

Demystifying the PLL

Antonio De Lima Fernandes, Cypress - April 01, 2013

Demystifying the PLL

The Phase Locked Loop (PLL) is an indispensable component in modern electronic systems. Its function is to generate an accurate output signal of frequency equal to, or a multiple of, the input signal frequency. It is mainly used in modulators/demodulators and in clock generation/multiplication. However, when designing a digital communications system on a mixed-signal chip, digital designers tend to avoid PLLs because of their inherent analog nature, and analog designers stay away from them because IDEs involve coding. This article presents a different way of designing a simple PLL.

PLL Basics

To begin, let us have a look at the block diagram of a PLL:

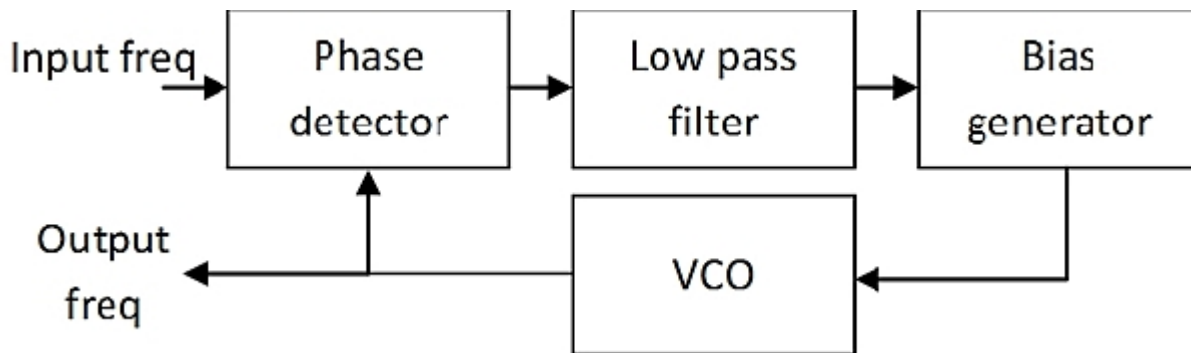


Figure 1: Block diagram of a PLL

Consider an input sine wave of frequency $\omega_i: \sin(\omega_i t)$ to the PLL. A square wave is an infinite summation of a sine and its harmonics, so although the circuit may be digital, results defined in terms of sines and cosines are equally valid. Let the output of the Voltage Controlled Oscillator (VCO) also be a sine wave of frequency ω_0 and phase $\varphi: \sin(\omega_0 t + \varphi)$. The input signal enters the PLL at the Phase detector, where its phase is compared to that of the VCO. In simple terms, this phase detector is a multiplier. Recall:

$$2 \sin A \sin B = \cos(A - B) - \cos(A + B)$$

Equation 1

Or if we had started out with cosines instead of sines, we would have

$$2 \cos A \cos B = \cos(A - B) + \cos(A + B)$$

Equation 2

Since $\cos(A + B)$ is a high frequency term, it is filtered out by the Low Pass Filter (LPF). Assuming that the PLL is closely following the input frequency, $\omega_i - \omega_0 \approx 0$ (low frequency signal), this passes through the LPF to generate an output proportional to $\cos((\omega_i - \omega_0)t + \varphi)$. If $\omega_i = \omega_0$, the output is

proportional to the phase difference ϕ between the VCO output and the input frequency. Thus the phase is detected.

For proper operation, a constant offset that is independent of frequency is added to this voltage by the bias generator, and then the voltage is input to the VCO. As its name suggests, the VCO generates an output voltage whose frequency is proportional to the input voltage:

$$\omega_0 = \omega_{0Q} + K_{VCO}V_c$$

Equation 3

Where ω_{0Q} is the quiescent or free-running frequency of the VCO (when input=0); K_{VCO} is the sensitivity (rad/s/Volt); V_c is the input voltage to the VCO.

Looking at the PLL from a high level, if the input frequency to the PLL is different from the VCO frequency, a voltage is generated by the phase detector. After being filtered and offset, this voltage adjusts the output frequency of the VCO to match the input.

Parameters of a PLL

The key parameters of a PLL include:

1. Type and order: Determined by the transfer function of the system.
2. Lock range: The frequency range the PLL is able to follow input frequency variations once locked. Mainly defined by the VCO range and limited by the phase detector.
3. Capture range: The frequency range the PLL is able to lock-in when starting from an unlocked condition. This range is usually smaller than the lock range and will depend on the LPF cut-off frequency.
4. Loop bandwidth: Defines the speed of the control loop.
5. Transient response: Peak overshoot and settling time.
6. Steady-state errors: Phase or timing error.
7. Output spectrum purity: Strength of the main frequency versus sidebands.
8. Phase-noise: Defined by noise energy in a certain frequency band. Dependent on VCO phase-noise, PLL bandwidth.
9. General parameters: Such as power consumption, supply voltage range, output amplitude.

S-Domain representation

S-Domain representation

In order to understand the PLL a little better, it is useful to get into the dirty details. Consider the s-domain representation of the PLL in the block diagram as below. For simplicity, the bias generator has been omitted.

