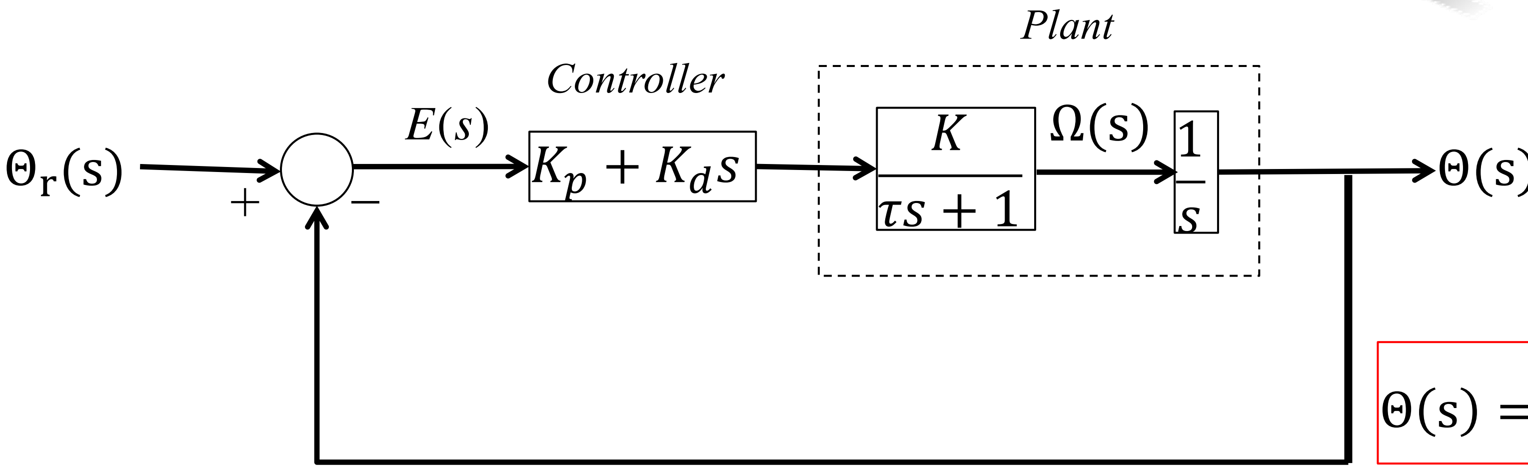
2.004 Lab 4 Intro
P and PD Control of Position

Fall, 2021

Today’s Tasks

- Closed-loop position control with derivative control action:
  - **Experiment #1:** Simulink Simulation
  - **Experiment #2:** Proportional Control of Position
  - **Experiment #3:** PD Control of Position and Comparison to Simulation
  - **Extra Credit:** Write code to command the wheel to tick 36 degrees every second
- Deliverable:
  - Lab 4 report (with prelab)

Plant Transfer Functions



$$\boxed{\Theta(s) = \frac{1}{s}\Omega(s)}$$

$$K = \frac{K_{dc}}{vc2pwm} \qquad vc2pwm = \frac{255\ [PWM]}{5\ [V]} = 51$$

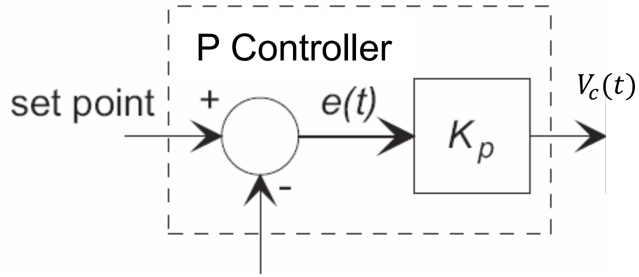
Plant TF:

$$\boxed{G_p(s) = \frac{\Theta(s)}{\Theta_r(s)} = \frac{1}{s}\frac{K}{\tau s + 1} = \frac{K}{\tau s^2 + s}}$$

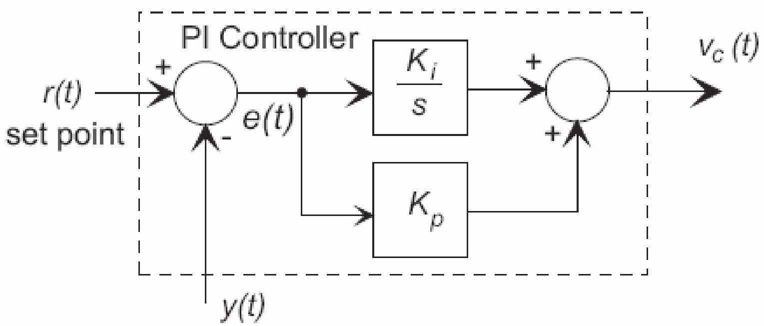
A “free” integrator

One open-loop zero: $z = \frac{-K_p}{K_d}$ Two open-loop poles: $\begin{cases} p_1 = 0 \\ p_2 = \frac{-1}{\tau} \end{cases}$

