6.2 System Description

The architecture proposed in this chapter is an extension for the architectures discussed in Chapters 4 and 5. While these designs represented an FPGA version, the one discussed here is the ASIC version. A block diagram based on Chapter 4 architecture is shown in Figure 6.1. The digital blocks, namely, accumulator and divider, are written in HDL, synthesized and implemented using standard digital IC design flow. The DTC is custom designed, thus we focus on it in this chapter. However, even in the custom designed blocks, standard cells where used whenever possible, in order to simplify the design process. The PLL plays a dual role. Not only it provides the oversampling clock, but also it generates 16 phases used for the phase interpolation process. Thus, an extra tapped delay line using a DLL is not required.

As an ASIC implementation, this design provides several enhancements over the FPGA version. First, the design is self-contained, i.e., the user is not asked to provide an oversampling clock as an input. Second, the system can operate at a much higher frequency, the oversampling clock (f_{OV}) is about 1 GHz, which is a 10x improvement over the FPGA version. Consequently, the system clock (f_{clk}) is also higher (120 MHz), which enables the synthesis of a wider range of frequencies. Third, the clock frequency allows a more fine timing resolution, thus, jitter performance is improved. Fourth, the width of the delay word (DW) is increased to 7 bits, i.e., the resolution of the DTC is improved, which means an increase in SFDR. A summary of the targeted specifications in the design is shown in Table 6.1.

6.3 Design of Phase Locked Loop

One of the main benefits of the proposed architecture is that the design of the PLL is greatly simplified. The PLL is working in a static mode at a fixed frequency. Therefore, there are no strict requirements on the dynamic

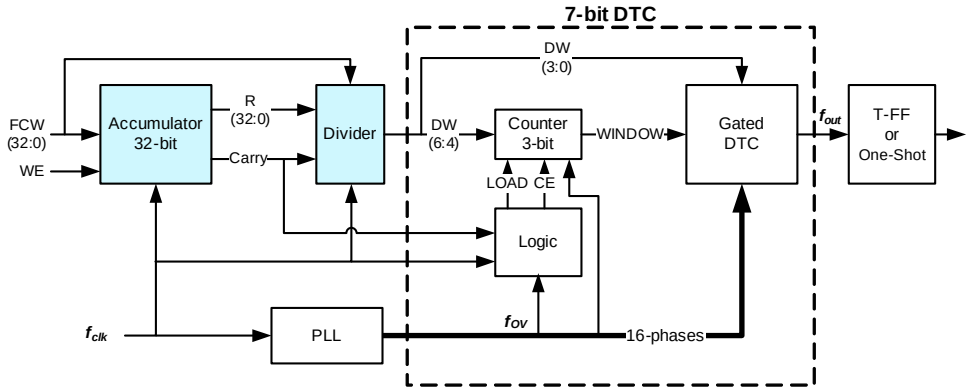


Figure 6.1: Block diagram of the proposed architecture.

Table 6.1: Summary of the DDS targeted specifications.

Implementation	0.13 μm CMOS process
Clock frequency (f_{clk})	120 MHz
No. of interpolation levels (2^D)	128
Effective clock (f_{eff})	15.36 GHz
Maximum output frequency	$f_{clk}/2$
Resolution (Δf)	0.028 Hz
Switching speed	< 0.1 μs
Maximum peak to peak deterministic jitter	65 ps

Table 6.2: Summary of the PLL targeted specifications.

Spec	Value
Input clock frequency (f_{clk})	120 MHz
Output clock (f_{OV})	960 MHz
Divider ratio (N)	8 (fixed)
VCO stages	8 (differential)
Switching speed	No requirement
Tuning resolution	No requirement

behavior of the PLL. Thus, the design process can be focused on loop stability, rather than the complex trade-offs that complicates the design problem. The targeted specifications for the PLL are shown in Table 6.2.

Although simplified, PLL design is still a challenging task involving many parameters to tune. Starting with transistor level simulations and using bottom-up design methodology is not the right way to follow. This methodology will result in a large number of design iterations, each one requires hours, if not days, of simulation time. In addition, the designer will be faced by a very large number of parameters to track, and he will not be able to understand the effect of each one on the system performance. To address these problems, a top-down design methodology is usually preferred.

Top-down flow [Kundert 00, Ahmed 02, Kundert 04] starts by system level design, where a particular architecture is chosen, and the system specifications are mapped to requirements on various blocks in the architecture. Next, the architecture is verified using system level behavioral simulations, e.g., MATLAB, Simulink, Verilog-A, HDL-AMS... etc. The following step is to design and verify each block as a unit. For digital blocks, this step can follow directly from HDL code written for system simulations, by using digital synthesis and implementation CAD tools. For analog blocks, designers still spend painful hours tweaking transistors till required specifications are met. In a bottom-up verification approach, the behavioral models should

be updated by results from blocks implementation. This allows a second more accurate iteration of system verification. However, trying to accurately model analog blocks is a difficult and time consuming task. Thus, this step is sometimes skipped [Kundert 00].

A more efficient and common approach for verification, is using mixed-signal simulation environment, together with the concept of incremental integration. To verify a block with mixed-level simulation, the model of the block in the top-level schematic is replaced with the transistor level schematic of the block. The system, described at a high level, acts as a test-bench for the block, which is described at the transistor level. Thus, the block is verified in the context of the system, and it is easy to see the effect of imperfections in the block on the performance of the system [Kundert 04]. After performing this step for each block, the system is integrated and verified incrementally [Ahmed 02], i.e., using the transistor level of two blocks, then three, then four, till we reach full chip simulation. Although it seems that this approach requires a larger number of simulation runs, it turns out to reduce the overall verification time substantially. Incremental integration helps to locate and fix many problems and bugs when simulation time is minutes rather than hours. On the other hand, an iteration of full chip simulation is very expensive and sometimes impractical. In addition, integrating the whole system in one step makes debugging a difficult task, if not impossible.

6.3.1 System Level Design

A block diagram of the ubiquitous charge pump PLL is shown in Figure 6.2. The charge pump PLL offers several advantages over the voltage phase detector and has almost replaced it [Banerjee 06]. Using the PFD, the PLL is able to lock to any input frequency, regardless of how far off it initially is in frequency, if not limited by the VCO range. In addition, for ideal charge pump, the PLL does not have a steady state phase error [Banerjee 06, Rhee 99]. A second order loop filter was chosen for simplicity. A third order

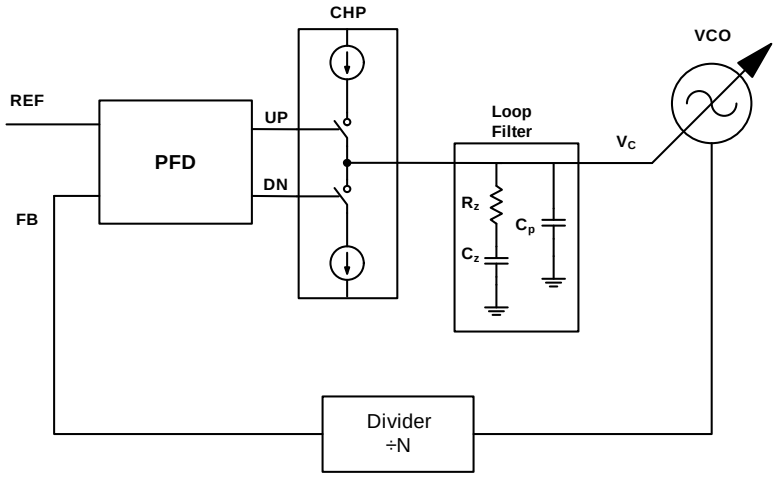


Figure 6.2: Block diagram of charge pump PLL.

filter is sometimes required for additional filtering of the VCO control voltage [Gardner 80].

