low instantaneously but go to zero after a certain delay. This is illustrated in Figure A-6 for the case of Figure A-3.

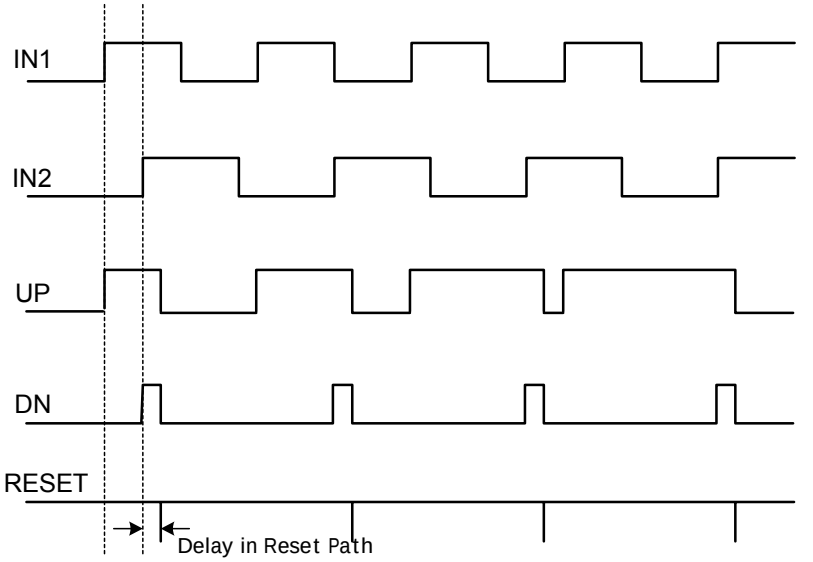


Figure A-6. Practical Timing Diagram of the PFD for IN2 Leading IN1. The Figure is not to Scale as in Reality the RESET Pulse is a Full Pulse and not an Impulse as the Figure might Suggest

One important disadvantage of the PFD described above is the dead zone. Dead zone occurs when IN1 and IN2 are very close and there is not enough delay in the reset path. This causes the UP and DN pulses not to fully turn ON and stay ON for a short time to allow for the CP switches to respond. Too much delay in the reset path, however, can adversely cause additional increase of phase noise due to the inherently noisy current sources that are ON for a longer time.

2 CHARGE PUMP

A behavioral model for a CP is shown in Figure A-7. The CP UP current source sources current into the loop filter whereas the loop filter sinks currents into the DN CP current source. The effect of sourcing and sinking the CP current is also illustrated in Figure A-7.

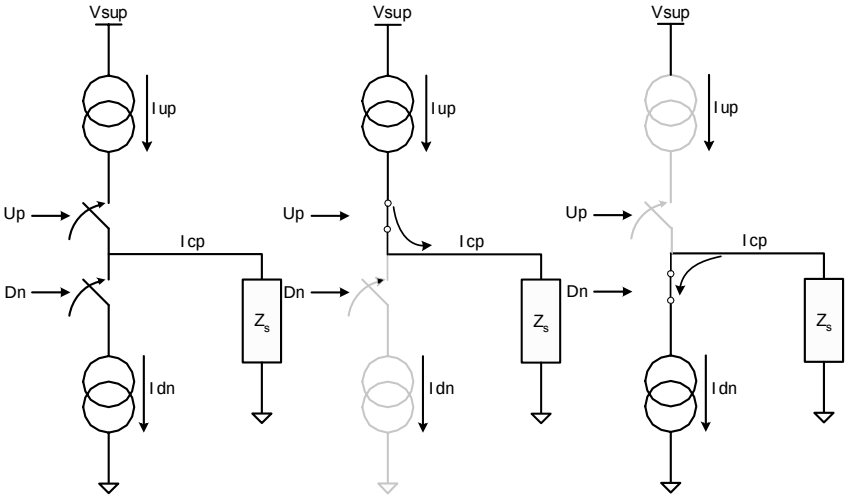


Figure A-7. Charge Pump Sourcing and Sinking Illustration

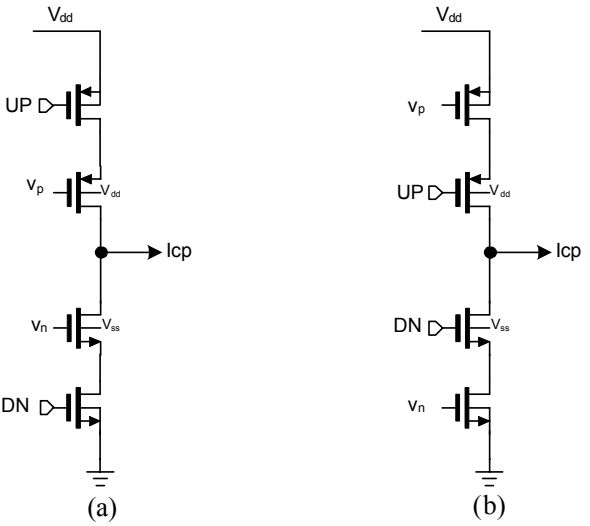


Figure A-8. Single-Ended Charge Pump and its Modification

A conventional single-ended CP circuit is shown in Figure A-8a. Figure A-8b shows a modification of the CP circuit to isolate the feed-through obtained by the switches' UP and DN pulses as the current sources act as buffers for the switches.

There are few issues that need to be considered when designing a CP. These are as follows:

- Output impedance of CP
- Current mismatch between UP and DN parts
- Switching time and switch feed-forward
- Leakage when UP and DN are off (CP $I_{cp} = 0$)

Example charge pump UP, DN, and net currents are shown in Figure A-9. It can easily be seen from the net current that there is a mismatch between the UP and DN currents for tuning voltage values less than 0.3 V and greater than 1.2 V.

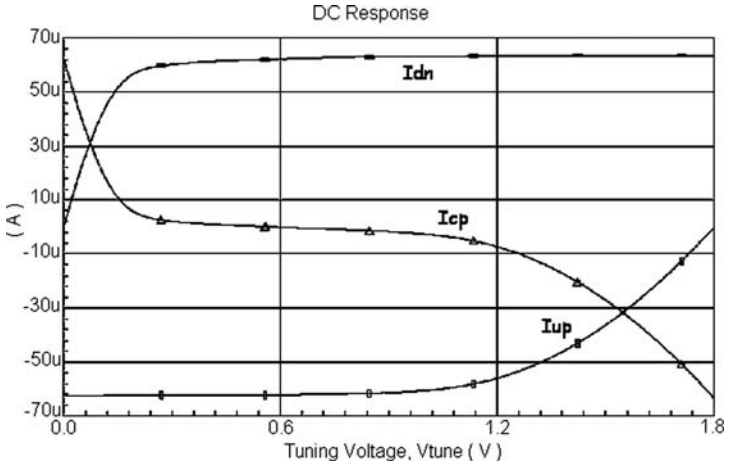


Figure A-9. Example DC Current Curves of the Charge Pump

3 PFD/CP CHARACTERISTICS

The combined PFD/CP performance is of particular interest. The characteristic curve is important because it can be used in system simulation to predict the PLL behavior in presence of the PFD/CP nonlinearity.

Figures A-10 and A-11 show the PFD/CP characteristics and the gain slope variations. Figure A-10 is very useful as it shows the areas where the CP gain has most of the variations for a specific PFD/CP architecture. The more variations in the gain the more susceptible the PLL loop will be to nonlinearities and noise folding. The average current here is the total charge multiplied by the reference frequency (40 MHz), i.e. the measured charge over the sampling period.

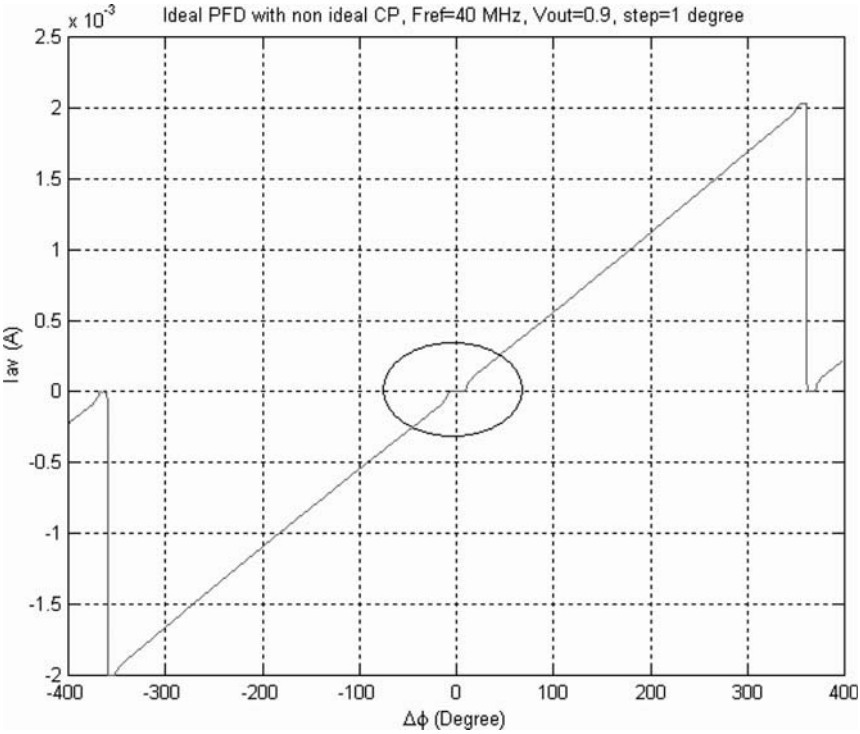


Figure A-10. PFD/CP Characteristic

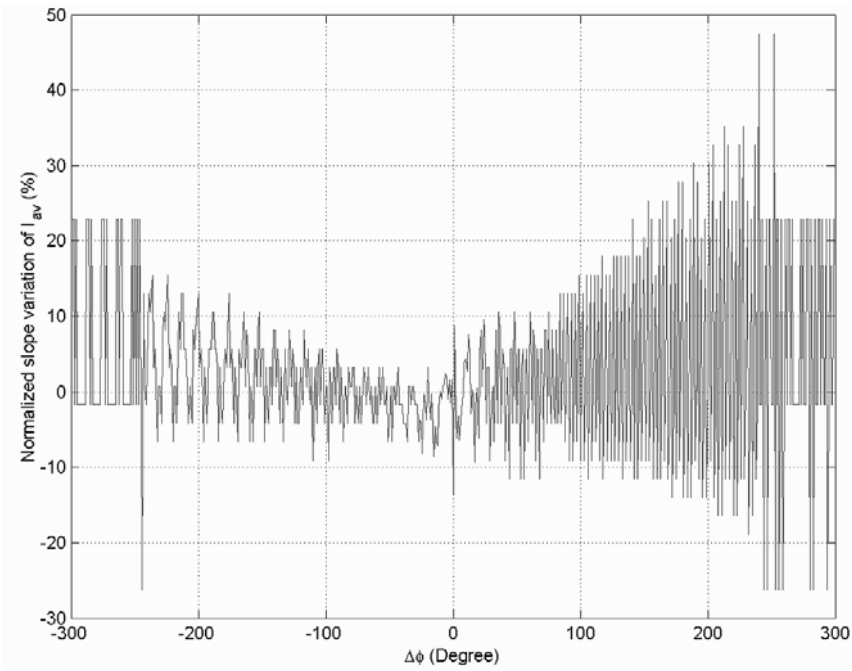


Figure A-11. PFD/CP Gain Slope Variations

Appendix B

CONTROLLED OSCILLATORS

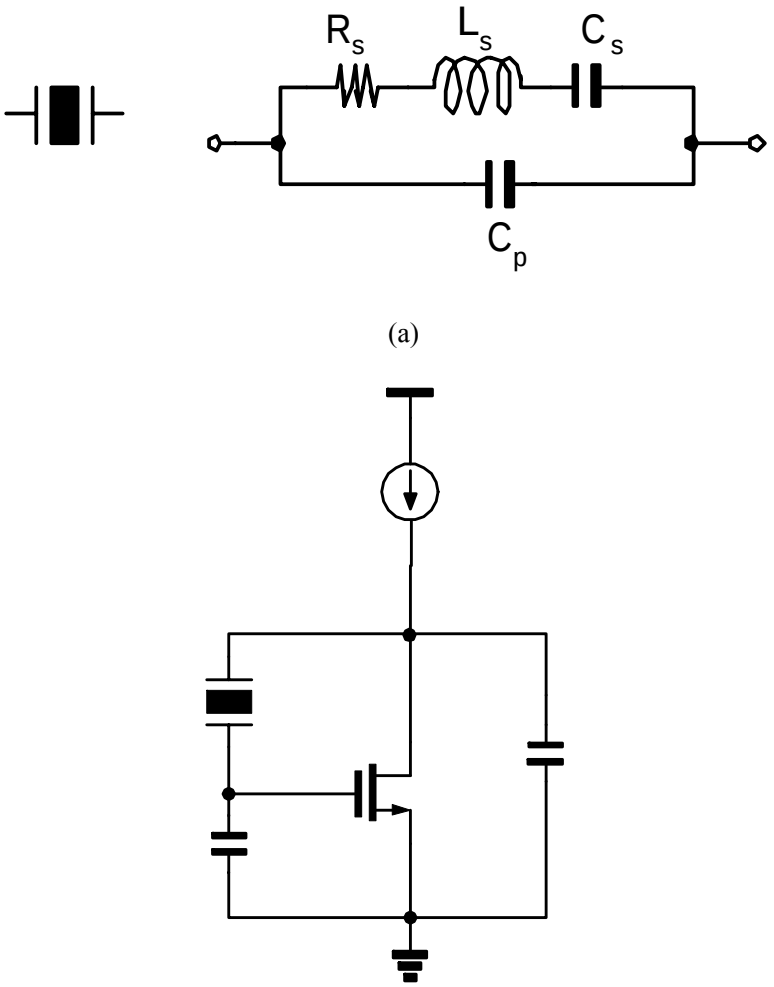
1 REFERENCE OSCILLATORS

Wireless frequency synthesizers have very stringent phase noise requirements. Working those requirements backwards to the various blocks of the PLL, the first block that determines the initial purity of the system is the reference oscillator. The main contributors to the phase noise plateau of the PLL are the reference oscillator, the PFD and the dividers. A low phase noise plateau warrants the usage of a clean and very stable reference oscillator.

1.1 Voltage-Controlled Crystal Oscillator

VCXOs are the most accurate, clean, and stable frequency sources that are used as frequency references in the PLL. The crystal is made of a piezoelectric resonator that is electromechanical in nature. Figure B-1 shows a basic crystal model and a configuration that forms the basis of a VCXO.

The motional capacitance C_s and the motional inductance L_s determine the series resonance of the crystal equation (B.1) as their impedances cancel out. The crystal impedance at the series resonance is approximated to R_s, the motional resistance. This resonant frequency is the operation frequency.



(a)

(b)

Figure B-1. Crystal Oscillator (a) Equivalent Circuit, (b) Simplified VCXO Circuit Implementation

$$f_s = \frac{1}{2\pi\sqrt{L_s C_s}} \tag{B.1}$$

Another crystal resonant frequency is achieved when the parallel capacitance becomes significant. This happens when the inductance

impedance becomes much larger than the capacitance impedance and dominates. The magnitude of the inductive impedance becomes equal to the magnitude of the parallel capacitive impedance, C_p, and cancels out at a parallel resonance frequency equation (B.2).

$$f_p = \frac{1}{2\pi}\sqrt{\frac{C_p + C_s}{L_s C_p C_s}} + \Delta f \tag{B.2}$$

where Δf is the shift in frequency from series to parallel resonance and is determined by the load capacitance, C_L, of the VCXO circuit configuration. This shift in frequency is given by:

$$\Delta f = \frac{C_s}{2(C_L + C_p)} \tag{B.3}$$

1.2 Temperature-Compensated Crystal Oscillator

VCXOs have a superior phase noise characteristic and hence are very suitable for PLL synthesizers with stringent phase noise requirements. However, crystal oscillators exhibit a fundamental drift in frequency with temperature. Equation B.4 [B1] shows frequency drifts due to ambient temperature variations of an AT-cut[1] crystal.

$$\Delta f = [\alpha_1 (T - T_0) + \alpha_2 (T - T_0)^2 + \alpha_3 (T - T_0)^3] f_0 \tag{B.4}$$

where f_0 is the nominal resonant frequency at $T_0 = 25^{\circ}C$ ambient room temperature, α_1, α_2 and α_3 are constants that depend on the physical properties of the crystal and its angle of cut, and T is the ambient temperature.

For highly accurate applications, a temperature-compensated crystal oscillator (TCXO) can be built where by the crystal temperature is kept constant. Compensation is achieved using temperature-dependent circuit elements such as thermistors and negative temperature coefficient (NTC) capacitors. Figure B-2 shows a typical circuit implementation of a TCXO using a colpitts oscillator topology [B2].

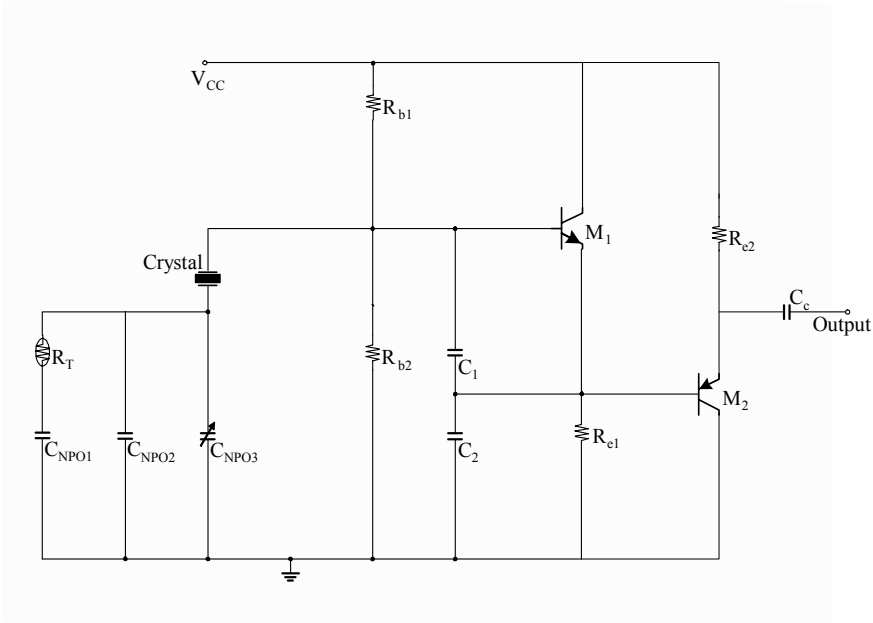


Figure B-2. TCXO Circuit

2 VOLTAGE-CONTROLLED OSCILLATORS

VCO are the heart of PLL frequency synthesizers. The stringent requirement on their spectral purity requires very good design practices and guidelines. Improving phase noise and jitter requires, however, the exploring of new techniques such as the ones cited in [B3–B6]. To design the VCOs with low noise performance, proper noise analysis have is required as presented in the next paragraphs.