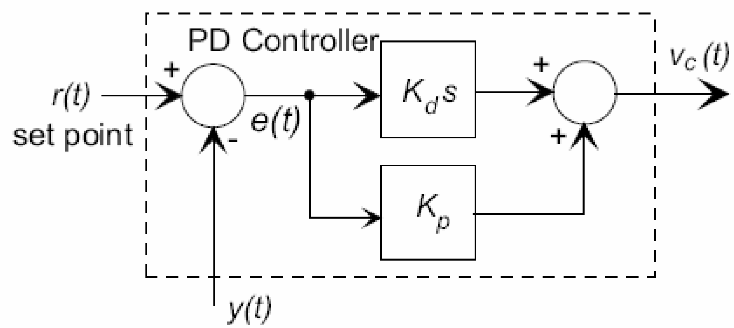
P, PI and PD Controllers



$$G_c(s) = K_p$$

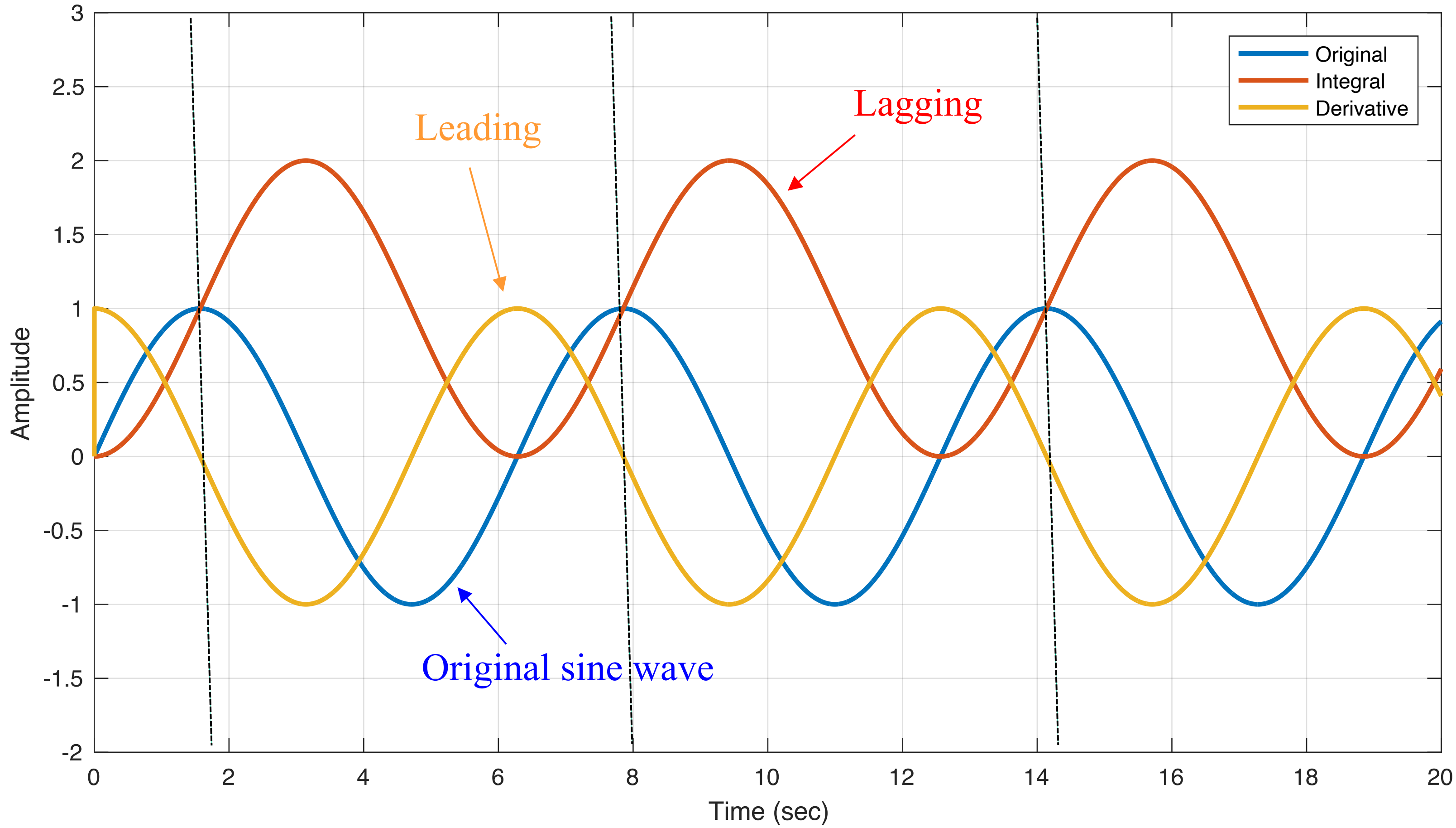


$$G_c(s) = K_p + \frac{K_i}{s}$$

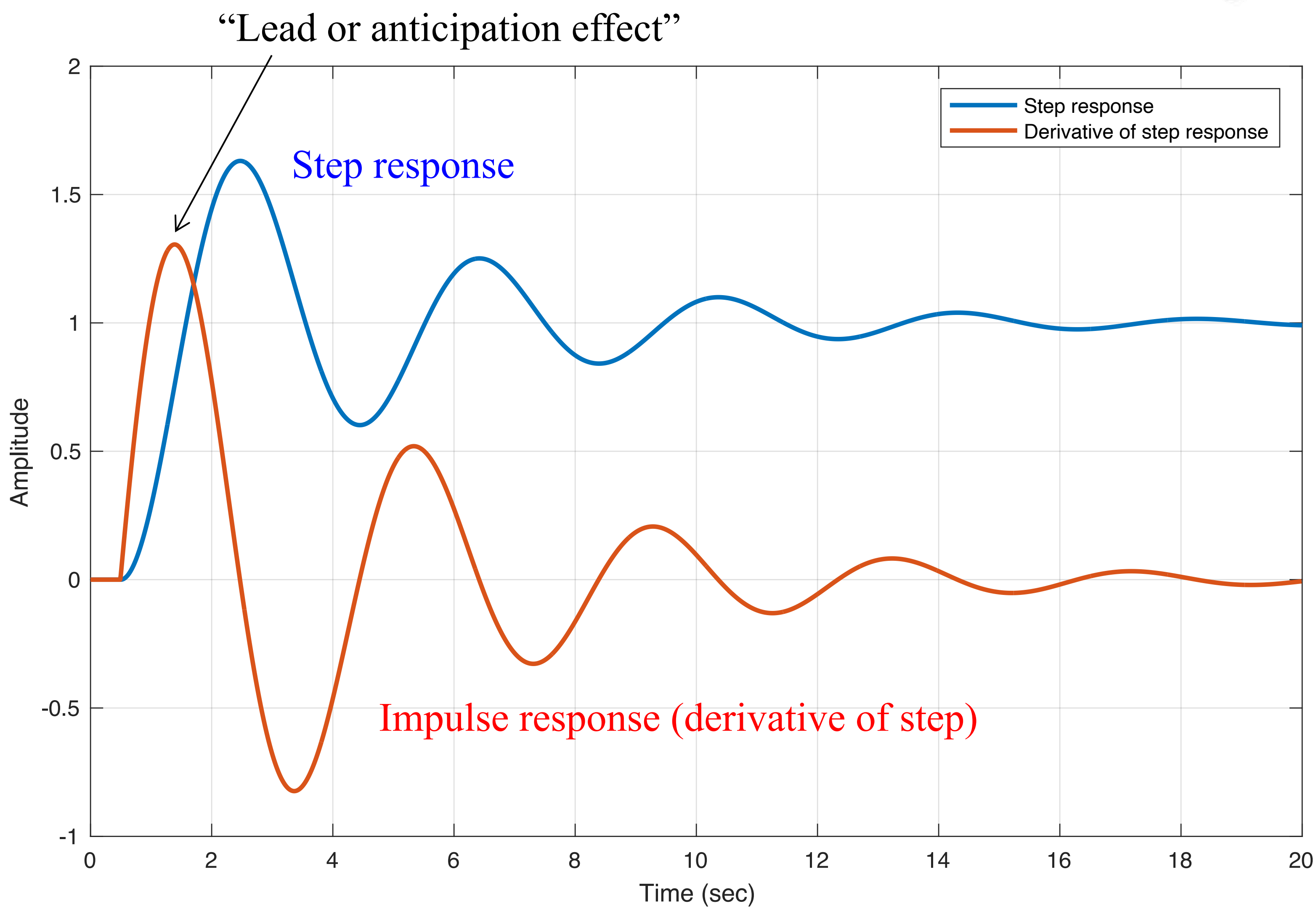


$$G_c(s) = K_p + K_d s$$

Integral and Derivative



Step Response and Its Derivative



Comparison of Closed-Loop Transfer Functions

P Control: $G_{cl}(s) = \dfrac{K_p K}{\tau s^2 + s + K_p K}$ $\Longrightarrow \zeta\omega_n = \dfrac{1}{2\tau}$