6.3.1.1 Choice of Loop Parameters

Detailed mathematical analysis of Charge Pump PLL with a second-order loop filter can be found in many excellent references [Keese 96, Best 03, Shu 05, Banerjee 06, Rogers 06]. For the purpose of brevity, some mathematical steps will be skipped in the following design procedure, while complete rigorous derivations can be found in the references mentioned before.

The open loop transfer function of the PLL is given by [Shu 05]:

$$H_{OL}(s) = \frac{I_{CHP} K_{VCO} Z(s)}{sN} \tag{6.1}$$

where I_{CHP} is the charge pump current, K_{VCO} is the VCO gain (Hz/V), N is the frequency divider ratio, and $Z(s)$ is the loop filter impedance given

by:

$$Z(s) = \frac{1}{C_z + C_p} \frac{sRC_z + 1}{s(sRC_s + 1)} \tag{6.2}$$

$$C_s = \frac{C_z C_p}{C_z + C_p} \tag{6.3}$$

The open loop gain can be rewritten as [Amin 09]:

$$H_{OL}(s) = \frac{K(s/\omega_z + 1)}{s^2(s/\omega_p + 1)} \tag{6.4}$$

where

$$K = \frac{I_{CHP} K_{VCO}}{(C_z + C_p) N} \tag{6.5}$$

$$\omega_z = \frac{1}{R_z C_z} = \frac{1}{\tau_z} \tag{6.6}$$

$$\omega_p = \frac{1}{R_z C_s} = \frac{1}{\tau_p} \tag{6.7}$$

Due to the pole associated with the VCO, the PLL is one order higher than the loop filter, i.e., third-order. C_z generates a pole at the origin and R_z is used to generate a zero for loop stability. C_p is usually much smaller than C_z as we will see later, and it is used to generate a second pole to remove high frequency ripples on the VCO control line.

The stability of the feedback loop can be analyzed by calculating the phase margin (ϕ_m). From Equation (6.4), we can write the phase margin as [Shu 05]:

$$\phi_m = \tan^{-1}\left(\frac{\omega_u}{\omega_z}\right) - \tan^{-1}\left(\frac{\omega_u}{\omega_p}\right) \tag{6.8}$$

where (ω_u) is the unity gain frequency, also known as cross-over frequency. It is clear that to maximize the phase margin, ω_u must be placed between ω_z and ω_p. By differentiating Equation (6.8) we can find that the maximum phase margin is obtained when ω_u is placed at the geometric mean of ω_z and

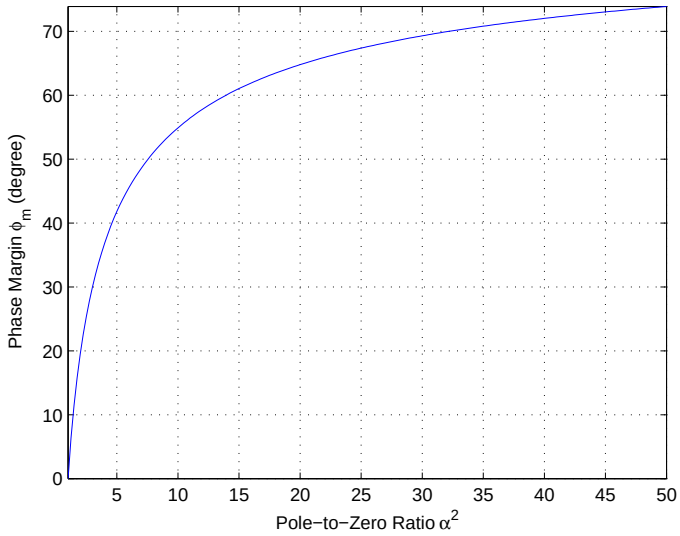


Figure 6.3: Optimum phase margin vs. pole-to-zero ratio.

ω_p [Vaucher 02]:

$$\omega_u = \sqrt{\omega_z \omega_p} = \alpha \omega_z = \frac{\omega_p}{\alpha} \tag{6.9}$$

where α^2 is the pole-to-zero ratio, which is approximately equal to the capacitance ratio [Ahmed 02, Shu 05]:

$$\alpha^2 = \frac{\omega_p}{\omega_z} = \frac{\tau_z}{\tau_p} = \frac{C_z}{C_s} = 1 + \frac{C_z}{C_p} \tag{6.10}$$

and C_s is given by Equation (6.3).

Substituting back into Equation (6.8) yields ϕ_m as a function of α [Shu 05]:

$$\phi_m = \tan^{-1}\left(\frac{\alpha^2 - 1}{2\alpha}\right) \tag{6.11}$$

This equation can be used to determine the capacitance ratio for a given phase margin. Equation (6.11) is plotted in Figure 6.3.

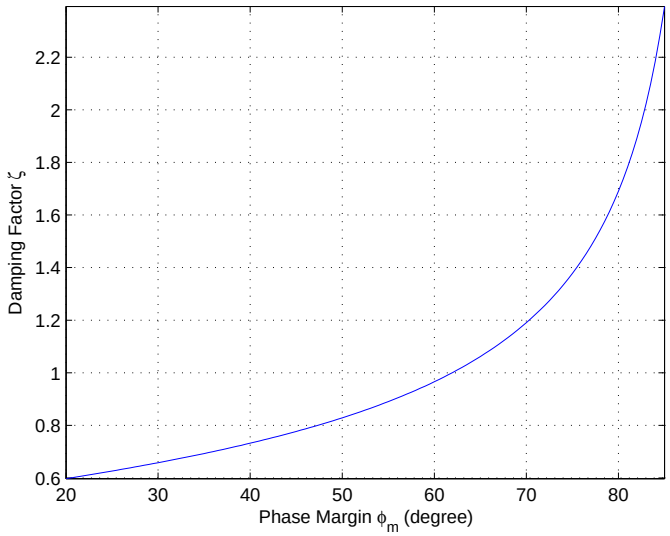


Figure 6.4: Damping factor vs. phase margin.

