MOKOLO RAYMOND EBUBECHUKWU
16/ENG04/067
ELECT-ELECT

**ANALOG MULTIPLIERS**

A more sophisticated type of double-balanced mixing employs analog multipliers. Certain electronic circuit configurations provide means of multiplying two analog signals. Gilbert cell is a commonly used analog multiplier configuration. The input/output relations in a Gilbert cell circuit are modeled in Figure *(a).

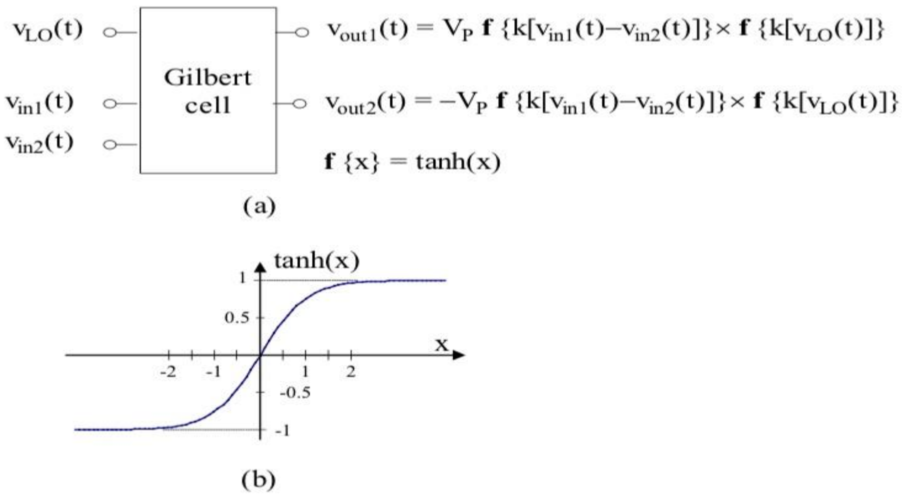



Figure * (a) Block diagram of Gilbert cell and (b) tanh(x)

The output of a Gilbert cell is a product of a function of the difference of two input signals and the same function of the local oscillator signal. The function is hyperbolic tangent, or tanh (·), given as

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}.$$

The variation of tanh(x) is depicted in Figure 7.6 (b). Gilbert cell is an electronic circuit and obviously it scales the input voltages in order to convert them into the argument of the tanh (·) function. The scaling factor is k and has a unit of (volt)-1.

Note that when x is small, tanh(x) is linear. Indeed, for $|x| < 0.3$, $\tanh(x) \approx x$. Therefore, for low amplitude levels of vin1− vin2 and vLO, vout1(t) ≈ Vpk2[vin1(t) − vin2(t)] [vLO(t)]. A direct analog multiplication can be performed.

MOKOLO RAYMOND EBUBECHUKWU
16/ENG04/067
ELECT-ELECT

The block diagram of a typical application of using an analog multiplier as a double balanced mixer is given in Figure ** (a).