2.1 Voltage-Controlled Oscillators: Phase Noise Analysis

Equation B5 shows the SSB power spectral density (PSD) of the total phase noise.

$$L_{\text{total}}\{\Delta\omega\} = 10 \cdot \log\left[\frac{P_{\text{sideband}}(\omega_0 + \Delta\omega, 1Hz)}{P_{\text{carrier}}}\right] (\text{dBc/Hz}) \qquad (B.5)$$

where $P_{\text{sideband}}(\omega_0 + \Delta\omega, 1\text{ Hz})$ represents the SSB power at a frequency offset of $\Delta\omega$ from the carrier over 1 Hz bandwidth. For LC tank oscillators, Leeson's equation, B6, models the phase noise based on a linear time invariant (LTI) system.

$$L(\Delta\omega) = 10.\log\left\{\frac{2FkT}{P_{\text{sig}}}\cdot\left[1+\left(\frac{\omega_0}{2Q\Delta\omega}\right)^2\right]\cdot\left(1+\frac{\Delta\omega_{1/f^3}}{|\Delta\omega|}\right)\right\} \qquad \text{(B.6)}$$

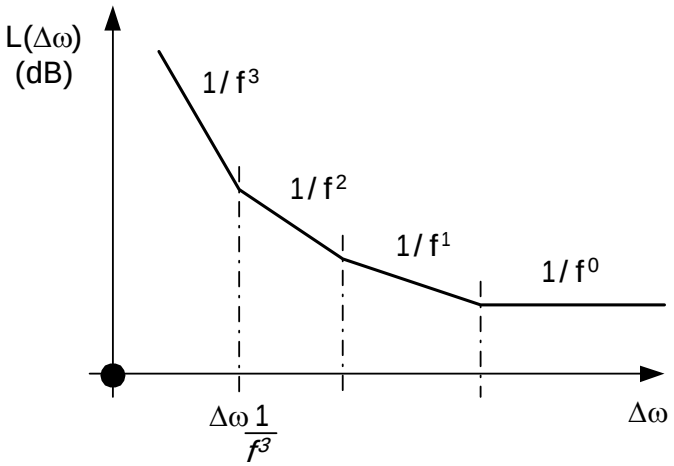


Figure B-3. Typical Curve of the Phase Noise of an Oscillator Versus Offset from Carrier

F is called the device excess noise number, k is Boltzman's constant, T is the absolute temperature, P_{sig} is the average power dissipated in the lossy resistive part of the tank, ω_0 is the oscillation frequency, Q is the loaded quality factor of the tank, $\Delta\omega$ is the offset from the carrier and $\Delta\omega_{1/f^3}$ is the frequency of the corner between $1/f^3$ and $1/f^2$ region (as shown in Figure B-3). This equation, however, makes use of F and $\Delta\omega_{1/f^3}$ that are fitting factors and cannot be calculated beforehand.

Using LTI method [B5] treats oscillator as a feedback system and considers each source noise as an input $X(j\omega)$ (Figure B-4). The phase noise at the output, $Y(j\omega)$, is made of the noise contributions of various elements in the circuit and the noise shaped by the feedback, equation (B.7).

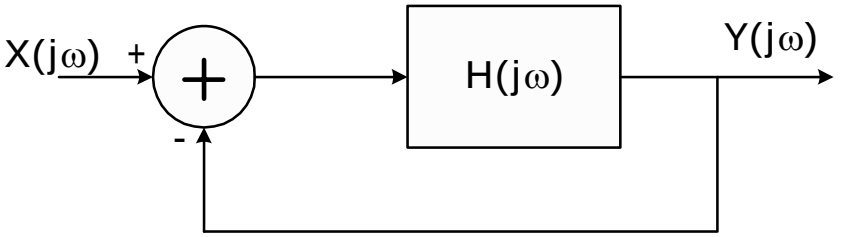


Figure B-4. Oscillator as a Linear System

$$\left|\frac{Y}{X}\left[\mathrm{j}(\omega_0+\Delta\omega)\right]\right|^2=\frac{1}{(\Delta\omega)^2\left|\frac{\mathrm{d}H}{\mathrm{d}\omega}\right|^2} \tag{B.7}$$

B7 can be used to get the output noise PSD. This approach, however, is based on linear analysis and cannot be used to predict realistic phase noise of real VCOs.

Linear time variant (LTV) method used in [B6], however, is more useful for predicting and optimizing phase noise in oscillators. This analysis makes use of a special function, ISF that describes how much phase shift results from applying a unit impulse at any point in time. Equation (B.8) shows the phase shift due to applying a unit impulse.

$$h_\Phi(t,\tau)=\frac{\Gamma(\omega_0 t)}{q_{\max}}u(t-\tau) \tag{B.8}$$

where $\Gamma(\omega_0 t)$ is the ISF function of the output and $q_{\max}$ is the maximum charge offset across the capacitor. The total excess phase due to a noise current can be described by the following equation:

$$\theta(t)=\int_{-\infty}^{+\infty}h_\Phi(t,\iota)i(\iota)d\iota=\int_{-\infty}^{t}\frac{\Gamma(\omega_0 \iota)}{q_{\max}}i(\iota)d\iota \tag{B.9}$$

The phase can hence be converted to voltage to get the SSBPSD, equation (B.10).

$$L\{\Delta\omega\} = 10 \cdot \log\left(\frac{\frac{\overline{i_n^2}}{\Delta f} \sum_{n=0}^{+\infty} c_n^2}{8 q_{\max}^2 \Delta\omega^2} \right) \tag{B.10}$$

where, $\frac{\overline{i_n^2}}{\Delta f}$ is the power spectral density of the input noise current, c_n is the coefficient of the Fourier transform of the ISF function, and $\Delta\omega$ is the frequency shift from the carrier frequency.

2.2 VCO Design Methodology

Figure B-5 shows the steady state parallel LC oscillator model. The tank loss is represented by g_{tank}, and the effective negative conductance of the active devices, required to compensate for the tank losses is represented by $-g_{acitve}$.

Typically, LC oscillators operate in two different modes: namely current- and voltage-limited modes [B7]. In the current-limited mode, the tank amplitude, $V_{tank,}$ linearly increases with the bias current according to $V_{tank} = I_{bias}/g_{tank}$ until the oscillator enters the voltage-limited mode where the amplitude is limited to V_{limit}, which is determined by the supply voltage. V_{tank} can be expressed in Equation (B.11) where I_{bias} is the independent variable.

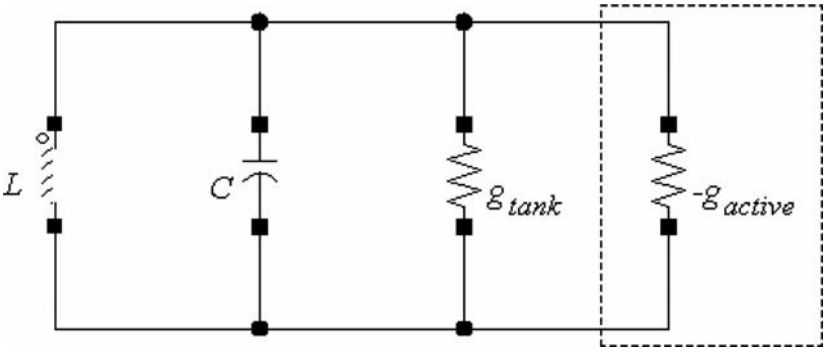


Figure B-5. Steady State parallel LC oscillator model

$$V_{\text{tank}} = \begin{cases} \dfrac{I_{\text{bias}}}{g_{\text{tank}}} & (I-Limited) \\ V_{\text{limit}} & (V-Limited) \end{cases} \tag{B.11}$$

Using the tank inductance, L, as the independent variable, we can express equation V_{tank} in equation B.12 as:

$$V_{\text{tank}}^2 = \frac{2E_{\text{tank}}}{C} = 2E_{\text{tank}}\omega_0^2 L \tag{B.12}$$

where E_{tank} is the tank energy defined in equation B.13 as:

$$E_{\text{tank}} = \frac{CV_{\text{tank}}^2}{2} \tag{B.13}$$

and, $\omega_0 = 1/\sqrt{LC}$ is the oscillation frequency. The tank amplitude grows with L for a given E_{tank} and ω_0. This is referred to as inductance-limited mode where the inductance is limited by the given size or available area. The V – limit is constrained by the supply voltage; hence V_{tank} can be expressed as:

$$\mathrm{V}_{\text{tank}}^2 = \begin{cases} 2\mathrm{E}_{\text{tank}}\omega_0^2\mathrm{L} & (\mathrm{L}-\text{Limited}) \\ \mathrm{V}_{\text{limit}}^2 & (\mathrm{V}-\text{Limited}) \end{cases} \tag{B.14}$$

2.2.1 VCO Design

Figure B-6 shows the building block of the used LC VCO core. In addition to the inductor dimensions, Figure B-7, the core can be optimized by optimizing the MOS W/L dimensions, the varactor tuning range, and the bias current.

Using the parallel oscillator model on the Figure B-7, the tank loss g_{tank}, the negative conductance $-g_{\text{active}}$, the tank inductance L_{tank}, and the capacitance C_{tank} are given by:

$$2g_{\text{tank}} = g_{\text{op}} + g_{\text{v}} + g_{\text{L}} \tag{B.15}$$

$$2g_{\text{active}} = g_{\text{mp}} \tag{B.16}$$

$$L_{\text{tank}} = 2L \tag{B.17}$$

$$2C_{\text{tank}} = C_{\text{PMOS}} + C_{\text{L}} + C_{\text{v}} + C_{\text{load}} \tag{B.18}$$

where g_{L} and g_{v} are the effective parallel conductance of the inductors and varactors, respectively.

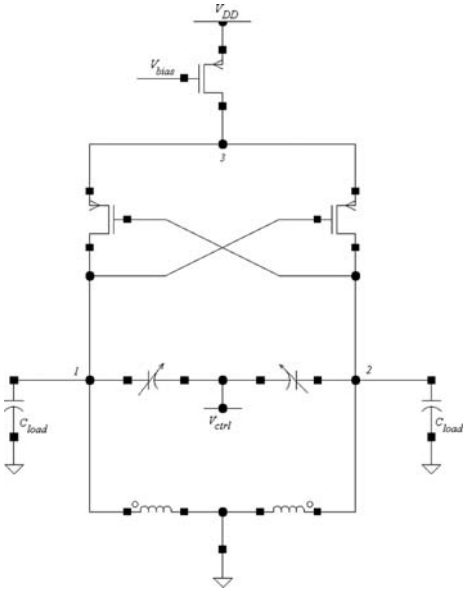


Figure B-6. VCO Core Schematic

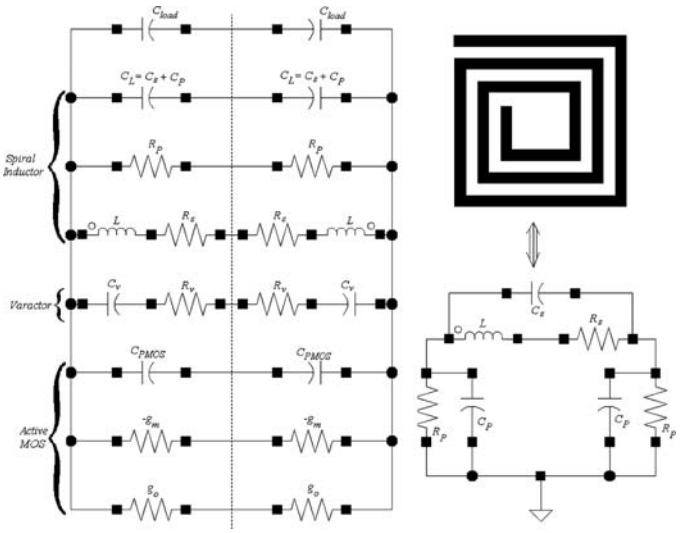


Figure B-7. Equivalent Oscillator Model and Symmetrical Spiral Inductor Model

The primary parameter in the oscillator specifications requiring optimization is the phase noise. This is followed by the power consumption, frequency tuning range, start-up conditions, tank amplitude, and diameter of spiral inductor.

Derived from the system power budget, the maximum power constraint is imposed in the form of maximum bias current $I_{\max}$ drawn from a given supply voltage, i.e.

$$I_{\text{bias}} \le I_{\max} \tag{B.19}$$

The tank amplitude is required to be large, $V_{\text{tank,min}}$ is to provide a large enough voltage swing to drive the next stage:

$$V_{\text{tank}} = \frac{I_{\text{bias}}}{g_{\text{tank,max}}} \ge V_{\text{tank,min}} \tag{B.20}$$

The tuning range of the oscillation frequency needs to be maximized for greater frequency coverage, hence:

$$L_{\text{tank}}\, C_{\text{tank, min}} \le \frac{1}{\omega_{\max}^2}, \textit{ Max. tuning frequency} \tag{B.21}$$

$$L_{\text{tank}}\, C_{\text{tank, max}} \geq \frac{1}{\omega_{\text{min}}^2},\ \textit{Min. tuning frequency} \tag{B.22}$$

To guarantee a reasonable start-up condition with a small-signal loop gain of at least α_{min}, the worst case condition can be expressed in equation (B.23), where $g_{\text{tank,max}}$ guarantees start-up in the worst case, hence:

$$g_{\text{active}} \geq \alpha_{\text{min}} g_{\text{tank,max}} \tag{B.23}$$

The final spec. is size or area which needs to be kept small. Being the most area-consuming component in the VCO core, the inductor has its size specified so as not to exceed a certain value and hence its diameter is specified as d_{max}, i.e.

$$d \leq d_{\text{max}} \tag{B.24}$$

For a given chip area or specified d_{max}, the inductance, L, is also constrained and so is the tank amplitude as shown earlier in equation (B.14).

2.2.2 Phase Noise Optimization

Using the LTV method of analysis, phase noise is given by:

$$L\{f_{\text{off}}\} = \frac{1}{8\pi^2 f_{\text{off}}^2} \cdot \frac{1}{q_{\text{max}}^2} \cdot \sum_{\text{n}} \left(\frac{\overline{i_{\text{n}}^2}}{\Delta f} \cdot \Gamma_{\text{rms,n}}^2 \right) \tag{B.25}$$

where f_{off} is the offset frequency from the carrier frequency. $\frac{\overline{i_n^2}}{\Delta f}$ represents the equivalent differential noise PSD due to drain current noise, inductor noise, and varactor noise, expressed as:

$$\frac{\overline{i_{M,d}^2}}{\Delta f} = 2kT\gamma(g_{d0,n} + g_{d0,p}) \tag{B.26}$$

$$\frac{\overline{i_{\text{ind}}^2}}{\Delta f} = 2kTg_{\text{L}} \tag{B.27}$$

$$\frac{\overline{i_{\text{var}}^2}}{\Delta f} = 2kTg_{\text{v,max}} \tag{B.28}$$

where γ is approximately $\frac{2}{3}$ and $\frac{5}{2}$ for long- and short-channel transistors, respectively. It has been proven [B8] that drain current noise is predominantly amongst the three noise sources. Taking only the drain current noise term into consideration in B.25, and replacing $q_{\max}$ by $\frac{V_{\text{tank}}}{L_{\text{tank}}\omega^2}$, and $g_{\text{d0}} = \frac{2I_{\text{drain}}}{L_{\text{channel}}E_{\text{sat}}}$ for short-channel transistors, ($\Gamma^2_{\text{rms}} = 1/2$ is used for pure sinusoidal waveform), the phase noise can be expressed as:

$$L\{f_{\text{off}}\} \propto \begin{cases} L^2 g_L^2 / I_{\text{bias}} & \text{(L – Limited)} \\ L^2 I_{\text{bias}} / V_{\text{supply}}^2 & \text{(V – Limited)} \end{cases} \tag{B.29}$$

B.29 states that for a given bias current, the phase noise rises with increasing L in the voltage-limited mode, hence once the V-limit is reached any excess in inductance, L, will worsen the phase noise. Additionally, for a given inductance, increasing the bias current translates to an increase of the phase noise in the voltage-limited mode, inducing power wastage. For a typical on-chip spiral inductor, the minimum effective parallel conductance g_{L} decreases with an increasing inductance; the factor $L^2g^2_{\text{L}}$ also increases. Thus, for a given I_{bias}, the phase noise increases with the inductance in the inductance-limited mode and hence a smaller inductance results in a better phase noise. So the design needs to be based on finding the smallest inductor that satisfies both the tanks amplitude and start-up condition for the maximum allowable bias current allowed by the current budget.

Here is a summary of the described design methodology:

- Set the bias current to I_{max}, and pick an initial guess for inductance value minimizing g_L.
- Plot C versus W of the active device for the chosen inductance
- If there are more than one possible points on the graph, reduce the inductance and replot until the possible region reduces to a single point. That optimum point represents the optimum C and W, corresponding to the optimum inductance yielding the optimum phase noise. A summary of constraint and design optimization methodology is given in Figure B-8 [B9].

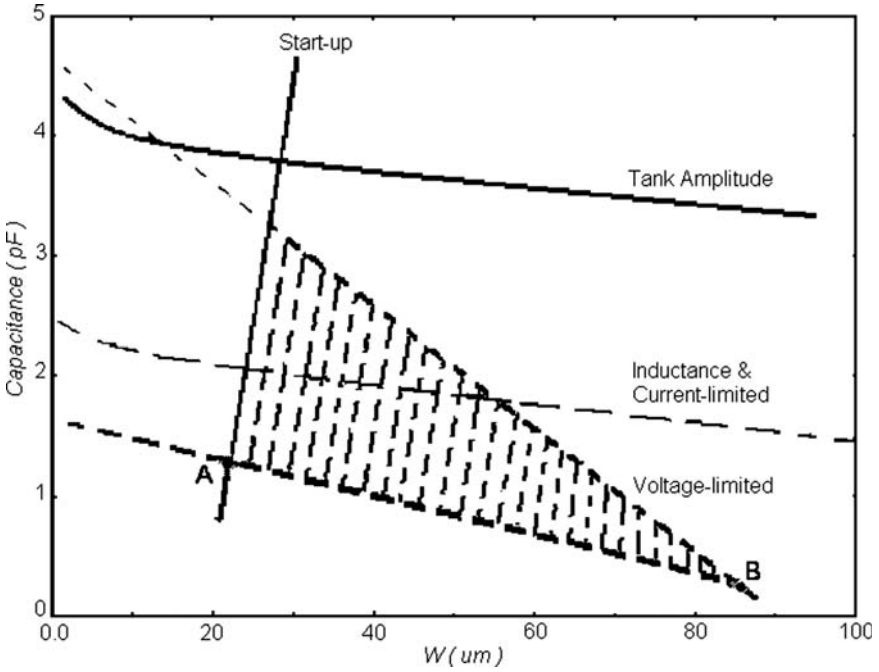


Figure B-8. C–W plane, Summary of Constraint and Design Optimization Methodology

REFERENCES

[B1] Frerking and E. Marvin, *Crystal Oscillator Design and Temperature Compensation*, Van Nostrand, New York, 1978.

[B2] M.A. Haney, "Design Technique for Analog Temperature Compensation of Crystal Oscillators", Master thesis, Faculty of the Virginia Polytechnic Institute and State University, Blacksburg, VI, 2001.