The phase margin is usually chosen between 30^o and 70^o [Keese 96]. The choice of phase margin is a trade-off between stability and dynamic behavior [Best 03]. Increasing ϕ_m means increased stability, but the damping factor (ζ) is increased[1], which means slower loop response time [Keese 96]. The relation between phase margin and damping factor is given by [Banerjee 06]:

$$\sec(\phi_m) - \tan(\phi_m) = \frac{1}{4\zeta^2} \tag{6.12}$$

and is plotted in Figure 6.4. Since we have no restrict requirements on the dynamic behavior, we can design for a relatively high phase margin (e.g., 70^o), in order to cope with the harsh PVT variations.

The second critical parameter in the design of the PLL is the loop band-

[1]It is worth noting that damping factor (ζ) and natural frequency (ω_n) are used to describe second-order systems, thus, using them with third-order PLL is a mere approximation which neglects the high frequency pole introduced by C_p [Vaucher 02, Banerjee 06].

width[2]. The choice of the loop bandwidth represents a trade-off between switching speed and the suppression of reference spurs [Ahmed 02, Banerjee 06]. In addition, all the previous continuous time analysis is valid only if the loop bandwidth is much smaller than the reference frequency (ω_{ref}) [Shu 05, Rogers 06]. A common rule of thumb is to choose the loop bandwidth as one tenth of the reference frequency [Gardner 05, Shu 05, Banerjee 06]. It should be noted as well, that the choice of the loop bandwidth affects the component values of the loop filter passive elements. This should be taken into consideration, such that these component values are reasonable and suitable for integration.

Given that the phase margin and the loop bandwidth are chosen, we can now proceed to calculate the time constants τ_z and τ_p [Keese 96, Banerjee 06]:

$$\tau_p = \frac{\sec(\phi_m) - \tan(\phi_m)}{\omega_u} \tag{6.13}$$

$$\tau_z = \frac{1}{\omega_u^2 \tau_p} \tag{6.14}$$

The values of the loop filter components can then be calculated as [Keese 96, Banerjee 06]:

$$C_p = \frac{\tau_p}{\tau_z} \frac{I_{CHP} K_{VCO}}{\omega_c^2 N} \sqrt{\frac{1 + (\omega_c \tau_z)^2}{1 + (\omega_c \tau_p)^2}} \tag{6.15}$$

$$C_z = C_p \left(\frac{\tau_z}{\tau_p} - 1 \right) \tag{6.16}$$

$$R_z = \frac{\tau_z}{C_z} \tag{6.17}$$

The above equations are relatively complex, thus they need to be simplified

[2]Many authors use the term "loop bandwidth" for the PLL open loop bandwidth, which is equal to the unity gain frequency, e.g., [Aktas 04, Gardner 05, Shu 05, Banerjee 06]. This is the convention that we are going to follow in our thesis. However, it should be noted that the relation between the unity gain frequency and the -3dB closed loop bandwidth (ω_{-3dB}) depends on the phase margin, e.g., $\omega_{-3dB} \approx 1.4\omega_u$ for $\phi_m = 70^o$ [Vaucher 02].

in order to give more insight to the designer. Using the approximation that $\alpha^2 \gg 1$, we can rewrite them as [Aktas 04]:

$$C_p = \frac{I_{CHP} K_{VCO}}{\alpha N \omega_u^2} \tag{6.18}$$

$$C_z = \frac{\alpha I_{CHP} K_{VCO}}{N \omega_u^2} \tag{6.19}$$

$$R_z = \frac{N \omega_u}{I_{CHP} K_{VCO}} \tag{6.20}$$

The value of K_{VCO} is usually forced by the VCO architecture. Circuit simulations show that K_{VCO} suffers from large PVT variations. It ranges from about 600 MHz/V to 6 GHz/V, with a typical value of 3.5 GHz/V, i.e., it varies by a factor of 10. Rather than designing for the typical value, the design value is taken as the geometric mean of the minimum and maximum values [Banerjee 06]. I_{CHP} gives the designer another degree of freedom to adjust the component values or compensate the variation in K_{VCO}. As we target a simple low-power design, we set I_{CHP} to 10 μA.

The above design procedure is written in a MATLAB script (Appendix B), and is used to compute loop parameters. Table 6.3 summarizes the results.

Table 6.3: Summary of PLL loop parameters.

	Parameter	Value
Input	ω_{ref}	$2\pi \times 120$ Mr/s
	ω_u	$2\pi \times 4$ Mr/s
	N	8
	ϕ_m	70^o
	K_{VCO}	600 MHz/V $\rightarrow$ 6 GHz/V
	I_{CHP}	10 μA
Output	$K_{VCO-mean}$	1.9 GHz/V
	ζ	1.2
	α^2	32
	C_p	0.662 pF
	C_z	20.6 pF
	R_z	10.9 $k\Omega$

6.3.1.2 MATLAB Simulation

The linear phase model is simulated using MATLAB to verify the open loop and closed loop behavior. Figure 6.5 shows the open loop characteristics for the extreme values of K_{VCO} along with their geometric mean. For the design value (i.e., the geometric mean), the phase margin is exactly 70^o, while it is about 60^o for both $K_{VCO-min}$ and $K_{VCO-max}$, which means that stability is ensured across PVT corners. The closed loop response is shown in Figure 6.6.

6.3.1.3 Behavioral Simulation

Behavioral time domain simulation of the PLL was done using Verilog-A. Transient simulation results for VCO control voltage while acquiring lock and PLL in locked state are shown in Figures 6.7 and 6.8, respectively.

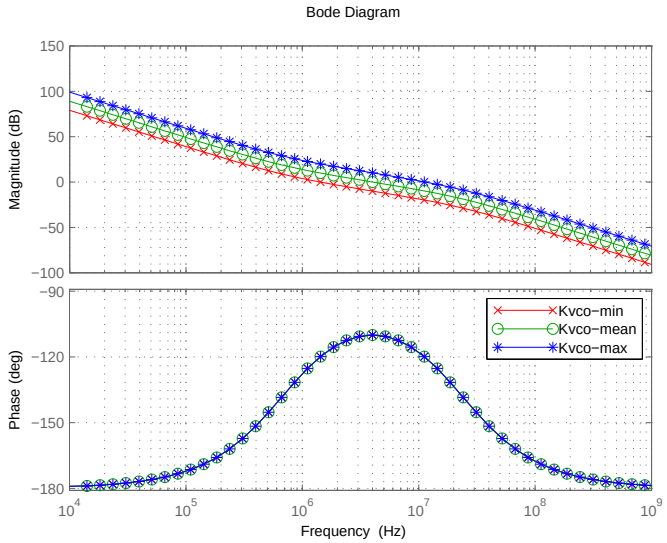


Figure 6.5: PLL open loop characteristics.

