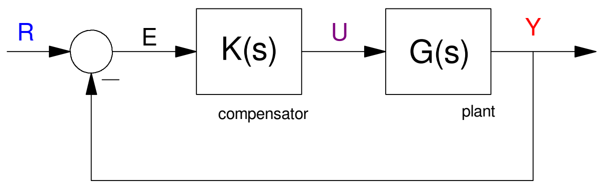
PID Compensators using Root Locus

Introduction

Given a feedback system around a plant, G(s), add a compensator K(s) to improve the closed-loop response.



One common type of compensator is a PID (Proportional Integral Derivative):

$$K(s) = P + \frac{I}{s} + Ds$$

Essentially, you're just adding poles and zeros as needed:

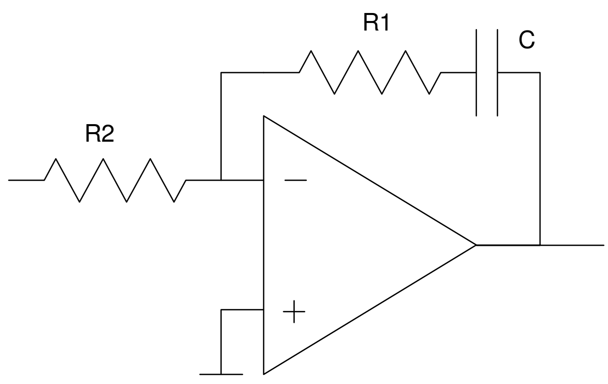
I: If the system is type-0, add a pole as s=0 to make it type-1. This reduces the steady-state error to zero.

PI: Add a pole at s=0 to make a type-0 system type-1. Add a zero to get rid of a bothersome pole.

PID: Ditto, but add two zeros to cancel two bothersome poles.

PID Circuits:

Circuits to implement these are as follows:



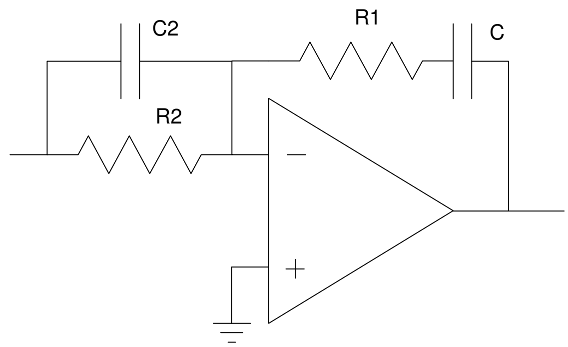
PI Compensator: $K(s) = -k\left(\frac{s+a}{s}\right)$

where

$$a = \left(\frac{1}{R_1 C}\right)$$

$$k = \frac{R_1}{R_2}$$

note: R1 = 0 for an I compensator, C = 0 for a P compensator.



PID Compensator: $K(s) = -\left(\frac{(R_1C_1s+1)(R_2C_2s+1)}{R_2C_1s}\right)$

Example: Take G(s) to be the 5th-order heat equation from before

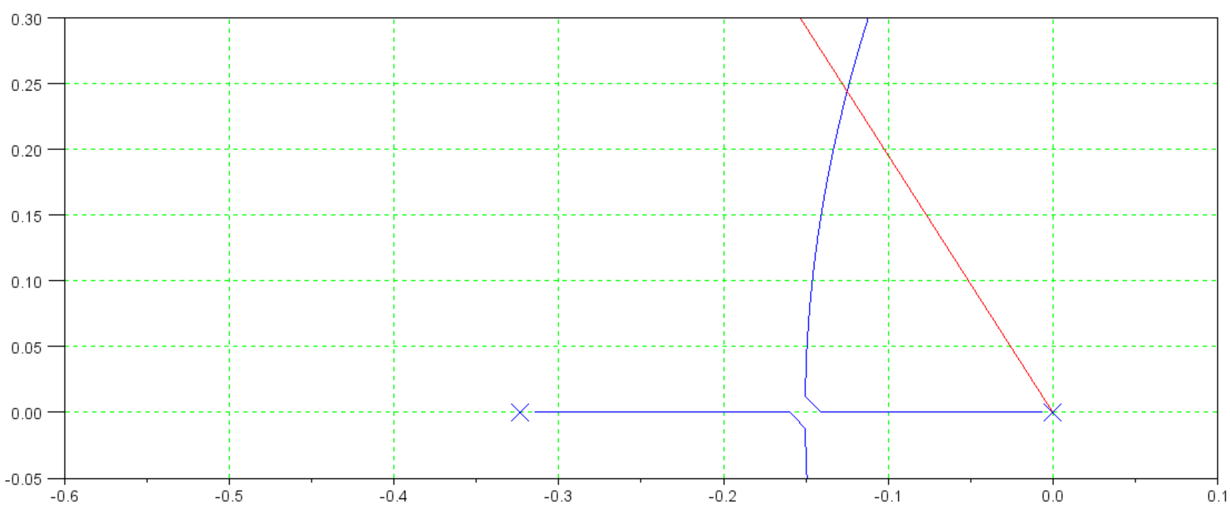
$$G(s) = \left(\frac{361.2378}{(s+15.65)(s+10.1)(s+5.439)(s+2.081)(s+0.3234)}\right)$$

