NATCAR – Background Information

Steering Control-Loop Design
Part II

Prof. Richard Spencer

The System Model

Remember that the overall model for our plant is given by

$$H_p(s) = \frac{Y_s(s)}{C(s)} = \frac{k_\phi}{1 + s/\omega_\phi}\left(\frac{v^2}{L}\right)\left(\frac{1 + sL'/v}{s^2}\right)$$

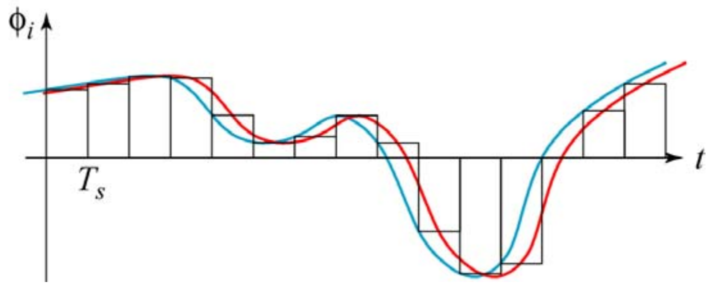
The controller transfer function is

$$H_c(s) = k_p\left(1 + sT_d + \frac{1}{sT_i}\right)$$

And we assumed the sensors have $H_s(s) = k_s$

The System Model Continued

- Also remember that the servo only samples the control voltage (about every 3ms) and then holds the value until it gets the next sample, so its input is really the sampled-and-held version of the control output, which we showed last time is roughly equivalent to adding some delay ($\sim T_S/2$) into the loop



Aside on Sampling

- We now want to quickly look at how to model this sample-and-hold operation mathematically
- The sample-and-hold function can be broken down into two separate functions; ideal impulsive sampling followed by a hold function
- Ideal sampling can be described mathematically by multiplying the waveform by a periodic string of impulses $s(t)$:

$$s(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT)$$

Sampling Continued

- Therefore, if we sample some waveform $x(t)$ and denote the sampled waveform by $x'(t)$ we find

$$x'(t) = x(t) \cdot s(t) = \sum_{k=-\infty}^{\infty} x(t)\delta(t-kT) = \sum_{k=-\infty}^{\infty} x(kT)\delta(t-kT)$$

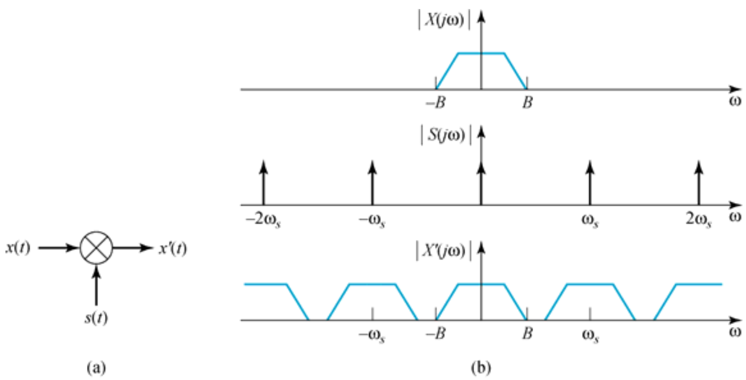
- The Fourier transform of this infinite comb of impulses, $s(t)$, is itself an infinite comb of impulses, $S(\omega)$:

$$S(\omega) = \frac{2\pi}{T} \sum_{k=-\infty}^{\infty} \delta(\omega - k\omega_s)$$