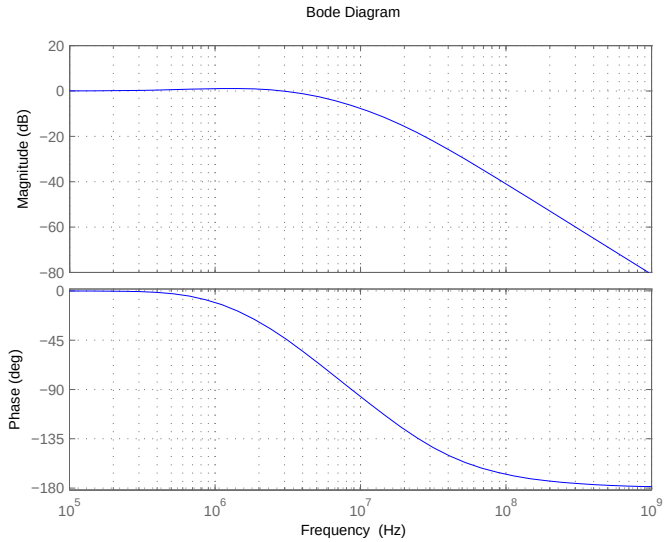


Figure 6.6: PLL closed loop characteristics.

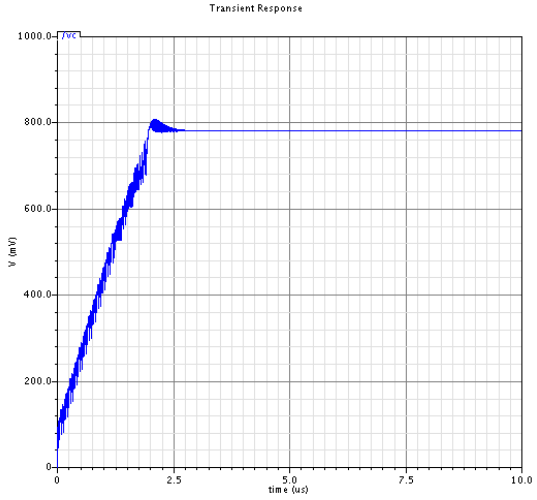


Figure 6.7: VCO control voltage (behavioral simulation).

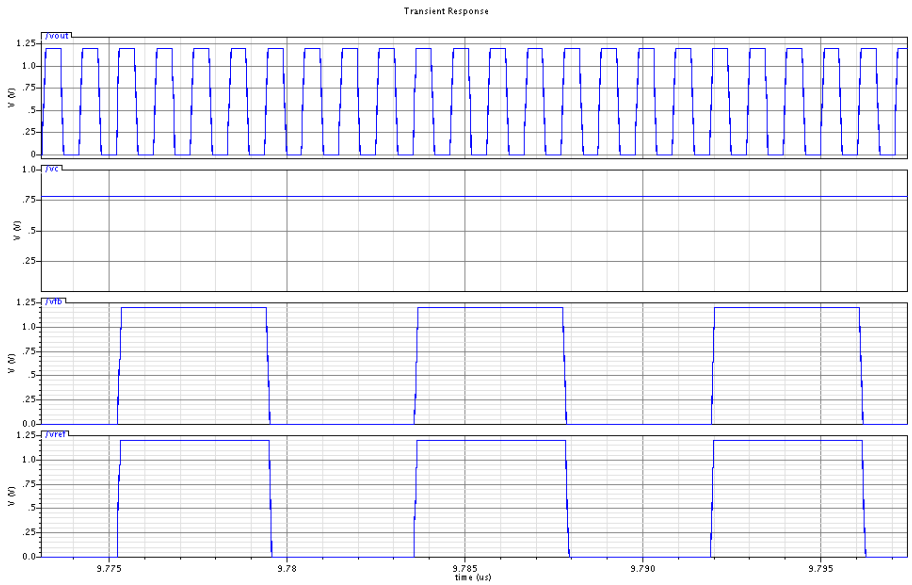


Figure 6.8: PLL in locked state (behavioral simulation). (1) VCO output (2) VCO control voltage (3) Divider output (4) Input reference.

6.3.2 Transistor Level Design

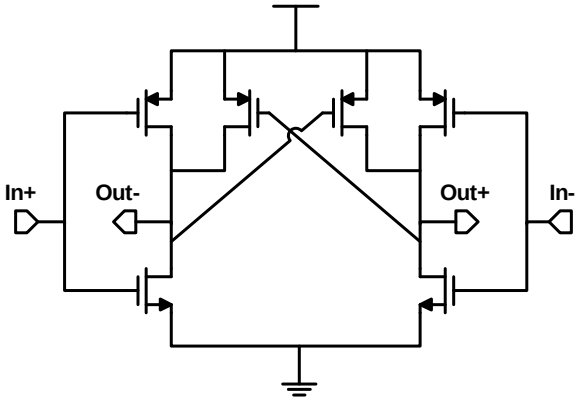
6.3.2.1 Voltage Controlled Oscillator

The voltage controlled oscillator (VCO) is designed in a ring architecture in order to provide the phases required in the phase interpolation process. As a 2^n number of phases is required, the ring must contain an even number of stages. Thus, a differential delay cell is necessary. A simple pseudo differential delay cell based on differential cascode voltage switch logic (DCVSL) was chosen [Rabaey 02, McNeill 09, Amin 09]. The schematic of the delay cell is shown in Figure 6.9a. The delay is controlled by varying the drive strength as shown in Figure 6.9b.

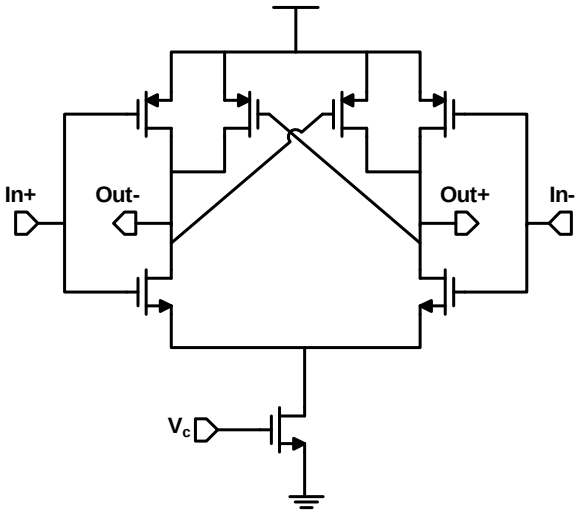
As 16 phases are required, an 8-stage differential ring VCO is used as shown in Figure 6.10. The VCO is working at 960 MHz across PVT variations. The tuning curves at worst cast PVT corners are shown in Figure 6.11.

6.3.2.2 Phase Frequency Detector

A typical digital tri-state phase-frequency detector (PFD) was used. The block diagram is shown in Figure 6.12. Standard cells were used for implementation. Transistor level simulation was carried out using netlists extracted from standard cells layout. It should be noted that the D-input of the flip-flop is driven by TIEHI standard cell, rather than using a direct connection to the positive supply, in order to provide ESD protection. Timing simulation of PFD is shown in Figures 6.13 and 6.14. The UP and DN signals are both high for a short time (t_{ON}), which is equal to the time delay in the reset path. An extra delay stage is usually inserted in the reset path to increase t_{ON}, in order to eliminate the dead zone problem. The dead zone problem is discussed in the next section.



(a) Basic delay cell.