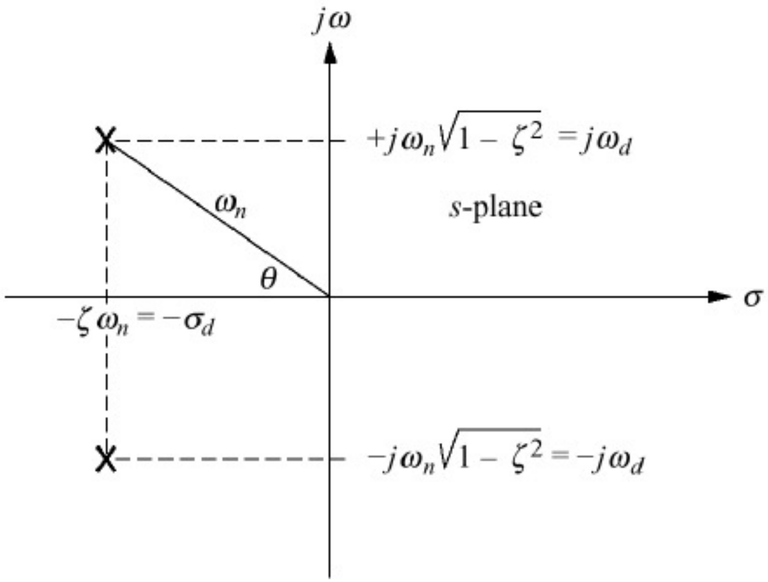
A "real" zero at $-\dfrac{K_p}{K_d}$

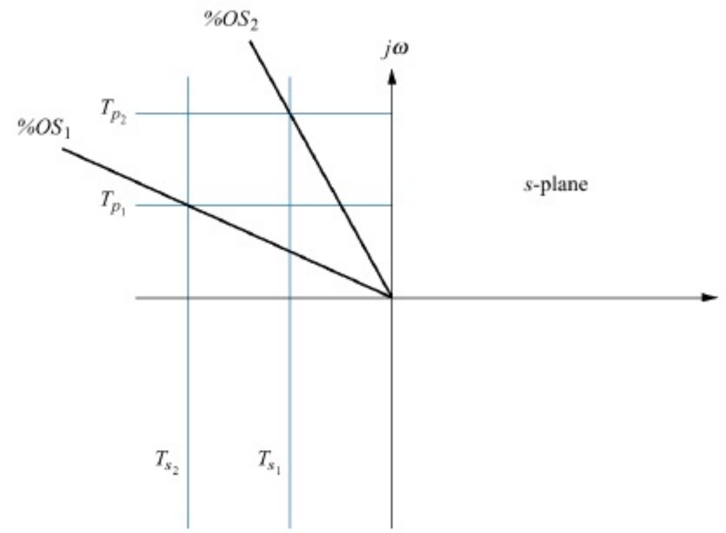
$s^2 + 2\zeta\omega_n s + \omega_n^2$

PD Control: $G_{cl}(s) = \dfrac{K(K_p + K_d s)}{\tau s^2 + (1 + K_d K)s + K_p K}$

Settling time ($\pm 2\%$): $T_s = \dfrac{4}{\zeta\omega_n}$

2nd Order System Poles





$$TF(s) = \frac{K_{dc}\cdot\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

$$s^2 + 2\zeta\omega_n s + \omega_n^2 = 0$$

$$\boxed{s_{1,2} = -\sigma_d \pm j\omega_d = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2}}$$

$$T_p = \frac{\pi}{\omega_n\sqrt{1-\zeta^2}}$$

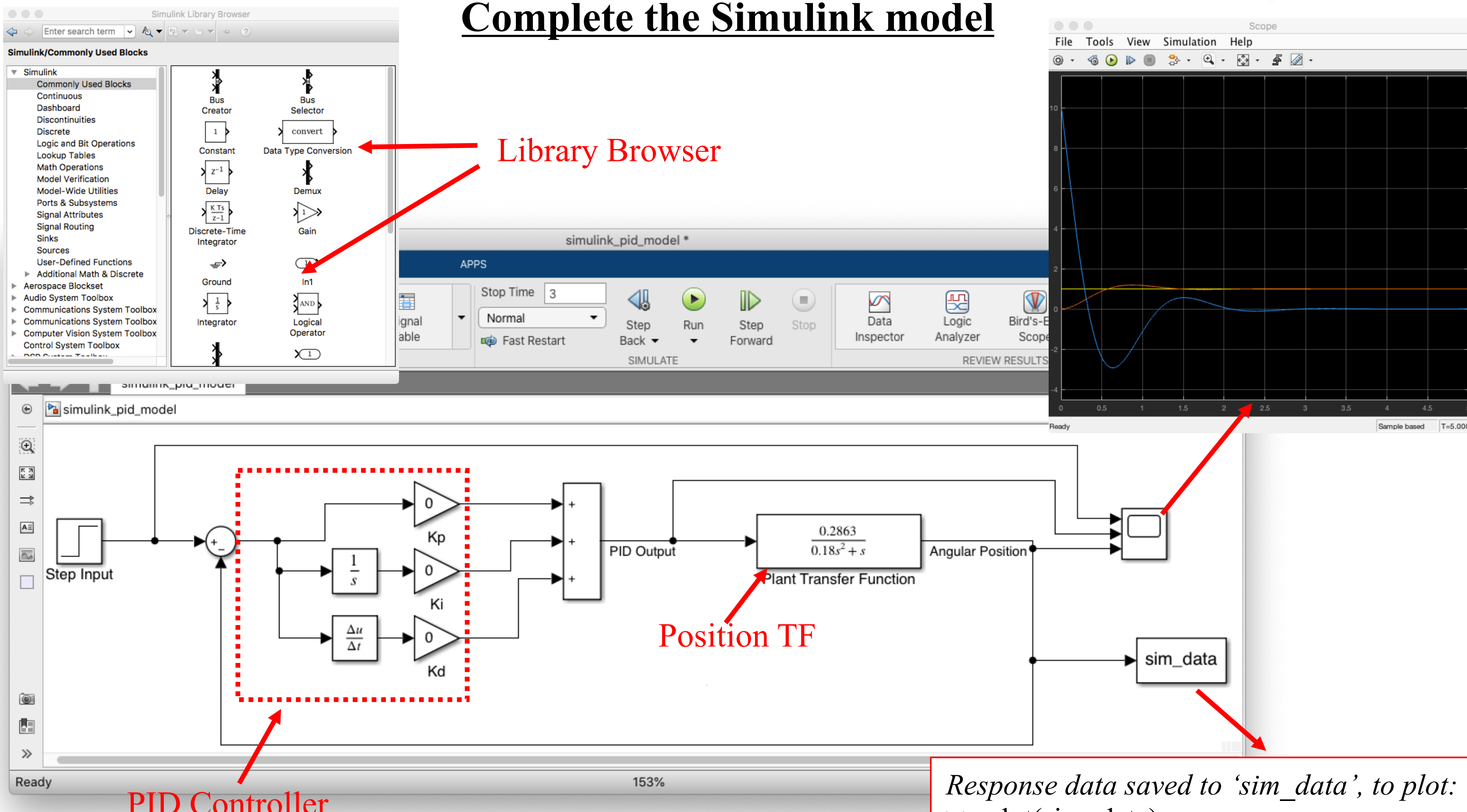
$$T_s = \frac{4}{\zeta\omega_n} \quad (\pm 2\%)$$

$$\%OS = e^{-\left(\frac{\zeta\pi}{\sqrt{1-\zeta^2}}\right)} \times 100$$

PID Simulink Model

MITMECHE

Complete the Simulink model



Response data saved to ‘sim_data’, to plot:
>> plot(sim_data)

