Analyzing Feedback Control with the RRC Circuit

Cole Sheyka

February 2023

1 Abstract

The goal of this experiment was to observe the real time step response of an RRC circuit with feedback control. This report analyzes the effects of implementing a proportional controller into a feedback loop, as well as a PI controller. After building both circuits in the lab, various methods and types of hardware were used to view and record the output of the system in feedback control.

2 Introduction

The RRC circuit is known for its filtering properties. It is composed of two resistors, one capacitor, and it can be wired as shown in Figure 1.

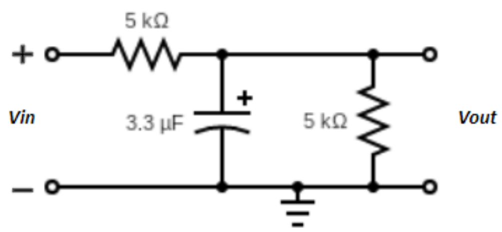

Figure 1: RRC circuit schematic

The resistor and capacitor values shown in Figure 1 are the nominal values reported by the manufacturer. When a signal passes through V_{in}, the frequency response can be visualized between the nodes at V_{out} and ground. By breaking up the circuit into its idealized elements, it is possible to derive the differential equation that governs the circuit, and its respective transfer function known as the plant G. Sending a signal through V_{in} would result in an open loop frequency response of the RRC circuit.

3 Theory

3.1 Transfer function of the RRC circuit

The general transfer function of the RRC circuit can be written as

$$G = \frac{1}{R_1Cs + (\frac{R_1}{R_2} + 1)}.$$

Given that $R_1 = R_2$, we have

$$G = \frac{1}{RCs + 2}.$$

The time constant, $\tau = R_fC$. Therefore,

$$G = \frac{1}{\tau s + 2} \tag{1}$$

Eqn. 1 represents the transfer function of the RRC circuit, where $R = 5k\Omega$ and $C = 3.3\mu F$.

3.2 Final Value Theorem

The final value theorem (FVT) is useful to determine where the system's response will settle as $t \to \infty$. Moreover, it is a method to calculate the DC gain of a transfer function. Since there are no roots on the imaginary axis in the open loop transfer function, in the right hand plane, or more than one pole at the origin, the FVT applies to the transfer function G. The FVT can be written as

$$z_{ss} = z(t)_{t\to\infty} = s\bar{z}(s)|_{s=0} \tag{2}$$

such that

$$z(t) \xrightarrow{\mathcal{L}} \bar{z}(s)$$

where z_{ss} is the steady-state value of a frequency response. This is commonly written as $\hat{k}$, which is also known as the DC gain of a system. In the case of the open loop RRC circuit, the DC gain is $\hat{k} = \frac{1}{2}$ by the FVT.This method of identifying the steady-state characteristics of transfer functions is used throughout this report.

3.3 Feedback Control

Feedback control is a robust method to model uncertainty. In a feedback control system, the output of the system is measured and compared to a desired reference signal, and the difference between them is used to adjust the system's output in order to achieve the desired behavior. With feedback control,

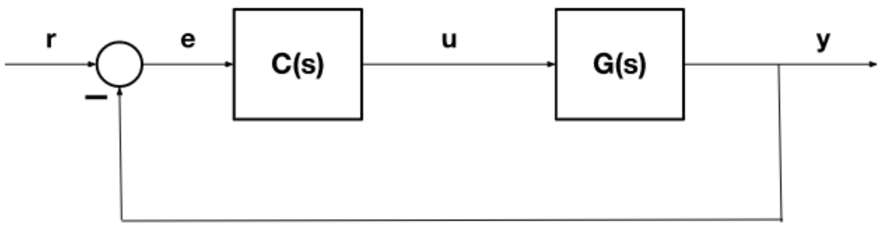

Figure 2: Basic feedback control diagram

it becomes possible to get a desired output from the system with minimal steady-state error. $G(s)$ can be any system and $C(s)$ is the controller. In the case of this report, $G(s)$ will always be the transfer function of the open loop RRC circuit derived in Eqn. 1.

There are four main types of controllers which affect the output. That is: proportional (P), proportional-derivative (PD), proportional-integral (PI), and proportional-integral-derivative (PID). Each type of controller is used to mitigate steady-state error and improve performance, although some controllers are better than others at doing that. For example, with a proportional controller, the steady state error with a step response can never be zero since $e = r - y$ is multiplied by a gain during each pass of the loop, essentially correcting itself forever. However, the proportional controller is not completely useless and it can still be used as a tool to achieve a desired output from a system.