[B3] T.C. Weigandt, B. Kim, and P.R. Gray, "Analysis of Timing Jitter in CMOS Ring Oscillators," Proceedings of ISCAS, June 1994.
[B4] J. McNeill, "Jitter in Ring Oscillators," *IEEE Journal of Solid-State Circuits*, 32, pp. 870–879, June 1997.
[B5] B. Razavi, "A Study of Phase Noise in CMOS oscillators," *IEEE Journal of Solid-State Circuit*, 31, pp. 331–343, Mar. 1996.
[B6] A. Hajimiri and T.H. Lee, "A General Theory of Phase Noise in Electrical Oscillators," *IEEE Journal of Solid-State Circuits*, 33, pp. 179–194, Feb. 1998.
[B7] D.B. Leeson, "A Simple Model of Feedback Oscillator Noises Spectrum," Proceedings on *IEEE*, 54, pp. 329–330, Feb. 1966.
[B8] D. Ham and A. Hajimiri, "Concepts and Methods in Optimization of Integrated LC VCOs," *IEEE Journal of Solid State Circuits*, 36, pp. 896–909, June 2001.
[B9] F. Fang and K. Phang, "Phase Noise Analysis of VCO and Design Approach to LC VCOs," Term Paper University of Toronto, pp. 3–14, 2001.

Appendix C

PHASE NOISE

Analysis

1 CALCULATION OF GLOBAL PHASE ERROR FROM L(f)

A signal $s(t)$ having a peak amplitude A, a radian frequency ω_c, and a periodic phase modulation with peak amplitude $\Delta\phi_{pk}$ and radian frequency ω_m is given by:

$$s(t) = A\cos(\omega_c t + \Delta\phi_{pk}\sin\omega_m t) \tag{C.1}$$

Using trigonometric identities,

$$s(t) = A\cos(\omega_c t)\cos(\Delta\phi_{pk}\sin\omega_m t) - A\sin(\omega_c t)\sin(\Delta\phi_{pk}\sin\omega_m t) \tag{C.2}$$

If the phase modulation index is small, $\Delta\phi_{pk} << 1$, then the second cosine term approaches unity and the second sine term approaches its argument.

$$\begin{aligned} s(t) &\approx A\cos(\omega_c t) - A\Delta\phi_{pk}\sin(\omega_c t)\sin(\omega_m t) \\ &= A\cos(\omega_c t) - \frac{A\Delta\phi_{pk}}{2}[\cos(\omega_c - \omega_m)t - \cos(\omega_c + \omega_m)t], \end{aligned} \tag{C.3}$$

which is a single carrier tone having two sideband tones with relative amplitude ($\Delta\phi_{pk}/2$), spaced by ω_m from the carrier.

Note that the peak phase error is related to the amplitude and power of the phase modulating signal. Similarly, the RMS phase error is given by the RMS amplitude of the phase modulating signal,

$$\Delta\phi_{rms} = \frac{\Delta\phi_{pk}}{\sqrt{2}} \tag{C.4}$$

Single-sideband phase noise (SSBN) is a measure of the power in a sideband region relative to the carrier power. The sideband noise ratio (SBNR) is defined numerically for the sideband tone as:

$$\text{SBNR} = \frac{\textit{power in sideband}}{\textit{power in carrier}} = \left(\frac{A\Delta\phi_{pk}}{2}\right)^2 \cdot (A)^{-2} = \left(\frac{\Delta\phi_{pk}}{2}\right)^2 = \frac{\Delta\phi_{rms}^2}{2} \tag{C.5}$$

SSBN, expressed in dBc/Hz is the most common measure of phase noise, describing the power in a 1-Hz bandwidth relative to the carrier at an offset frequency f from the carrier. The script-L symbol is used.

$$L(f) = 10\log_{10}\left(\frac{\Delta\phi_{rms}^2(f)}{2}\right) \tag{C.6}$$

where the variable (f) indicates the power is measured in a 1-Hz bandwidth at an offset f Hz from the carrier. $L(f)$ is a convenient term because it can be measured directly using a spectrum analyzer or similar setup in the laboratory.

We can also calculate the total RMS phase error contribution for phase noise in a span of frequencies offset from the carrier. In that case the sideband noise power is integrated over the bandwidth of interest.

$$\Delta\phi_{rms} = \sqrt{\int_{f_1}^{f_2} \Delta\phi_{rms}^2(f)\mathrm{d}f} = \sqrt{2\int_{f_1}^{f_2} 10^{L(f)/10}\mathrm{d}f} \tag{C.7}$$

This is the relationship between total, or "global," RMS phase error and the phase noise spectrum $L(f)$. The equation returns the RMS phase error in radians.

As an example, the GSM system requires a total global RMS phase error of 5° degree symbol. To allow margin for other transmit path impairments, we can arbitrarily set a specification of less than 2° for the PLL and VCO. Using the above relationship, and assuming that the in-band phase noise is constant and will dominate the result, we find:

$$L(f) = 10\log_{10}\left(\frac{\Delta\phi_{\text{rms}}^2}{2 \cdot BW}\right) \tag{C.8}$$

where BW is the loop bandwidth. For a 150 kHz bandwidth and 2° error, $L(f)$ must be less than 84 dBc/Hz in-band.

2 PHASE NOISE AND PHASE MODULATION

Random noise in the vicinity of a signal causes both amplitude and phase noise modulation of that signal.

A single noise sideband (at any offset from the carrier), produces both amplitude and phase modulation and can be represented vectorially (Figure C-1).

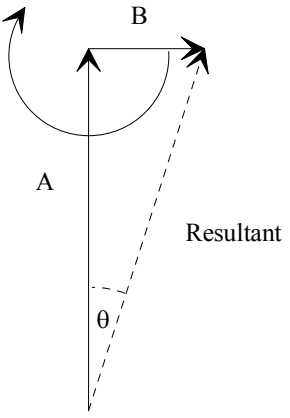


Figure C-1. Single Sideband and Carrier

The resultant vector changes in both amplitude and phase, the amplitude varying from A − B to A + B, with a modulation depth of B/A multiplied by 100%. The phase varies by ± Φ, where Φ = B/A, in radians, provided B is small compared to A.

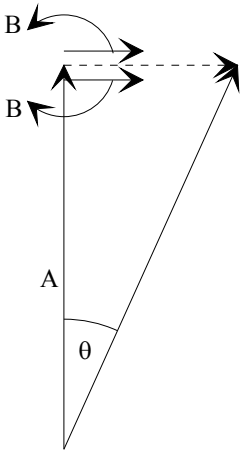


Figure C-2. Carrier and Phase Modulating Sidebands

Two sidebands (Figure C-2), with equal initial phase, each in quadrature with the carrier A, but at equal frequency offsets at either side of it can be represented as positively rotating vectors, producing only phase modulation. The peak phase variation is ±2B/A. This is representative of what occurs when we measure the noise of signals where phase noise dominates, which is the usual case (the natural limiting of the oscillator in a phase-locked loop restricts the amplitude noise, but not the phase noise). The single-sideband power level is thus $20\log_{10}$ (B/A) dBc.

At a particular offset frequency, the sum vector is:

$$e(t) = A\cos(\omega_c t) + B\cos(\omega_c - \omega_m)t + B\cos(\omega_c + \omega_m)t \qquad \text{(C.9)}$$

If we assume that 2B/A << π/2, then we can arrive at a value for $e(t)$, representative of the phase modulation of this carrier at this particular frequency offset, m:

$$e(t) = A\cos(\omega_c t + \tfrac{2B}{A}\sin(\omega_m t)) \qquad \text{(C.10)}$$

where 2B/A $\sin(\omega_m t)$ is the phase modulation.

3 RMS PHASE ERROR FROM PHASE NOISE

The output of a PLL is a sinusoid with phase noise. The ultimate goal is to calculate from this phase noise spectrum the RMS phase error of this sinusoid. First considering a pure sinusoid with white noise, such as that associated with a resistor, in communications terminology there is an associated signal-to-noise ratio, (S/N), and bandwidth. Here *C* will be used to represent the carrier replacing *S*, the sinusoid, in the commonly accepted 1 Hz bandwidth of SSB phase noise, L(f).

If we look at a 1 Hz bandwidth of this noise power at an offset frequency f_m, from the carrier, and calculate the effect of this small amount of noise, the effect of all the noise in the total bandwidth can be found. The relationship between the vector and spectral representations of the carrier with noise is shown in Figure C-3;

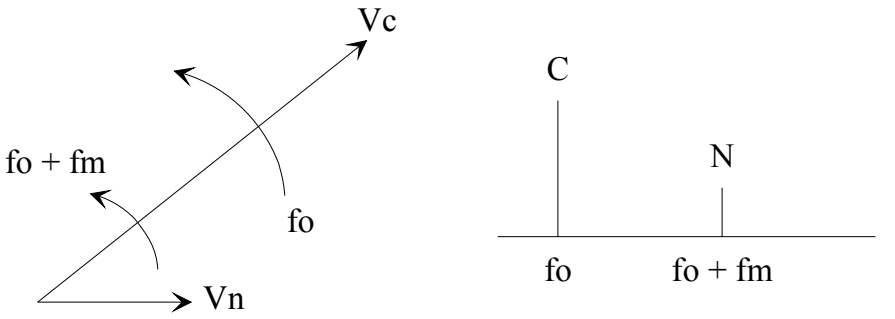


Figure C-3. Single Sideband and Carrier

where *C*, *N* are the powers V_c^2/R , V_n^2/R .

Calculation of the ratio of carrier to noise is as follows:

$$\theta_{pk} = \tan^{-1}\left(\frac{V_n}{V_c}\right) \approx \frac{V_n}{V_c}, \text{ in power } \sqrt{\frac{N}{C}} \qquad \text{(C.11)}$$

The RMS phase deviation is:

$$\phi = \frac{\theta_{pk}}{\sqrt{2}} = \sqrt{\frac{N}{2C}} \qquad \text{(C.12)}$$

In the case of two sidebands at ± f_m with respect to f_o, the incoherent random phase noise contributions add in a sum of squares fashion (Figure C-4).

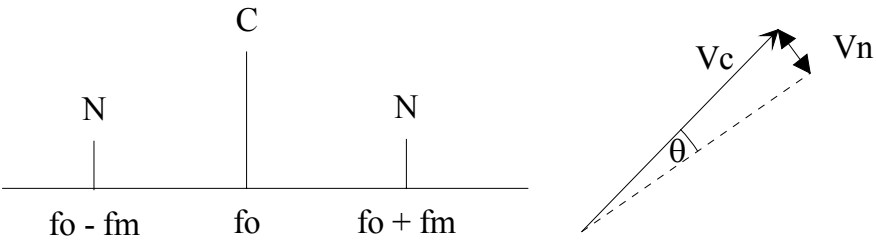


Figure C-4. Carrier and Phase-Modulating Sidebands

$$V_{n\text{Total}} = \sqrt{V_{n1}^2 + V_{n2}^2} = \sqrt{2V_n^2} = \sqrt{2}\,V_n \tag{C.13}$$

$$\text{Thus } \theta_{\text{pk}} = \frac{\sqrt{2}V_n}{V_c} = \sqrt{\frac{2N}{C}} \text{ in RMS} \phi = \frac{\theta_{\text{pk}}}{\sqrt{2}} = \sqrt{\frac{N}{C}} \text{ rad RMS} \tag{C.14}$$

Before looking at the total noise in a given bandwidth, it is important to note what kind of noise spectrum is being considered. Superimposed noise vectors, from such things as amplifiers or thermal noise sources, create both AM and FM sidebands. The diagrams above considered this case. However, if the spectrum is phase noise AM noise is insignificant (because of the self-limiting action of the VCO), requiring the above equations to be altered from noise, N, to phase noise N_P. Since the noise power is split evenly into AM and PM noise, N has twice the power of N_P. Thus, the equations can be altered with the relationship $N = 2N_P$.

For white noise from amplifiers or resistors, the noise can be viewed as many single sidebands and integrated over the required bandwidth.

$$\text{Phase Error} = \frac{180^0}{\pi}\sqrt{\int_0^b \frac{N}{C}\,df} = \frac{180^0}{\pi}\sqrt{\int_{-b}^b \frac{N}{2C}\,df} \tag{C.15}$$

For the case of a phase noise spectrum:

$$\text{Phase Error} = \frac{180^0}{\pi}\sqrt{\int_{-b}^b \frac{N_P(f)}{C}\,df} \text{ Deg RMS} \tag{C.16}$$

Thus, it is possible to integrate the normal SSB phase noise spectrum, which expresses phase noise as a noise power at a particular offset from the carrier, with respect to the total carrier noise power.

4 RESIDUAL FM

Residual FM is a measure of frequency instability related to $S\phi(f)$ (the spectral density of phase fluctuations), expressing the total RMS frequency deviation within a specified bandwidth. Commonly used bandwidths are 50 Hz–3 kHz, 300 Hz–3 kHz and 20 Hz–15 kHz. Only the short-term frequency instability occurring at rates within the bandwidth is indicated and no information regarding relative instability rates is conveyed. The presence of large spurious signals at frequencies near the carrier frequency can greatly exaggerate the measured level of residual FM, since the spurious signals are detected as FM sidebands.

$$S_{\Delta f}(f) = \frac{\Delta f_{\text{rms}}^2(f)}{B} = \frac{f^2 S_\phi(f)}{B} \tag{C.17}$$

$$\text{res FM} = \int_a^b \sqrt{S_{\Delta f}(f)} = \int_a^b \sqrt{\frac{f^2 S_\phi(f)}{B}} \ \text{[Hz]} \tag{C.18}$$

To convert the quantity $S\phi(f)$ to $L(f)$ (SSBN), the following can be used.

$$\text{L}(f) = \frac{\textit{Power density}\ (\text{one phase modulation sideband})}{\textit{Carrier Power}} \left[\frac{\text{dBc/Hz}}{\text{Hz}}\right] \tag{C.19}$$

$$L(f) = \frac{S_\phi(f)}{2} \tag{C.20}$$

$L(f)$ is the ratio of the power in one phase-modulated sideband per Hz, to the total signal power.

Appendix D

FREQUENCY DIVIDERS

1 REFERENCE DIVIDER

The reference divider operates at low frequencies (frequencies below 500 MHz for 0.18 um CMOS technology). This type of divider is constructed as a programmable counter designed in standard CMOS standard cell logic. This divider usually divides the reference frequency down to the sampling frequency of the PFD.

There are two types of counters used as reference frequency dividers. These are either synchronous or asynchronous. Both dividers will be described below.

1.1 Synchronous Dividers

Synchronous frequency dividers can be constructed as pseudorandom binary sequence (PRBS) counters. A 6-bit PRBS-based counter is shown in Figure D-1.

These dividers are formed from a shift register of length N with some taps (either 2 or 4) fed to a modulo-2 adder whose output is connected to the input of the shift register. If the correct taps are used, the shift register will clock through a PRBS of length 2^{N-1}. It also has a forbidden state of all zeros; a state if it falls into then it never leaves.

The sequence of 2^{N-1} states does not include the all-zeros state. Therefore, the divider cannot get into the forbidden state from its normal counting mode. A problem can possibly occur at switch-on, which may yield all zeros, or if the counter is accidentally loaded with all zeros. A good way to protect the divider from this state is to fix the

terminal count as a code which will be zero. This makes the bit which is "1" a "don't care" bit. With this arrangement, terminal count is detected at the defined value and for the forbidden all-zeros state. The terminal count forces a parallel load on the shift register as part of the normal divider operation, so it will load a permissible code if the forbidden state occurs.

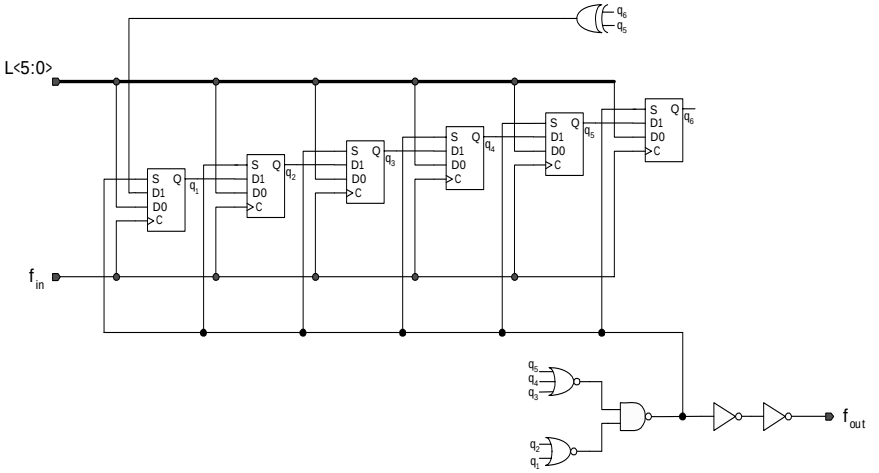


Figure D-1. A 6-bit PRBS Synchronous Divider

These dividers are fast, simple, and fully programmable. They are nearly as fast and simple for big division ratios as for small ones. The disadvantages are that they count in pseudorandom fashion (not binary) so a lookup table is required for division ratios versus parallel load value. They also consume more power than ripple counters because all the D-type flip-flops used are clocked at the input clock-rate. If designed carefully however, the maximum frequency of operation is not much less than that of a single divide by two using the same D-type flip-flop.

1.2 Asynchronous Reference Frequency Divider

A simple 6-bit programmable reference divider is shown in figure D-1. Figure D-2 shows an asynchronous divider comprised of six toggle flip-flop (TFF) stages. Each TFF can be designed from a simple set/reset D-type flip-flop and some logic (see Figure D-3).

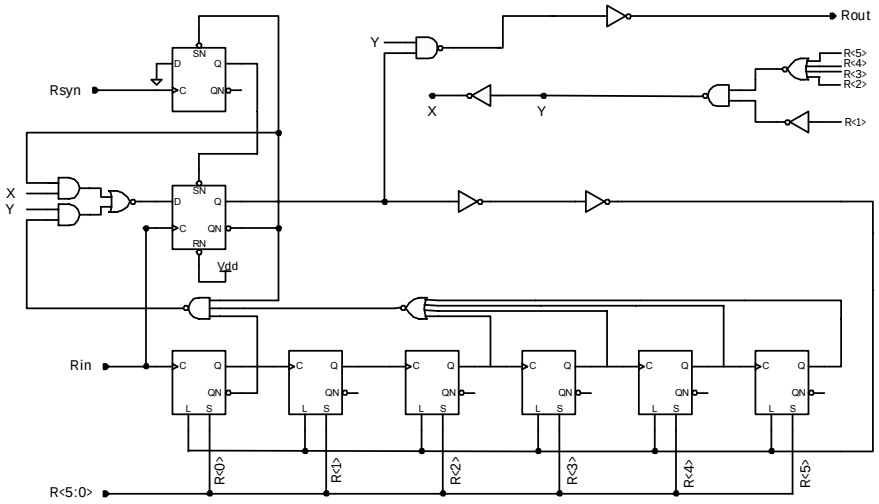
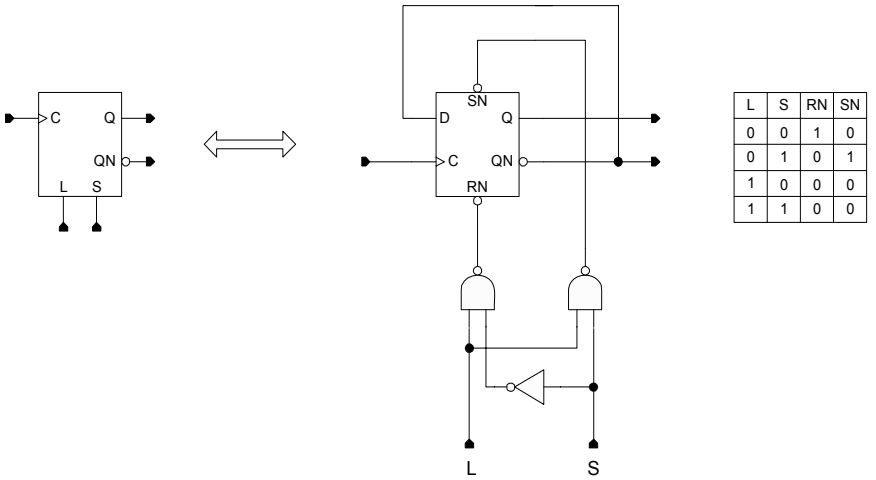


Figure D-2. A Simple Programmable Reference Divider



L	S	RN	SN
0	0	1	0
0	1	0	1
1	0	0	0
1	1	0	0

Figure D-3. A Basic Toggle Flip-Flop Implementation

The clock drives only the first TFF and each subsequent stage is driven by the previous output stage. Figure D-4 shows the timing diagram of the reference divider programmed to divide by 16. Additional flip-flops and gates are used as decision logic to detect when the count reaches final count and to reload the divider value.