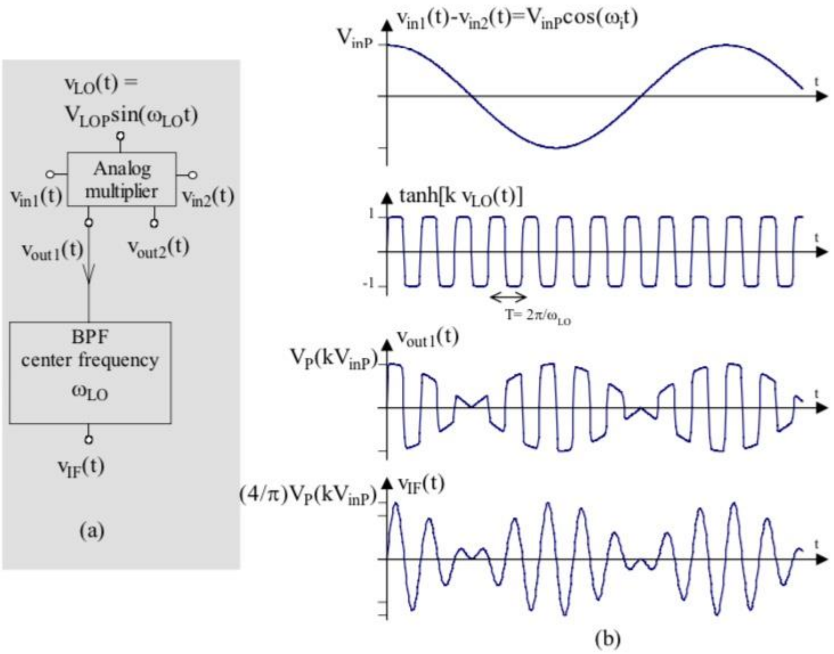



Figure ** Analog multiplier and IF filter outputs with large vLO amplitude (a) block diagram and (b) waveforms.

Base band signals (such as audio, for example) vin1(t) and v in2(t) and a sine wave from the local oscillator vLO(t) are applied as inputs to the analog multiplier. The output of the multiplier is filtered using a BPF centered at the local oscillator frequency. IF signal vIF is obtained at the filter output.

The amplitude of vin1(t) and v in2(t) is normally kept low in order to avoid distortion. A low amplitude sine wave is assumed in the waveforms given in Figure ** (b).

When analog multipliers are used as double-balanced mixers for frequency conversion, a high amplitude sinusoidal vLO(t) signal is employed. Then tanh[kvLO(t)] is almost like a square wave with ±1 levels, as depicted in Figure ** (b).

With such choices for vin1(t) − vin2(t) and vLO(t), the output becomes

$$v_{out1}(t) \approx V_P k(v_{in1} - v_{in2})\{\sum_{n=1}^{\infty} b_n \sin(n\omega_s t)\} = V_P \sum_{n=1}^{\infty} b_n \sin(n\omega_s t)\{k(v_{in1} - v_{in2})\}.$$

MOKOLO RAYMOND EBUBECHUKWU
16/ENG04/067
ELECT-ELECT
The signal vout1(t) is given in Figure ** (b). When this signal passes through the BPF, all harmonics of tanh[kvLO(t)] are eliminated and we obtain the IF signal

vIF(t) ≈ b1VP(kVinP)cos(ωit) sin(ωst).

The fundamental component amplitude in this square wave is b1=4/π (and is larger than 1). Therefore, the p-p amplitude of vIF(t) is b1 times larger than the p-p amplitude of vout1(t).

Gilbert cell analog multipliers are difficult to implement at frequencies above few hundred MHz. SA602A is a very good integrated circuit implementation of Gilbert cell, which operates very well up to UHF. Such implementations are active circuits and they provide gain as well as analog multiplication.

**Conversion gain**

An important parameter in the evaluation of mixers is conversion gain. Conversion gain is defined as

Gc = Po/Pi

where Po is the total power delivered to a matched load at output and Pi is, again, the total available power at the input. This expression is similar to the gain of an amplifier, except here the input and output frequencies are different.

**PHASE LOCKED LOOPS (PLL) FREQUENCY SYNTHESIS**

Configuring a phase locked loop (PLL) for a given frequency synthesis application can simultaneously be both a quick-and easy-process as well as a time-consuming, tedious, and iterative process. This dual nature in PLL system design arises from the number of loop parameters that need to be appropriately dialled in for a given application. As will be discussed in this article, there are two categories of loop parameters that must be considered: frequency synthesis parameters and performance parameters. The former sets up the loop to generate the correct frequency while the later dictates the quality of output frequency (with “quality” being a term relative to the given application).

The interplay between these two categories of parameters is where designers spend the bulk of their time. After determining a set of frequency synthesis parameters that meet the system needs, we then attempt to dial in the performance parameters. However, when we reach the end of optimizing the loop, there is always the doubt: did I choose the best possible frequency synthesis parameters?  Perhaps there is a different set that will run cleaner and consume less power or have more margin? It is these design choices upon which this paper will attempt to shed some common-sense design principles.