Solving for the general closed loop transfer function, we have

$$H = \frac{Y}{R} = \frac{GC}{GC + 1}. \tag{3}$$

Using Eqn. 3, the closed loop transfer function, H, can be calculated by substituting in the values for G and C. The closed loop transfer functions for a P-controller and a PI-controller are

$$H_P = \frac{k_P}{\tau s + (k_P + 2)} \tag{4}$$

and

$$H_{PI} = \frac{s k_P + k_I}{\tau s^2 + (k_P + 2)s + k_I} \tag{5}$$

respectively, where

$$C_P = k_P$$

and

$$C_{PI} = k_P + \frac{k_I}{s}.$$

The constant k_I in the above equation can be defined as $\frac{1}{RC_f} = \frac{1}{\tau}$. Respectively, k_P and k_I are the gains to the proportional and integral controllers.

3.4 Open Loop Frequency Response

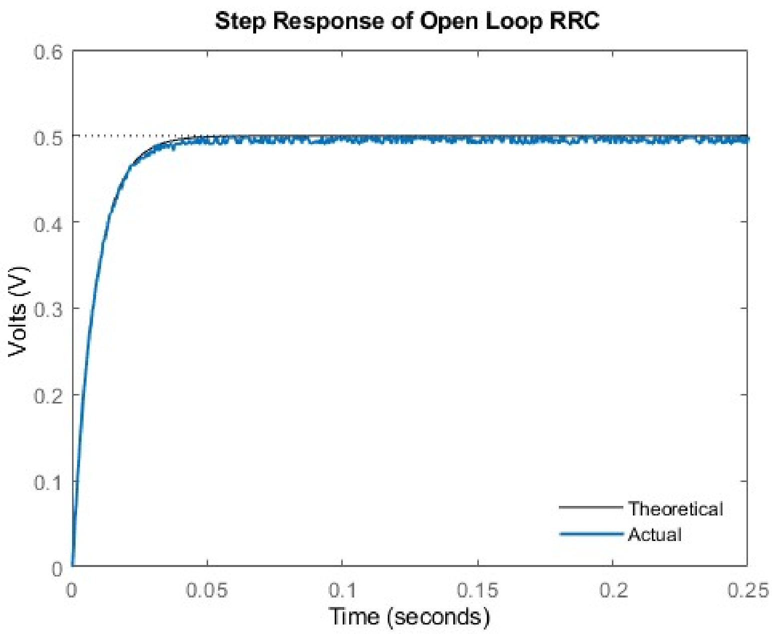

Figure 3: Open loop response of the RRC circuit

Figure 3. shows the theoretical and actual step response of the RRC circuit with a square wave input signal of 500mHz, 50% duty cycle, 1 V_{pp}, and a 500mV offset, which simulates a periodic step response. Both of these plots converge to $V = 0.5V$, which aligns with the FVT. Without increasing V_{in}, implementing feedback control can scale the DC gain of a frequency response. Building the controller seen in Figure 2. is possible by connecting various op-amps combined with various external components like resistors and capacitors.

4 Hardware

The ITLL at CU Boulder has various resources required to build and observe this circuit and its behavior. In order to transmit signals, a function generator was used to generate the same signal mentioned earlier to obtain the open-loop step response of the RRC circuit. However, for the PI-controller, a smaller frequency of 500mHz was used to make comparing to the numerical solution easier. Multiple single output DC power sources were used to power op-amps and test the individual stages of the circuit to make sure that the results being recorded accurately reflect theory. A digital multi meter

(DMM) was frequently used to check continuity on connections, voltages at different points in the circuit, and real-resistance values for numerical solutions. Lastly, a digital oscilloscope was used to observe and record the output behavior of the system under feedback control and open loop conditions.

The control diagram seen in Figure 2. can be build out of a selection of signal processing circuits. For this to work, three LM358-N op-amp chips were used to amplify, sum, and integrate incoming signals. This section will discuss the different op-amp circuits used to modify signals.

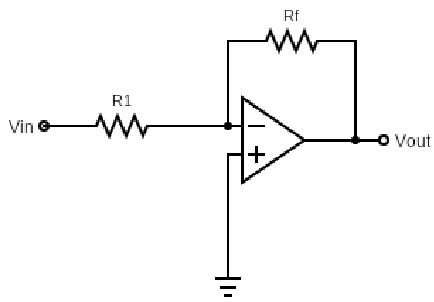

Figure 4: Amplifier (Inverting) circuit

The inverting amplifier is useful to multiply the incoming signal by a gain. The governing equation for this amp is

$$V_{out} = -\frac{R_f}{R_1} V_{in}. \tag{6}$$

If $R_f = R_1$, then

$$V_{out} = V_{in}. \tag{7}$$

Figure 4. shows the correct wiring of the amplifier to its inverting and non-inverting inputs. For the need of creating larger gains, notice that R_f can be swapped for a higher resistance value.