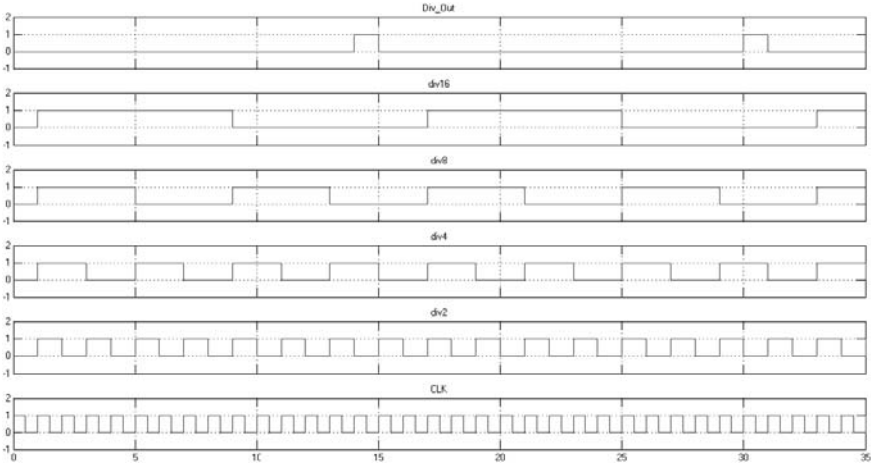


Figure D-4. Reference Divider Timing Diagram

2 FEEDBACK DIVIDER

The RF feedback divider is the only high-frequency block that operates at the VCO frequency. Driven by the VCO output clock frequency, the divider scales the feedback RF signal to the PFD input clock frequency for comparison with the reference VCXO clock.

2.1 Specification and Different Architecture Evaluation

High speed, power consumption, and low noise are the main requirements of the VCO frequency divider. The divider needs to operate at the VCO frequency making it unavoidably power hungry. However, some architectures have shown to consume less power than others. A good design is one that can be low noise and consume low power. These requirements are important as most wireless communication devices are portable and battery operated.

2.1.1 Direct Division versus Prescaler Method

Ideally a MMD such as the one in Figure D-5 driven directly by the RF VCO frequency (RF_{in}) will give minimal noise as the ratio

between RF_{in} and the clock frequency of the fractional Δ–Σ modulator is at a maximum. However, the MMD is implemented using synchronous CML and hence each stage is driven and operated at the RF VCO frequency. This implementation yields maximum power consumption and is difficult to design especially as the divider modulus is increased from 6 to 7 bits to cover frequency bands such as 802.11b, 802.11g, and Bluetooth™ (2.4–2.5 GHz) or higher frequency bands such as 802.11a (4.9–5.805 GHz). Additionally, if the divider is designed to marginally meet its highest frequency of operation, this leads to worst noise performance.

The following example illustrates the speed constraint of the feedback divider: taking the 802.11b band, if the RF channel frequency is:

$$f_{\mathrm{RF}} = 2.46\,\mathrm{GHz} \tag{D.1}$$

and the crystal oscillator frequency is:

$$f_{\mathrm{VCXO}} = 40\,\mathrm{MHz} \tag{D.2}$$

then the required divider ratio is:

$$N = \frac{f_{\mathrm{RF}}}{f_{\mathrm{VCXO}}} = \frac{2460}{40} = 61.5 \tag{D.3}$$

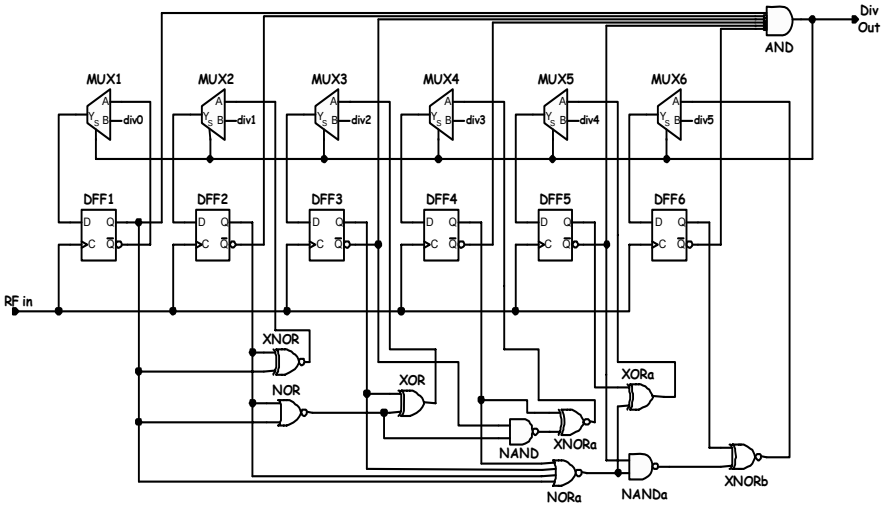


Figure D-5. Multimodulus Divider Implementation

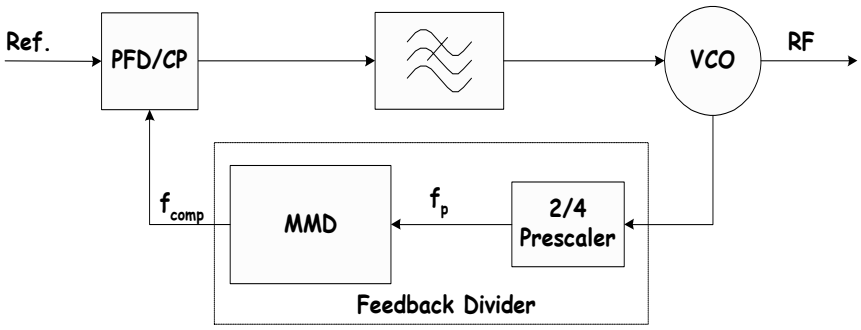


Figure D-6. Divide-by-2/4 Prescaler and MMD in the Feedback Divider

Using a simple third-order Mash modulator, the divisor can have a maximum value of 65 which clearly cannot be covered by a 6-bit MMD. To overcome this constraint, a prescaler such as a divide-by-2 or a divide-by-4, as shown in Figure D-6, can be used.

Having an intermediate RF prescaler in the feedback path reduces the speed constraints of the MMD, this in turn reduces the power consumption of the MMD as it operates at half the VCO frequency for the 2.4 GHz bands and a quarter of the VCO frequency for the 5–6 GHz bands. However, this prescaling in the feedback increases the noise of the PLL as the quantization noise of the Δ–Σ modulator gets amplified in the loop by a factor of:

$$20\log_{10} N_{\mathrm{p}} \tag{D.4}$$

where N_p is the intermediate prescaler value. The additional phase noise due to this phenomenon is approximately 12 dB for a 5 GHz synthesized frequency which significantly increases the in-band phase noise of the fractional-N PLL.

Other architectures that have been used in the past based on dual-modulus architectures such as the phase switching dual-modulus divider (PSDMD) are reported in [D1, D2].

The PSDMD architecture of Figure D-7 has a much lower power than the conventional direct MMD as only the first and the second divide-by-2 prescalers operate at full and at half VCO speeds, respectively. The divide-by-2 prescalers are single ended input to differential output and operate from rail-to-rail and hence do not require level shifting to interface to the low frequency CMOS phase multiplexers.

The main drawbacks of this architecture are the possibility of spikes occurring as the multiplexer switch transitions between phases.

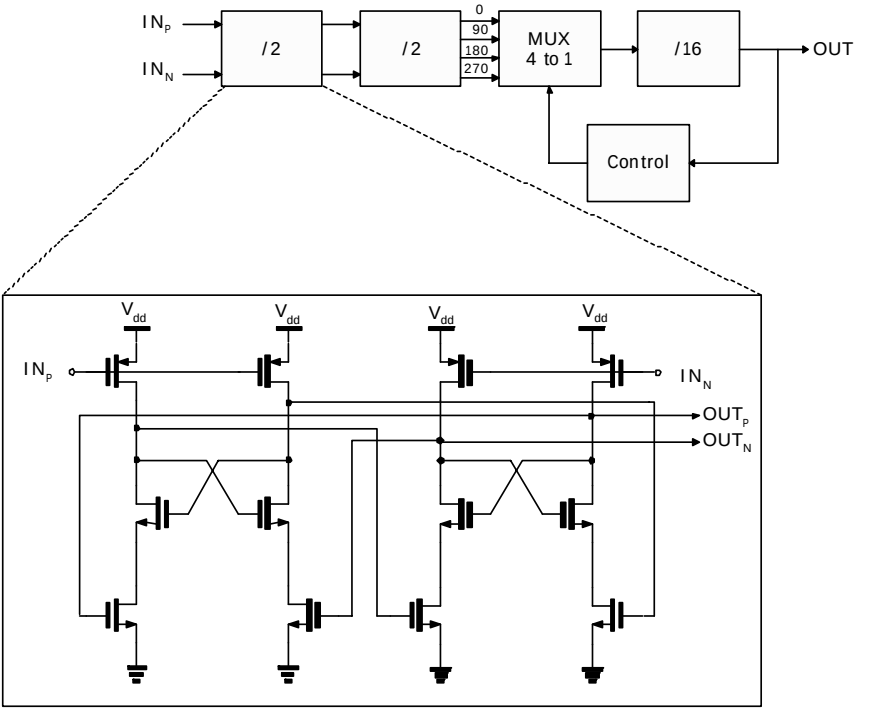


Figure D-7. The Phase Switching Dual-Modulus Prescaler

This could send the wrong information to the modulus control and hence divide by the wrong divisor. Also the input sensitivity of the divide-by-2 is much worse than its CML counterpart [D3].

The low frequency CMOS divide-by-16 counter needs to operate at a quarter of VCO frequency which might be fine for a 2 GHz range VCO frequency. However, as we move higher in the spectrum to cover the 802.11a high band of (5.805 GHz), that might be difficult to achieve. From a noise standpoint, this architecture is similar to that of the divide-by-4 prescaler-driven MMD as it exhibits similar quantization noise amplification of 12 dB, equation (D.4).

The ordinary DMD in Figure D-8 works somewhat differently from the PSDMD. A and B counters are loaded with the desired divider ratio and the prescaler divides by $P + 1$ as long as the modulus control is low. When the content of the B counter reaches 0, the MC goes high and the prescaler divides by P until the content of the counter A reaches 0. For proper division, counter A has to be greater

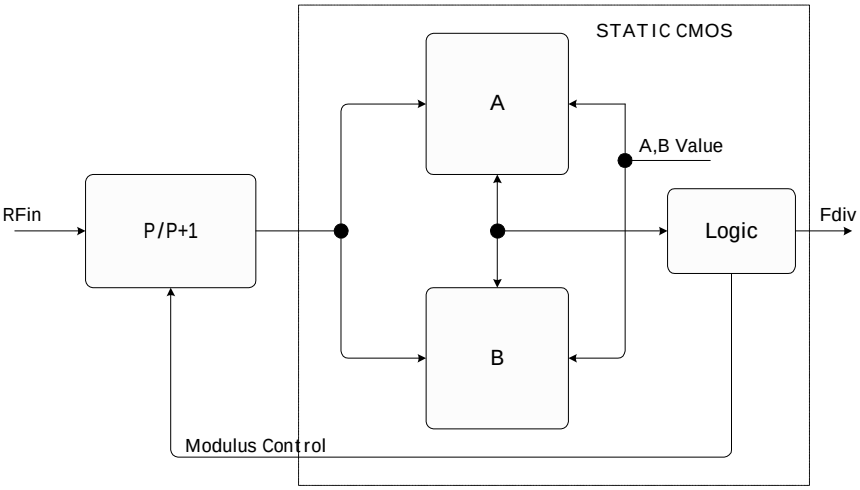


Figure D-8. P/P + 1 Dual-Modulus-Based Prescaler architecture

or equal to B. Hence the prescaler divides by $P + 1$ for B and by P for A – B, yielding a division of:

$$B(P+1)+(A-B)P = AP + B \tag{D.5}$$

where A and B values can be determined as follows:

$$A = \frac{\left(\frac{\text{RFin}}{\text{Fdiv}} - B\right)}{P} \text{ and } B = \frac{\text{RFin}}{\text{Fdiv}} - AP \tag{D.6}$$

The advantage of this method of feedback division is illustrated in the synchronized operation of the critical high-frequency $P/P + 1$ prescaler with the low-frequency A/B counters. Its feedback operation makes it behave as if it were a direct division MMD without the disadvantage of high power consumption. Consequently, this method does not amplify the quantization noise. Its major disadvantage is its continuous minimum divider value that is limited to $P^2 - P$. The maximum divider value however is only limited by the size of the A and B counters.

Figure D-9 shows logic implementation of 2/3 DMD. Having the reset signal at logic high, the divider divides by 3 for MC signal at logic high and divides by 2 for MC signal at logic low. Figure D-10 shows a simulated operation of a 2/3 DMD.

Using the designed divide-by-2/3 dual-modulus prescaler architecture of Figure D-9, we can design any higher modulus divider. Figures D-11 and D-13 show the implementation of a 4/5 and 8/9 dual-modulus prescalers, respectively. Figures D-12 and D-14 show their respective simulated behaviors.