Sampling Continued

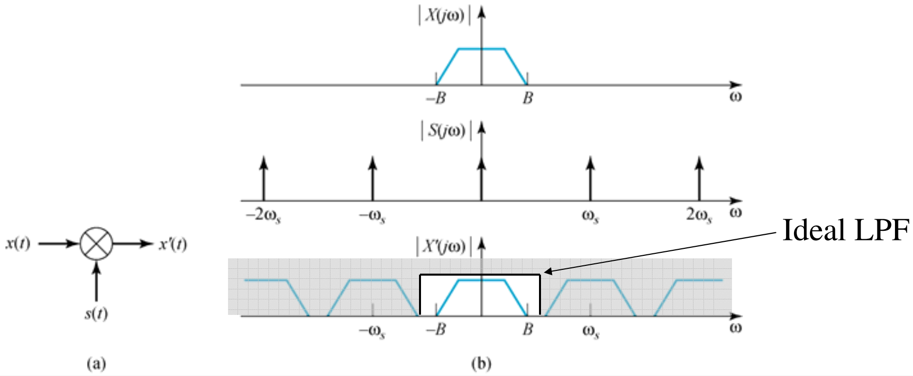
- Multiplication in the time domain implies convolution in the frequency domain, so we find:

$$X'(\omega) = \frac{1}{2\pi} X(\omega) * S(\omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X(\omega - k\omega_s)$$



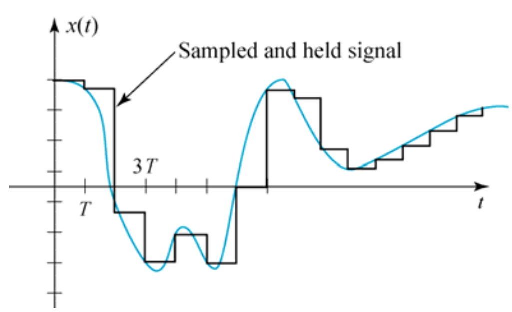
Sampling Continued

- So long as the sampling frequency, ω_s, is more than twice the signal bandwidth, B, we can recover the signal by low-pass filtering the sampled waveform (B is sometimes called the Nyquist frequency, so ω_s must be $> 2 \cdot \omega_{\text{Nyquist}}$)



Sampling Continued

- If the time samples are held, you get a waveform that looks like this:



- This can be generated mathematically by convolving the sampled signal $x'(t)$ with a pulse

Sampling Continued

- Assuming we have a pulse $p(t)$ defined by

$$p(t) = \begin{cases} 0;\ t < 0 \ \text{ or } \ t > T \\ 1\ ;\ 0 < t < T \end{cases}$$

- We find the sampled-and-held waveform is

$$x_{SH}(t) = x'(t) * p(t) = \sum_{k=-\infty}^{\infty} x(kT)\,p(t - kT)$$

- The Fourier transform of $p(t)$ is

$$P(\omega) = \int_{-\infty}^{\infty} e^{-j\omega t} p(t)dt = \left[\frac{-e^{-j\omega t}}{j\omega}\right]_0^T = \frac{1 - e^{-j\omega T}}{j\omega}$$

Sampling Continued

- We had $$P(\omega) = \frac{1 - e^{-j\omega T}}{j\omega}$$
- And we had

$$x_{SH}(t) = x'(t) * p(t) = \sum_{k=-\infty}^{\infty} x(kT)\,p(t - kT)$$

- Therefore, their spectra are multiplied in the frequency domain and we have

$$X_{SH}(\omega) = X'(\omega) \cdot P(\omega)$$

Sampling Continued

- Note that

$$P(\omega)=\frac{1-e^{-j\omega T}}{j\omega}=\frac{2e^{-j\frac{\omega T}{2}}}{\omega}\left(\frac{e^{j\frac{\omega T}{2}}-e^{-j\frac{\omega T}{2}}}{2j}\right)=\frac{2e^{-j\frac{\omega T}{2}}}{\omega}\sin\frac{\omega T}{2}$$

$$=Te^{-j\frac{\omega T}{2}}\frac{\sin\frac{\omega T}{2}}{\frac{\omega T}{2}}=Te^{-j\frac{\omega T}{2}}\operatorname{sinc}\frac{\omega T}{2}$$

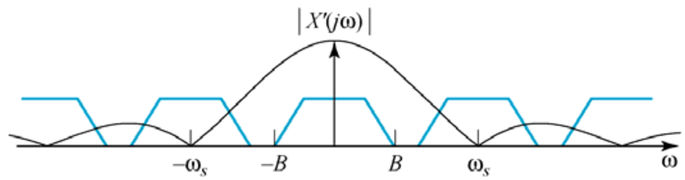
- And we had $X'(\omega)=\frac{1}{T}\sum_{k=-\infty}^{\infty}X(\omega-k\omega_s)$