(b) Voltage controlled delay cell.

Figure 6.9: Ring oscillator/VCO delay cell.

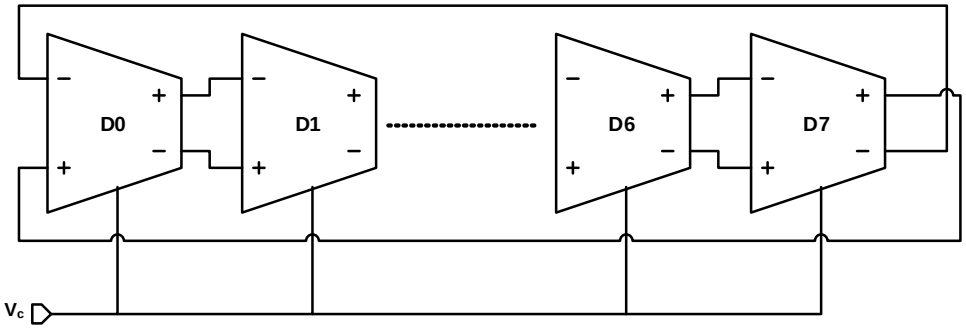


Figure 6.10: 8-stage differential ring VCO.

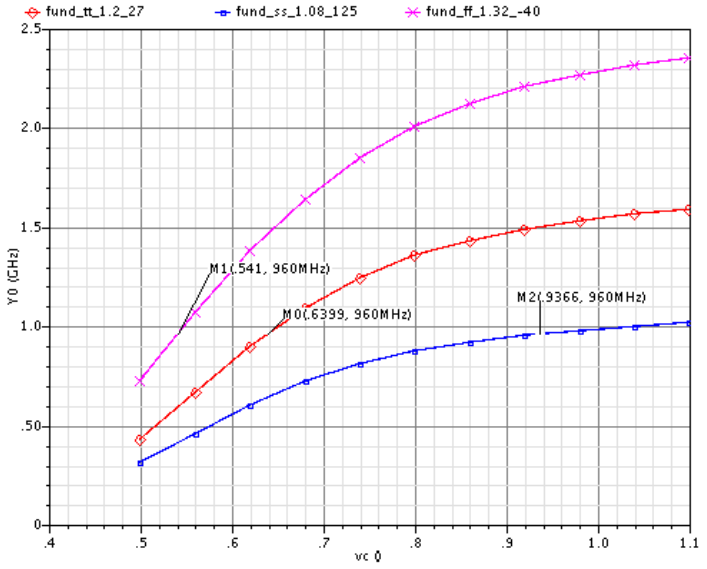


Figure 6.11: VCO tuning curves for PVT corners.

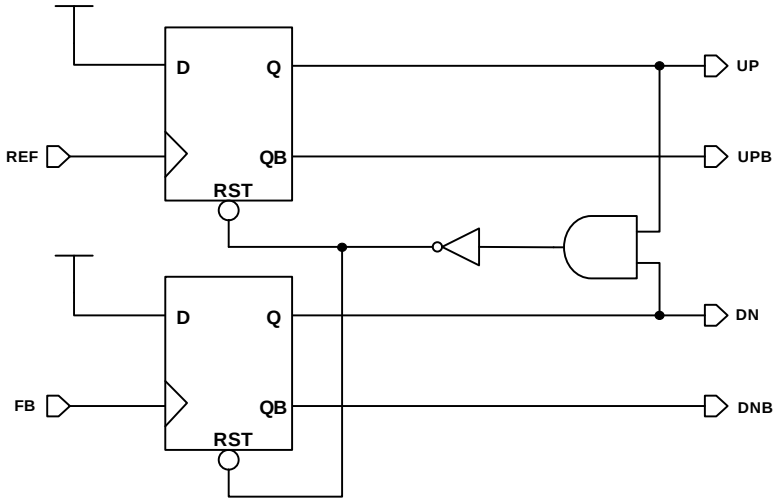


Figure 6.12: Tri-state phase frequency detector (PFD).

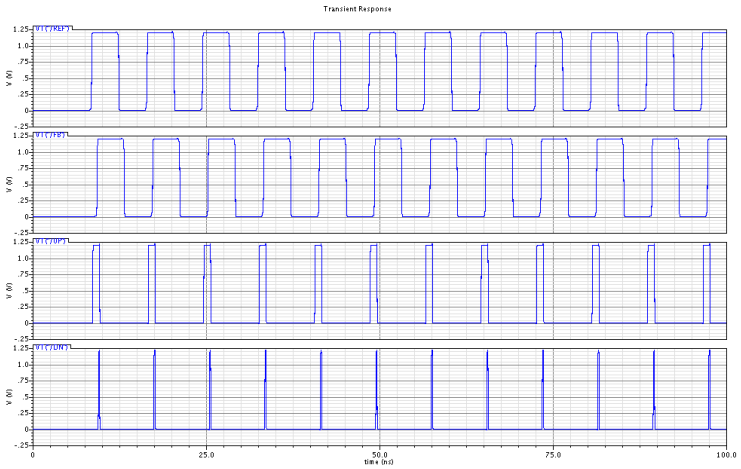


Figure 6.13: PFD timing when REF is leading (transistor level transient simulation). (1) REF (2) FB (3) UP (4) DN.

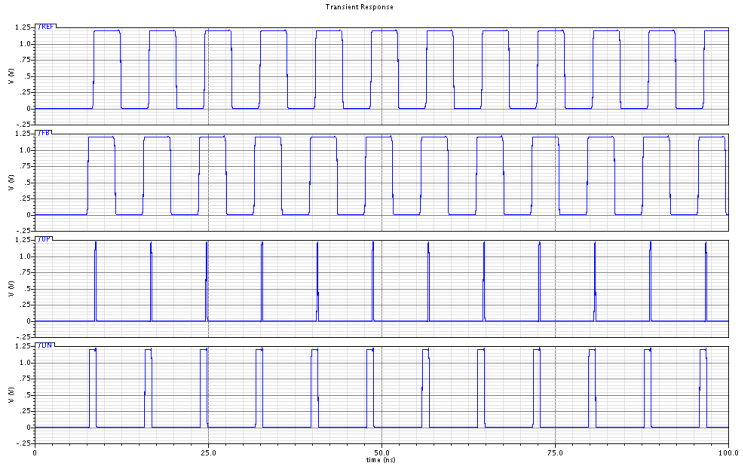
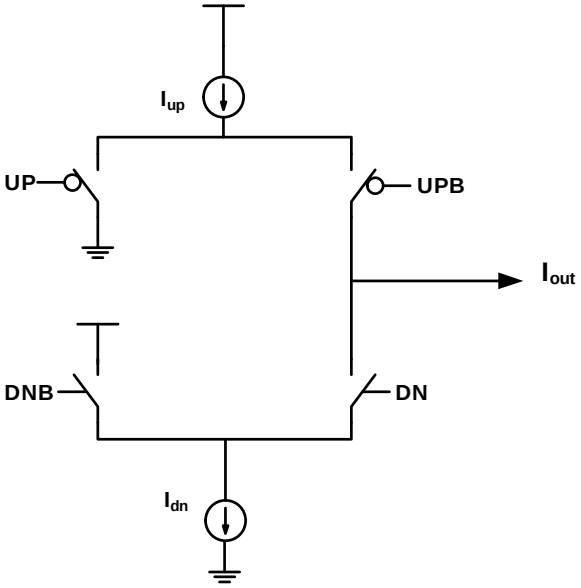


Figure 6.14: PFD timing when FB is leading (transistor level transient simulation). (1) REF (2) FB (3) UP (4) DN.