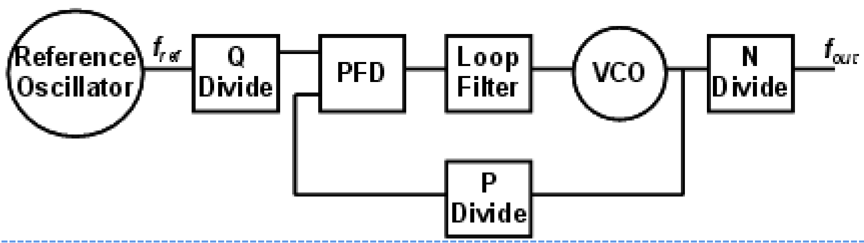
At the most fundamental level, the goal of any frequency synthesizer is, based on a given reference frequency, to generate a desired output frequency. That is, solve:

$$f_{out} = \kappa \cdot f_{ref} \tag{1}$$

where κ is the frequency scaling constant, sometimes referred to as the normalized frequency. Any frequency synthesizer circuit is simply a mechanism for approximating κ. A PLL frequency synthesizer approximates κ by inserting divide blocks between the reference oscillator and the output clock.

Then, using a feedback loop with a phase detector to maintain phase coherence between the two dividers, the desired frequency is generated. The block diagram for this is shown in Figure 1. This is the general form of a charge pump integer divide phase locked loop (a very common topology used for frequency synthesis).

Figure 1. Block diagram of a basic integer divide PLL.




Three divide blocks are used to approximate the value of κ: the reference divider (Q), the feedback divider (P), and the output divider (N). It can be readily shown that κ is defined for this type of frequency synthesizer as:

$$\kappa = \frac{P}{Q \cdot N} \tag{2}$$

Combining Equations 1 and 2, the relationship between input and output frequency is:

MOKOLO RAYMOND EBUBECHUKWU
16/ENG04/067
ELECT-ELECT

$$f_{out} = \frac{P}{Q \cdot N} \cdot f_{ref} \tag{3}$$

It is these P, Q, and N divide values that we refer to as the “frequency synthesis parameters”. These values set up the gross functionality of the loop and must be chosen to set the desired output frequency. One common way of determining these values is to divide the output frequency by the reference frequency, and reduce the fraction:

$$\frac{f_{out}}{f_{ref}} = \frac{P}{Q \cdot N} \tag{4}$$

The difficulty in solving Equation 4, for any arbitrary reference and output frequency, is that there are three degrees of freedom (limited only by the range of divide values P, Q, and N can take on). The most common technique for solving Equation 4 is a search algorithm. Such algorithms work by searching the solution space, looking for sets of P, Q, and N values that will result in the desired κ value. They are, in essence, triple nested loops that search all possible P, Q, and N values.

Common simplifications are to set N equal to one, Q equal to one, or both. These simplifications are based on system design needs. Analysis of all three of these simplifications is a subset of the general case shown in Figure 1 and represented by Equation 4.

If both Q and N are set equal to one, then the maximum resolution of the output frequency is limited to the reference frequency, making it possible to synthesize only integer multiples of the reference. In this case determining the value of P is reduced to a simple matter of arithmetic.

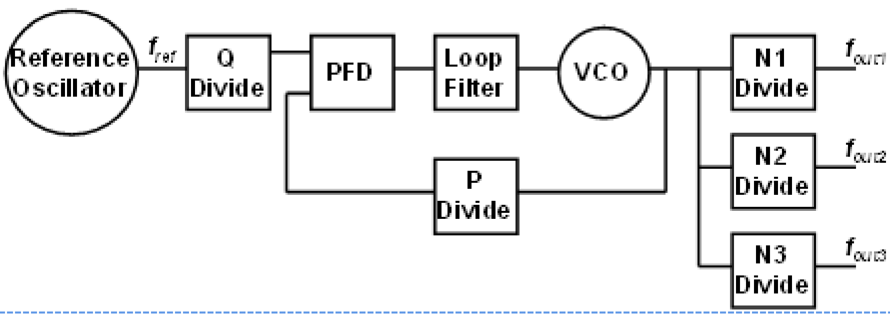
If just Q or N is set equal to one, then only a single configuration exists (with respect to a minimum Q/P or N/P ratio) for synthesizing the desired output. Determining this ratio is then a matter of fraction reduction.