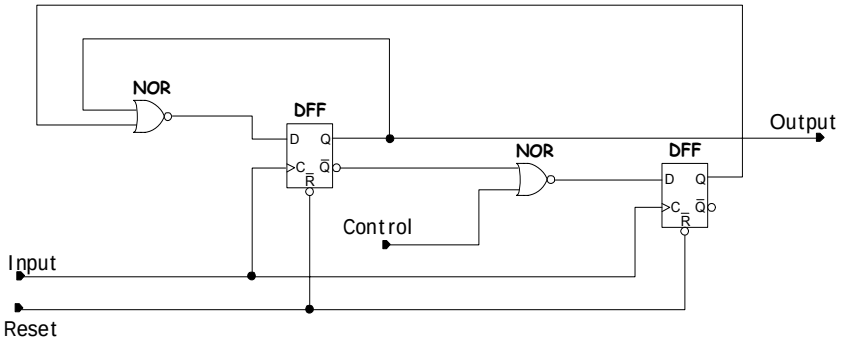


Figure D-9. Divide-by-2/3 Dual-Modulus Prescaler Logic

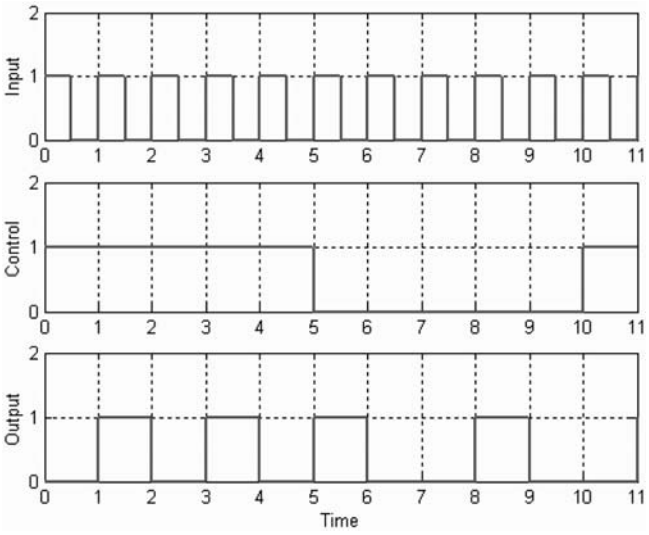


Figure D-10. Simulated Divide-by-2/3 DMD

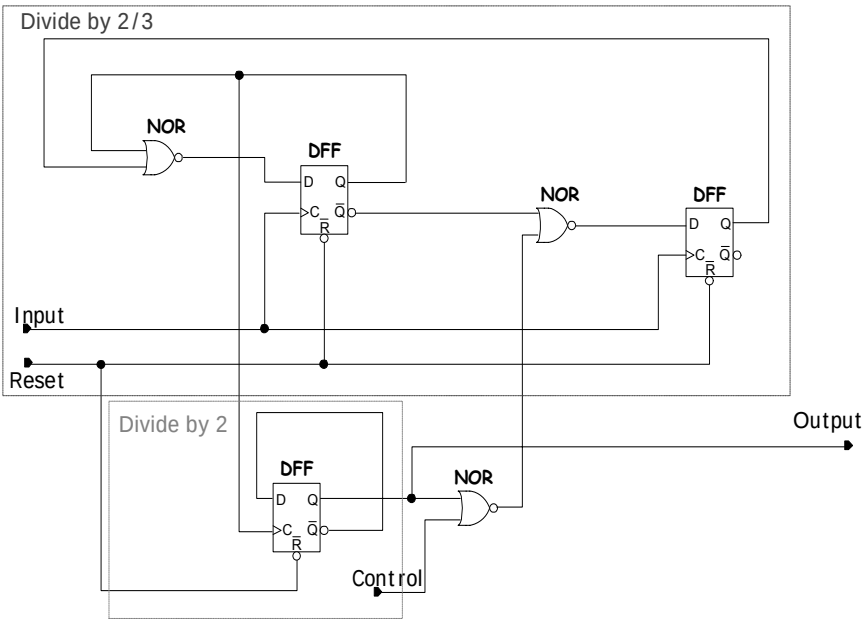


Figure D-11. Divide-by-4/5 MMD implementation

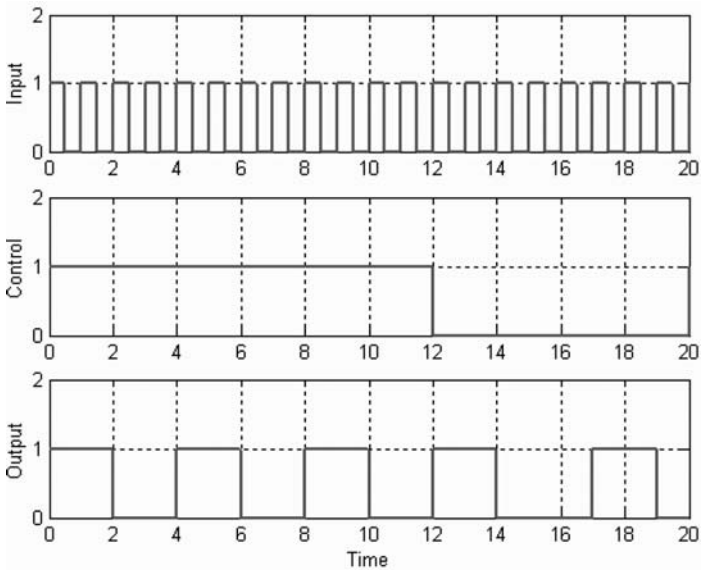


Figure D-12. Simulated 4/5 Divider

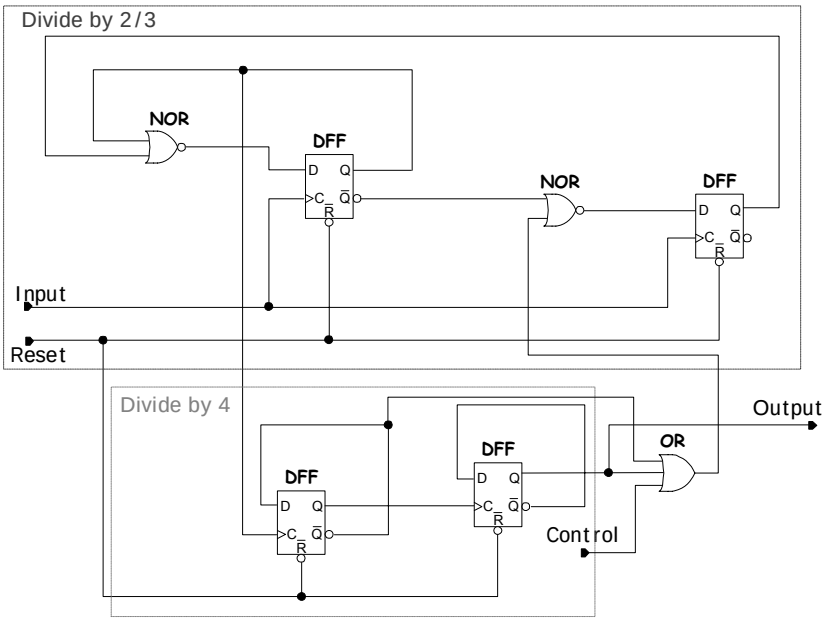


Figure D-13. Divide-by-8/9 DMD implementation

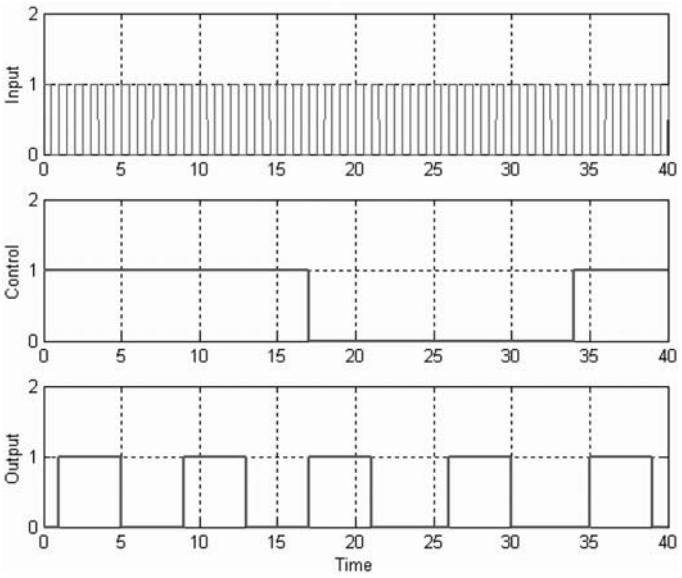


Figure D-14. Simulated 8/9 DMD divider

In implementing the multistandard WLAN of Figure 5-2, the synthesized frequency is two-thirds of the transmitter frequency. This indirect method of transmission is used to prevent power amplifier (PA) pulling of the VCO. The minimum and the maximum synthesized frequencies are 3.2 and 3.9 GHz, respectively. Bearing in mind that the reference used is 40 MHz, the minimum divider value is:

$$\frac{RF_{\min}.\frac{2}{3}}{\mathrm{Re}\,f}-n_{\min}=\frac{4800.\frac{2}{3}}{40}-3=77 \tag{D.7}$$

where $RF_{\min}$ is the minimum transmitter frequency in MHz, Ref is the reference frequency in MHz and $n_{\min}$ is the order of the Δ–Σ MASH modulator. The maximum divider value is:

$$\frac{RF_{\max}.\frac{2}{3}}{\mathrm{Re}\,f}+n_{\max}=\frac{5805.\frac{2}{3}}{40}+4=100 \tag{D.8}$$

$RF_{\max}$ is the maximum transmitter frequency in MHz and $n_{\max}$ is the maximum Δ–Σ MASH modulator offset value. Examining all the required divisors, the divider modulus needs to be 7 bits with the MSB set to logic high.

Another observation is the fact that only the first three bits of the divisor are modulated due to the implemented third-order Δ–Σ modulator. Having this in mind, we can use a 8/9 prescaler method with the A/B swallow counter with four and three bits, respectively and the MSB of the A counter (fourth bit) set to a logic high. Figure D-15, shows the implemented divider architecture.

Having the MSB of counter A tied to V_{dd} (logic high), the fractional divider value can be programmed as follows:

$$N_{\mathrm{frac}}=8\cdot\left[8+N<5:3>\right]+N<2:0>\right] \tag{D.9}$$

where N<5:0> is the output of the Δ–Σ modulator described in chapter 5.

The A and B counters are full-scale static CMOS and operate at a maximum frequency of 500 MHz. The 8/9 prescaler is the bottleneck

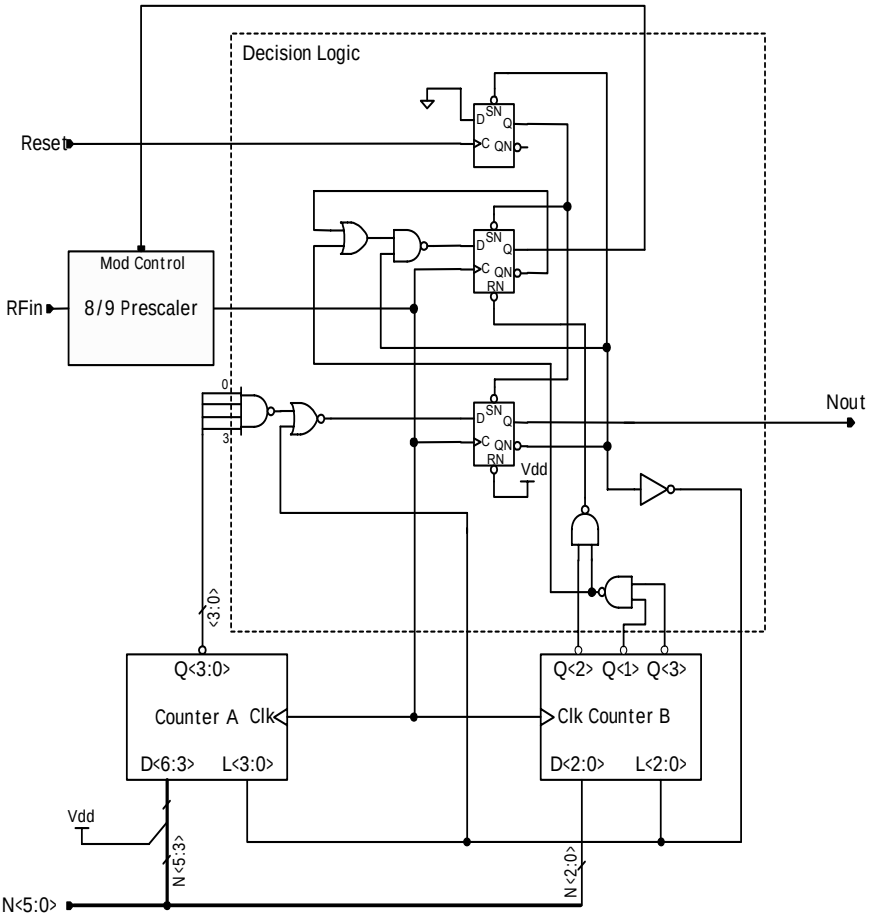


Figure D-15. Multimodulus Divider (MMD) Architecture using 8/9 Prescaler

of the MMD operating at high VCO frequency. It is implemented using CML. The design of the 8/9 prescaler is described and covered in the next section.

3 HIGH-SPEED CMOS DIVIDER DESIGN

Implementing the high-speed prescaler in CMOS requires careful attention. The high-frequency operation means high power consumption.

However, using the architecture outlined in Figure D-13, only the flip-flops of the front divide-by-2/3 stage operate at full VCO speed. The rest of the logic operates at half speed. The flip-flops of the divide-by-4 stages operate at half and quarter speeds. All 8/9 prescaler blocks are designed in high-speed CML.

3.1 Current-Mode Logic Design: An Overview

Unlike static CMOS, CML gates can operate at high frequency in the GHz region. To help us understand the various elements that set the speed and hence the power consumption of a CML gate, let us examine a simple differential CMOS amplifier also a CML inverter. Figure D-16 shows a basic CML inverter gate.

The speed of the gate is determined by the transient characteristics, the current drive capability of its load and the propagation delay.

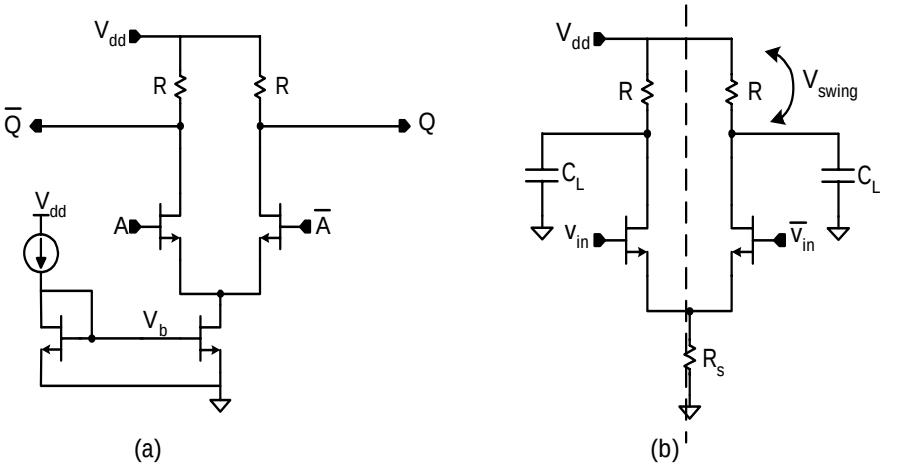


Figure D-16. Basic CML Gate (a) Biased Gate (b) Simplified Gate with Capacitive Load

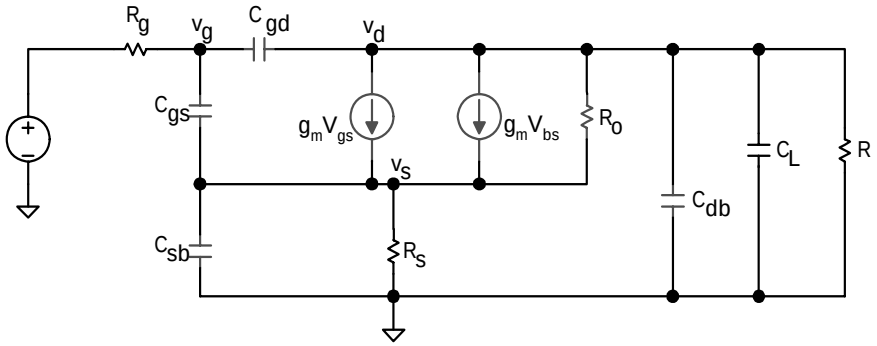


Figure D-17. CML Gate Half-Circuit Model

Using the half circuit of Figure D-16b, we can derive the propagation delay of the CML inverter gate. Figure D-17 shows half the CML gate circuit model.

We can treat the delay as four components for simplicity:

$$\tau = \tau_1 + \tau_2 + \tau_3 + \tau_4 \tag{D.10}$$

Each delay component can be studied separately. For τ_1, delay term due to gate–source capacitance, use Figure D-18, we can ignore g_{mb} and R_o:

$$R_1 = \frac{v_T}{i_T} = \frac{R_g + R_s}{1 + g_m R_s} \tag{D.11}$$

$$\tau_1 = R_1 C_{gs} = \left(\frac{R_g + R_s}{1 + g_m R_s} \right) C_{gs} \tag{D.12}$$

where V_T is the input voltage at the gate of the MOS gate.

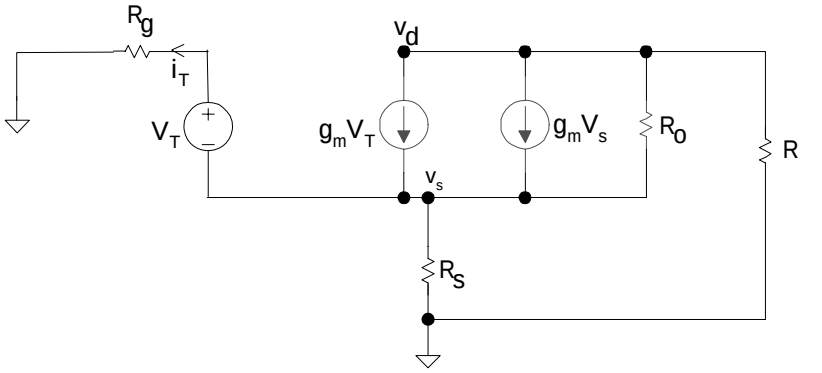


Figure D-18. Simplified Model to Derive First Delay Term

To derive the second delay term due to gate–drain junction capacitance, we ignore g_{mb} and R_o, Figure D-19 is used as follows:

$$R_2 = \frac{v_T}{i_T} = R_g + R + \frac{g_m}{1 + g_m R_s} R_g R \tag{D.13}$$

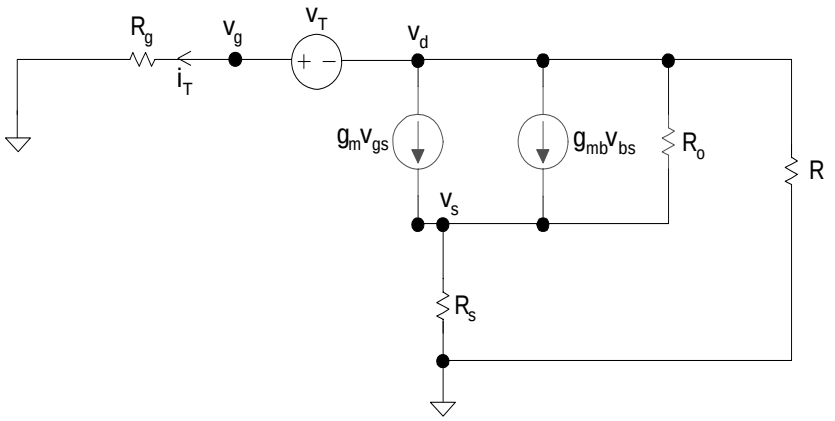


Figure D-19. Simplified Model to Derive Second Delay Term

$$\tau_2 = \left(R_g + R + \frac{g_m}{1 + g_m R_s} R_g R \right) C_{gd} \tag{D.14}$$

For the third delay term, τ_3 due to source–bulk junction capacitance, the simplified circuit model of Figure D-20 is used:

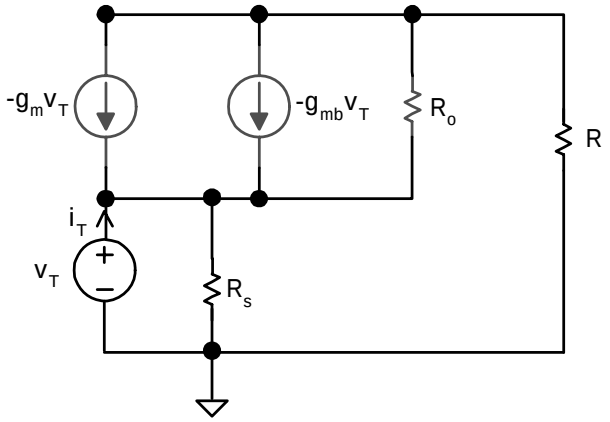


Figure D-20. Simplified Model to Derive Component due to Source–bulk Capacitance

$$R_3 = \frac{R_s}{1 + g_m R_s} \tag{D.15}$$

$$\tau_3 = R_3 C_{sb} = \left(\frac{R_s}{1 + g_m R_s} \right) C_{sb} \tag{D.16}$$

Similarly for the fourth delay term, τ_3 due to the capacitive load C_L and drain–bulk junction capacitance, Figure D-21 shows the simplified circuit model, where we can ignore R_o:

$$\tau_4 = R\left(C_{db} + C_L \right) \tag{D.17}$$

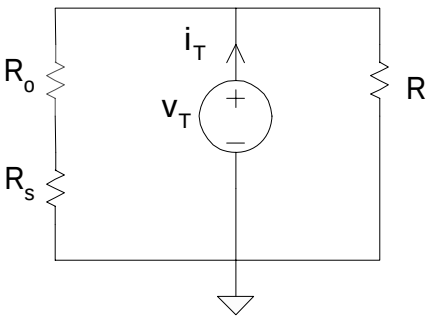


Figure D-21. Simplified Circuit Model to Derive fourth Delay Term

The total propagation delay of the CML inverter then becomes:

$$\tau = \frac{R_g + R_s}{1 + g_m R_s} C_{gs} + \left(R_g + R + \frac{g_m}{1 + g_m R_s} R_g R \right) C_{gd} + \frac{R_s C_{sb}}{1 + g_m R_s} + \left(C_{db} + C_L \right) R \qquad \text{(D.18)}$$

$R_s = 0$ (AC ground for differential mode) and consider $R_g = 0$ negligible series gate resistance. This leads to:

$$\tau = R\left(C_{gd} + C_{db} + C_L \right) = \frac{V_{swing}}{I}\left(WC_{ov} + WL_d C_{jd} + C_L \right) \qquad \text{(D.19)}$$

Where C_{ov} is the oxide capacitance, C_{jd} the drain junction capacitor, W and L are transistor dimensions. V_{swing} is the required output voltage swing. Knowing the required swing and the required speed we can choose the transistor dimensions and the differential tail current required to drive the CML gate.