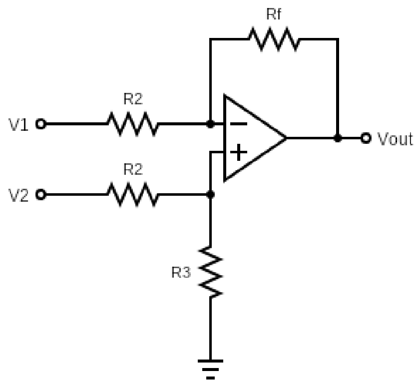

Figure 5: Differential circuit

The equation that governs the differential amp is

$$V_{out} = (\frac{R_f + R_1}{R_1})(\frac{R_3}{R_3 + R_2})V_2 - \frac{R_f}{R_1}V_1.$$

If $R_f = R_1 = R_2 = R_3$, the above equation can be simplified as

$$V_{out} = V_2 - V_1 \tag{8}$$

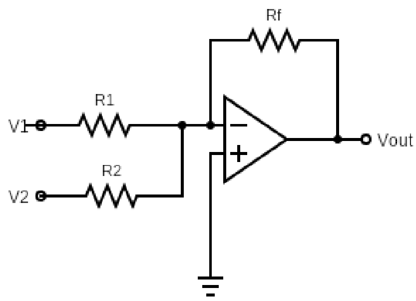

Figure 6: Summing circuit

The summing circuit is integrated into the circuit as additional control is added to $C(s)$. For example, with a PID controller, the outputs of the respective integrator, proportional gain, and differentiator circuits are summed with this op amp configuration. To make a PID controller out of op amps, a total of four LM358s would have to be used. Nonetheless, the general form of the summing circuit is

$$V_{out} = -R_f(\frac{V_1}{R_1} + \frac{V_2}{R_2}).$$

If $R_1 = R_2 = R_f$, then

$$V_{out} = -(V_1 + V_2). \tag{9}$$

The integrator op amp circuit is used later for creating an integral controller. The governing equation of the integrator is

$$-\frac{1}{R_1 C_f}\int V_1 dt = -\frac{1}{\tau s}. \tag{10}$$

4.1 Feedback with a Proportional Controller

Figure 8. shows a schematic with three op amps and an RRC circuit. Although there are three amps, this schematic represents two transfer functions. The first op amp on the left hand side is part of a differential circuit. It takes the difference of voltage out of the RRC circuit and the input signal. The

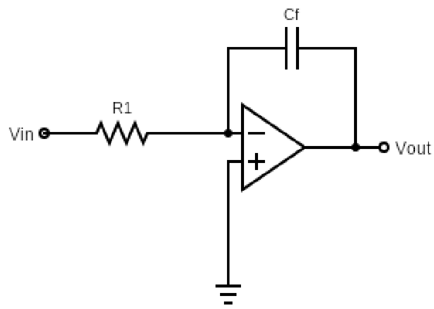

Figure 7: Integrating circuit

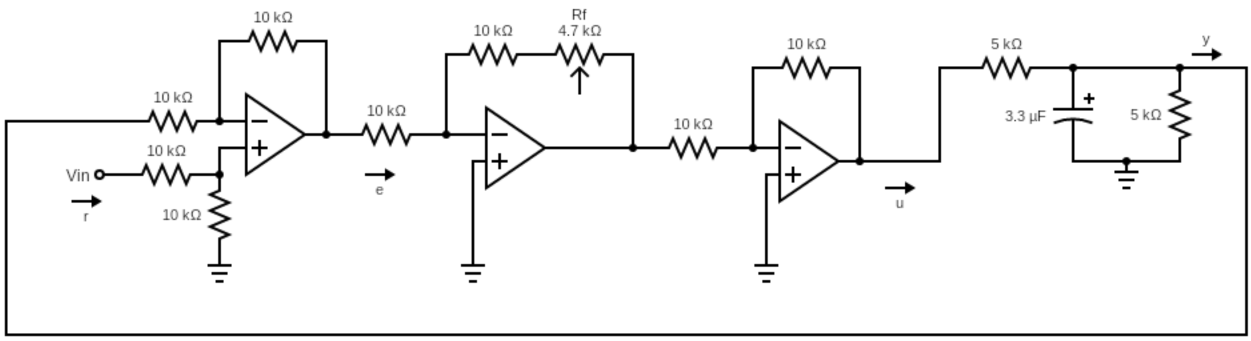

Figure 8: Circuit diagram of feedback control with the RRC circuit and P controller

output can be expressed as

$$e(t) = r(t) - y(t)$$

where r is the input signal and y is the voltage out of the RRC circuit. The error function, $e(t)$, is the deviation from the desired output to the actual output of the system.