I Compensation:

$$K(s) = \frac{I}{s} = k\left(\frac{1}{s}\right)$$

$$GK = \left(\frac{361.2378k}{(s+15.65)(s+10.1)(s+5.439)(s+2.081)(s+0.3234)s}\right)$$

The root locus becomes the following:



Root Locus for Integral Compensation

The point on the root locus which intersects the 0.4559 damping line is

s = -0.1249 + j0.2483

k at this point is

$$\left(\frac{361.2378k}{(s+15.65)(s+10.1)(s+5.439)(s+2.081)(s+0.3234)s}\right)_{s=-0.1249+j0.2483} = -1$$

k = 0.3974219

resulting in

$$K(s) = \left(\frac{0.3974219}{s}\right)$$

The steady-state error for a step input is zero: it's a type-1 system.

The error constant Kv is

$$K_v = \lim_{s\to 0}(s \cdot G \cdot K)$$

$$K_v = \left(\frac{361.2378k}{(s+15.65)(s+10.1)(s+5.439)(s+2.081)(s+0.3234)}\right)_{s\to 0} = 0.2481$$

The input at t=0 is

$$U_{t=0} = (K(s))_{s\to\infty} = 0$$

The resulting specifications for the I-compensated system are

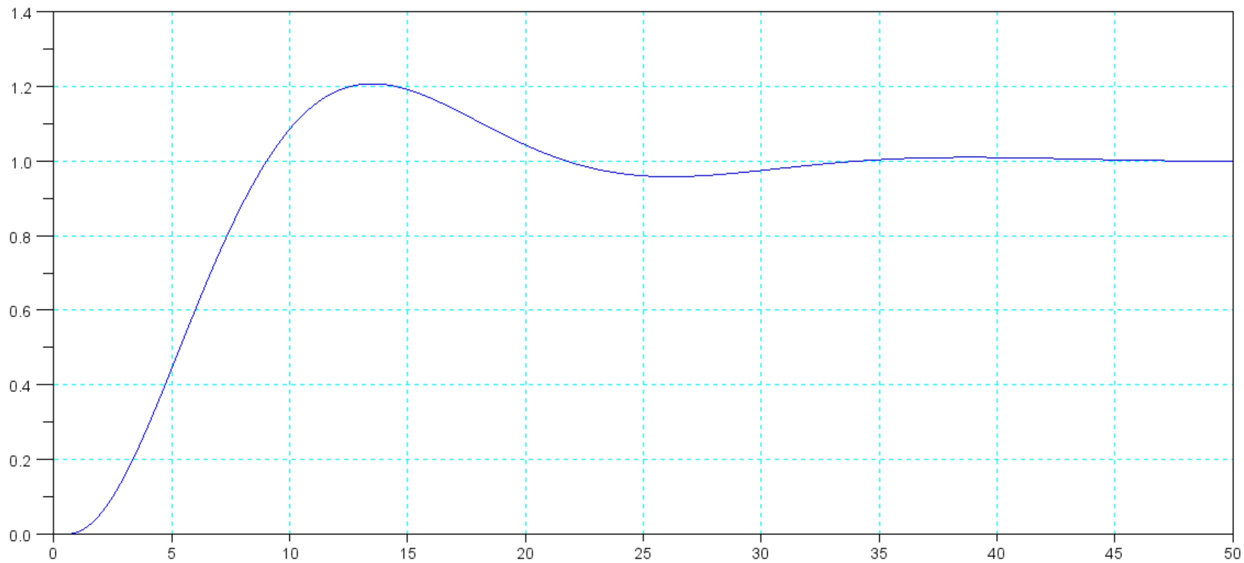
P and I Compensation				
K(s)	Closed-Loop Dominant Pole(s)	U at t=0 K(s) as s -> infinity	Kv	2% Settling Time seconds
5.5117	-0.6942 + j1.3884	5.5117	0	5.76
$\left(\frac{0.3974219}{s}\right)$	-0.1249 + j0.2483	0	0.2481	32.02