4 IMPLEMENTED CML GATES

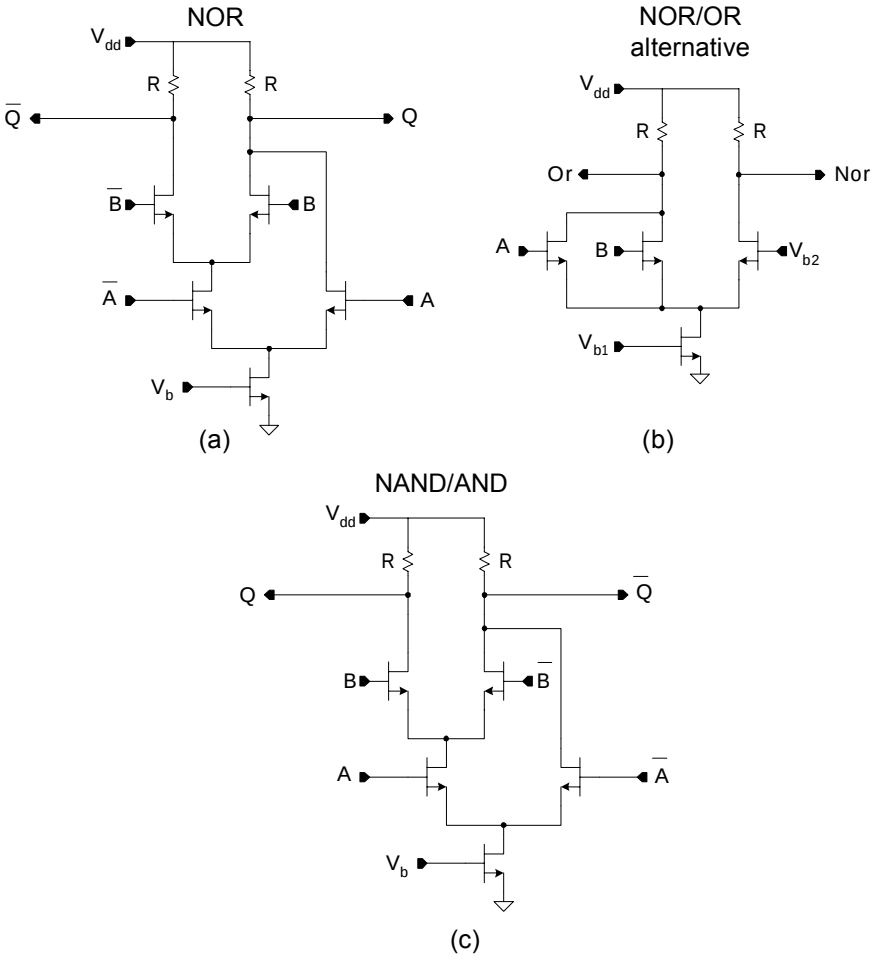


Figure D-22. Basic CML Gates Implementation (a), (b) NOR/OR, and (c) NAND/AND

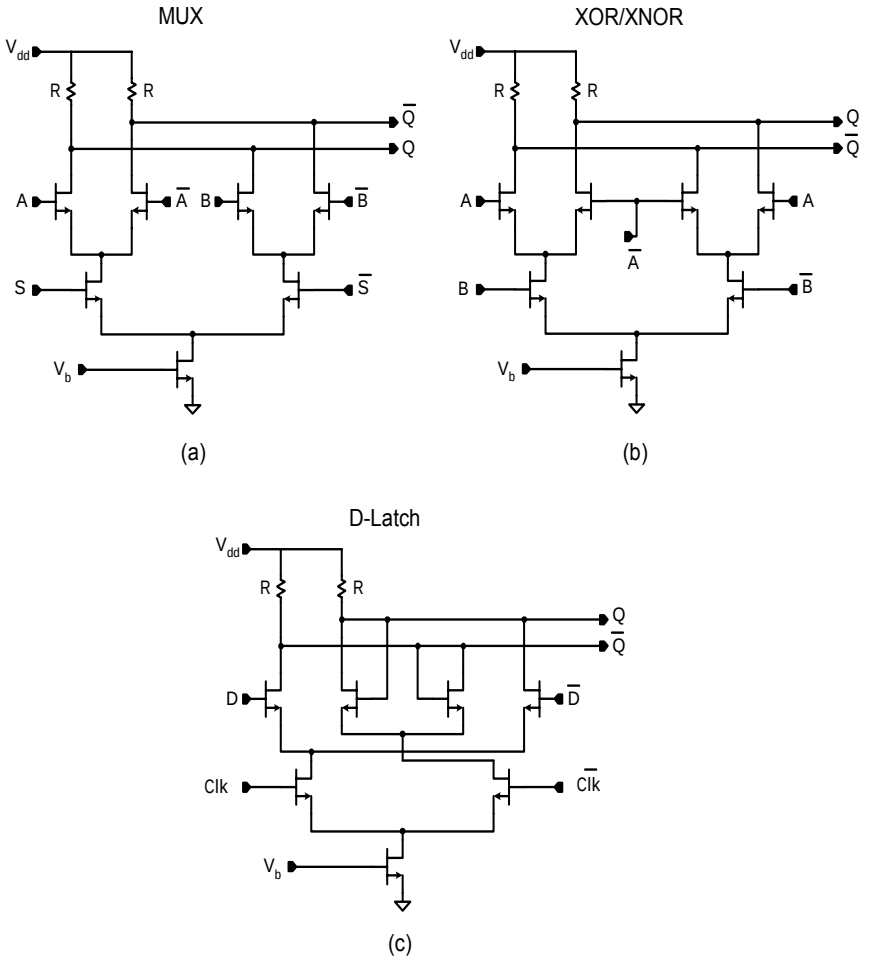
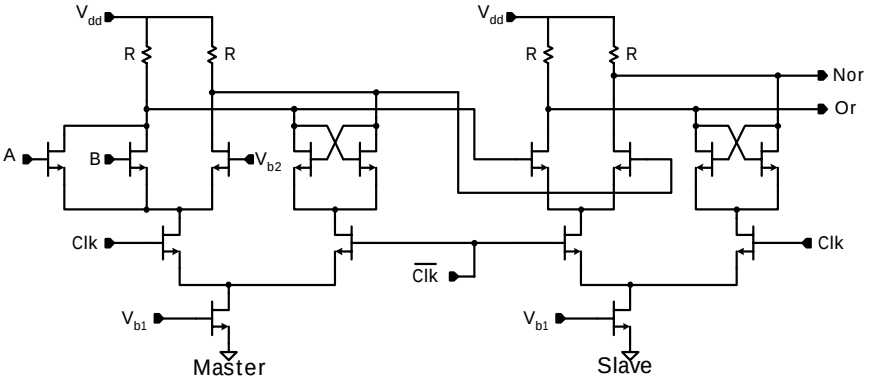
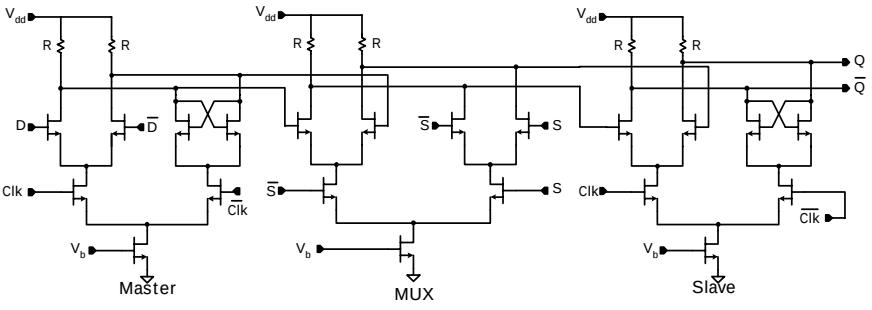


Figure D-23. CML gates (a) Multiplexer, (b) XOR/XNOR, and (c) Latch



(a)



(b)

Figure D-24. Combination CML Logic (a) NOR/OR Mater/Slave D-Type Flip-Flop (b) MUX/D-Type Flip-Flop

REFERENCES

[D1] J. Craninckx and M. Steyaert "A 1.75-GHz/3-V Dual-Modulus Divide-by-128/129 Prescaler in 0.7-μm CMOS," *JSSC,* 31, 7, pp. 890–897, July 1996.

[D2] R. Ahola and K. Halonen, "A 4 GHz CMOS Multiple Modulus Prescaler," *Proceedings of IEEE ICECS 1998*, Lisbon, Portugal, Sept. 1998, pp. 2.323–2.326.

[D3] T. Seneff et al., "A Sub-1 mA 1.6 GHz Silicon Bipolar Dual Modulus Prescaler," *IEEE Journal of Solid-State Circuits*, 29, pp. 1206–1211, Oct. 1994.

Appendix E

PROGRAMS AND CODES

1 MATHCAD™ PROGRAM USED FOR THE SIMULATIONS OF ALL THE MATHCAD FIGURES

Mathcad program used for the simulation of all Mathcad figures presented in the book.

Defining the Units:-

$f \equiv 10^{-15}$ $p \equiv 10^{-12}$ $n \equiv 10^{-9}$ $\mu \equiv 10^{-6}$ $m \equiv 10^{-3}$ $k \equiv 10^{3}$

$M \equiv 10^{6}$ $G \equiv 10^{9}$ $T \equiv 10^{12}$ $R := 10$ Set Resolution:- Hz

Defining the Loop Parameters:-

$K_{vco} := 100 \cdot M$ MHz/V

$K\phi := 2.0 \cdot m$ mA

Samples per decade

Start_Freq $:= 10$ Hz

Stop_Freq $:= 10 \cdot M$

Duty Cycle $:= 2 \cdot \pi$ Duty Cycle

$F_{max} := 1725 \cdot MHz$

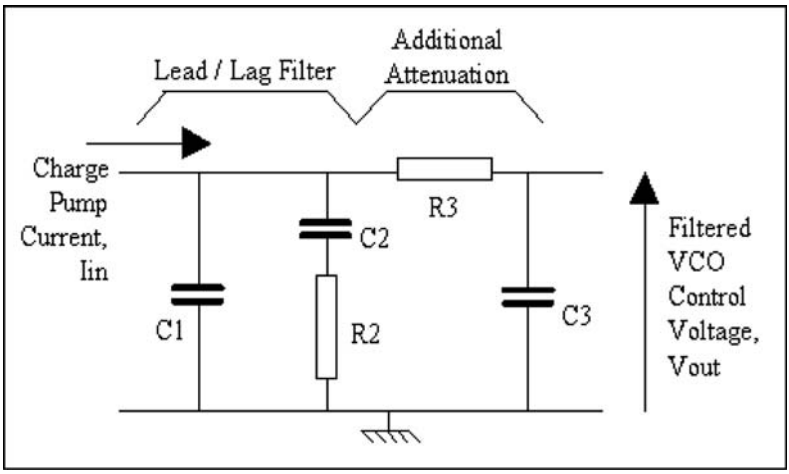
$F_{min} := 1725 \cdot MHz$

VCXO_Ref_Freq $:= 40 \cdot MHz$

$F_{samp} := 40 \cdot MHz$

The Log sweep...

$XScale(F) := 10^{\frac{F}{10}}$



The Linear Sweep...

$$F := 10\cdot\log(Start\_Freq), 10\cdot\log(Start\_Freq) + \frac{10}{R} .. 10\cdot\log(Stop\_Freq)$$

If FM modulation is applied at the VCO
then this value must be entered.

$K_{fm} := K_{vco}$

$K_{fm} := 200\cdot k$

Enter values for calculating the loop filter components:-

$LBW := 100\cdot k\ ^{Hz}$

$\phi p := 56\ ^{Degrees}$

$ATTEN := 15\ \ dB$

(Make this value"0"to remove the
additional R3 C3 low pass filter)

RF Noise Source Values:-

VCO Noise

$VCO\_9dB := 1.3\cdot k$ $VCONoise\_Plateau := -159\ ^{dBc/Hz}$

$VCO\_3dB := 8\ M$ $VCO\_6dB := 8\cdot M$

Noise Sources

Reference Oscillator Noise VCO Divider Noise

$RefNoise\_Plateau := -143\ dBc/Hz$ $VCODiv\_Plateau := -173\ dBc/Hz$ $VCODiv\_Noise\_3dB := 0.3\cdot$

$Ref\_3dB := 10\cdot k$ $Ref\_6dB := 1\cdot k$ RF Amplifier Noise dBm

$Ref\_9dB := 100$ $Ref\_12dB := 10$ $Amp\_NF := 8\ dB$ $Power\_in := -15\ Hz$

Reference Divider Noise

$RefDiv\_Plateau := -173\ dBc/Hz$ $R\_Div\_3dB := 0.3\cdot k$ $R\_Div\_6dB := 10$

$Amp\_Gain := 25\ dB$ $RFAmp\_noise\_3dB := 100$

$\Phi n\_Plateau := -216\ dBc/Hz$

$\Phi n_{3dB} := 8\cdot M$ $\Phi\_3dB := 1\cdot k$ $\Phi\_6dB := 50$ $\Phi\_9dB := 20$ $\Phi\_12dB := 5$

Loop Filter Value CALCULATIONS

Type 2 Second-Order Filter

$K_V := K_{vco}$ $\omega p := 2\cdot\pi\cdot LBW$ $N := \frac{\sqrt{Fmax\cdot Fmin}}{Fsamp}$

$K_V = 1\times 10^{8}$ $LBW = 1\times 10^{5}$ $N = 43.125$

$$T1 := \frac{\sec\left(\phi p\cdot\frac{\pi}{180}\right) - \tan\left(\phi p\cdot\frac{\pi}{180}\right)}{\omega p} \qquad T2 := \frac{1}{\omega p^{2}\cdot T1}$$

$$\omega p_{rad} := \sqrt{\frac{1 + (2\cdot\pi\cdot LBW)^{2}\cdot T2^{2}}{1 + (2\cdot\pi\cdot LBW)^{2}\cdot T1^{2}}}$$

$T1 = 4.866\times 10^{-7}$

$T2 = 5.206\times 10^{-6}$

$$C1_{T22O} := \frac{T1\cdot\frac{K_{vco}\cdot K\phi}{N}\cdot\omega p_{rad}}{T2\cdot(2\cdot\pi\cdot LBW)^{2}} \qquad C2_{T22O} := C1_{T22O}\cdot\left(\frac{T2}{T1} - 1\right) \qquad R2_{T22O} := \frac{T2}{C2_{T22O}}$$

$C1_{T22O} = 3.591536\times 10^{-9}$ $C2_{T22O} = 3.483242\times 10^{-8}$ $R2_{T22O} = 149.451$

Type 2 Third-Order Filter

$$T3 := \sqrt{\frac{10^{\frac{ATTEN}{20}} - 1}{(2\cdot\pi\cdot F_{samp})^{2}}}$$

$$T2 := \frac{1}{(T1+T3)\cdot\left[\frac{(T1+T3)\cdot\tan\left(\phi p\cdot\frac{\pi}{180}\right)}{(T1+T3)^2+T1\cdot T3}\cdot\left[\sqrt{\left[\frac{(T1+T3)^2+T1\cdot T3}{(T1+T3)^2\cdot\tan\left(\phi p\cdot\frac{\pi}{180}\right)^2}\right]}+1-1\right]\right]^2}$$

$$C1_{T23O} := \left(\frac{T1}{T2}\right)\cdot\left[\frac{1}{\left[\frac{(T1+T3)\cdot\tan\left(\phi p\cdot\frac{\pi}{180}\right)}{(T1+T3)^2+T1\cdot T3}\cdot\left[\sqrt{\left[\frac{(T1+T3)^2+T1\cdot T3}{(T1+T3)^2\cdot\tan\left(\phi p\cdot\frac{\pi}{180}\right)^2}\right]}+1-1\right]\right]^2}\right]\cdot\frac{K_{vco}\cdot K\phi}{N}\cdot$$

$$\sqrt{\frac{T2^2\cdot\left[\frac{(T1+T3)\cdot\tan\left(\phi p\cdot\frac{\pi}{180}\right)}{(T1+T3)^2+T1\cdot T3}\cdot\left[\sqrt{\left[\frac{(T1+T3)^2+T1\cdot T3}{(T1+T3)^2\cdot\tan\left(\phi p\cdot\frac{\pi}{180}\right)^2}\right]}+1-1\right]\right]^2+1}{\left[1+\left[\frac{(T1+T3)\cdot\tan\left(\phi p\cdot\frac{\pi}{180}\right)}{(T1+T3)^2+T1\cdot T3}\cdot\left[\sqrt{\left[\frac{(T1+T3)^2+T1\cdot T3}{(T1+T3)^2\cdot\tan\left(\phi p\cdot\frac{\pi}{180}\right)^2}\right]}+1-1\right]\right]^2\cdot T1^2\right]}}$$

$$\sqrt{\frac{1}{\left[1+\left[\frac{(T1+T3)\cdot\tan\left(\phi p\cdot\frac{\pi}{180}\right)}{(T1+T3)^2+T1\cdot T3}\cdot\left[\sqrt{\left[\frac{(T1+T3)^2+T1\cdot T3}{(T1+T3)^2\cdot\tan\left(\phi p\cdot\frac{\pi}{180}\right)^2}\right]}+1-1\right]\right]^2\cdot T3^2\right]}}$$

$$C2_{T23O} := C1_{T23O}\cdot\left[\left(\frac{T2}{T1}\right)-1\right] \qquad R2_{T23O} := \frac{T2}{C2_{T23O}} \qquad C3_{T23O} := \frac{C1_{T23O}}{10} \qquad R3_{T23O} := \frac{T3}{C3_{T23O}}$$

$$C1_{T23O} = 3.665\times10^{-9} \quad C3_{T23O} = 3.665\times10^{-10} \quad C2_{T23O} = 3.635\times10^{-8} \quad R2_{T23O} = 146.14 \quad R3_{T23O} = 23.342$$

$$C1 := \text{if}\left(ATTEN = 0, C1_{T22O}, C1_{T23O}\right) \qquad C2 := \text{if}\left(ATTEN = 0, C2_{T22O}, C2_{T23O}\right) \qquad C3 := \text{if}\left(ATTEN = 0, 0, C3_{T23O}\right)$$

$$R2 := \text{if}\left(\text{ATTEN} = 0, R2_{T22O}, R2_{T23O}\right) \qquad R3 := \text{if}\left(\text{ATTEN} = 0, 0, R3_{T23O}\right)$$

Giving the following values:-

NOISE CALCULATIONS

Reference Noise Calculation

$$R := \frac{\text{VCXO\_Ref\_Freq}}{F_{\text{samp}}} \quad R = 1$$

This line calculates the *R* divider logic noise floor including the 1/*f* corners for this *R* divider.

$$\text{R\_Div\_Noise}(\text{FrqPoint}) := \text{Ref Div\_Plateau} + 10\cdot\log\left(\left|1 + j\cdot\frac{\text{R\_Div\_3dB}}{\text{FrqPoint}}\right|\right) + 10\cdot\log\left(\left|1 + j\cdot\frac{\text{R\_Div\_6dB}}{\text{FrqPoint}}\right|\right)$$

This line calculates the ideal divided down (due to the *R* term) of the incoming reference VCXO source. Note that this is a direct curve fit on the specification and makes no attempt to correct for the usual oscillator 1/*f* corners, giving 10Log breakpoints.