Control Action Comparison

- ***Proportional action*** – improves speed but with steady-state error in some cases
- ***Integral action*** – improves steady state error but with less stability; may create overshoot, longer transient, or integrator windup
- ***Derivative action*** – improves stability but sensitive to noise; may create large output when the input is not a continuous signal

<u>*PID controller transfer function*</u>

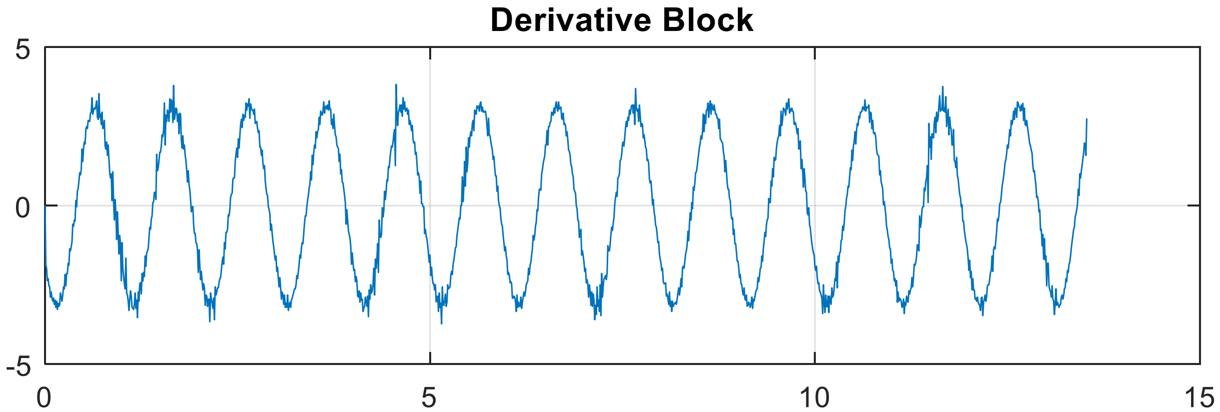
$$G_c(s) = K_p + \frac{K_i}{s} + K_d s$$

$$= \frac{K_d s^2 + K_p s + K_i}{s}$$

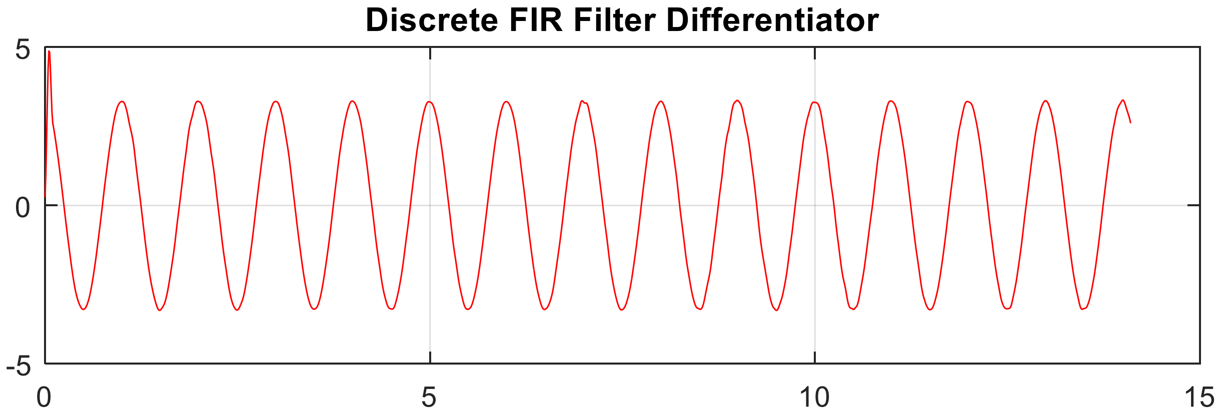
$$= K_d \left(\frac{s^2 + \left(K_p / K_d \right) s + \left(K_i / K_d \right)}{s} \right)$$