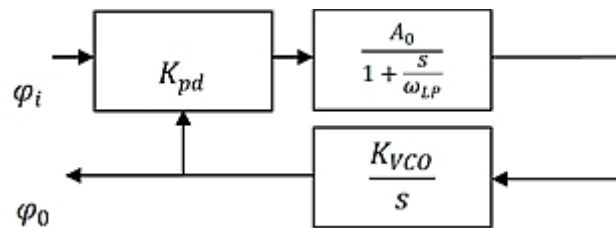


Figure 2. S-domain representation of PLL

Assuming that the Phase Detector is linear (true for frequencies close to quiescent), the signal flow is as follows:

The phase detector generates an output proportional to the phase difference of input and VCO signal. Let this proportionality constant be K_{pd} . This phase-dependent output is then sent to a first-order LPF with gain A_0 and cut off frequency ω_{LP} to remove the unwanted high frequency components. If this LPF is implemented as an RC circuit, $\omega_{LP} = (RC)^{-1}$. Since we need to convert this phase dependent voltage to a frequency, we introduce the sensitivity parameter K_{VCO} . Furthermore, the output of VCO should respond to phase changes - hence we add the first order integrator term - $1/s$ (Refer to the Appendix for details). The entire system is restricted to be second order in order to not create stability problems.

Response of the PLL to various inputs

Let us consider how a PLL responds to various kinds of signals (see Figure 2).

Case 1: $\omega_i = \omega_0 = \omega_{0Q}$

Input frequency is equal to the free-running frequency of the VCO, i.e. the VCO frequency when there is no input signal. This implies that the VCO should see no voltage at its input due to the input frequency. But as we have derived earlier, the output at the LPF when $\omega_i = \omega_0$, is $\cos((\omega_i - \omega_0)t + \varphi)$. Hence clearly, $\varphi = 90^\circ$ to ensure that $\cos \varphi = 0$.

Thus we learn an important aspect of the PLL - the output of the VCO, with input at the quiescent frequency is a signal at the quiescent frequency with a 90° phase shift with respect to the input. Thus the term 'Phase-Locked' is somewhat misleading - the frequency matches, but the phase does not.

Case 2: $\omega_i \neq \omega_0$ but $\omega_i \approx \omega_0$

Now from the quiescent frequency, assume we gently increase the input frequency by a small amount. Since $\omega_i - \omega_0 \approx 0$, the signal passes through the LPF, and appears at the VCO input. This causes the output of the VCO to change and follow the input frequency. This keeps happening continuously until $\omega_i = \omega_0$. I highlight the word continuously, because adjustments happen as fast as the loop will allow.

A similar result is obtained if one decreases the input frequency. If the change is slow, and the VCO was 'following' the previous change, it will keep tracking input variations. Now two interesting questions spring forth.

1. Until what point will the VCO track input changes? - **Lock Range**
2. When will it lock if I start with $\omega_i \neq \omega_0$ and $\omega_i - \omega_0 \gg 0$? - **Capture Range**

Derivation of lock range

The first question is dependent upon the phase detector. Say you use an XOR gate as the phase detector (see Figure 3) and assume that logic level 1 is +1V, logic 0 is -1V. Let us begin at the quiescent frequency. The VCO and input signals are 90° phase shifted. In the figure, when A and B are 90° phase shifted, the output is a 50% duty cycle square wave with average value of 0.

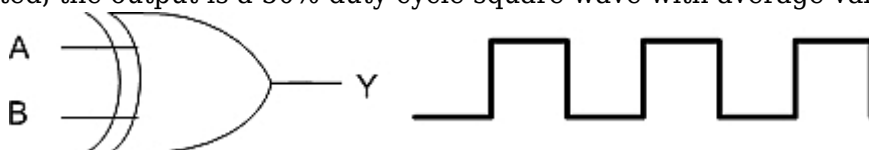


Figure 3. XOR gate as phase detector

However, as we increase the phase difference beyond 90° towards 180° , the average output value

starts to rise linearly (XOR=1 if $A \neq B$ i.e. 180° phase shift). Similarly, if we decrease the phase difference between A and B, towards 0° the output Y starts decreasing (XOR=0 if $A=B$). However, beyond 0, the average Y starts increasing (work it out with the XOR) though, ideally, since the phase is still decreasing, we want the average output to continue decreasing. Similarly, beyond 180° , average Y values start falling though they should be rising.

Thus, the average value of the XOR gate is a good indicator of phase difference only in the open interval (0,180). Only inside of that frequency range will the PLL be able to work, once the VCO is locked on to the input frequency. This is what usually determines the lock range (if the VCO is capable of hitting all frequencies implied by (0,180)).

We will now calculate the actual lock range value. Going through the loop, the maximum phase difference (from quiescent) at the input is $\pi/2$, therefore, the voltage at the output of the phase detector is $K_{PD}(\pi/2)$. After the LPF, it becomes $V_C = K_{PD}(\pi/2)A_0$, and this is converted to a frequency whose value is $K_{PD}(\pi/2)A_0K_{VCO}$. This is the maximum deviation possible from the quiescent frequency. Since this can be both positive and negative, the lock range is twice this value.

Understanding capture range

Understanding capture range

Case 3: $\omega_i \neq \omega_0$ and $\omega_i \gg \omega_0$

In this case, we arbitrarily change the input frequency starting far from ω_{0Q} . Our aim is to find when the PLL will lock onto the input frequency. Since $\omega_i - \omega_0 \gg 0$, this error voltage does not show up at the VCO input, and thus the VCO does not respond to any frequencies $\omega_i - \omega_0$ that are beyond its cut-off ω_{LP} . Hence, the capture range is a very small window that depends on the LPF.

For the derivation, see the sidebar: Further Derivations. Note also, for a visual understanding of lock/capture range and the actual waveforms out of the XOR gate, please refer to [Reference 4](#).

High level design view

Using this theory, we can now get to the process of designing the PLL. Below is an implementation of a PLL, with all the blocks that we saw in Figure 1. The individual building blocks are explained one-by-one below.