Note that the I-compensator

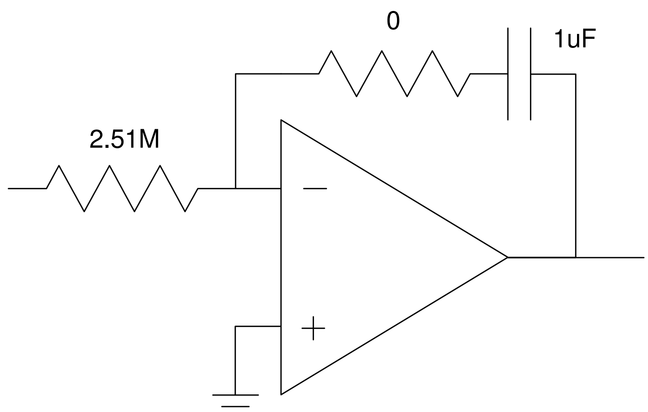
- Reduced the steady-state error to zero, but
- Resulted in a much slower system.

This is typical for I-type compensators.

Verifying this by taking the step response of the closed-loop system results in:



Step Response with I Compensation



Circuit to implement an I compensator

PI Compensation

If you add a proportional (P) term to K(s) you get a PI compensator.

$$K(s) = P + \frac{I}{s}$$

With a little algebra, this can be rewritten as

$$K(s) = \left(\frac{Ps+I}{s}\right) = k\left(\frac{s+a}{s}\right)$$

Essentially, with a PI compensator, you

- Add a pole at s=0 to make the system type-1
- Add a zero to cancel a pole.

From before, the pole at -0.3234 was the problem-child, slowing up the system. So, let's cancel that pole with the zero of the PI compensator

$$K(s) = k\left(\frac{s+0.3234}{s}\right)$$

Then

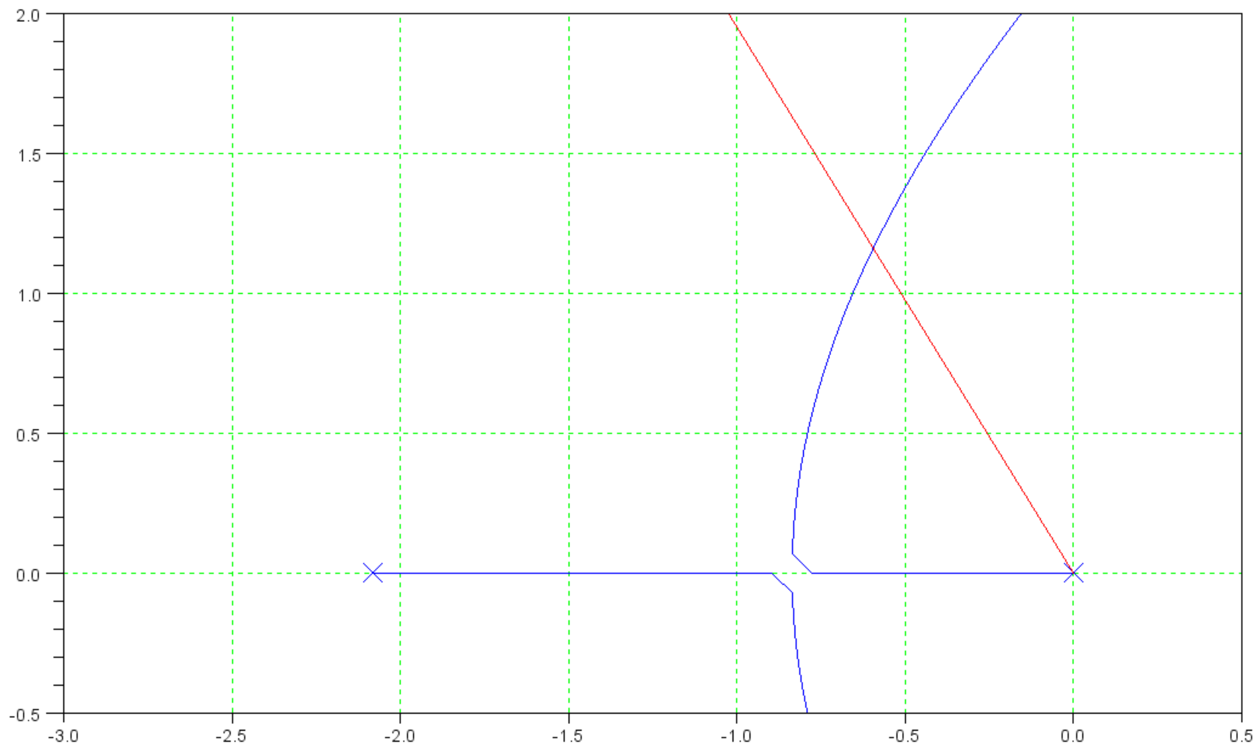
$$GK = \left(\frac{361.2378k}{(s+15.65)(s+10.1)(s+5.439)(s+2.081)s}\right)$$