Practical Consideration

Use a Pseudo Differentiator



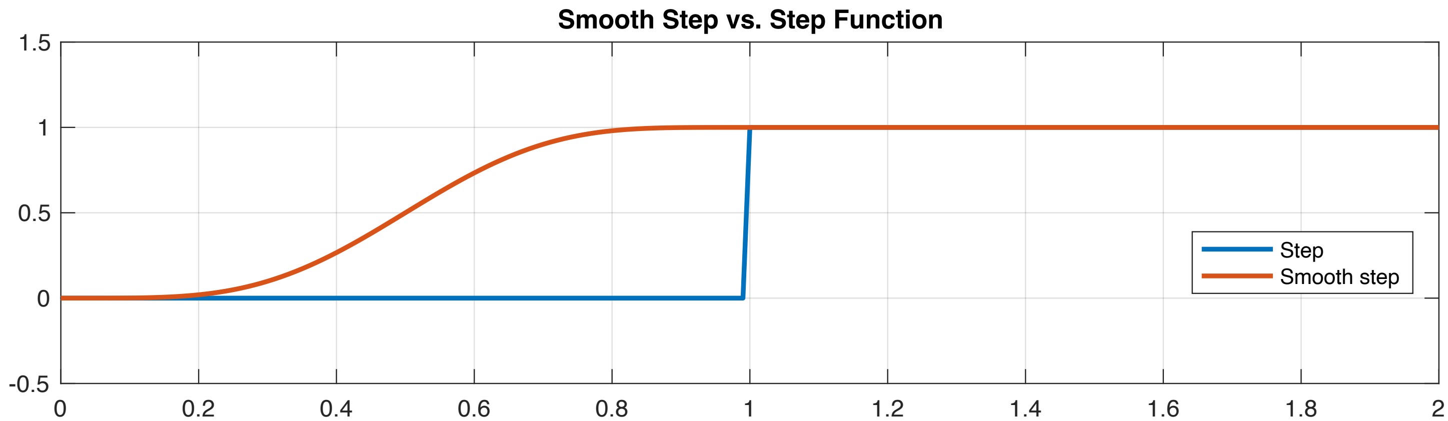
Finite difference method

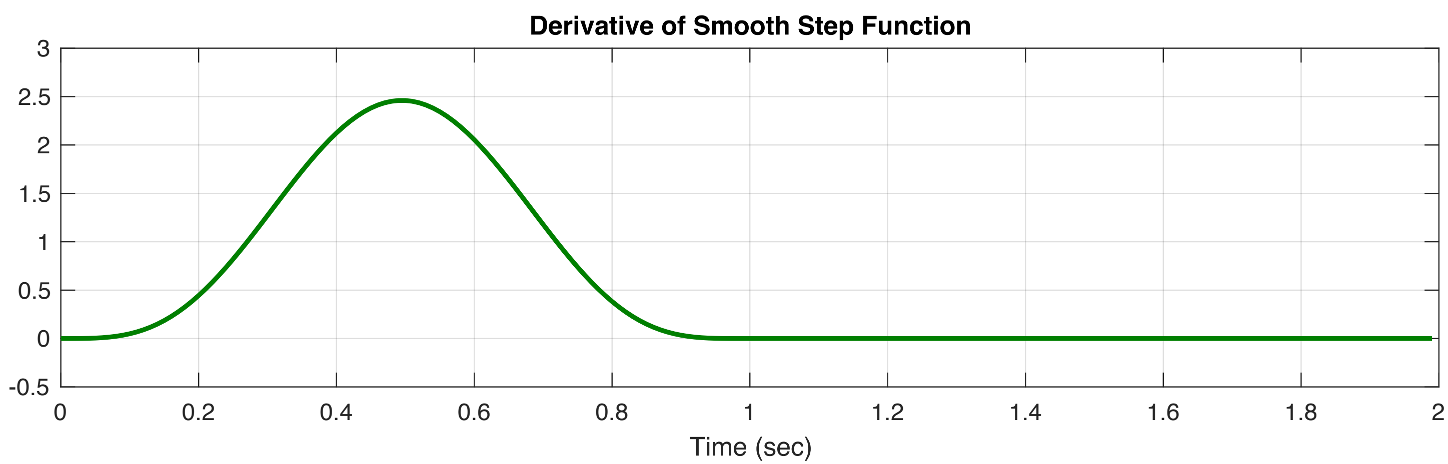


Band-limited differentiator to eliminate excess noise in the signal.