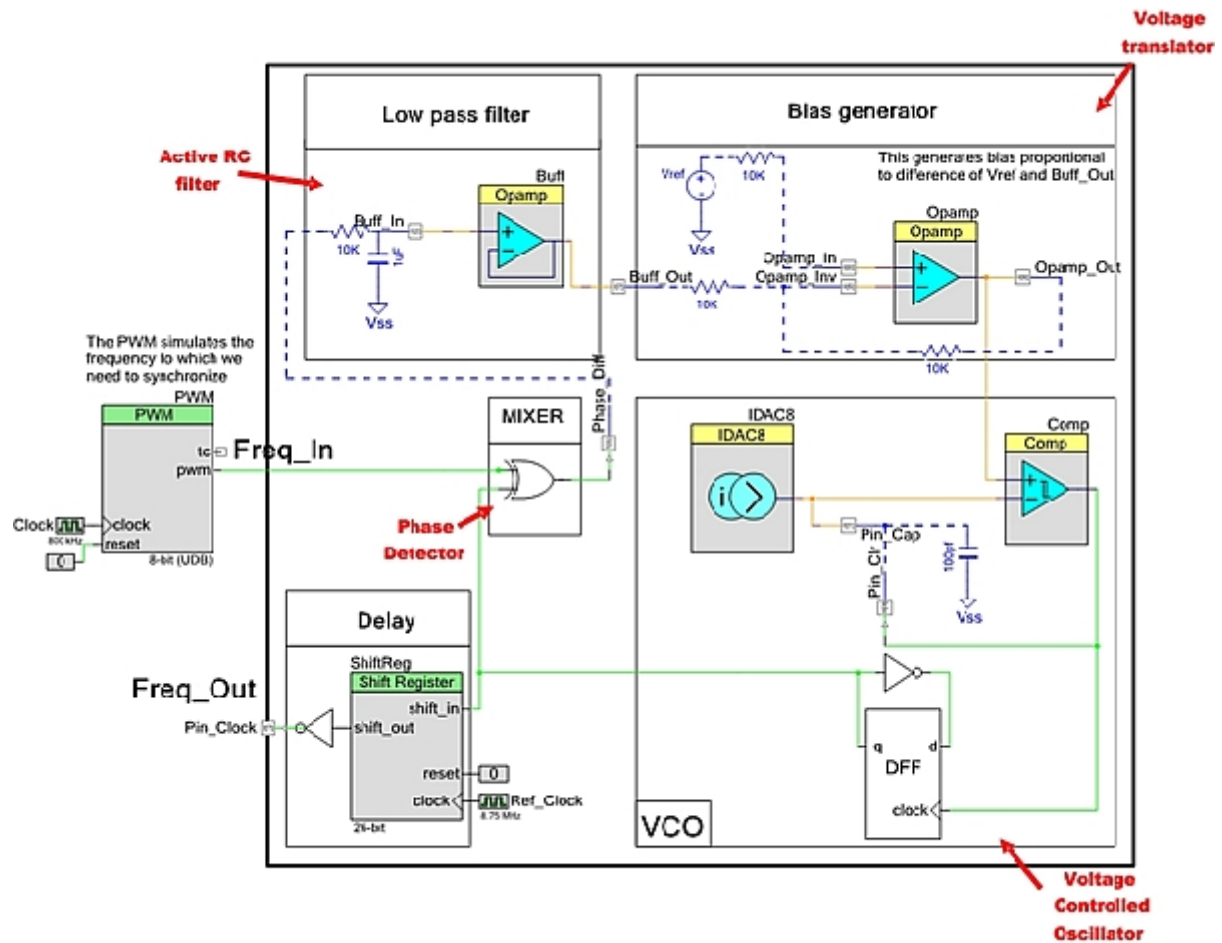


Figure 4. PSoC Creator PLL Schematic

Implementing a PLL Design

As shown in Figure 4, the PLL contains all the basic blocks as in Figure 1, albeit in their simplest schematic form. The heart of any Phase Locked Loop is its Voltage Controlled Oscillator (VCO). So we begin the design with the VCO:

VCO

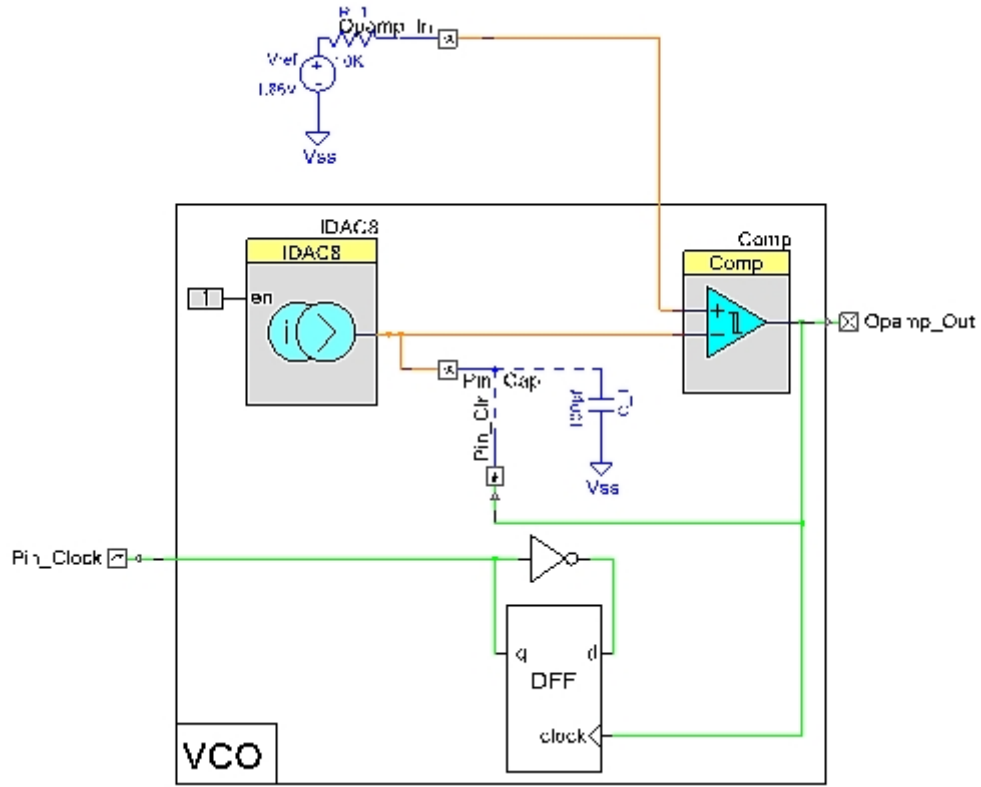


Figure 5. Voltage Controlled Oscillator Schematic

The working of the VCO is straightforward. The IDAC (Digital to Analog Converter with Current as output) charges the capacitor, whose voltage increases linearly. When this voltage becomes larger than the positive input (to the comparator), the output of the hysteretic comparator falls. This shorts the capacitor to ground through Pin_Clr. The moment the capacitor is discharged, the output of the comparator goes high again, thus sending a rising edge to the DFF (D Flip Flop), and also re-enabling the IDAC. The DFF thus generates an output clock at half the rate of the comparator edges, with 50% duty cycle.

The quiescent frequency of the VCO is decided by a combination of the IDAC, the V_{ref} and the Capacitor; following the equation:

$$I = C(dv/dt)$$

Or

$$f = I/(Cdv)$$

Where dv is the voltage that the capacitor charges to before discharging, i.e $V_{ref} - 0$; I is IDAC current; and f is the frequency of edges at comparator output. For example, choosing $I=50\mu A$, $V_{ref}=1V$, $C=100pF$, we can calculate the frequency of the edges at the comparator to be 500kHz. However, actual values of frequency seen are lower because of the finite discharge time of the capacitor, and additional board/pin capacitances. Below are the waveforms obtained for the VCO.