note: What placing the zero at s = -0.3234 means is

$$\left(\frac{Ps+I}{s}\right) = P\left(\frac{s+I/P}{s}\right) = k\left(\frac{s+0.3234}{s}\right)$$

You're specifying the ration if I/P when you place the zero.

To determine k, sketch the root locus of GK



Root locus of $\left(\frac{361.2378k}{(s+15.65)(s+10.1)(s+5.439)(s+2.081)s}\right)$

The point on the root locus that intersects the 0.4559 damping line is

s = -0.5886 + j1.1772

At this point

$$\left(\frac{361.2378}{(s+15.65)(s+10.1)(s+5.439)(s+2.081)s}\right)_{s=-0.5886+j1.1772} = 0.1998\angle 180^0$$

k is then

$$k = \frac{1}{0.1998} = 5.0050$$

resulting in

$$K(s) = 5.0050\left(\frac{s+0.3234}{s}\right)$$

The error-constant, Kv, is then

$$K_v = \lim_{s \to 0} (s \cdot G \cdot K) = 1.0106$$

and the specifications for the PI compensated system are:

P, I, and PI Compensation				
K(s)	Closed-Loop Dominant Pole(s)	U at t=0 K(s) as s -> infinity	Kv	2% Settling Time seconds
5.5117	-0.6942 + j1.3884	5.51	0	5.76
$\left(\frac{0.3974219}{s}\right)$	-0.1249 + j0.2483	0	0.2481	32.02
$5.0050\left(\frac{s+0.3234}{s}\right)$	-0.5886 + j1.1772	5.00	1.0106	6.79