The use of all three divide blocks introduces an added layer of generality to the hardware that enables the direct reuse of the PLL through programming for many different frequency synthesis applications. However, this generality also results in a significantly more complex problem in determining the values of P, Q, and N to use. Specifically, it results in multiple frequency synthesis parameter sets that are valid for a given reference and output frequency, all of which can have drastically different performance characteristics (band width, phase margin, jitter, phase noise, power consumption, etc).

An additional configuration that is commonly used in programmable systems on a chips (SoCs) is to have multiple output dividers. This allows for the synthesis of multiple outputs at different frequencies.

Figure 2 illustrates this configuration. It is important to note that each output is an integer multiple of the VCO frequency. This topology emphasizes the importance of selecting the right VCO frequency so as to maximize the number of system clocks that can be generated off of the single PLL.

Figure 2. Block diagram of a basic integer divide PLL with multiple output dividers.




A Quick Walk Around the Loop

With the basic input-output relationship in hand, we next need to take a walk around the PLL and consider the steady-state operating condition. When the loop is in lock, the output of the Q divider and the P divider have matched phase and frequency and the output of the PFD is tri-stated, leaving the loop filter voltage unchanged and the VCO frequency steady. As the loop filter voltage changes (due to noise, charge pump leakage, and capacitor charge leakage), the VCO frequency will drift.

The PFD observes this drift by comparing the VCO output to the reference oscillator frequency (through the P and Q divides respectively) and causes the charge pump to either pump up or pump down the loop filter voltage. Once the correct loop filter voltage is reached again, the PFD output is tri-stated and the loop filter voltage is left unmodified.

During this normal mode of operation, the PFD frequency (the frequency at which the Phase-Frequency Detector runs) plays a central role. The PFD frequency is set by the reference oscillator frequency divided by the reference divider:

$$f_{pfd} = \frac{f_{ref}}{Q} \tag{5}$$

In the steady-state condition, the PLL loop dynamics force the VCO output frequency, divided by the P divider, to also be equal to the PFD frequency:

$$f_{pfd} = \frac{f_{VCO}}{P} \tag{6}$$

By re-arranging Equations 5 and 6, we can now make a key observation: the reference frequency and the VCO frequency must both have the PFD frequency as a common divisor:

$$Q = \frac{f_{ref}}{f_{pfd}}, \quad P = \frac{f_{VCO}}{f_{pfd}} \tag{7}$$

Any valid choice of P and Q will result in a PFD frequency that is a common divisor of both the reference frequency and VCO frequency. This allows us to take a fundamentally different approach to determining the P and Q divide values than the fraction reduction method illustrated in equation (4). It is also worth noting that the PFD frequency will also play a central role in the PLL performance parameters, which we will discuss in more detail later. For now, let's look at selecting the P and Q divide values based on the PFD frequency.

Determining P and Q Divide Values Based on the PFD Frequency

Since we know that the PFD frequency must be a common divisor of the reference frequency and VCO frequency, let's choose it to be the greatest common divisor. By doing this we will minimize P and Q, and maximize the PFD frequency:

$$Q = \frac{f_{ref}}{GCD(f_{vco}, f_{ref})} \tag{8}$$

$$P = \frac{f_{vco}}{GCD(f_{vco}, f_{ref})} \tag{9}$$

The range of valid VCO frequencies and N divide values is dictated by the following relationship:

$$\frac{f_{vco(\min)}}{f_{out}} \leq N \leq \frac{f_{vco(\max)}}{f_{out}} \tag{10a}$$

