**Phase Locked Loop Circuits**

A PLL is a feedback system that includes a VCO, phase detector, and low pass filter within its loop. Its purpose is to force the VCO to replicate and track the frequency and phase at the input when in lock.

Applications:

1. Clock generation
2. Frequency synthesizer
3. Clock recovery in a serial data link

Phase detector: compares the phase at each input and generates an error signal, ve(t), proportional to the phase difference between the two inputs. KD is the gain of the phase detector (V/rad).

VCO. In PLL applications, the VCO is treated as a linear, time-invariant system. Excess phase of the VCO is the system output. The VCO oscillates at an angular frequency, ωout. Its frequency is set to a nominal ω0 when the control voltage is zero. Frequency is assumed to be linearly proportional to the control voltage with a gain coefficient KO or KVCO (rad/s/v).

PLL dynamic response: The frequency step will cause the phase difference to grow with time since a frequency step is a phase ramp. This in turn causes the control voltage, Vcont, to increase, moving the VCO frequency up to catch up with the input reference signal. In this case, we have a permanent change in ωout since a higher Vcont is required to sustain a higher ωout.

Lock Range. Range of input signal frequencies over which the loop remains locked once it has captured the input signal. This can be limited either by the phase detector or the VCO frequency range.

Capture range: Range of input frequencies around the VCO center frequency onto which the loop will lock when starting from an unlocked condition. Sometimes a frequency detector is added to the phase detector to assist in initial acquisition of lock.

Approach: We will discuss the details of phase detectors and loop filters as we proceed. But, at this point, we will treat the PLL as a linear feedback system. We assume that it is already “locked” to the reference signal, and examine how the output varies with the loop transfer function and input. A frequency domain approach will be used, specifically describing transfer functions in the s-domain.

**Frequency and phase tracking loop:**

First we will consider the PLL with feedback = 1; therefore, input and output frequencies are identical. The input and output phase should track one another, but there may be a fixed offset depending on the phase detector implementation.

Phase error function: For the frequency synthesis application, we want to have ideally perfect phase tracking for phase and frequency steps. When the synthesizer frequency is changed, it is a discontinuous step in modulus, and we want to have zero steady state phase error in this case. We know that the phase detector will be producing an output equal to or at twice the carrier frequency, thus some low pass filtering will be needed. Let’s start with a simple RC lowpass network.

Bode Plot: If the loop filter frequency is lower than the crossover frequency, which you might want to do to attenuate the high frequency ripple from the phase detector, then the phase margin can become unacceptably small. And, if we increase the loop gain, KV = KDKO, to reduce the residual phase error, we get even smaller phase margin. Thus, we have a conflict between stability of the loop and minimizing the phase error. However, the loop can be made to work if ω1 > ωcrossover. But, then we may have insufficient filtering of the phase detector output.

**FREQUENCY RESPONSE**

we can see how the 3 dB frequency and gain flatness varies with ζ. Also, we see that the natural frequency must be significantly greater than the maximum frequency of phase variation for the reference (φin) when ζ < 1 in order to avoid gain peaking. This is a consequence of the zero added to the transfer function. For applications that require very small gain peaking (such as clock recovery), ζ > 2 is often employed.

**PHASE ERROR**

There is no frequency error when the loop is locked • Input frequency = output frequency

But, it is possible to have a phase error for some input transient phase conditions. The phase error must remain bounded in order to keep the loop locked. To analyze in the frequency domain, we assume a sinusoidal phase variation at the input.

**TRANSIENT PHASE ERROR**

1. Phase step. there is only a transient phase error for a phase step. This is reasonable, because the control voltage must return to the same value after the phase step is completed. The frequency will be the same before and after the step.

2. Frequency step. There is a static “error”, but it can be made small by increasing KV. This is consistent with the idea that a shift in control voltage is needed to give a step in frequency. The phase error needed to generate this control voltage step varies inversely with the loop gain.

3. Frequency ramp. We could do the same exercise for a frequency ramp (Doppler shift). This gives an unlimited steady state error. So, a type I loop is not suitable for tracking a moving source.

Bandwidth: The loop 3 dB bandwidth is important for noise considerations. It is determined by ωn and ζ, so bandwidth must be determined in conjunction with the overshoot and settling time specifications. We find again that the formula is different for the case with a forward path zero as opposed to the feedback zero case that we discussed in the feedback lectures.

**Phase Frequency Detector**

The phase-frequency detector shown below is a widely used architecture in frequency synthesizers. As opposed to the XOR phase detector that we first considered, this one produces two outputs: QA and QB, or as is customary, UP and DOWN respectively.

PFD characteristic.

When the phases coincide, both outputs produce minimum width pulses. When there is a phase or frequency error, the width of the UP or DOWN pulses increases. When integrated by the loop filter, this causes the control voltage of the VCO to move toward the locked condition of equal frequency and phase.

Because both outputs must be combined to obtain the desired output, the loop filter must be modified for differential inputs as shown below. F(s) is the same as that of the single ended version.

**Charge Pump Loop Filter**

An alternative loop filter implementation called the charge pump is widely used for many applications. It is very convenient to implement in CMOS.

• The PFD output produces UP (QA) and DOWN (QB) pulses whose width is proportional to the phase error.

• Charge pump current sources I1 and I2 must produce exactly equal currents. They charge and discharge the capacitor, CP, in discrete steps.

• If there is a static phase error ∆φ at the PFD input, the capacitor, C, will be charged indefinitely – therefore, the DC gain is infinite: an ideal integrator. So, we expect to have zero static phase error. This is unlike the type I loop which gave ∆φ = ∆ω/KV steady state phase error.

• The CP PLL will detect small phase errors and correct them as long as the frequency of the phase error (jitter frequency) is within the loop 3 dB bandwidth. This phase comparison occurs on every cycle.

**Closed Loop Frequency Response**

The closed loop frequency response can be evaluated from H(jω). In Fig. 2-3, Gardner has plotted the magnitude as a function of ω/ωn.

Bandwidth: The loop 3 dB bandwidth is important for noise considerations. It is determined by ωn and ζ, so bandwidth must be determined in conjunction with the overshoot and settling time specifications. We find again that the formula is different for the case with a forward path zero as opposed to the feedback zero case that we discussed in the feedback lectures.

**Third-order CP PLL**

There is still one residual problem that we have overlooked. The phase detector produces pulses of variable width that activate the switches to either charge or discharge the capacitor CP. Now that we have added the resistor, however, we find that the control voltage coming out of the charge pump will jump up or down before settling to its steady state value. This occurs because you cannot change the voltage across a capacitor instantaneously, so the initial voltage drop occurs across RP, which then charges CP exponentially. This jumpy control voltage frequency modulates the VCO at the reference frequency, creating reference spurs. This is not such a big problem if N = 1 because the jump will be at the same frequency as the VCO. But, at larger N values, it creates sidebands and jitter.

**PLL Phase Noise**

We have considered how the bandwidth of the loop affects things like settling time and capture range. But it also plays a role in the PLL noise behavior.

For frequency synthesis, we are interested in low phase noise. There are at least 2 main sources:

1. reference noise – usually small since we frequently use a crystal oscillator

2. VCO noise – often high. We hope that the PLL will suppress most of the noise, at least close to the carrier.