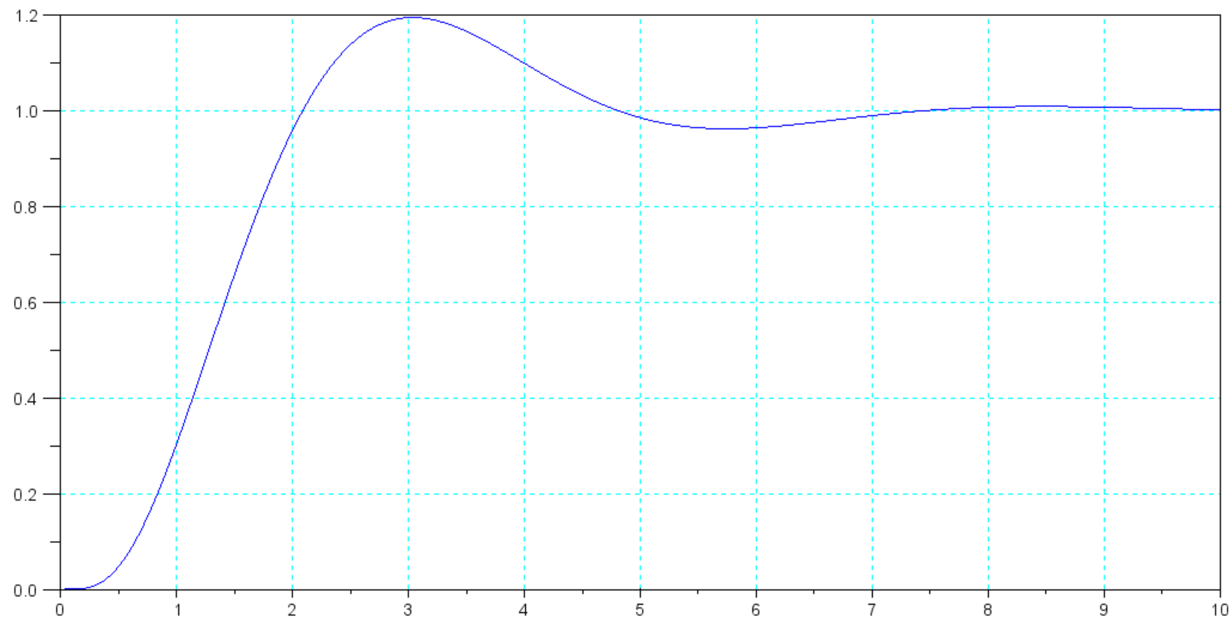
Note that the PI compensator results in a much faster system with better tracking.

What the integrator does is it searches to find the input, U, that forces the output to match the set point.

- With an I compensator, the initial guess for U is zero
- With a PI compensator, the initial guess for U is 5.00

This helps speed up the system. Moving the dominant pole left also helps improve the settling time.

The faster system can be verified in Matlab by plotting the step response:



Step Response of the PI Compensated system.

As expected

- The steady-state error is zero
- The overshoot is 20% (actually slightly below 20%), and
- The 2% settling time is around 7 seconds.

A circuit to implement K(s) is

$$K(s) = 5.0050\left(\frac{s+0.3234}{s}\right) = \left(\frac{R_1+\frac{1}{C_1 s}}{R_2}\right) = \left(\frac{R_1}{R_2}\right)\left(\frac{s+\frac{1}{R_1 C_1}}{s}\right)$$

Let

R1 = 1M

Then

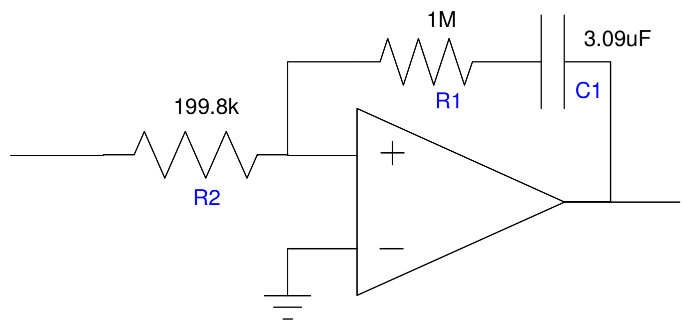
$$\left(\frac{R_1}{R_2}\right) = 5.005$$

$$R_2 = 199.8k$$

and

$$\frac{1}{R_1 C_1} = 0.3234$$

$$C_1 = 3.09uF$$



Circuit to implement a PI compensator. Just add R1 and I becomes PI

PID

If you add the derivative term to K(s) you get

$$K(s) = P + \frac{I}{s} + Ds$$

With a little algebra

$$K(s) = \left(\frac{Ds^2+Ps+I}{s}\right) = D\left(\frac{s^2+\frac{P}{D}s+\frac{I}{D}}{s}\right)$$