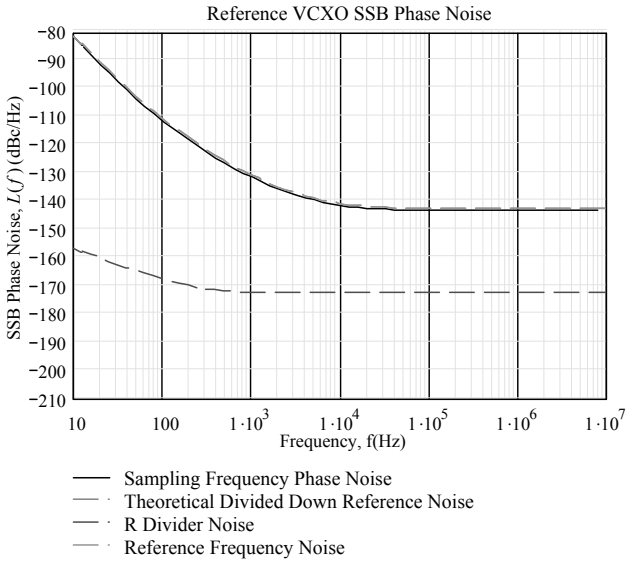
$$\text{Ref\_Div}(\text{FrqPoint}) := \text{Ref Noise\_Plateau} + 10\cdot\log\left(\left|1 + j\cdot\frac{\text{Ref\_3dB}}{\text{FrqPoint}}\right|\right) + 10\cdot\log\left(\left|1 + j\cdot\frac{\text{Ref\_6dB}}{\text{FrqPoint}}\right|\right) + \blacksquare$$

$$\blacksquare + 10\cdot\log\left(\left|1 + j\cdot\frac{\text{Ref\_9dB}}{\text{FrqPoint}}\right|\right) + 10\cdot\log\left(\left|1 + j\cdot\frac{\text{Ref\_12dB}}{\text{FrqPoint}}\right|\right) - 20\cdot\log(R)$$

This VXCO SSB Phase Noise Plot represents the free running VCXO which is considerably cleaner than the 16 MHz clean upused. This is because the 16 MHz cleanup loop has a loop bandwidth of the order of 40 Hz, there after the phase noise follows the logic noise and Op-Amp noise profile of the other components in that loop. These values are not modeled here, giving an optimistic account of this loop's output. This will reflect in a better than measured value in the main synthesizer output for this part of the phase noise profile.

Hence, the actual phase noise profile of the signal at the sampling frequency is:-

$$\text{Samp\_Freq\_Noise}(\text{FrqPoint}) := 10\cdot\log\left(10^{\frac{\text{Ref\_Div}(\text{FrqPoint})}{10}} + 10^{\frac{\text{R\_Div\_Noise}(\text{FrqPoint})}{10}}\right)$$



The effect of the *R* divider logic noise plateau becomes particuraly important when considering low sampling frequencies and very large VCO frequencies and hence multiplication of this reference noise. The clear distinction has to be made between the sampling frequency phase noise and the phase frequency logic noise.

VCO Noise Calculation

$$VCONoise(FrqPoint) := VCONoise\_Plateau + 10\cdot\log\left(\left|1 + j\cdot\frac{VCO\_3dB}{FrqPoint}\right|\right) + 10\cdot\log\left(\left|1 + j\cdot\frac{VCO\_6dB}{FrqPoint}\right|\right) + \blacksquare$$

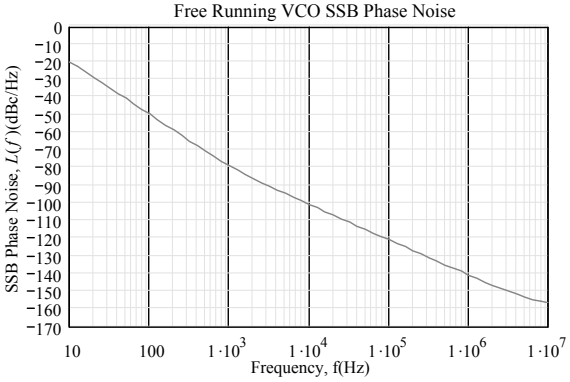
$$10\cdot\log\left(\left|1 + j\cdot\frac{VCO\_9dB}{FrqPoint}\right|\right)$$

Phase Noise at different offsets:-

$$VCONoise\left(10^{\frac{50}{10}}\right) = -120.937 \quad VCONoise\left(10^{\frac{40}{10}}\right) = -100.902 \quad VCONoise\left(10^{\frac{30}{10}}\right) = -78.789$$

L(800 kHz)=

$$VCONoise\left(10^{\frac{10\cdot\log(800\cdot k)}{10}}\right) = -138.957$$



Phase Detector Noise Calculations

This first line calculates the frequency profile of the normalized phase detector noise.

$$PD\_Noise\_Norm(FrqPoint) := \Phi n\_Plateau + 10\cdot\log\left(\left|1 + \frac{\Phi\_3dB}{FrqPoint}\right|\right) + 10\cdot\log\left(\left|1 + \frac{\Phi\_6dB}{FrqPoint}\right|\right) + \blacksquare$$

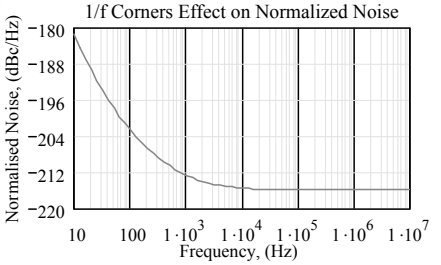
$$\blacksquare + 10\cdot\log\left(\left|1 + \frac{\Phi\_9dB}{FrqPoint}\right|\right) + 10\cdot\log\left(\left|1 + \frac{\Phi\_12dB}{rqPoint}\right|\right)$$

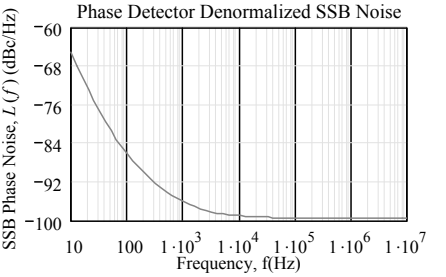
This line denormalizes this normalized phase detector noise profile.

$$PD\_Noise(FrqPoint) := PD\_Noise\_Norm(FrqPoint) - 10\cdot\log(Fsamp) + 20\cdot\log\left(\sqrt{Fmax\cdot Fmin}\right) + 10\cdot\log\left(\left|1 + \frac{Fsamp}{\Phi n_{3dB}}\right|\right)$$

Phase Noise at 1 MHz offset:-

$$PD\_Noise\left(10^{\frac{60}{10}}\right) = -99.499$$





VCO Divider Noise Calculation

$$VCODiv\_Noise(FrqPoint) := VCODiv\_Plateau + 10\cdot\log\left(\left|1 + j\cdot\frac{VCODiv\_Noise\_3dB}{FrqPoint}\right|\right)$$

RF Amplifier Noise Calculation

Some constants...

$K \equiv 1.38066210^{-23}$ J/K $T \equiv 273.14$ K $B \equiv 1$ Hz

Calculation of Amplifier Noise

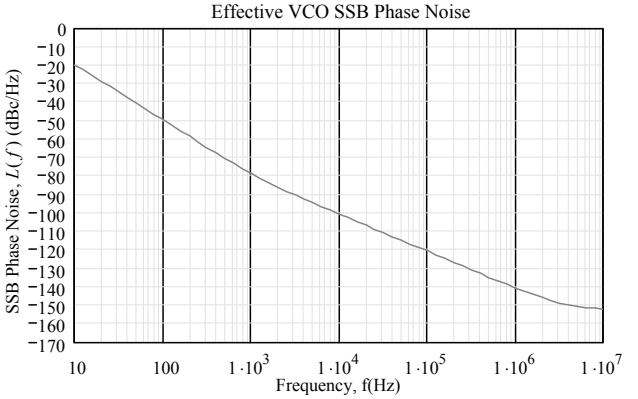
$$RFAmp\_Plateau := Amp\_NF + 10\cdot\log(K\cdot T\cdot B) + 30 - 3 - Power\_in \quad \text{dBc/Hz}$$

giving... $RFAmp\_Plateau = -154.235$

Hence TOTAL RF Noise of the VCO and the RF Amplifier is calculated...

$$RFAmp\_noise(FrqPoint) := RFAmp\_Plateau + 10\cdot\log\left(\left|1 + j\cdot\frac{RFAmp\_noise\_3dB}{FrqPoint}\right|\right)$$

$$VCOEffective\_Noise(FrqPoint) := 10\cdot\log\left(10^{\frac{VCONoise(FrqPoint)}{10}} + 10^{\frac{RFAmp\_noise(FrqPoint)}{10}}\right)$$



LOOP GAIN PARAMETER CALCULATIONS

$N_m := \sqrt{\frac{F_{min}}{F_{samp}} \cdot \frac{F_{max}}{F_{samp}}}$ $N_m = 43.125$ $F_{vco} := \sqrt{F_{max} \cdot F_{min}}$

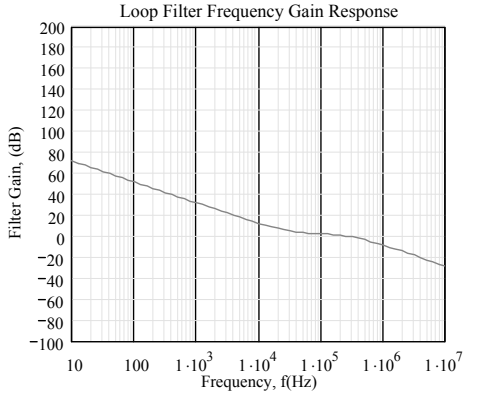
$K_{pd} := \frac{K\phi}{DutyCycle}$ $K_{pd} = 3.183 \times 10^{-4}$ A/Rad

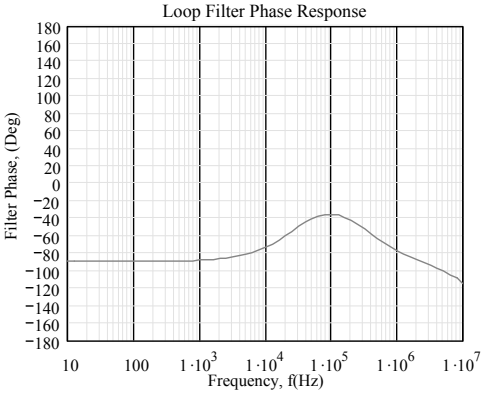
$K_v := Kvco \cdot 2 \cdot \pi$ $K_v = 6.283 \times 10^{8}$ Rad/V R2 = 146.14

Filter Transfer Calculation

$$Fs(FrqPoint) := if\left[ATTEN = 0, \frac{1 + j\cdot(2\cdot\pi\cdot FrqPoint)\cdot R2\cdot C2}{\left[j\cdot(2\cdot\pi\cdot FrqPoint)\cdot(C1 + C2)\right]\cdot\left[1 + j\cdot(2\cdot\pi\cdot FrqPoint)\cdot R2\cdot\frac{C1\cdot C2}{C1 + C2}\right]}, \frac{1}{\left[j\cdot(2\cdot\pi\cdot FrqPoint)\cdot C3\right]}\right]$$

$$\left[\frac{1}{\left[1 + j\cdot(2\cdot\pi\cdot FrqPoint)\cdot R2\cdot C2 + \left[1 + j\cdot(2\cdot\pi\cdot FrqPoint)\cdot R3\cdot C3\right]\cdot(C1 + C2)\cdot\left(\frac{1}{C3}\right)\cdot\left[1 + j\cdot(2\cdot\pi\cdot FrqPoint)\cdot R2\cdot\frac{C1\cdot C2}{C1 + C2}\right]\right]}\right]$$

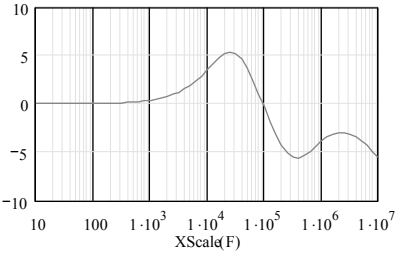


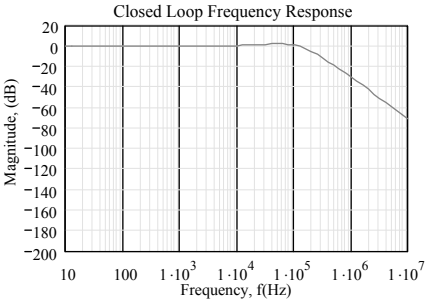


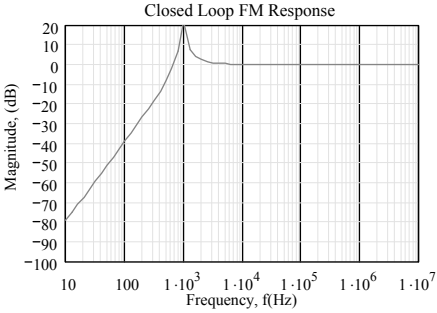
Calculating the Loop Gain Terms

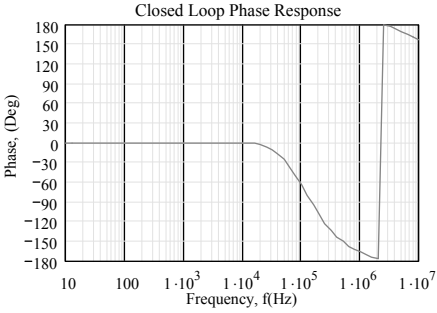
$$\text{Aol}(\text{FrqPoint}) := \text{Fs}(\text{FrqPoint}) \cdot \frac{K_v}{j \cdot (2 \cdot \pi \cdot \text{FrqPoint})} \cdot K_{pd} \qquad \text{FB} := \frac{1}{N_m} \qquad K_{pd} = 3.183 \times 10^{-4}$$

$$\frac{d}{dF} \arg\left(F_S\left(10^{\frac{F}{10}}\right)\right) \cdot \frac{180}{\pi}$$









Final Noise Summation Calculation

$$VCO\_LoopGain\_dB(FrqPoint) := 20\cdot\log\left(\left|\frac{1}{1 + Aol(FrqPoint)\cdot FB}\right|\right)$$

$$Ref\_in\_LoopGain\_dB(FrqPoint) := 20\cdot\log\left(\left|\frac{Aol(FrqPoint)}{1 + Aol(FrqPoint)\cdot FB}\right|\right)$$

$$PD\_LoopGain\_dB(FrqPoint) := 20\cdot\log\left(\left|\frac{Aol(FrqPoint)\cdot FB}{1 + Aol(FrqPoint)\cdot FB}\right|\right)$$

$$VCODiv\_LoopGain\_dB(FrqPoint) := 20\cdot\log\left(\left|\frac{Aol(FrqPoint)}{1 + Aol(FrqPoint)\cdot FB}\right|\right)$$

$$VCO\_Loop\_Noise\_dB(FrqPoint) := VCO\_LoopGain\_dB(FrqPoint) + VCOEffective\_Noise(FrqPoint)$$

$$Samp\_Freq\_Loop\_Noise\_dB(FrqPoint) := Ref\_in\_LoopGain\_dB(FrqPoint) + Samp\_Freq\_Noise(FrqPoint)$$

$$VCODiv\_Loop\_Noise\_dB(FrqPoint) := VCODiv\_LoopGain\_dB(FrqPoint) + VCODiv\_Noise(FrqPoint)$$

$$PD\_Loop\_Noise\_dB(FrqPoint) := PD\_Noise(FrqPoint) + PD\_LoopGain\_dB(FrqPoint)$$

$$Total\_Noise(FrqPoint) := 10\cdot\log\left(10^{\frac{VCO\_Loop\_Noise\_dB(FrqPoint)}{10}} + 10^{\frac{Samp\_Freq\_Loop\_Noise\_dB(FrqPoint)}{10}} + \blacksquare\right) + \blacksquare + 10\cdot\log\left(\blacksquare + 10^{\frac{PD\_Loop\_Noise\_dB(FrqPoint)}{10}} + 10^{\frac{VCODiv\_Loop\_Noise\_dB(FrqPoint)}{10}}\right)$$

$$Total\_Noise\left(10^{\frac{10\cdot\log(1.7\cdot k)}{10}}\right) = -96$$

$$Small\_Angle(FrqPoint) := -10\cdot\log(FrqPoint) - 30$$

$$Marker1 := \frac{1}{577\cdot\mu} \qquad Marker1 := 10^{\log(Marker1)} \qquad Marker2 := 135\cdot k \qquad Marker2 := 10^{\log(Marker2)}$$

Phase Noise at:- L(1.5 kHz)= **Phase Noise at:-** L(800 kHz)=

$$Total\_Noise\left(10^{\frac{10\cdot\log(1.5\cdot k)}{10}}\right) = -95.647 \text{ dBc/Hz} \qquad Total\_Noise\left(10^{\frac{10\cdot\log(800\cdot k)}{10}}\right) = -125.955 \text{ dBc/Hz}$$