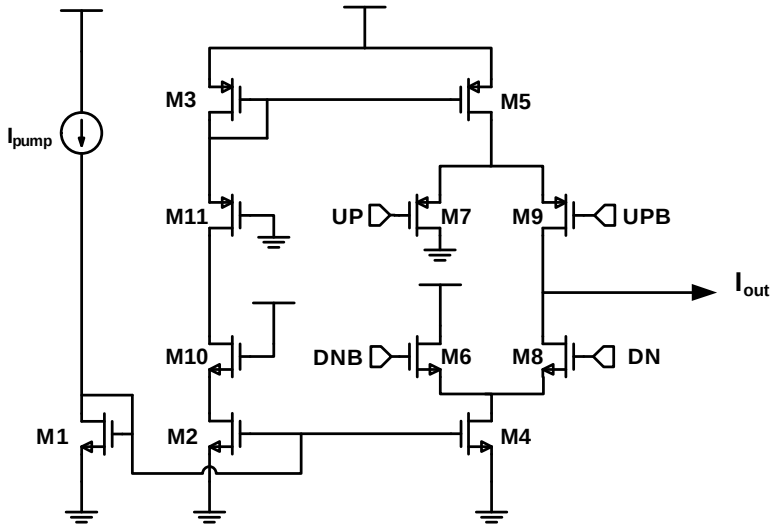
6.3.2.3 Charge Pump

A current steering charge pump was used as it provides fast transient response [Rhee 99, Magnusson 03, Shu 05]. The basic block diagram is shown in Figure 6.15a and the circuit implementation is shown in Figure 6.15b [Rhee 99, Amin 09]. M1 to M5 provide current mirror operation. These transistors are designed with large channel length to increase the output resistance. M6 to M9 are switches to provide the current steering function. These are designed with minimum channel length to increase the switching speed [Ahmed 02], and relatively large width to reduce their "on" resistance. M10 and M11 are added to improve matching between UP and DN currents, by making the voltage drop (V_{DS}) across the current mirror transistors as equal as possible [Sansen 06].

Charge pump non-ideal behavior is the main cause of reference spurs in PLL. Current mismatch, leakage current, and timing mismatch are only few of many non-ideal issues affecting the performance of the charge pump [Rhee 99, Shu 05]. Due to the finite output resistance of the current mirror,



(a) Conceptual diagram.



(b) Circuit implementation.

Figure 6.15: Current steering charge pump.

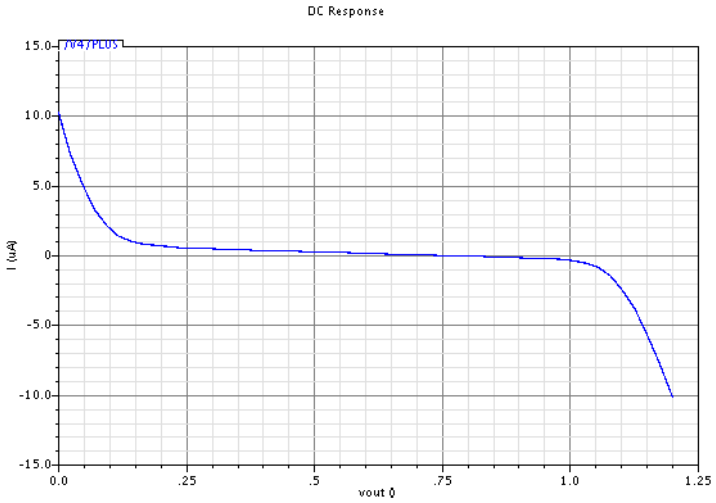


Figure 6.16: Charge pump DC current mismatch vs. output voltage.

a DC current mismatch between UP and DN currents exists. To test DC current mismatch, a DC source is placed at the output of the charge pump. When both UP and DN controls are set to logic high, the output current, ideally, is equal to zero. Thus, a plot of the output current vs. output voltage indicates the DC current mismatch [Ahmed 02]. The mismatch current of the designed charge pump is shown in Figure 6.16. The figure indicates acceptable current matching, as the error is less than 1% at the expected output voltage in typical conditions. The figure also shows wide compliance range, which was achieved by increasing the W/L ratio of the current mirror transistors.

To estimate the static phase error in steady state, dynamic mismatches should be taken into account as well. A transient simulation was run with both REF and FB in phase. Ideally, the output current should be zero. The average current is due to static and dynamic mismatches of PFD and charge pump. The average current was found to be less than 35 nA. From this infor-

mation we can calculate the phase/timing error in steady state [Ahmed 02]:

$$\Delta\phi_{error} = 2\pi \times \frac{\Delta I_{avg}}{I_{pump}} \approx 1.26^{o} \tag{6.21}$$

$$\Delta t_{error} = T \times \frac{\Delta I_{avg}}{I_{pump}} \approx 30\,ps \tag{6.22}$$

which is much less than the CLK-to-Q delay of the frequency divider in the FB path.

Another important phenomenon related to the PFD and the charge pump is the dead zone problem. When the phase error is very small, the charge pump cannot respond to the narrow UP and DN pulses. This causes a non-linear flat region in charge pump characteristics as shown in Figure 6.17. In the flat region, the gain is lower than the ideal one and it tends to zero. For this reason, it is usually called the dead zone. The dead zone should always be avoided in the design of a PLL. The open-loop gain would be highly reduced at small phase errors, leading to potential stability problems. The phase error at the steady state may wander around the region where the gain is almost zero, which translates into spurious tones at the VCO output. To solve this problem, an extra delay stage is usually inserted in the reset path to increase the minimum pulse width of UP and DN signals. This linearizes the charge pump characteristics and eliminates the dead zone problem. However, better linearity has been traded with potentially higher in-band noise. For a zero time error, even if the UP and DOWN current pulses cancel out, their uncorrelated noises add in power. The longer the reset delay, the higher the charge noise injected into the loop [Lacaita 07]. To investigate the dead zone problem in the designed charge pump, a parametric analysis was carried out with phase shift as a parameter. Each time the average output current of the charge pump is calculated. The result is plotted in Figure 6.18. The characteristics are clearly linear, which indicates that the inherent delay in the reset path is enough for the charge pump to respond and an extra delay stage is not required.