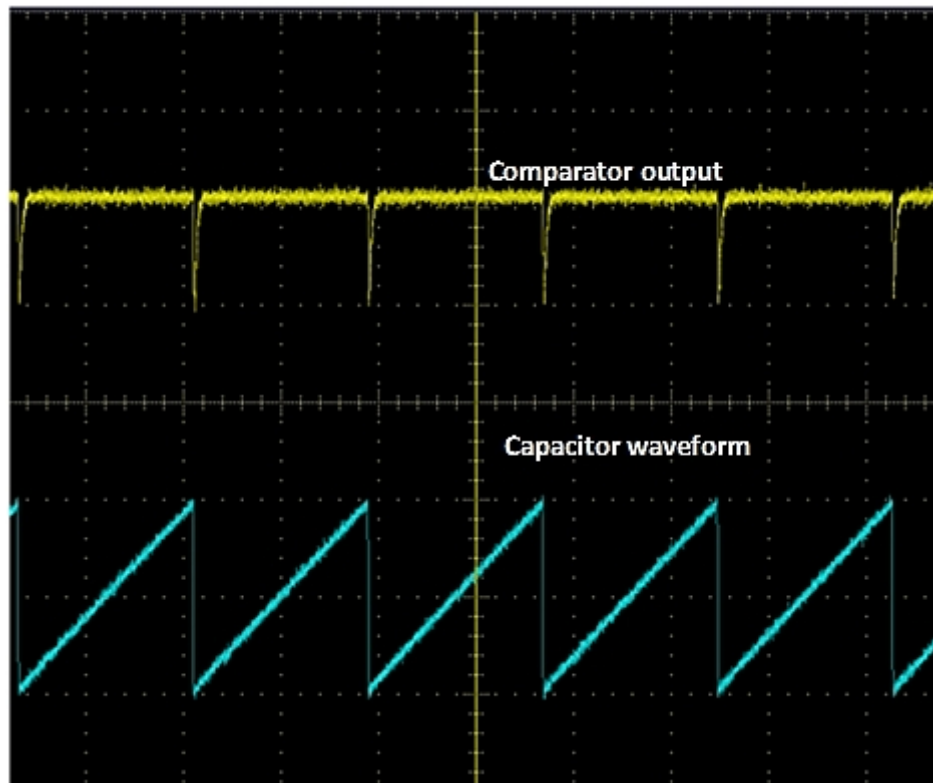


Figure 6. VCO Capacitor and Comparator waveforms

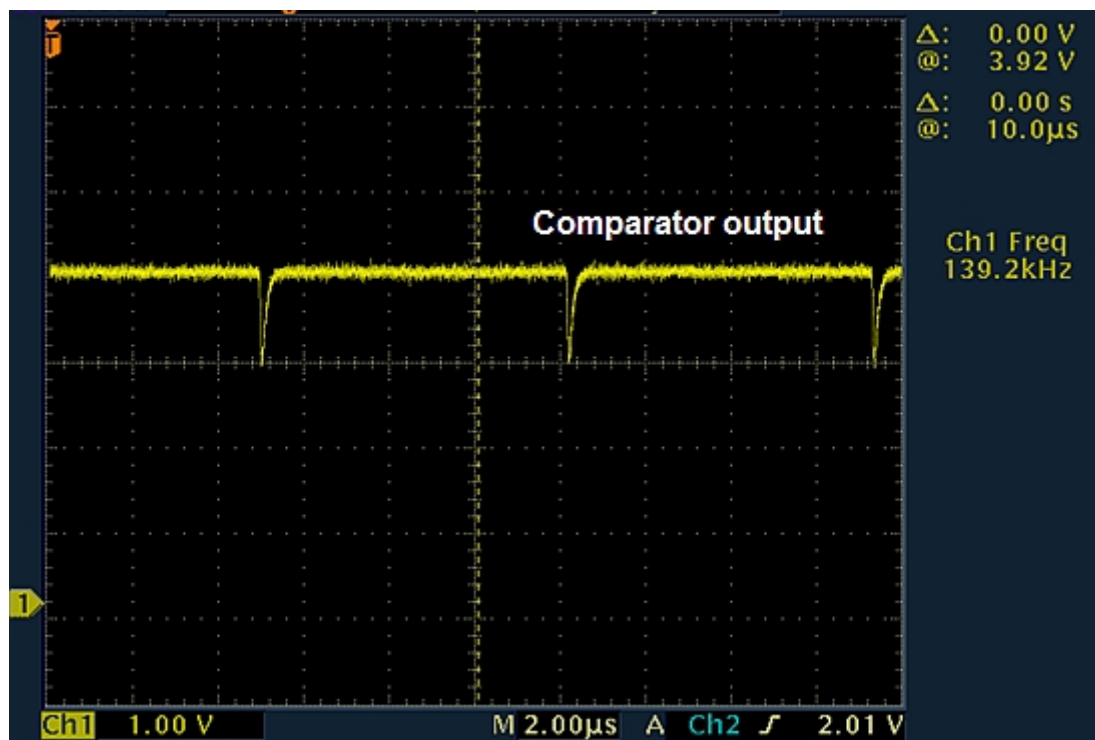


Figure 7. Comparator output

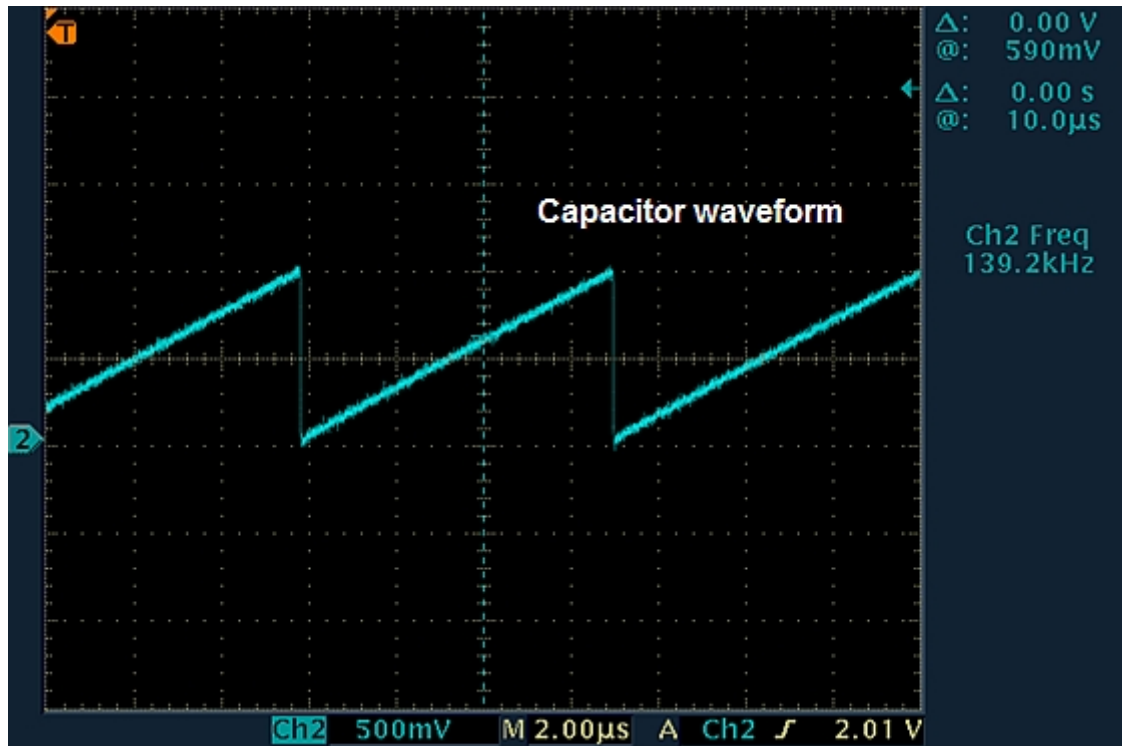


Figure 8. Capacitor charging and discharging

LPF
LPF

The next component we will look at is the LPF. In order to restrict the PLL to be a second-order system, the Low Pass Filter is implemented as an active RC filter with $f_{LP} = 100\text{Hz}$ ($R=10\text{k}$, $C=1\mu\text{F}$).