Sampling Continued

- Therefore,

$$X_{SH}(\omega)=X'(\omega)\cdot P(\omega)=\left[\frac{1}{T}\sum_{k=-\infty}^{\infty}X(\omega-k\omega_s)\right]Te^{-j\frac{\omega T}{2}}\operatorname{sinc}\frac{\omega T}{2}$$

$$=e^{-j\frac{\omega T}{2}}\operatorname{sinc}\frac{\omega T}{2}\sum_{k=-\infty}^{\infty}X(\omega-k\omega_s)$$

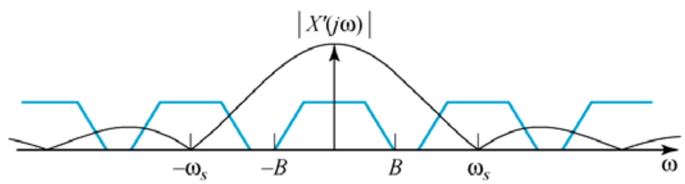


Sampling Continued

- Note that if we focus on the region around zero (because we will low-pass filter the output), we can restrict our attention to the $k = 0$ term,

$$X_{SH}(\omega) = X'(\omega) \cdot P(\omega) = \left[\frac{1}{T} \sum_{k=-\infty}^{\infty} X(\omega - k\omega_s) \right] P(\omega)$$

$$= \frac{1}{T} X(\omega) P(\omega) \equiv X(\omega) P'(\omega) \text{ for } k = 0$$



Sampling Continued

- In other words, the sample-and-hold transfer function has a strange low-pass transfer characteristic that we must allow for in our simulations
- We do *not* need to model the actual sampling operation if we use a continuous-time model and assume that the sampling frequency is high enough (since we can perfectly reconstruct the signal)
- In the Laplace domain, we have $P'(s) = \frac{1 - e^{-sT}}{sT}$
- This particular type of hold is called a zero-order hold (higher-order holds do a better job of interpolating between adjacent samples)

Simulation Model for the Loop

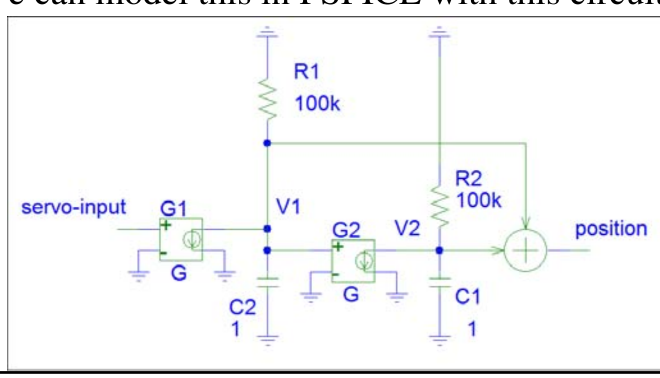
- But, we can't enter the zero-order hold directly into Matlab as a transfer function – it won't allow the perfect delay represented by the exponential (unless we do a discrete-time approximation to the entire control system in the *z* domain)
- An alternative to using MATLAB is to model the loop using SPICE – this method also has the advantage of allowing us to model nonlinearities
- The Laplace blocks available in PSPICE work in the frequency domain, but not in the time domain – so we must use controlled sources and elements to build up a behavioral model of the system

PSPICE Car Model

- The transfer function for the car is

$$Y_s(s) \approx \left(\frac{v^2}{Ls^2} + \frac{vL'}{Ls} \right) \Phi(s) = \left(\frac{v^2}{L} \right) \left(\frac{1 + sL'/v}{s^2} \right) \Phi(s)$$

