Lab 5 - Phase Lock Loops

Goal

Carrier phase estimation of passband signals using phase-locked loops.

Pre- lab

Reading

1. ZT 3.4.1, 3.4.3,(All of this should be review from 154a)
2. Additional notes on PLLs on the class Web site. Be sure to study Figure 11 in detail, as this is the VI you will implement in week 2.

Problems and Simulations

1. *Correcting for phase offset θ*
   Eq.(3.269) in ZT was derived for an arbitrary phase offset between the received signal and the local oscillator shown in Figure 3.69.

   a) Re-derive Eq.(3.269), which is $y_{DD}(t)$ in Figure 3.69 if

   $$x_c(t) = A_c\left(m_1(t)\cos(\omega_c t) - m_2(t)\sin(\omega_c t)\right)$$

   This is the standard form for a quadrature signal from the notes on passband modulation.

   b) Derive the amplitude $A(t)$ and the phase $\theta(t)$ from $m_1(t)$ and $m_2(t)$. (See the notes on passband modulation.)

   c) Derive $y_{DQ}(t)$ using the same form for $x_c(t)$.

   d) Derive expressions for $m_1(t)$ and $m_2(t)$ in terms of $y_{DD}(t)$, $y_{QD}(t)$ and $\widehat{\theta}$. (Hint: The phase offset is a rotation the signal "axes" used to represent the signal. This rotation is $e^{j\theta(t)}$ in complex form.)

   e) Sketch a block diagram of how to recover $m_1(t)$ and $m_2(t)$ from $y_{DD}(t)$, $y_{QD}(t)$ and $\widehat{\theta}$.

2. *Block diagrams for three forms of PLLs.*
   Print out three copies of Figure 11 in the PLL notes. This is the figure of the PLL implemented in lab.

a) On the first copy circle the parts of the VI that implement:

   i. The I and Q demodulation

   ii. The loop filter

   iii. The VCO

   Examining the VI, what part requires the most computational resources from the processor?

b) On the second copy hand draw the modifications to the VI to create a "standard" PLL as shown in Figure 2 of the PLL notes.

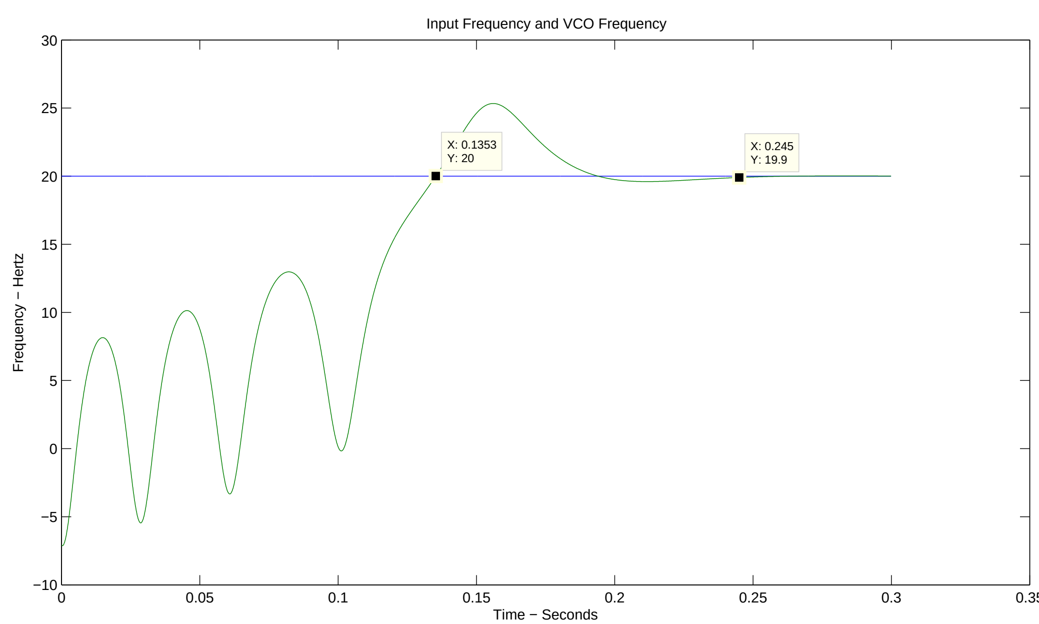
c) On the third copy of the VI, hand draw the modifications to implement a hard limiter in the in-phase arm of the Costas loop as shown in Figure 12 of the PLL notes.