With a PID compensator

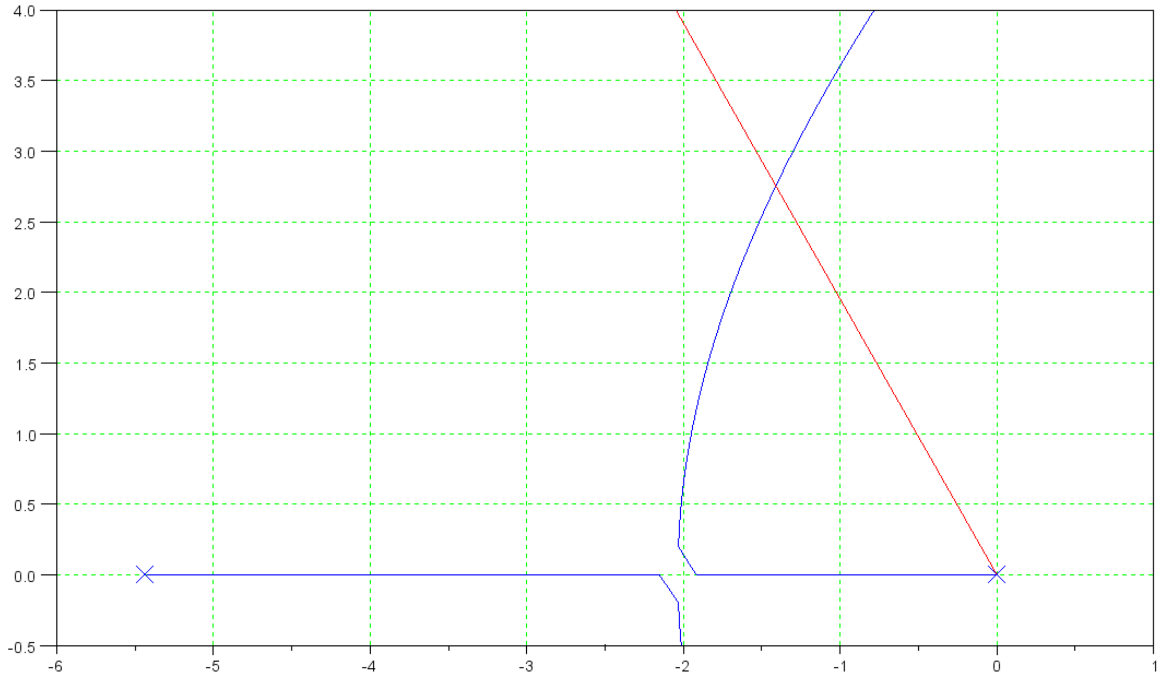
- You add a pole at s = 0, making the system type-1
- You also add two zeros, allowing you to cancel two poles

Since we're adding a pole at s=0, we don't need to keep the pole at s = -0.32 any more. The two slowest poles can then be canceled with the zeros, resulting in

$$K(s) = k\left(\frac{(s+0.3234)(s+2.081)}{s}\right)$$

$$GK = \left(\frac{361.2378k}{(s+15.65)(s+10.1)(s+5.439)s}\right)$$

To find k, find the point on the root-locus that intersects the damping line:



Root Locus Plot for $\left(\frac{361.2378k}{(s+15.65)(s+10.1)(s+5.439)s}\right)$

The point on the root locus is

s = -1.3947 + j2.7895

At this point

$$\left(\frac{361.2378}{(s+15.65)(s+10.1)(s+5.439)s}\right)_{s=-1.3947+j2.7895} = 0.1776\angle 180^0$$

meaning

$$k = \frac{1}{0.1776} = 5.6321$$

and

$$K(s) = 5.6321\left(\frac{(s+0.3234)(s+2.081)}{s}\right)$$

This results in the following parameters for the PID compensated system:

PID Controllers				
K(s)	Closed-Loop Dominant Pole(s)	U at t=0 K(s) as s -> infinity	Kv	$T_{2\%}$ seconds
5.5117	-0.6942 + j1.3884	5.51	0	5.76
$\left(\frac{0.3974219}{s}\right)$	-0.1249 + j0.2483	0	0.2481	32.02
$5.0050\left(\frac{s+0.3234}{s}\right)$	-0.5886 + j1.1772	5.00	1.0106	6.79
$5.6321\left(\frac{(s+0.3234)(s+2.081)}{s}\right)$	-1.3947 + j2.7895	infinity	2.3665	2.87