- We can model this in PSPICE with this circuit



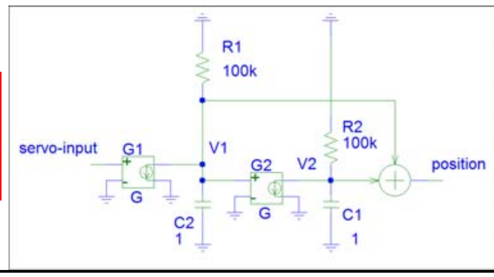
PSPICE Car Model Continued

- Ignore R_1 and R_2 at first – PSPICE requires them, but they don't affect the operation much. So

$$V_1(s) = \frac{-G_1}{sC_1} V_{in}(s) \text{ and } V_2(s) = \frac{-G_2}{sC_2} V_1(s) = \frac{G_1 G_2}{s^2 C_1 C_2} V_{in}(s)$$

- Therefore, $V_o(s) = V_1(s) + V_2(s) = \left(\frac{G_1 G_2}{s^2} - \frac{G_1}{s} \right) V_{in}(s)$

Note that both capacitors are 1 Farad

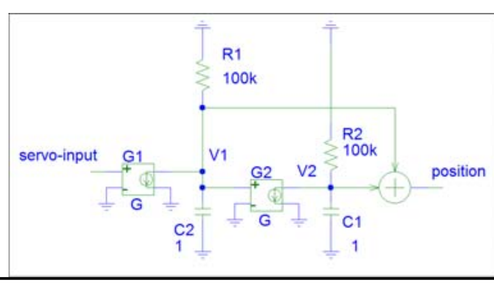


PSPICE Car Model Continued

- Setting $V_{in} = \Phi$ and $V_o = Y$, we have

$$Y_s(s) \approx \left(\frac{v^2}{Ls^2} + \frac{vL'}{Ls} \right) \Phi(s) = \left(\frac{G_1 G_2}{s^2} - \frac{G_1}{s} \right) \Phi(s)$$

- which requires $G_1 = \frac{-vL'}{L}$ and $G_2 = \frac{-v}{L'}$



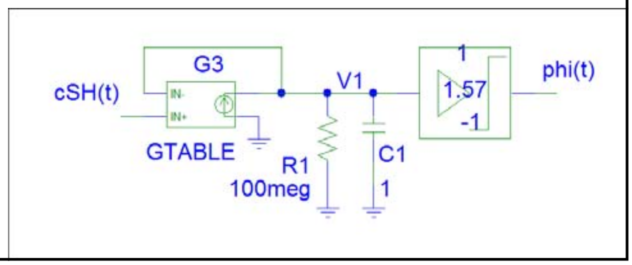
PSPICE Servo Model

- The servo can be modeled in PSPICE as shown below – this model includes the servo slew rate, small-signal bandwidth and limits
- The first block has negative feedback, so

$$V_1 = \frac{G_3 Z_1 C_{SH}}{1+G_3 Z_1} = \frac{G_3 R_1 C_{SH}}{1+sR_1C_1+G_3R_1} = \left(\frac{G_3R_1}{1+G_3R_1}\right)\frac{C_{SH}}{1+s/\omega_p}$$

- where

$$\omega_p = \frac{1+G_3R_1}{R_1C_1}$$



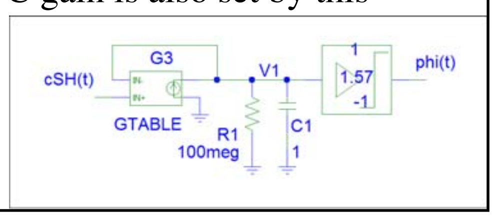
PSPICE Servo Model

- The slew rate is set by the maximum current available out of the GTABLE block; assuming it all flows into C_1, the maximum rate of change of V_1 is

$$\left.\frac{dV_1}{dt}\right|_{\max} = \frac{I_{\max}}{C_1}$$

- The steering limits are given by the voltage limit block, and the overall DC gain is also set by this block since we set