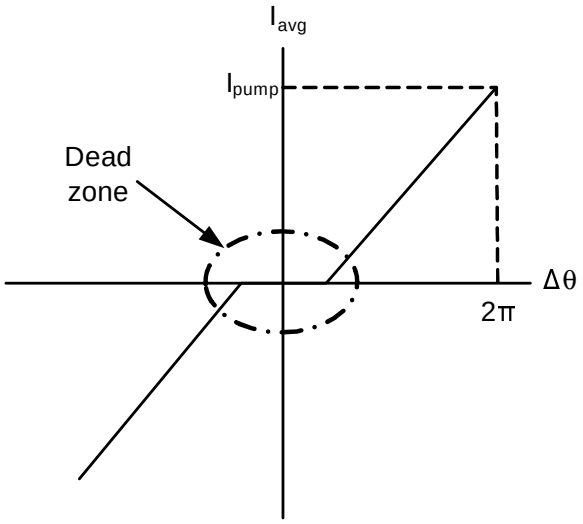


Figure 6.17: An exaggerated illustration of charge pump dead zone problem.

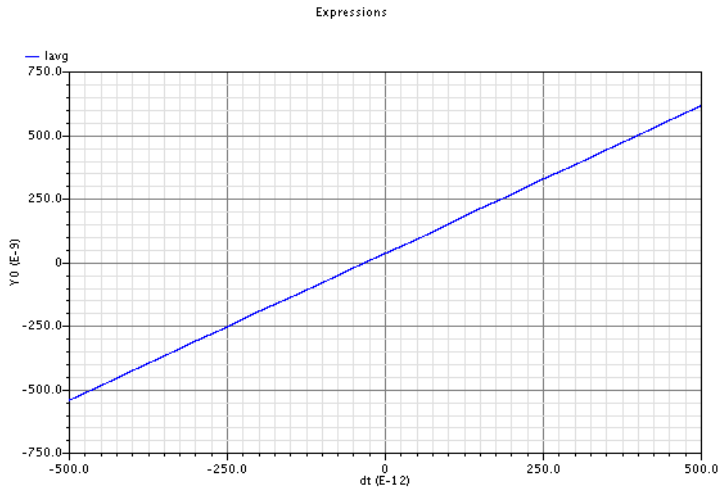


Figure 6.18: Charge pump average output current vs. phase error (in ps) between REF and FB. The period of REF and FB is $\approx$ 8 ns.

6.3.2.4 Loop Filter

As the reference frequency is relatively high, this allows large loop bandwidth. Consequently, the component values of the loop filter elements are relatively small as illustrated before in Table 6.3. Thus, the loop filter can be fully integrated without needing any off-chip components. Capacitors are implemented using MOS cap cell available in the design kit. The resistance is implemented using HRIPOLY cell (high resistance poly).

6.3.2.5 Loop Divider

The loop divider is a simple divide by 8 frequency divider. The simplest implementation is achieved by using a 3 stage asynchronous counter [Mano 02, Lacaita 07]. However, asynchronous counter suffers from large delay and jitter which is accumulated from one stage to the next. For this reason, a retiming (synchronization) operation is necessary. The output phase noise using a synchronizer will only be given by the input clock jitter and by the jitter introduced by the synchronizer [Vaucher 02, Fahim 05, Lacaita 07].

As the VCO output frequency (~ 1GHz) is within CMOS capabilities, standard cells were used to implement the divider. The schematic of the divider is shown in Figure 6.19. It should be noted that this architecture cannot be used for a large divider ratio, because the accumulated delay may cause timing hazards at the synchronizer stage. Thus, care should be taken to assure that the total delay does not approach the input clock period across all PVT corners. Transistor level simulation was carried out using extracted netlists. Transient simulation results are shown in Figure 6.20.

6.3.3 Integration and Verification

As we pointed out at the start of Section 6.3, incremental integration will be used. This step is simplified if the same CAD tools are used for both

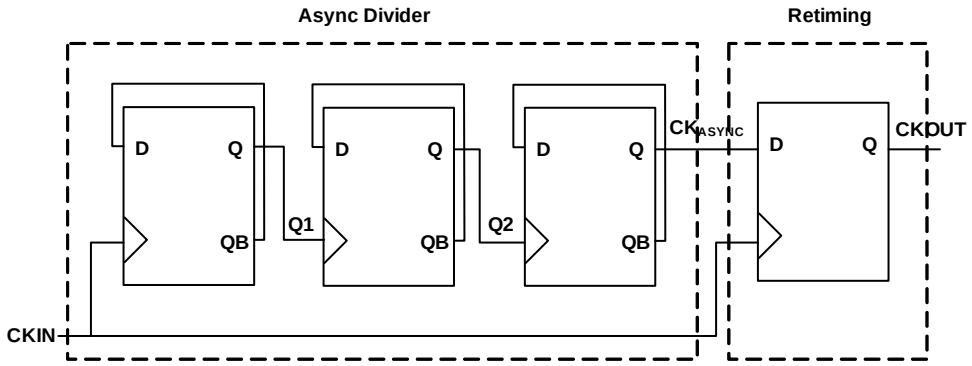


Figure 6.19: Frequency divider schematic.

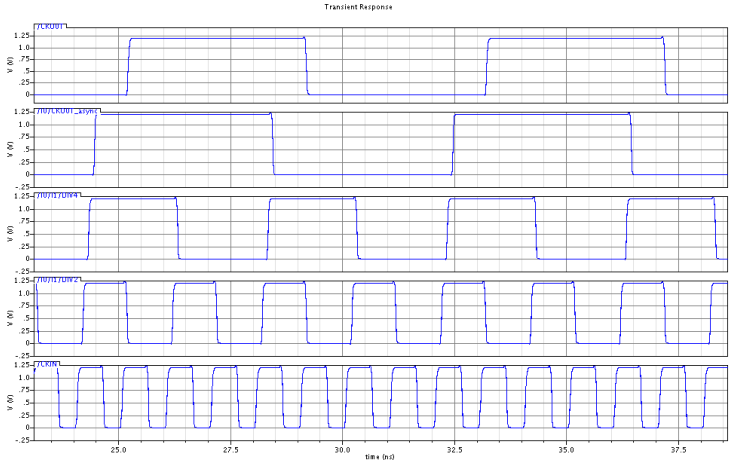


Figure 6.20: Frequency divider transistor level transient simulation. (1) CKOUT (2) CK_{ASYNC} (3) Q2 (4) Q1 (5) CKIN.

system and circuit simulations and "pin-accurate" symbols are used for the behavioral models. Using the Hierarchy Editor and configuration view in Cadence custom IC design tools is a very powerful mean in this stage. Without changing the test-bench, the Hierarchy Editor allows the designer to choose the "view" to use for each block in the system, e.g. Verilog-A, schematic, extracted... etc. This allows seamless incremental integration without distracting the designer with a dozen of test-benches for each stage.