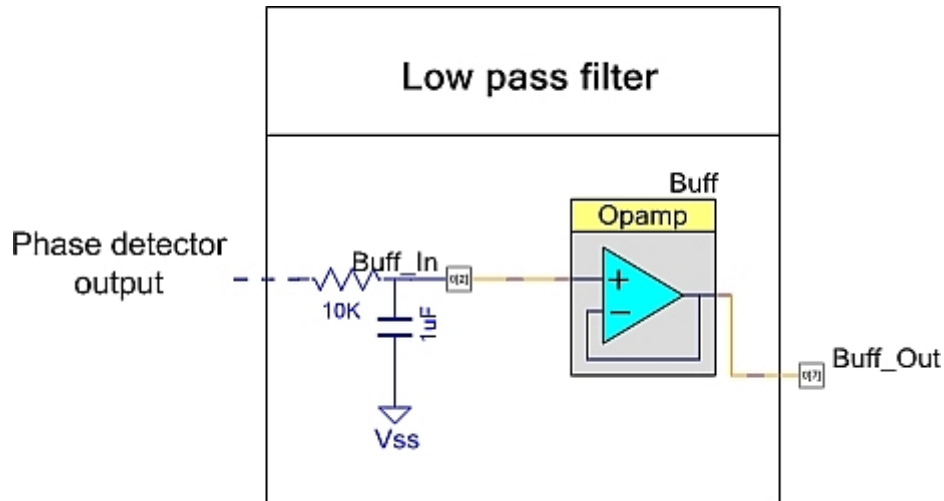


Figure 9. Low Pass Filter

Phase detector and bias generator

As seen earlier, an XOR gate is used as the phase detector. The bias generator sets the quiescent frequency of the VCO, and hence the captured signal. It is implemented as a difference amplifier.

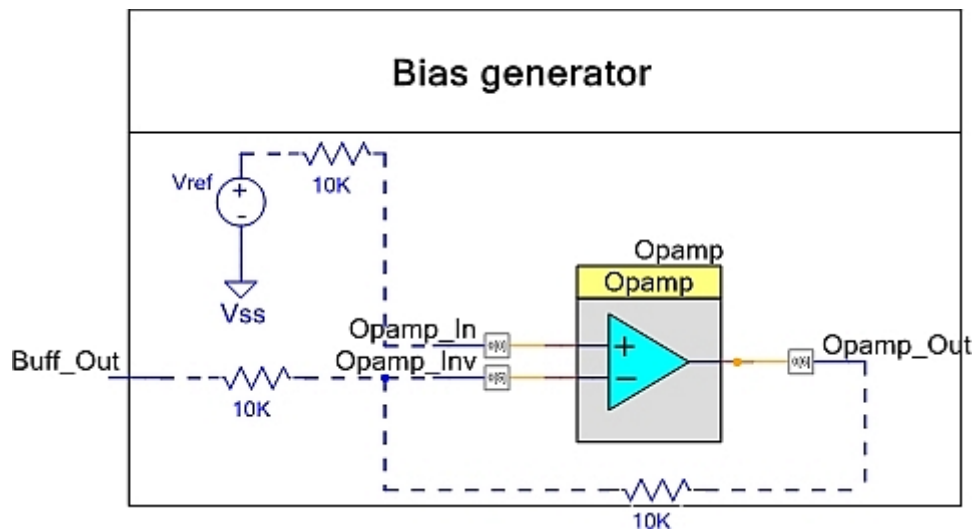


Figure 10. Bias Generator

Here, $Opamp_Out = 2V_{ref} - Buff_Out$ and V_{ref} can be chosen according to the input frequency range.

Input simulator and Shift Register

Two additional blocks can be seen in Figure 11 – the PWM which acts as the external frequency to which the PLL must lock, and a shift register used to generate a delay in the output of the PLL so that the phases of input and output match.

The final schematic, along with output delay block and external frequency simulator (PWM) is shown below:

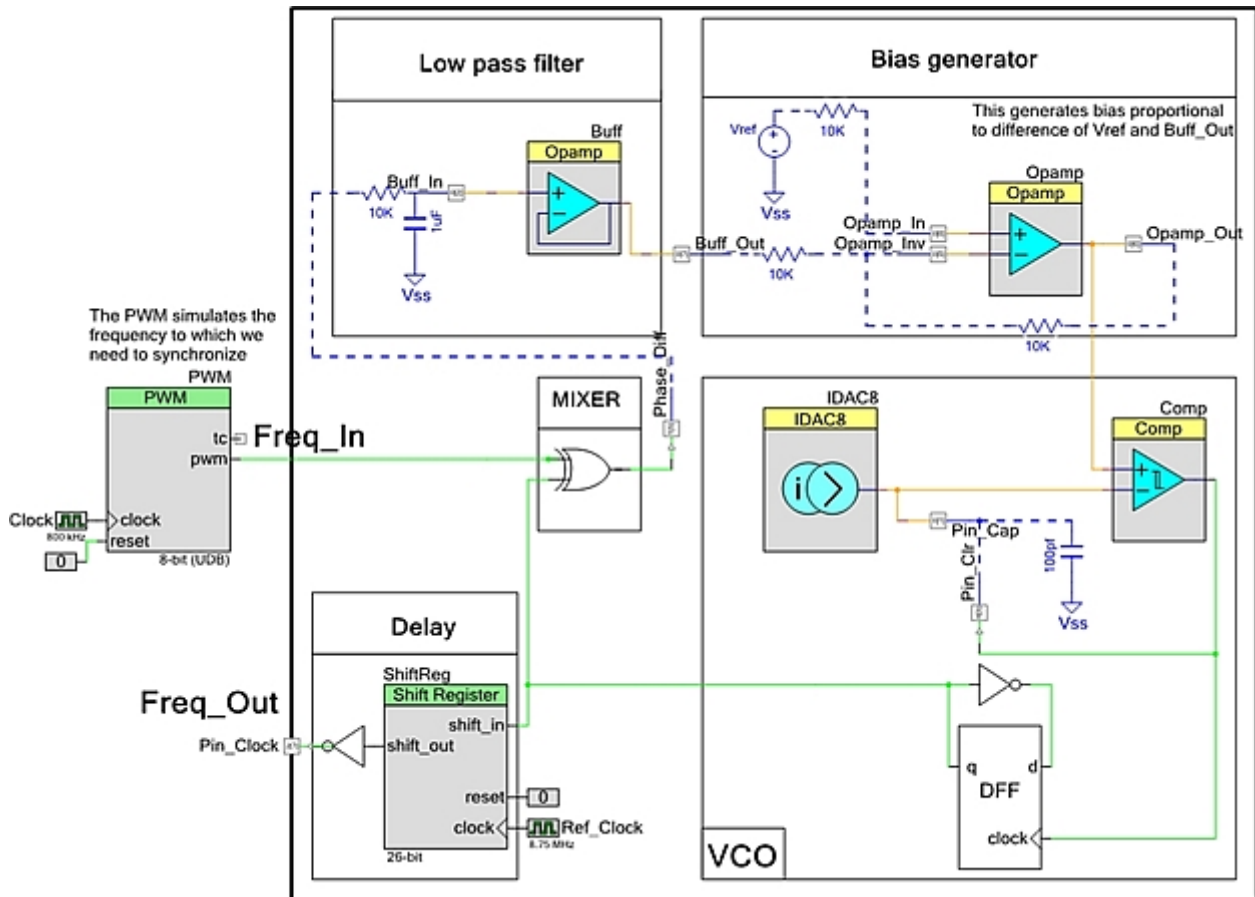


Figure 11. PLL along with delay and external stimulator (PWM)

All the blocks needed for this design were dragged onto the schematic from a reference library and wired up. The nature of this process allows the designer to make a very rough design, test it, and then just as easily modify the original design. Additionally, in such a design, there is effectively no code. Thus the design cycle is greatly simplified.

```

#include <device.h>

void main()
{
    /* Start all used components */
    IDAC8_Start();
    Comp_Start();
    Opamp_Start();
    Buff_Start();
    PWM_Start();
    ShiftReg_Start();

    for(;;)
    {

    }
}

```

Figure 12. C-code required for the PLL

The following two scope-shots show the PLL output for 2 frequencies - 125kHz and 25kHz, both at quiescent frequency (90° phase shift clearly visible).

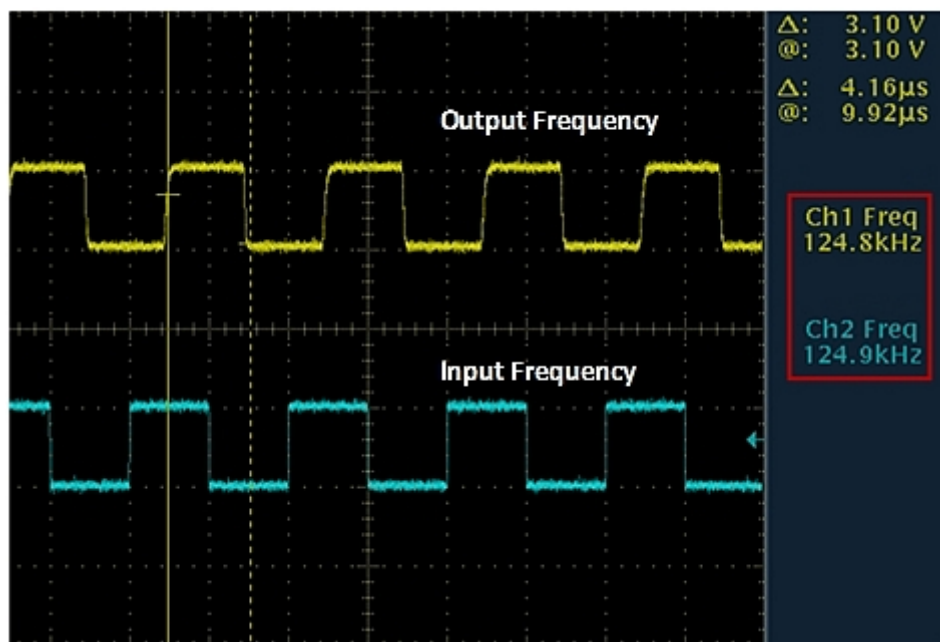


Figure 13. Oscilloscope screen capture showing PLL input and output frequencies at 125kHz