and all valid VCO frequencies are found by multiplying the list of N divide values by fout:

$$f_{vco} = f_{out} \cdot N \tag{10b}$$

Thus, given a reference frequency and desired output frequency, we can use equations 8, 9, and 10 to determine all possible sets of frequency synthesis parameters (sets of P, Q and N). There is only one small problem with equations 8 and 9: The Greatest Common Divisor function is only defined on the set of integers. We usually want to synthesis real frequencies, not integer frequencies. This means that we need to make a few mathematical adjustments to equations 8 and 9 to make them useful in practical applications.

This is done by first recognizing that the reference frequency and output frequency are both real numbers, which implies that they can always be represented as ratios of integers:

$$f_{ref} = \frac{\alpha_{ref}}{\beta_{ref}}, \; f_{vco} = \frac{\alpha_{vco}}{\beta_{vco}} \tag{11}$$

By multiplying both fractions by their denominators we will transform them into integers (since the product of two integers is always an integer):

$$\frac{\alpha_{ref}}{\beta_{ref}} \cdot \beta_{ref} \cdot \beta_{vco} = \alpha_{ref} \cdot \beta_{vco} \tag{12}$$

$$\frac{\alpha_{vco}}{\beta_{vco}} \cdot \beta_{ref} \cdot \beta_{vco} = \alpha_{vco} \cdot \beta_{ref} \tag{13}$$

We then can take the greatest common divisor of the result:

$$GCD(\alpha_{ref} \cdot \beta_{vco}, \alpha_{vco} \cdot \beta_{ref}) \tag{14}$$

This is almost the result we want. It is the GCD of two frequencies that are related to our original reference and VCO frequency. We can get back to the original values by dividing out the term we used to transform them:

$$\frac{GCD(\alpha_{ref} \cdot \beta_{vco}, \alpha_{vco} \cdot \beta_{ref})}{\beta_{ref} \cdot \beta_{vco}} \tag{15}$$

This result is equal of the greatest common divisor of our original reference frequency and VCO frequency. If we define a function called the greatest common divisor of rational numbers (GCDR) as:

$$GCDR(f_{ref}, f_{vco}) = \frac{GCD(\theta \cdot f_{ref}, \theta \cdot f_{vco})}{\theta} \qquad (16)$$

where

$$\theta = LCM(den(f_{ref}), den(f_{vco})) \qquad (17)$$

(note that LCM is the Least Common Multiple and den is the denominator part of fref and fvco respectively), then we can solve for P and Q using the following equations:

$$Q = \frac{f_{ref}}{GCDR(f_{vco}, f_{ref})} \qquad (18)$$

$$P = \frac{f_{vco}}{GCDR(f_{vco}, f_{ref})} \qquad (19)$$

By solving these equations for all VCO frequencies found in equation 10, the set of all divide values that will synthesize our desired output frequency from our given reference frequency is found.

**Power**

Power is dominated by the VCO frequency, charge pump current, and divide block settings. Most VCO architectures require larger tail currents to achieve higher frequencies. So as frequency increases, so does power consumption. Charge pump current is discharged once for each PFD period. When larger charge pump currents are required (for loop stability or fast startup / settling time) more power is consumed per PFD period. Clock dividers dissipate power at each clock edge. Larger clock divide values require more divide cells to transition, consuming more power.

**Startup Time / Settling Time**

The startup and settling time for a charge pump PLL is dominated by the loop natural frequency. This parameter can be thought of as the frequency slew rate of the PLL. It quantifies how fast the PLL can change the output frequency. It is proportional to the VCO gain and charge pump current, and inversely proportional to the feedback divide value and loop filter capacitance. Since the PLL output frequency is set by the VCO frequency, when we want to force a large step in the output frequency (either from zero at startup or from one setting to another) we need to force a large step in the VCO control voltage. This is accomplished by the charge pump dumping a large amount of charge onto the loop filter cap. The amount of frequency change per volt increase on the loop filter is set by the VCO gain. The rate at which the loop filter voltage is updated is set by the PFD frequency.