3. *Simulation of a Costas loop*

   a) Run Computer Example 3.3 and reproduce Figure 3.52 and 3.53 for $\Delta f = 20$Hz and Δf=40 Hz using $f_n = 10$ Hz and $\zeta = 1/\sqrt{2}$. Using Figures 8 and 9 in the PLL notes as an example determine:

      i. The frequency pull-in time and the phase pull-in time until the frequency is within 0.5% of the final value. (Note: these values can be determined quickly using the *Data Cursor* feature in the Figure tools. More information is here)

      ii. Repeat for $\Delta f = 2$ Hz and compare the frequency pull-in and phase pull-in for all three cases: What is the same? What is different?
      Turn in the three figures with the two data cursors on the curves as shown below:



   b) Now modify the Computer Example 3.3, to implement a Costas loop shown in Figure 3.57. Generate curves for $\Delta f = 2$Hz, $\Delta f = 20$Hz and Δf=40 Hz and again measure the frequency pull-in time and the phase pull-in time. Compare with the curves generated in part (a) and determine what is different about the two loops and what is the same.

i. Change $f_n$to 20 and 40 Hz and measure the phase-pull in time in each case. What is the approximate relationship between the phase pull-in time and f_n?

c) Now add an independent zero-mean gaussian noise with rms value $\sigma = 0.05$ to both the in-phase and quadrature arm of the Costas loop before the multiplier. Plot the VCO frequency vs. time for a frequency for a step of $\Delta f = 2$ Hz, $\zeta = 1/\sqrt{2}$, for $f_n = 2$, 5 and 10 Hz. What is the qualitative relationship between the phase noise (spread of frequency values of the VCO frequency) and f_n?

1 "Manual" PLL

Insight to into the operation of a PLL can be obtained by replacing the VCO in the feedback loop with you. The VI in this section will simulate a carrier wave with an unknown combination of phase offset, frequency offset, and noise. The output of the phase detector $e_d(t) = K_f \sin(\phi(t) - \theta(t))$ will be displayed as a time graph. You will be given a software control (knob) that emulates the output of the output of the VCO which is the estimated phase of the carrier.

The instantaneous frequency out of the VCO is then $f = K_v e_v(t)$ and input the phase detector is $\theta(t) = K_v \int_{-\infty}^{t} e_v(\tau) d\tau$ which is the integral of the input that you control.

1. Open "ManualPLL.vi" . Try and lock the signal using the following settings. You should only spend about a minute on each one:

   a) Freq Offset =0, Bandwidth =10, Noise =0,
   b) Freq Offset =0, Bandwidth =10, Noise =0.04. (You may not be able to lock.)
   c) Freq Offset =0, Bandwidth =2, Noise =0.04. (Should be able to lock.)
   d) Freq Offset ~2%, Bandwidth =10, Noise =0. (Harder to lock)
   e) Freq Offset ~2%, Bandwidth =10, Noise =0.04. (Very hard to lock.)
   f) Freq Offset ~2%, Bandwidth =2, Noise =0.04. (Hard to lock.)
   g) Freq Offset ~4%, Bandwidth =10, Noise =0.04. (Good luck!)
   h) Freq Offset ~4%, Bandwidth =2, Noise =0.04. (May be possible.)

   The explanation of these results is required for the lab write-up.

2 Performance of PLLs

We will now study the performance of two PLLs:

1. Standard PLL with a PI filter

2. Costas loop with a PI filter

How to modify the PLL VI for each case was asked as a Prelab question. A signal generation VI that sets the frequency and the noise added before detection will be provided. Connect the output of the AWG to the digitizer.

2.1 Basic Measurements

Start with the PLL VI configured to a "standard" PLL. You will have to modify the supplied PLL code. Set the IF frequency to 1 MHz. Display the VCO frequency out vs. time and measure or estimate the frequency pull-in and phase pull-in times for each of the cases using cursors on the plots. Dump the first three cases as figures.