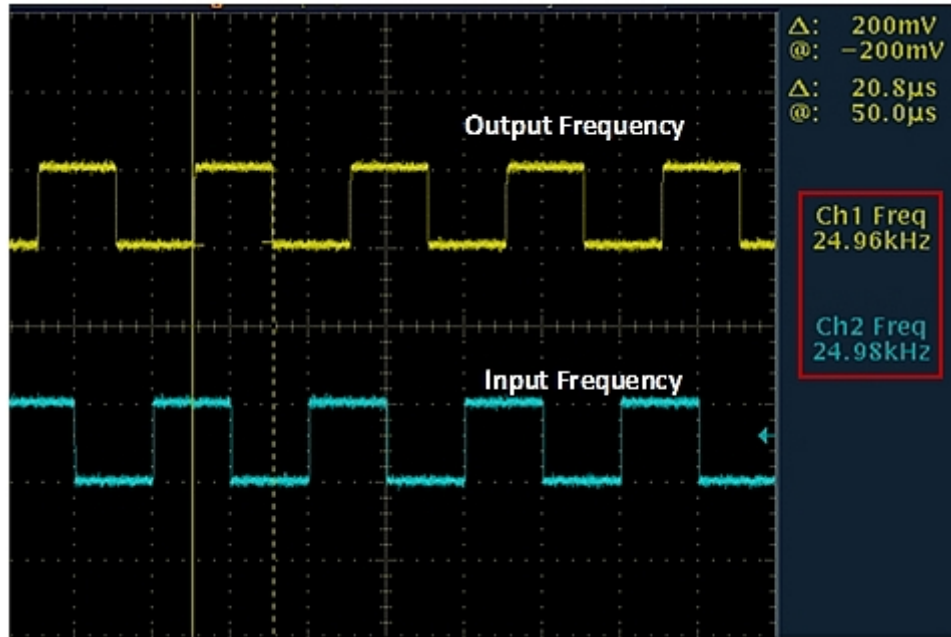


Figure 14. Oscilloscope screen capture showing PLL input and output frequencies at 25kHz

Thus we have seen how to de-mystify the sometimes intimidating process of designing modular sub-systems as a part of a bigger, mixed-signal application. Note that the PLL designed in this article offers many opportunities for optimization.

Sidebar: Further Derivations

Sidebar: Further Derivations

VCO Equations and Transfer functions:

To understand the relation between the s-domain transfer function

$$K_{VCO}/s$$

And the K_{VCO} term seen in Equation 3, let's look back at the analog sine-cosine paradigm. The basic behavior of the VCO is to generate a frequency at its output, proportional to the input voltage. This sentence is Equation 3 put into words. Thus, Equation 3, the output of the VCO can be written as:

$$\begin{aligned} & \sin(\omega_0 t + \varphi) \\ &= \sin((\omega_{0Q} + K_{VCO} V_c)t + \varphi) \end{aligned}$$

Equation 4

But as calculated earlier, $\varphi = 90^\circ$, so we rewrite Equation 4 as

$$\cos((\omega_{0Q} + K_{VCO} V_c)t)$$

Equation 5

ω_{0Q} is a constant, but $K_{VCO} V_c$ is better written as $K_{VCO} V_c(t)$.

Since we are interested in the phase change of the output of the VCO wrt input voltage, let us rewrite Equation 5 as:

$$\cos(\omega_{0Q}t + \varphi_o(t))$$

Equation 6

This is the φ_o for which we need to find the transfer function. We know that

$$\omega = d\varphi/dt$$

Hence

$$\frac{d\varphi_o(t)}{dt} = K_{VCO}V_c(t)$$

Or in s-domain

$$s\varphi_o(s) = K_{VCO}V_c(s)$$

$$\therefore \frac{V_c(s)}{\varphi_o(s)} = \frac{K_{VCO}}{s}$$

Equation 7

Which is what we set out to prove.

Derivation of capture range:

Assuming that $\omega_i \neq \omega_0$ and $\omega_i - \omega_0 > \omega_{LP}$, the output of the phase detector is

$$K_{PD} \cos((\omega_i - \omega_0)t + \varphi)$$

Using $\varphi = 90^\circ$,

$$K_{PD} \sin(\omega_i - \omega_0)t$$

This signal, after passing through the LPF, will look like:

$$V_c = \frac{A_0 K_{PD} \sin(\omega_i - \omega_0)t}{\sqrt{1 + (\frac{\omega_i - \omega_0}{\omega_{LP}})^2}}$$

Now this voltage is converted to a frequency by the VCO according to Equation 3

$$\omega_o(t) = \omega_{0Q} + K_{VCO} \frac{A_0 K_{PD} \sin(\omega_i - \omega_0)t}{\sqrt{1 + (\frac{\omega_i - \omega_0}{\omega_{LP}})^2}}$$

The maximum value of sine is ± 1 , so the peak variation of ω_o is:

$$\omega_{0Q} \pm \frac{A_0 K_{PD} K_{VCO}}{\sqrt{1 + (\frac{\omega_i - \omega_0}{\omega_{LP}})^2}}$$

So thus, we can say that the capture frequency with ω_{0Q} as reference is:

$$\omega_c = \pm \frac{A_0 K_{PD} K_{VCO}}{\sqrt{1 + \left(\frac{\omega_i - \omega_0}{\omega_{LP}}\right)^2}}$$

Equation 8

So Equation 8 is a quadratic in $\omega_i - \omega_0 = \omega_c$. Assuming that $\left(\frac{\omega_c}{\omega_{LP}}\right)^2 \gg 1$, we simplify Equation 8 to:

$$\omega_c = \pm \sqrt{A_0 K_{PD} K_{VCO} \omega_{LP}}$$

Equation 9

Thus, we have capture range as $2\omega_c$.

One should verify if the last assumption was valid after obtaining ω_c .

About the author



Antonio Rohit De Lima Fernandes is an Electrical and Electronics Engineer from BITS, Pilani. He has worked as a co-op at Cypress Semiconductor, D-Link, Nextgen PMS and the Univ. of British Columbia.

References

For anyone interested, the following links (especially the videos) are very useful:

1. [Unlocking the Phase locked loop](#), Charan Langton
2. PLL video lecture series, Prof. Radhakrishna Rao (IIT-Chennai), [link1](#), [link2](#), [link3](#)
3. [Phase Locked Loop](#), Wikipedia
4. Univ. Colorado ECE Lab: [PLL](#), Dragan Maksimovic