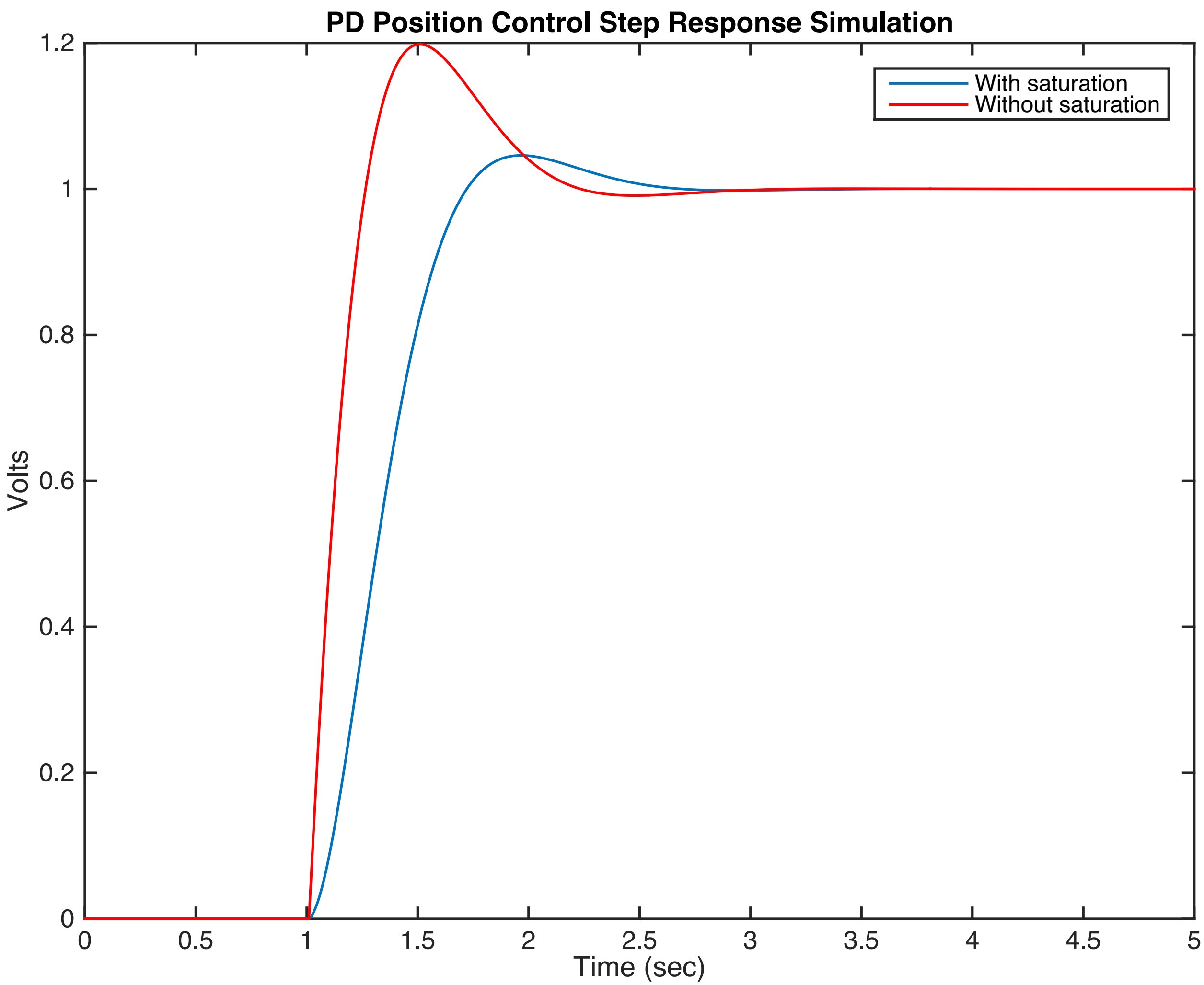
Practical Consideration

<u>Use a Smooth Step Input</u>





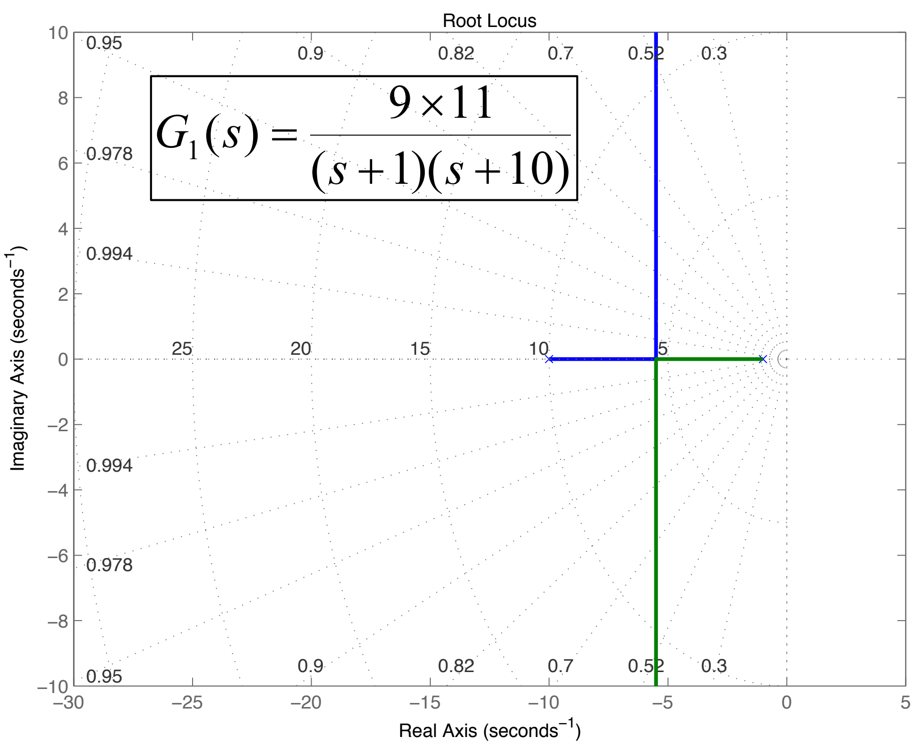
Effect of Saturation

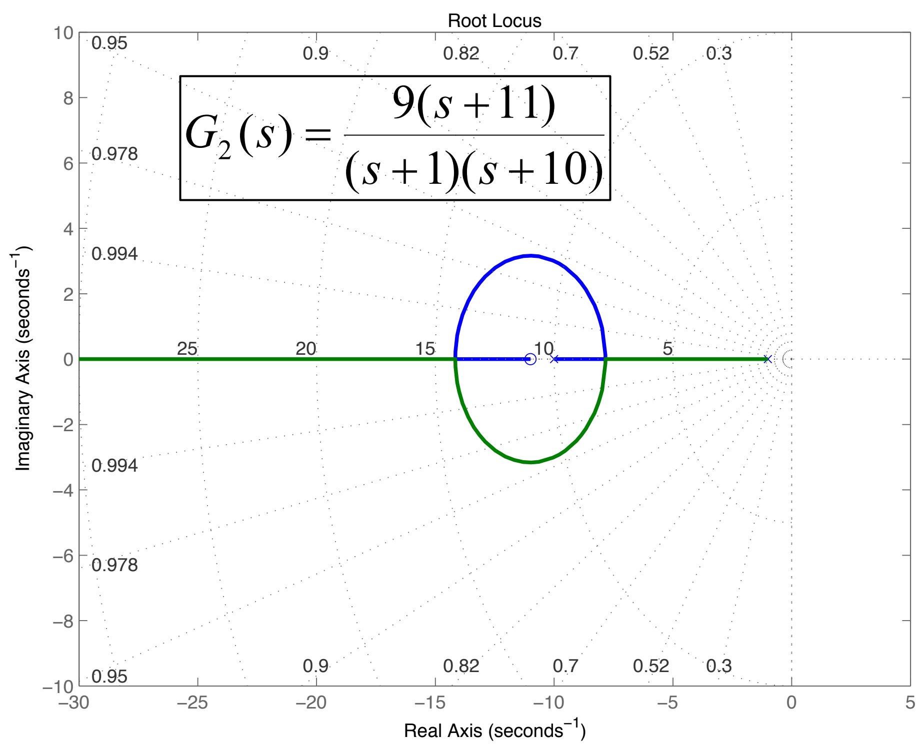


Effects of Zeros

In control design, zeros play a major role

- Zeroes reduce the effect of nearby poles
- The derivative (lead) effect

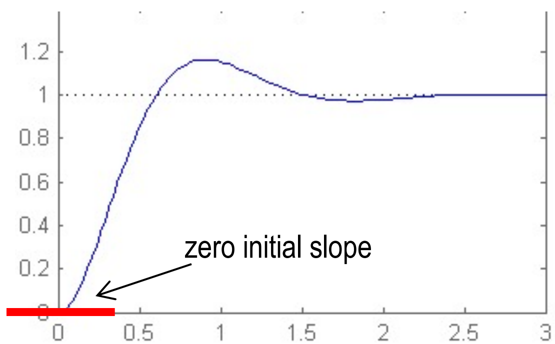




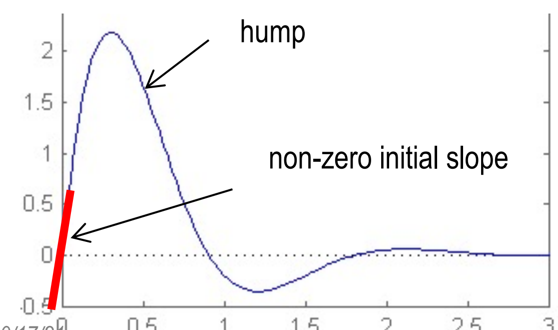
Effects of Zeros

$$G_0(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \qquad G(s) = (\alpha s + 1)G_0(s) \qquad z = -\frac{1}{\alpha}$$