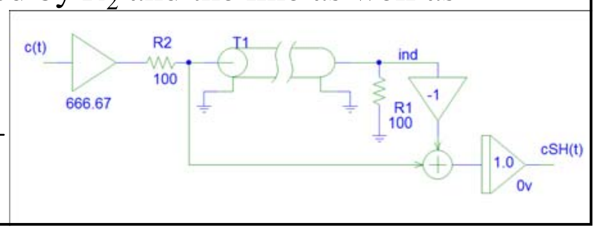
$$\left(\frac{G_3R_1}{1+G_3R_1}\right) \approx 1$$



PSPICE Zero-Order Hold Model

- The zero-order hold can be modeled using an ideal transmission line to provide the delay
- The transmission line must be terminated properly to prevent reflections – I used a 100Ω line with a 3ms delay (T1 in the schematic)
- The integrator block provides the $1/s$ term
- The input gain includes a factor of 2 to allow for the divider formed by R_2 and the line as well as the $1/T$ term

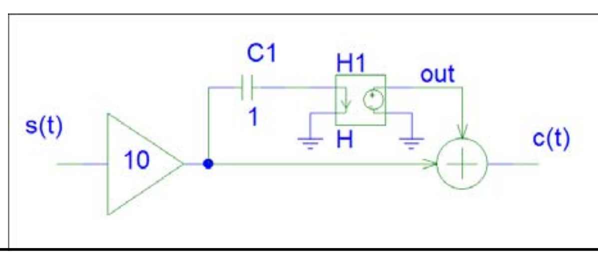
$$\frac{C_{SH}(s)}{C(s)} = P'(s) = \frac{1-e^{-sT}}{sT}$$



PSPICE Controller Model

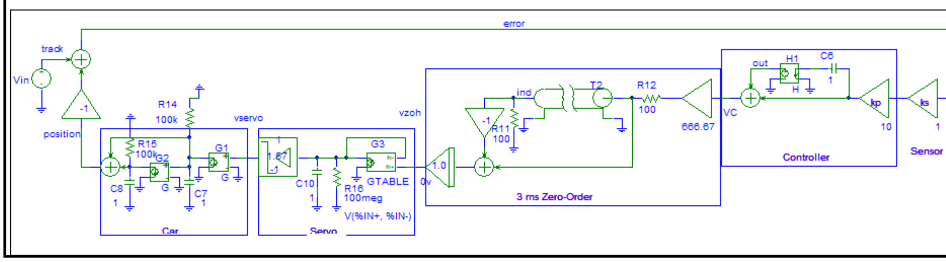
- I modeled a PD controller as shown below
- The input gain block sets k_p (10 in the schematic)
- The current in C_1 is the derivative of $k_p \cdot s(t)$, so the gain of the $H1$ block sets T_d
- The output is the sum of the derivative and proportional terms:

$$c(t) = k_p\left(1 + sT_d\right)s(t)$$



Complete PSPICE Model

- The complete PSPICE model for the feedback loop is shown below
- The sensors are modeled by a gain block and the track by an independent source
- The gain of -1 is used to make the summer take the difference between V_{in} and the car's position



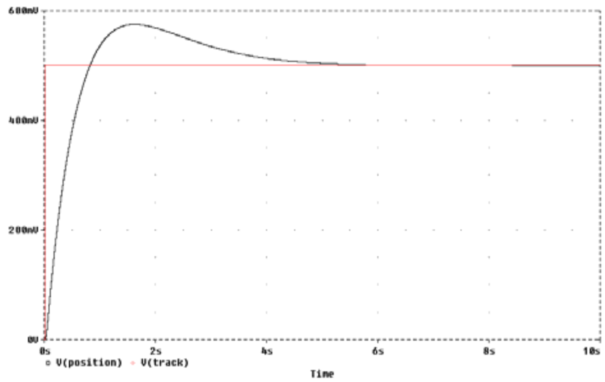
System Model; Example Numbers

- Assume the following:
  - The sensor gain is one (V/ft)
  - The servo gain is $\pi/2 = 1.57$ (rad/V)
  - The servo has a bandwidth of 100 rad/sec, a slew rate of 20 rad/sec (120º change in 0.1 sec) and limits of ±1 V (±60º)
  - The wheelbase is $L = 1$ foot and $L' = 1.5$ feet
- Let's use our PSPICE model to examine the step response for different velocities and controllers WARNING! You have to set the maximum step size in the transient simulations to a small value (I used 0.005ms most everywhere) and, perhaps, reduce the charge tolerance or you will not get accurate step responses

Step Response

- Here is the step response (6 in. step) with $k_p = 1$, $T_d = 0$ s and $v = 1$ ft/s

1V = 1 ft.