1. $f_n = 50$ kHz, $\Delta f = 10$ kHz, $\sigma = 0$ $\zeta = 1\sqrt{2}$. (linear no noise, optimal damping).
2. $f_n = 50$ kHz, $\Delta f = 50$ kHz, $\sigma = 0$ $\zeta = 1\sqrt{2}$. (nonlinear, no noise, optimal damping).
3. $f_n = 50$ kHz, $\Delta f = 100$ kHz, $\sigma = 0$ $\zeta = 1\sqrt{2}$. (more nonlinear, no noise, optimal damping).
4. $f_n = 50$ kHz, $\Delta f = 10$ kHz, $\sigma = 0$ $\zeta = 0.1$. (linear, no noise, underdamped).
5. $f_n = 50$ kHz, $\Delta f = 10$ kHz, $\sigma = 0$ $\zeta = 2$. (linear, no noise, overdamped).
6. $f_n = 50$ kHz, $\Delta f = 10$ kHz, $\sigma = 0.1$ $\zeta = 1\sqrt{2}$. (linear, noise, optimal damping).
7. $f_n = 100$ kHz, $\Delta f = 10$ kHz, $\sigma = 0.1$ $\zeta = 1\sqrt{2}$. (linear, more noise, optimal damping).
8. $f_n = 20$ kHz, $\Delta f = 10$ kHz, $\sigma = 0.1$ $\zeta = 1\sqrt{2}$. (linear, less noise, optimal damping).

Reconfigure the PLL as a Costas loop and repeat Steps 1-3 again dumping the figures. All together, you should have 11sets of measurements and six figures.

2.1.1 High natural frequency -effect of sampling and the processor

When the rate of change of the frequency is large with respect to the maximum sampling rate (2 Msamples/s for the VI we are using) or the speed of the processor, the PLL will begin to degrade and eventually fail to achieve lock. In this section we examine this.

1. Start with the following setting: $f_n = 250$ kHz, $\Delta f = 50$ kHz, $\sigma = 0$, $\zeta = 1\sqrt{2}$. With the exception of the scaling factor of 5, this curve should be identical to that of Step 1 in the last part. Confirm this.
2. Increase $\Delta f = 250$ kHz. This should be similar to the Figure for Step 2 in the last section. Dump the trace and record the pull-in times.
3. Open Matlab in another window and run the following command in command line mode `data=normrnd(0, sigma,[1 100000]);` This command will generate 10^5 random numbers and will take some time to complete. While it is running, repeat Step 2 and dump the trace. Are the pull-in times different? Why?
4. Stop Matlab and increase Δf in steps of 50 kHz starting at 300 kHz until Δf= 500 kHz. Are the results the same as in the last part? At what value of Δf does the loop fail to acheive lock?

2.2 Tracking the frequency of an external function generator

Connect the FGEN VI to the PLL. Set the frequency to about 1 MHz and start the PLL VI. Using the frequency adjust knob on the front of the the FGEN VI, observe the output of the PLL and how the signal changes as the frequency deviation increases. Can you change the frequency fast enough to have the PLL fail to lock? Why?

2.3 Response to BPSK signal.

Using $f_n = 250$ kHz, and the signal generation VI with $\sigma = 0$, display the VCO frequency vs. time for a BPSK waveform modulated at 100 kbit/s for each of the two PLLs.

3 Post Lab

1. Qualitatively explain the results of the experiments using the manual PLL with respect to the bandwidth of the loop filter, the amount of noise, and the speed of the response of the loop.

2. Using the data, discuss the relative performance of the PLLs with respect to the following parameters:

   a) Linear regime vs. nonlinear regime as a function of $\Delta f / f_n$.

   b) Frequency pull-in time vs. phase pull-in with respect to damping term ζ.

   c) Phase pull-in time vs. total amount of phase noise with respect to f_n.

3. Accounting for the scaling factors, what parameter sets used in lab agree with the Matlab simulations? Which parameters sets do not agree?

4. For a sampling rate of 2 Msample/s and a natural frequency of 500 kHz, the loop fails to lock for a Δf= 500 kHz. Scaling the values by a factor of 1000, try and reproduce the experimental results using the Matlab simulation code.

   a) Are all of the effects that you see in lab attributable to the sampling rate?

   b) Are some attributable to the processor speed?

   c) Discuss an experiment that you could run to separate the two effects.