Step Response: $$Y(s) = (\alpha s + 1)G_0(s)\frac{1}{s} = \underbrace{G_0(s)\frac{1}{s}}_{\text{The step response of } G_0(s)} + \underbrace{\alpha G_0(s)}_{\text{Impulse response}}$$



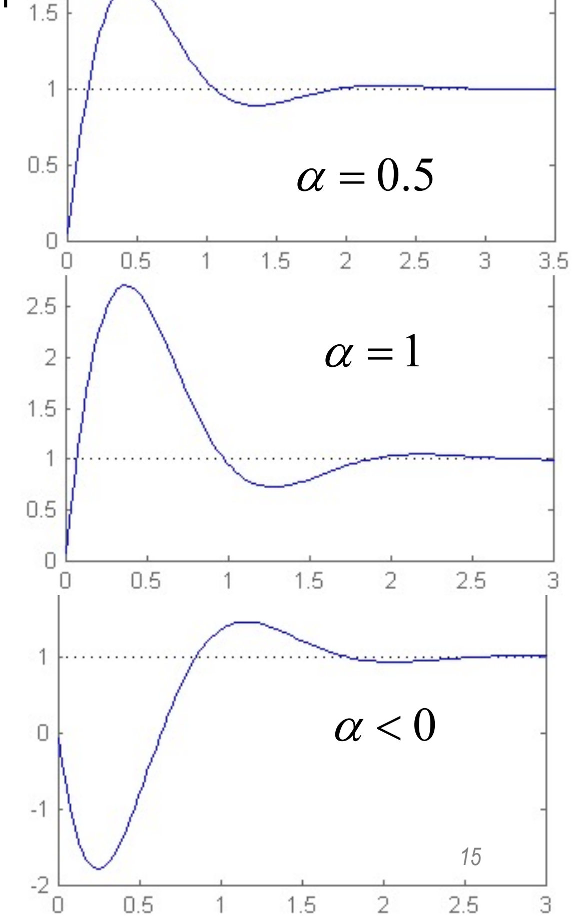
Step Response



Impulse Response

By adding the impulse response, i.e. the derivative of the step response, the resultant response becomes faster, but more overshoot.

For $\alpha < 0$, the step response exhibits an undershoot. The output first moves in the opposite direction. The response is also very slow.



Extra Credit Task

Write code to command the wheel to tick 36 degrees every second