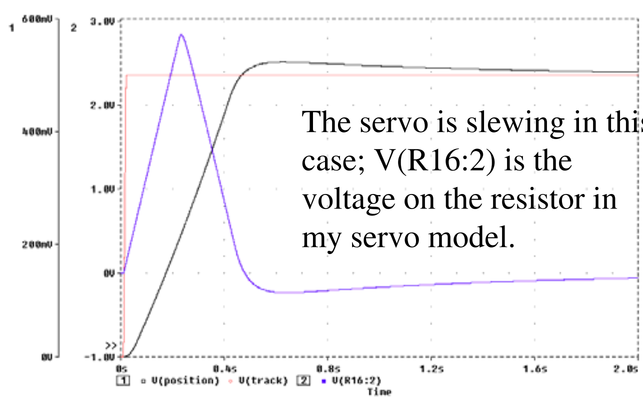


1 s = 1 foot

Step Response

- Here is the step response (6 in. step) with $k_p = 10$, $T_d = 0$ s and $v = 1$ ft/s. Note the change in x axis scale and that this has *less* overshoot than $k_p = 1$!

1V = 1 ft.



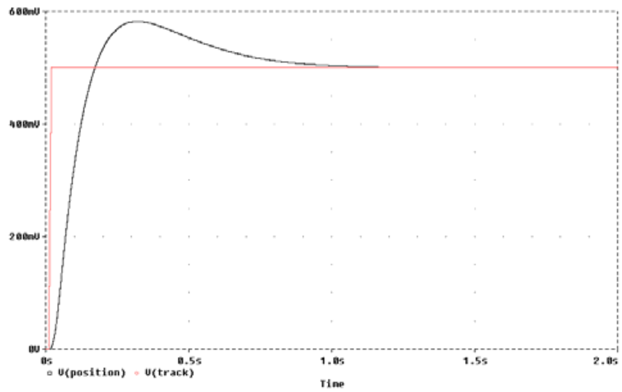
The servo is slewing in this case; V(R16:2) is the voltage on the resistor in my servo model.

1 s = 1 foot

Step Response

- Here is the step response (6 in. step) with $k_p = 1$, $T_d = 0$ s and $v = 5$ ft/s

1V = 1 ft.

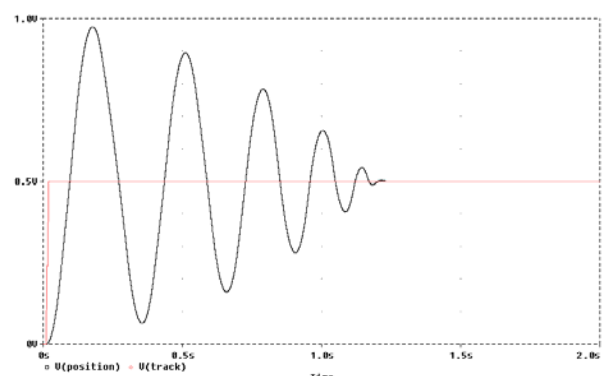


200 ms = 1 foot

Step Response

- Here is the step response (6 in. step) with $k_p = 10$, $T_d = 0$ s and $v = 5$ ft/s. The servo is again slewing. The loop is, in practice, unstable.

1V = 1 ft.