**Jitter (Cycle-to-Cycle)**

Cycle-to-Cycle jitter (the change in period length from one period to the next) can easily be dominated by the individual blocks of the PLL (VCO, dividers, reference oscillator), creating a situation where no loop parameter changes can improve performance. If you are working with a low noise PLL, then loop parameter settings can make a significant improvement.

Similar to startup time / settling time, the PFD frequency and VCO gain play a key role. Higher PFD frequencies mean that the PLL loop filter voltage is refreshed at a higher rate. This prevents the loop filter voltage from drifting. By using a large loop filter capacitance, the amount of voltage drift per PFD period is minimized. Because the VCO gain dictates how far the output frequency drifts per unit voltage drift on the loop filter, lower VCO gain makes the PLL less sensitive to loop filter voltage drift.

**Phase Noise**

Optimizing phase noise is highly application dependent, but a few general observations can be made. Phase noise contributed by the reference oscillator can be suppressed by setting the PLL to a lower closed loop bandwidth. Phase noise contributed by the VCO can be suppressed by setting the PLL to a higher closed loop bandwidth.

Phase noise divides down proportional to the output divide setting. If the output divider is a low noise divider, then running the VCO at a higher frequency and dividing the output frequency down will result in a phase noise improvement.

Table 2. Summary of key PLL performance parameters

| Parameter | Key Loop Parameters | Key Design Equations | Optimization |
|---|---|---|---|
| Power | VCO frequency ($f_{vco}$)<br>PFD Frequency ($f_{pfd}$)<br>Charge pump Current ($I_{chp}$)<br>Divide Values (P, Q, N) | $f_{vco} = f_{out} N$<br>$\omega_n = \sqrt{\frac{K_{VCO} I_{CP}}{P \cdot (C_L + C_S)}}$ | Minimize $f_{vco}$<br>Minimize $I_{chp}$<br>Minimize $f_{pfd}$<br>Minimize P, Q, N |
| Startup Time<br>Settling Time | PFD Frequency ($f_{pfd}$)<br>Charge Pump Current ($I_{chp}$)<br>Loop Filter Capacitance (C)<br>VCO Gain ($K_{vco}$) | $f_{pfd} = \frac{f_{ref}}{Q}$<br>$\omega_n = \sqrt{\frac{K_{VCO} I_{CP}}{P \cdot (C_L + C_S)}}$ | Maximize $f_{pfd}$<br>Maximize $I_{chp}$<br>Minimize C<br>Maximize $K_{vco}$ |
| Jitter<br>(cycle-to-cycle) | PFD Frequency ($f_{pfd}$)<br>Loop Filter Capacitance (C)<br>VCO Gain ($K_{vco}$) | $f_{pfd} = \frac{f_{ref}}{Q}$<br>$\omega_n = \sqrt{\frac{K_{VCO} I_{CP}}{P \cdot (C_L + C_S)}}$ | Maximize $f_{pfd}$<br>Maximize C<br>Minimize $K_{vco}$ |
| Phase Noise | Closed Loop Bandwidth<br>PLL Component Phase Noise | $H(s) = \frac{G(s)}{1+G(s)}$<br>$E(s) = \frac{1}{1+G(s)}$ | Depends on component noise figures. Use loop bandwidth to suppress reference noise and VCO noise. |

$\omega_n$ = natural frequency
$K_{vco}$ – VCO gain (Hz/V)
$I_{chp}$ – charge pump current (A/rad)
$C_L$, $C_S$ – Loop filter large cap, and loop filter small cap respectively
H(s) – closed loop transfer function
G(s) – open loop transfer function
E(s) – error transfer function

## CONCLUSION

Configuring a PLL for system applications can be an arduous task with lots of iteration. By first solving for all frequency synthesis parameters that meet our needs, we can then make well founded design choices that maximize flexibility and minimize cost.