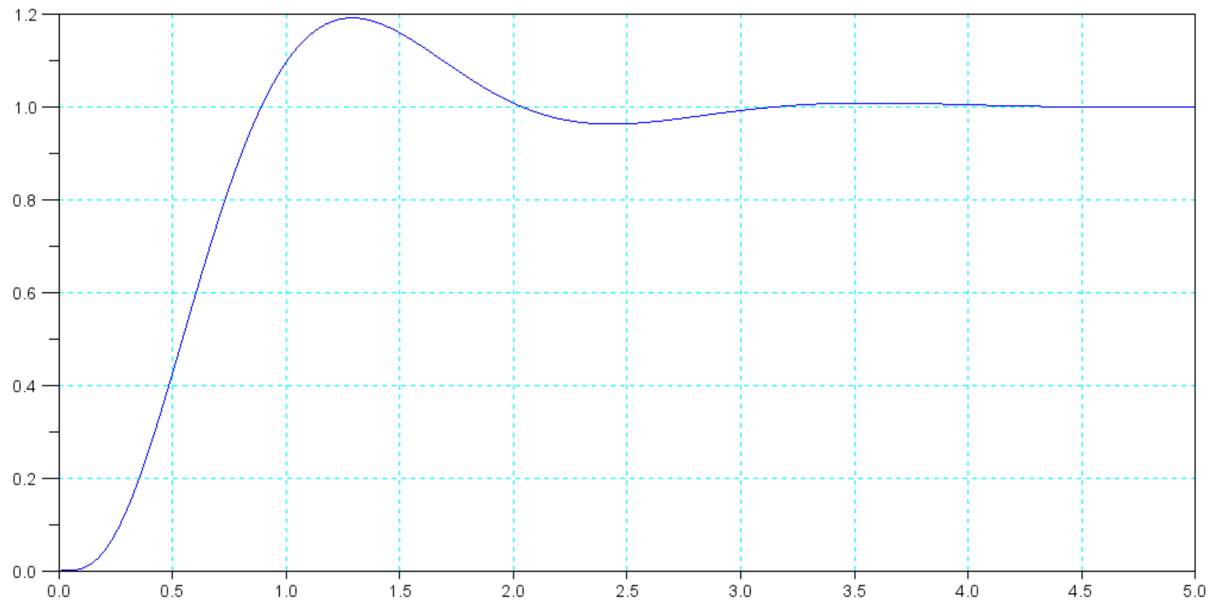
Note that by adding a second zero, the system is faster with better tracking. Differentiating the input causes problems, however

- If there is a step input, you apply the derivative of a step to the system (infinite input).
- If there is noise on the sensors, you differentiate this noise, amplifying it.

The step response verifies the compensator design:

- The steady-state error is zero (courtesy of the I compensator)
- The overshoot is 20%, and
- The 2% settling time is about 2.87 seconds.

On paper, a PID compensator works better than a PI compensator. In practice, differentiation causes lots of problems. Usually just PI compensator is used.



Step Response with a PID Compensator

If you *did* want to implement a PID compensator, a circuit to do this is as follows. Note that

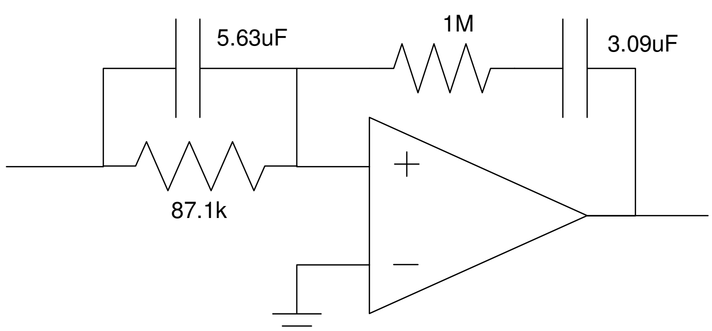
$$K(s) = 5.6321\left(\frac{(s+0.3234)(s+2.081)}{s}\right) = \left(\frac{(R_1C_1s+1)(R_2C_2s+1)}{R_2C_1s}\right)$$

Let R1 = 1M

$$\frac{1}{R_1C_1} = 0.3234 \qquad \text{C1 = 3.09uF}$$

$$5.6321 = \frac{R_1C_1R_2C_2}{R_2C_1} = R_1C_2 \quad \text{C2 = 5.63uF}$$

$$\frac{1}{R_2C_2} = 2.081 \qquad \text{R2 = 87.1k}$$



PID Compensator Circuit