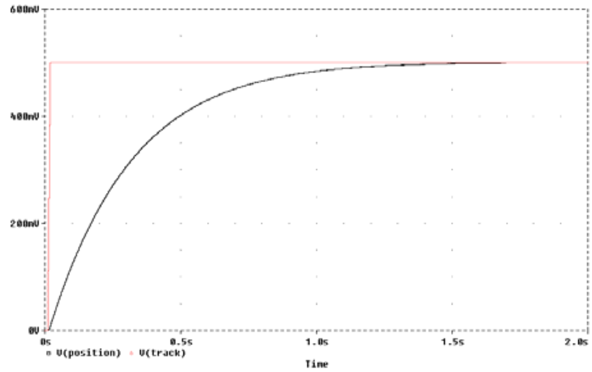


200 ms = 1 foot

Step Response

- Here is the step response (6 in. step) with $k_p = 10$, $T_d = 0.3$ s and $v = 5$ ft/s. The derivative has helped a lot!

1V = 1 ft.

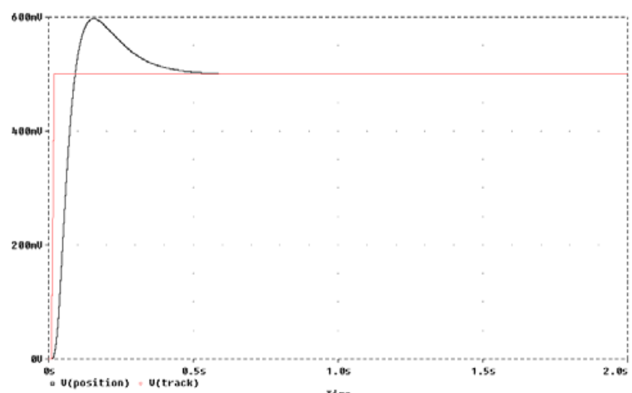


200 ms = 1 foot

Step Response

- Here is the step response (6 in. step) with $k_p = 1$, $T_d = 0$ s and $v = 10$ ft/s

1V = 1 ft.

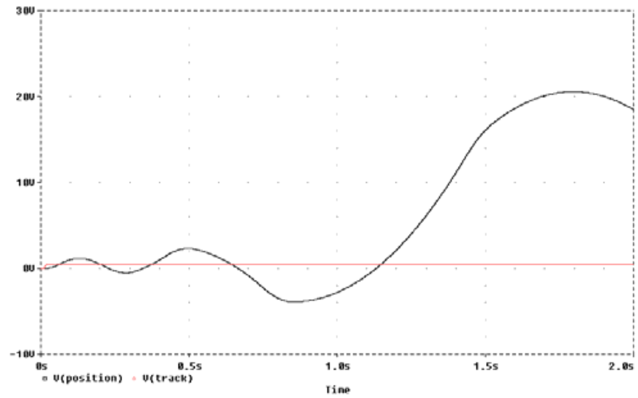


100 ms = 1 foot

Step Response

- Here is the step response (6 in. step) with $k_p = 10$, $T_d = 0$ s and $v = 10$ ft/s. The loop is unstable here!

1V = 1 ft.

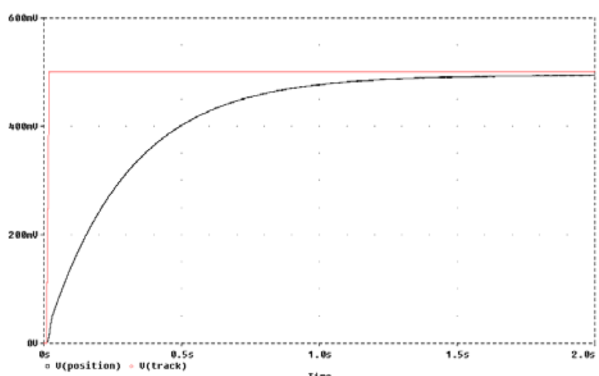


100 ms = 1 foot

Step Response

- Here is the step response (6 in. step) with $k_p = 10$, $T_d = 0.3$ s and $v = 10$ ft/s. The derivative has again cured the instability.

1V = 1 ft.



100 ms = 1 foot

Bode Plots

- You can look at the Bode plots, but because of the nonlinearity in the servo and the fact that the phase is not monotonically non-decreasing in magnitude, the results are of questionable value and we will not use them
- PSPICE will not plot a Nyquist plot