Filter Values:-

$C1 = 3.665\times10^{-9}$ F

$C2 = 3.635\times10^{-8}$ F

$R2 = 146.14$ Ohms

$R3 = 23.342$ Ohms

$C3 = 3.665\times10^{-10}$ F

Loop Gain Values:-

$K_{vco} = 1\times10^{8}$ Hz/V

$K\phi = 0.002$ Amps

$F_{samp} = 4\times10^{7}$ Hz

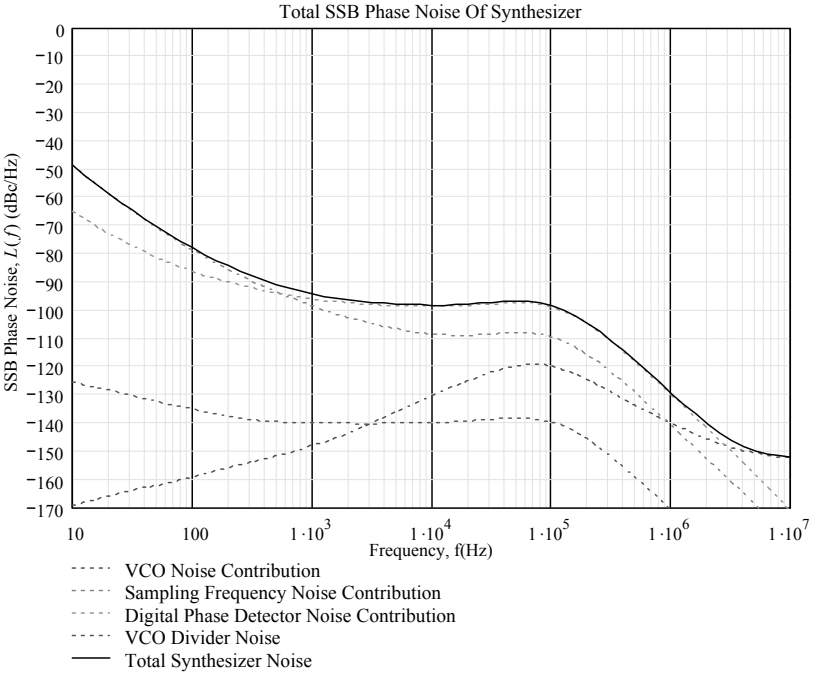
$F_{vco} = 1.725\times10^{9}$ Hz

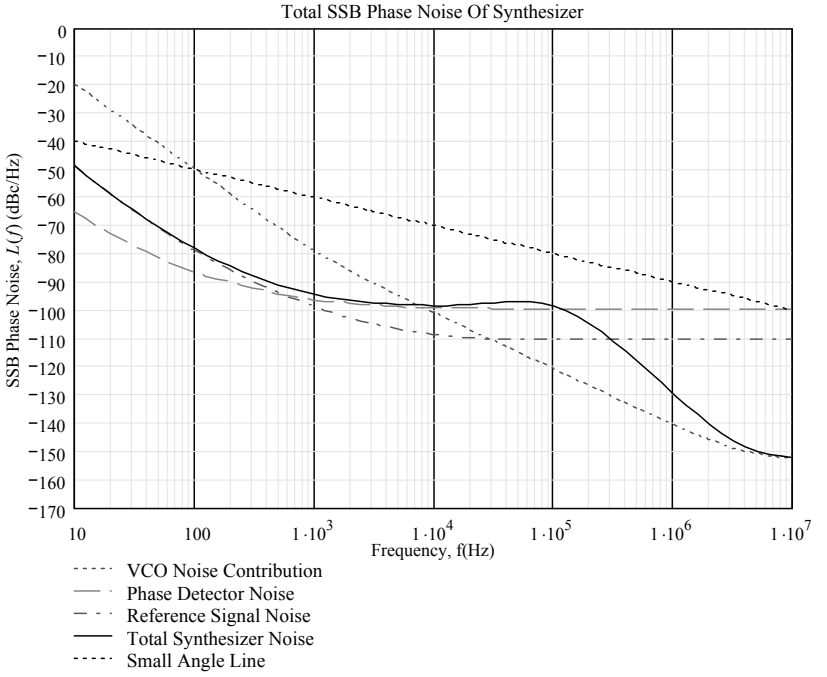
Loop Filter Dynamic Values:-

Phase Margin:-

$LBW = 1\times10^{5}$ Hz

$\phi p = 56$ Degrees


Total SSB Phase Noise Of Synthesizer
SSB Phase Noise, L(f) (dBc/Hz)
0
−10
−20
−30
−40
−50
−60
−70
−80
−90
−100
−110
−120
−130
−140
−150
−160
−170
10
100
1·10^3
1·10^4
1·10^5
1·10^6
1·10^7
Frequency, f(Hz)
VCO Noise Contribution
Sampling Frequency Noise Contribution
Digital Phase Detector Noise Contribution
VCO Divider Noise
Total Synthesizer Noise


Total SSB Phase Noise Of Synthesizer
SSB Phase Noise, L(f) (dBc/Hz)
0
−10
−20
−30
−40
−50
−60
−70
−80
−90
−100
−110
−120
−130
−140
−150
−160
−170
10
100
1·10^3
1·10^4
1·10^5
1·10^6
1·10^7
Frequency, f(Hz)
VCO Noise Contribution
Phase Detector Noise
Reference Signal Noise
Total Synthesizer Noise
Small Angle Line

2 MATLAB™ PROGRAM USED FOR THE SIMULATIONS OF THE FRACTIONAL-N PLL NOISE SPECTRUM

```
% Definition of the PLL elements
f_max=1898.208/90;
f_min=1880.928/92;
f_s=20e06;

N=91;
V_max=5;
V_min=0;

% Wanted T_L approx 2*pi/Wn <= 30 us
% <=> Wn >= 2*pi/ 30 us ~ 209.439 krads
T_L=30e-06;
Wn=2*pi/T_L;

xi=0.7;

K_pfd=(V_max-V_min)/(4*pi);
% gain for classic PFD with CP
K_vco=2*pi*20e06/(V_max-V_min);
K_filt=1;
K_t=K_vco*K_pfd*K_filt;
K=K_t/N;
K_gain=1000;
% K> 1.3614 = Wn*N/(2*xi*K_t) insure that Wz >=0
K=K*K_gain;

Wp= (Wn^2)/K;

% Wp/(Wn)^2 =1/K;
% xi = 0.5* (Wp/Wn + Wn/Wz)
% 1/Wz= 2 *xi/Wn – 1/ K

if (–1/K  +2 *xi/Wn) >=0
   Wz= 1/(–1/K + 2 *xi/Wn);
else
   Wz=0;
   sprintf('Increase the loop gain or the lock in time')
end
```

```
dsm_order=5;
f_ref=20e06;

% Noise contribution in the  PLL

f_min=1;    % 10^fmin=10 Hz
f_max=7;    % 10^fmax=10 MHz
N_f=100;        % Number of points
f=logspace(f_min,f_max,N_f);
s=j*2*pi*f;

% Definition of the close-loop transfer function H and 1-H

Hcl_2nd_order=Wn^2*[s/Wz + 1]./[(s.^2) + 2*xi*Wn*s+ Wn^2];
% Closed-Loop Gain of 2nd order
Gol_2nd_order=Hcl_2nd_order./(1-Hcl_2nd_order);
 % Open-Loop Gain of 2nd order
Wp2=10*Wp;
% Further suppression at 10 times the frequency
Gol_3rd_order=Gol_2nd_order./(1+s/Wp2);
% Open-Loop Gain of Loop Filter + Suppression Filter
Hcl_3rd_order=Gol_3rd_order./(1+Gol_3rd_order);
% Closed-Loop Gain of 3rd Order
Hcl=Hcl_3rd_order;
% Closed-Loop Gain of 3rd Order
one_min_H_cl=1-Hcl;
% Complement of Closed-Loop Gain of 3rd order
%semilogx(f,10*log10(abs(Hcl)))

% contribution from the VCO
f_p1=50;
f_z1=10^(6.5);
s_phi_vco1_f=100./((f/f_p1).^3).*((f/f_z1).^3+1);
% article micro and RF nov 1994
% does not qualify as the phase noise is too high at high frequency
% ... and S_phi_out = S_phi_vco at high frequency !!

s_phi_vco2_f= 10^-14.5+10^-1.5 ./f.^3+10^-1.5 ./f.^2;
% formula p 171 poly goldberg 1998
s_phi_vco_f=s_phi_vco2_f;
figure(1)
```

```
subplot(2,1,1),semilogx(f,10*log10(s_phi_vco1_f),'b',f,10*
log10(s_phi_vco2_f),'r')
title('different Phase noise characteristic for the crystal oscillator')

% contribution from the crystal frequency reference
% p 93 poly Goldberg, state of the art Xtal
s_phi_state1(1)=10^(−10);
s_phi_state1(2)=10^(−13);
s_phi_state1(3)=10^(−14.3);
s_phi_state1(4)=10^(−15.8);
s_phi_state1(5)=10^(−16.4);
s_phi_state1(6)=10^(−17);
s_phi_state1(7)=10^(−17);
f1=logspace(0,7,8);
s_phi_in1_f=s_phi_state1(1)./(f.^3).*((f/10) + 1).*
((f/100) + 1).*((f/5e03) + 1);
% s_phi_in1_f is the linear approximation from the vco p93

% Microwave and RF 1994
s_phi_in2_f=1e-04./((f).^3).*((f/10^3).^2+1).*((f/(4*10^4))+1);
% (Pretty noise at low frequencies !!)

% too ideal xtal below !! (from example)
s_phi_in3_f=1e-011./((f).^3).*((f/10).^2 + 1).*((f/100) + 1);

% phase noise of Hy-Q oscillator
s_phi_in4_f=10^(-5.3)./(f.^4).*((f.^3/5e04) + 1).*((f/8e03) + 1);

figure(1)
subplot(2,1,2),semilogx(f,10*log10(s_phi_in1_f),'b',...
f,10*log10(s_phi_in2_f),'r',f,10*log10(s_phi_in3_f),'g',f,10*
log10(s_phi_in4_f),'m')
title('different Phase noise characteristic for the crystal oscillator')

s_phi_in_f=s_phi_in1_f;
% s_phi_in3 is too ideal  and  s_phi_in2 is too bad (too much noise
% with in the loop bandwidth, one should achieve −80 dBc for DECT

% contribution from the loop filter
s_phi_loop=10^(-11)./((f).^3).*((f/(30)).^2+1).*((f/1e02)+1);

% contribution from the frequency detector
```

```
% sphi= 10^{−10.6 +/−0.3}/f −22 dB

s_phi_pd_f=10^(−2.2)*10^(−10.6)./f;

%s_phi_pd=10^(−22/10)* tf([10^(−10.6)],[1 0]);

% contribution from the frequency divider
% Integer-N case
s_phidn=(10^(−14.7)./f + 10^(−16.5));

s_phi_dn_int=s_phidn;
%s_phi_dn_int=s_phidn + s_phidn.*abs(Hcl).^2;

% plot of the phase noise source before filtering, N integer
figure(2)
subplot(2,1,1),semilogx(f,10*log10(s_phi_dn_int),'b',f,10*
log10(s_phi_pd_f),'r',...
f,10*log10(s_phi_loop),'g',...
f,10*log10(s_phi_in_f),'m',f,10*log10(s_phi_vco_f),'c');
grid on
set(gcf,'DefaultTextColor','k')
xlabel('frequency (Hz)')
ylabel('S_\Phi (dB)')
title('Phase noise source before filtering N integer')
set(gcf,'DefaultTextColor','c')
text(1e03,-70,'S_\Phi_{ vco}')
set(gcf,'DefaultTextColor','m')
text(1e01,-120,'S_\Phi_{ ref}')
set(gcf,'DefaultTextColor','b')
text(1e06,-170,'S_\Phi_{ dn}')
set(gcf,'DefaultTextColor','r')
text(1e03,-150,'S_\Phi_{ pfd}')
set(gcf,'DefaultTextColor','g')
text(1e06,-154,'S_\Phi_{ loop filter}')
set(gcf,'DefaultTextColor','k')

% Fractional-N case
% depends on the variable dsm_order
s_phi_switch=(2*pi)^2/(12*f_ref).*abs(2*sin(f*pi/f_ref)).^(2*
(dsm_order-1));
s_phi_dn_frac=s_phidn+s_phi_switch;
```

```
% plot of the phase noise source before filtering , N Fractional
figure(2)
subplot(2,1,2),semilogx(f,10*log10(s_phi_dn_frac),'b',f,
10*log10(s_phi_pd_f),'r',...
f,10*log10(s_phi_loop),'g',...
f,10*log10(s_phi_in_f),'m',f,10*log10(s_phi_vco_f),'c');
grid on
set(gcf,'DefaultTextColor','k')
xlabel('frequency (Hz)')
ylabel('S_\Phi (dB)')
title('Phase noise source before filtering, N frac')
set(gcf,'DefaultTextColor','c')
text(1e03,-70,'S_\Phi_{ vco}')
set(gcf,'DefaultTextColor','m')
text(1e01,-120,'S_\Phi_{ ref}')
set(gcf,'DefaultTextColor','b')
text(1e06,-170,'S_\Phi_{ dn}')
set(gcf,'DefaultTextColor','r')
text(1e03,-150,'S_\Phi_{ pfd}')
set(gcf,'DefaultTextColor','g')
text(1e06,-154,'S_\Phi_{ loop filter}')
set(gcf,'DefaultTextColor','k')

% Summation of all the contribution
S_phi_inband_int=(s_phi_in_f+s_phi_dn_int+s_phi_pd_f)*N^2.*
(abs(Hcl)).^2;
% multiply by |H|^2
S_phi_out=(s_phi_vco_f+s_phi_loop./f.^2).*(abs(one_min_H_cl)).^2;
% multiply by |1-H|^2
S_phi_tot=S_phi_inband_int+S_phi_out;

figure(3)

subplot(2,1,1),semilogx(f,10*log10(s_phi_dn_int*N^2.
*(abs(Hcl)).^2),'b',...
f,10*log10(s_phi_pd_f*N^2.*(abs(Hcl)).^2),'r',...
f,10*log10(s_phi_loop./f.^2 .*(abs(one_min_H_cl)).^2),'g',...
f,10*log10(s_phi_in_f*N^2.*(abs(Hcl)).^2),'m',...
f,10*log10(s_phi_vco_f.*(abs(one_min_H_cl)).^2),'c',
f,10*log10(S_phi_tot),'b-.');
grid on
set(gcf,'DefaultTextColor','k')
```

```
xlabel('frequency (Hz)')
ylabel('S_\Phi (dB)')
title('Phase noise source after filtering')
set(gcf,'DefaultTextColor','c')
text(1e06,-130,'(1-H)^2 S_\Phi_{ vco}')
set(gcf,'DefaultTextColor','m')
text(1e01,-70,'N^2 H^2 S_\Phi_{ ref}')
set(gcf,'DefaultTextColor','b')
text(1e07,-165,'N^2 H^2 S_\Phi_{ dn}')
set(gcf,'DefaultTextColor','r')
text(1e05,-200,'N^2 H^2 S_\Phi_{ pfd}')
set(gcf,'DefaultTextColor','g')
text(1e02,-250,'(1-H)^2 f^{-2} S_\Phi_{ loop filter}')
set(gcf,'DefaultTextColor','k')

% DECT phase noise mask
N_l=4*N_f;
f_l=logspace(log10(1.2e06),f_max,N_l);
dect_mask(1:2)=10^-(8);
dect_mask(3:4)=10^-(8.5);
 % -85 dBc @ 100 kHz
decal=4;
dect_mask(decal+1:decal+174)=10^(−9.5);
f_l(1)    =1.2 MHz
dect_mask(decal+175:decal+257)=10^(−11.7);
 % f_l(174)   ~3 MHz
dect_mask(decal+258:decal+N_l)=10^(−13.5);
% f_l(257)   4.67 MH < 4.7 MHz

figure(4)

subplot(2,1,1),semilogx(f,10*log10(S_phi_tot),'b',f,10*
log10(s_phi_vco_f),'c',...
[10,1e05-1,1e05,1.2000e+06,f_l],10*log10(dect_mask),'k')
title('Overall phase noise and phase mask');
grid

% fractional case
S_phi_inband_frac=(s_phi_in_f+s_phi_dn_frac+s_phi_pd_f)*N^2.*
(abs(Hcl)).^2;
S_phi_tot_frac=S_phi_inband_frac+S_phi_out;
```

```
figure(3)
subplot(2,1,2),semilogx(f,10*log10(s_phi_dn_frac*N^2.
*(abs(Hcl)).^2),'b',...
f,10*log10(s_phi_pd_f*N^2.*(abs(Hcl)).^2),'r',...
f,10*log10(s_phi_loop./f.^2 .*(abs(one_min_H_cl)).^2),'g',...
f,10*log10(s_phi_in_f*N^2.*(abs(Hcl)).^2),'m',...
f,10*log10(s_phi_vco_f.*(abs(one_min_H_cl)).^2),'c',
f,10*log10(S_phi_tot_frac),'b-.');
grid on
set(gcf,'DefaultTextColor','k')
xlabel('frequency (Hz)')
ylabel('S_\Phi (dB)')
title('Phase noise source after filtering')
set(gcf,'DefaultTextColor','c')
text(1e06,-130,'(1-H)^2 S_\Phi_{ vco}')

set(gcf,'DefaultTextColor','m')
text(1e01,-70,'N^2 H^2 S_\Phi_{ ref}')
set(gcf,'DefaultTextColor','b')
text(1e07,-165,'N^2 H^2 S_\Phi_{ dn}')
set(gcf,'DefaultTextColor','r')
text(1e05,-200,'N^2 H^2 S_\Phi_{ pfd}')
set(gcf,'DefaultTextColor','g')
text(1e02,-250,'(1-H)^2 f^{-2} S_\Phi_{ loop filter}')
set(gcf,'DefaultTextColor','k')

figure(4)
subplot(2,1,2),semilogx(f,10*log10(S_phi_tot_frac),'b',f,10*
log10(s_phi_vco_f),'c',...
[10,1e05-1,1e05,1.2000e+06,f_l],10*log10(dect_mask),'k')
title('Overall phase noise and  phase mask');
grid
```

The following M-file reads in the captured output of the simulated HDL of the delta—sigma block as applied to the MMD.

```
clear all;
Ts=25; %sampling period is 24nS
fid=fopen('HDL_deltasigma_out_data70p5.dat','r');
out=fscanf(fid,'%i %i %i\n',[3 inf]);
fclose(fid);
```

```
%M=1000044;
%M=length(out(1,:));

frac=out(2,:);
int=out(1,:);
inst_div=out(3,:);

%M=length(frac);
M=2^19;
y2=inst_div;

y2=y2(20001:M);
mean(y2)
M=length(y2);
fs=1/(Ts*1e-9);
f=(1:M)*fs/M/1e3;
win=hanning(M);
win=win';
a2=abs(fft(y2.*win));
a2=(a2.*a2)/M;
a2=a2(1:M/2);
f=f(1:M/2);
a2=10*log10(a2/max(a2));
figure(1),plot(f,a2);
xlabel('FREQUENCY (KHz)');
ylabel('MAGNITUDE (dB)');
axis([min(f) max(f) -350 0]),grid;
title('Output spectrum of Nemo Delta—Sigma Modulator (VHDL)');
delta_f=fs/M;
place=floor(1000e3/delta_f);
a3=a2(1:place);
f3=f(1:place);
figure(2),plot(f3,a3);
xlabel('FREQUENCY (KHz)');
ylabel('POWER SPECTRUM (dB)');
axis([min(f3) max(f3) -350 0]),grid;
title('Baseband output spectrum of Nemo Delta—Sigma Modulator (VHDL)');

fB=200e03;            % The cutoff frequency = fpass
fstop=2000e03;        % The stop frequency
rp=3;                 % pass band attenuation in dB
rs=30;                % stop band attenuation in dB
wp=fB*2/fs;       % pass normalized frequency
ws=fstop*2/fs;    % stop normalized freuqnecy
[filt_order,wn]=buttord(wp,ws,rp,rs);
[b1,a1]=butter(filt_order,wn);  % calculate coefficents of butterworth filter
filtered=filter(b1,a1,y2);
```

```
a4=abs(fft(filtered.*win));
a4=(a4.*a4)/M;
a4=a4(1:M/2);
a4=10*log10(a4/max(a4));

figure(3),plot(f,a4);
xlabel('FREQUENCY (KHz)');
ylabel('MAGNITUDE (dB)');
axis([min(f) max(f) -350 0]),grid;
title('Filtered spectrum of MASH output');

% to be able to print the results you have to decimate the results before
presenting
a5=a2(1:64:length(a2));
f5=f(1:64:length(f));
figure(4),plot(f5,a5);
xlabel('FREQUENCY (KHz)');
ylabel('MAGNITUDE (dB)');
axis([min(f5) max(f5) -350 0]),grid;
title('Decimated output spectrum of MASH');

a6=a3(1:64:length(a3));
f6=f3(1:64:length(f3));
figure(5),plot(f6,a6);
xlabel('FREQUENCY (KHz)');
ylabel('MAGNITUDE (dB)');
axis([min(f6) max(f6) -350 0]),grid;
title('Decimated baseband output spectrum of MASH');

a7=a4(1:64:length(a4));
figure(6),plot(f5,a7);
xlabel('FREQUENCY (KHz)');
ylabel('MAGNITUDE (dB)');
axis([min(f5) max(f5) -350 0]),grid;
title('Decimated filtered spectrum of MASH output');
```

Index