First, the transistor level implementation of the VCO was embedded in the PLL behavioral model to verify its operation "in loop". Transient simulation result showing multiphase clock signals generated from VCO is shown in Figure 6.21. The VCO generates eight differential clock signals. Each differential clock signal carries the information of two distinct phases; one in the rising edge and another in the falling edge. Thus we have 16 different phases separated by $\frac{T_{OV}}{16}$, where T_{OV} is the oversampling clock generated by the PLL. If T_{clk} is 1 ns, then the timing resolution, which represents the maximum peak to peak deterministic jitter, is equal to 62.5 ps. Transient simulation of PLL in the locked state using the VCO transistor level schematic is shown in Figure 6.22.

Next, the transistor level schematic of the charge pump is simulated with the PLL behavioral model. The charge pump is the most difficult block to model, thus, simulating its transistor level schematic is often easier and more accurate than modeling its non-idealities. The simulation results of full transistor level simulations are shown in Figures 6.23 and 6.24. PLL power breakdown is shown in Figure 6.25.

6.4 Design of DTC

6.4.1 Interpolation Counter

The block diagram of the DTC is repeated in Figure 6.26 for convenience. The 3-bit interpolation counter VHDL code was synthesized using Cadence

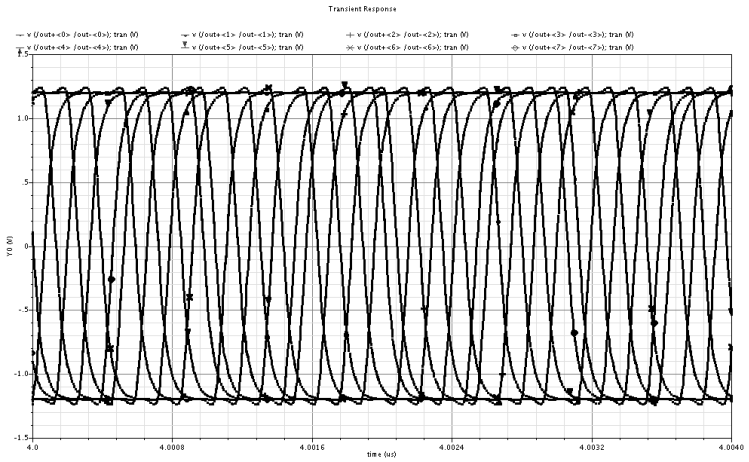


Figure 6.21: Multiphase clock output for VCO transistor level embedded in PLL behavioral model.

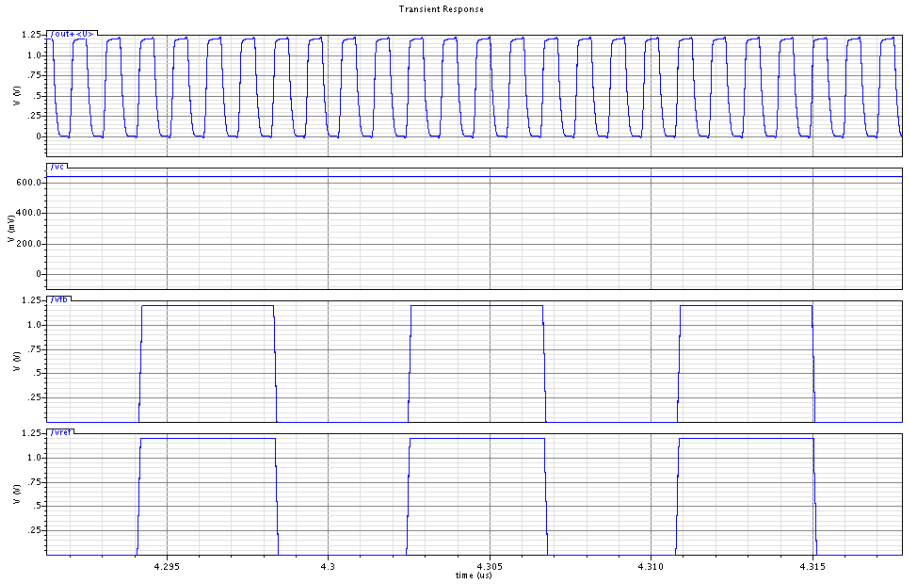


Figure 6.22: PLL in locked state for VCO transistor level embedded in PLL behavioral model. (1) VCO output (2) VCO control voltage (3) Divider output (4) Input reference.

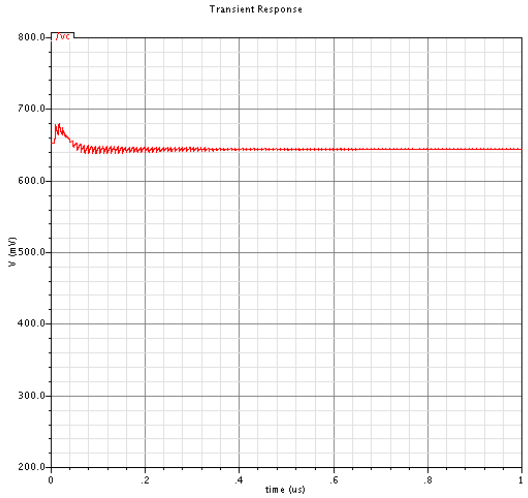


Figure 6.23: VCO control voltage for full transistor level simulation.

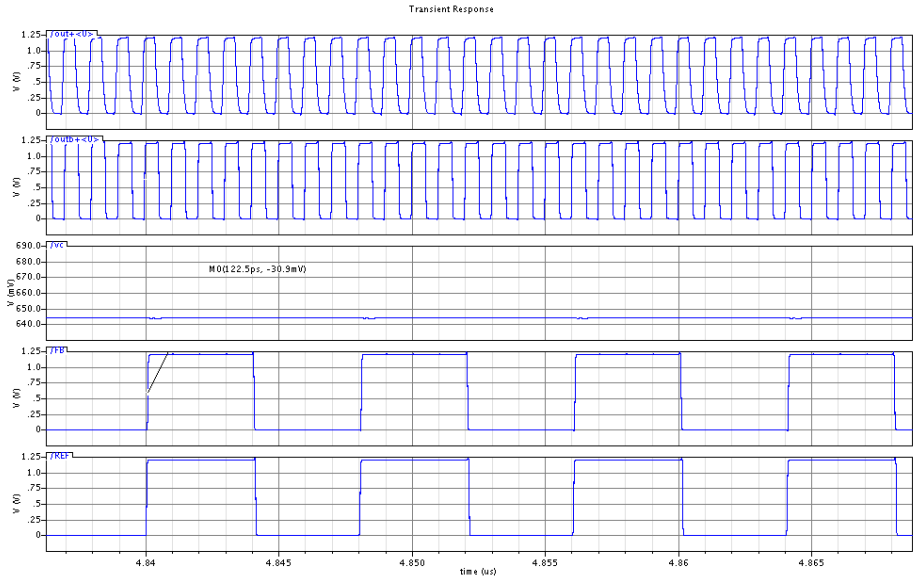


Figure 6.24: PLL in locked state for full transistor level simulation. (1) VCO output (2) VCO output after buffer (3) VCO control voltage (4) Divider output (5) Input reference.