This deviation is then sent into the P controller circuit, which acts to amplify the error signal and adjust to the desired output. The P controller in Figure 8. is the two other inverted amplifiers to the right of the differential. The first increases the amplitude of the error signal by maually adjusting the variable resistor's resistance (actually swapping the variable resistor symbol with a resistor of known value), and the second multiplies by a gain of -1 to have the output of $C(s)$, or $u(t)$, be positive. The control signal, or the manipulated variable $u(t)$ then passes through the RRC circuit. The output of the RRC circuit then passes through the differential again and the loop continues.

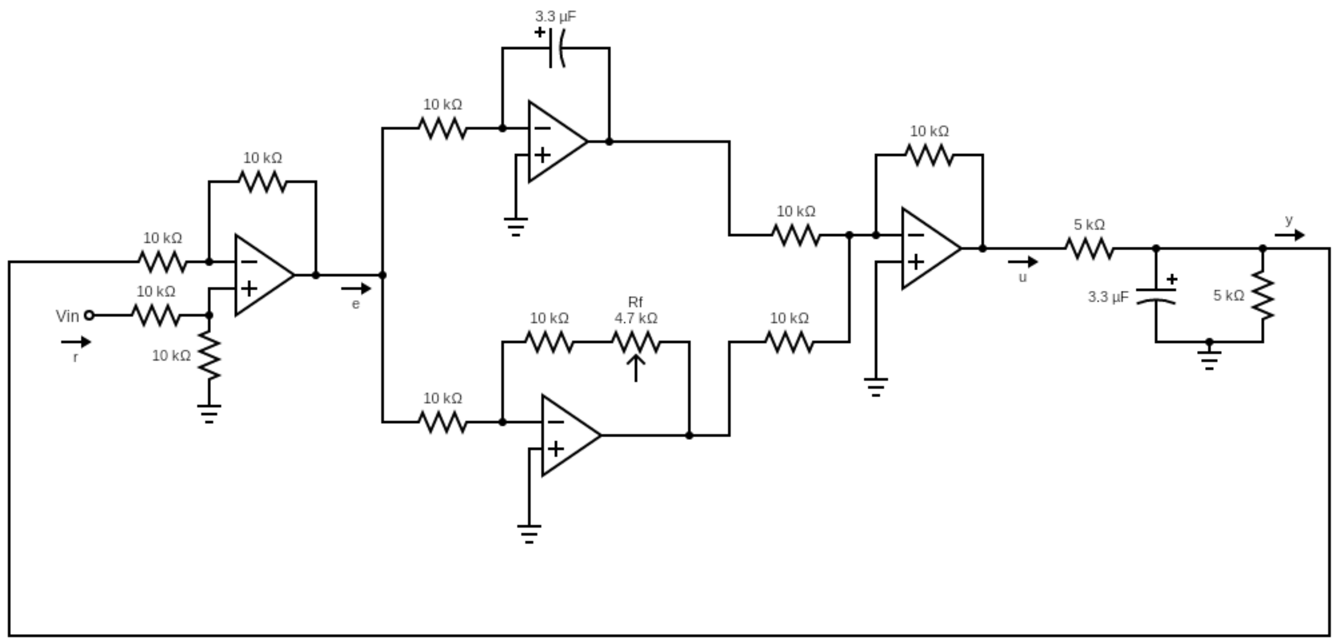

Figure 9: Circuit diagram of feedback control with the RRC circuit and PI controller

4.2 Feedback with a PI Controller

The PI controller is a slightly more complicated circuit with an additional op amp being used. The differential amp is still there, and it does the same thing as it did in the P controller. However, now there is a node on the output of the differential which feeds the same voltage from the output of the differential amp into an integrator and proportional gain. Since both of the outputs from these op amp circuits are negative voltages, the sum of these two signals in a summing op amp circuit will result in the control signal to be positive. These three circuits together form the PI controller.

The P control does a good job at decreasing the rise time of the frequency response, but by nature the system will always have steady state error. The integrator is able to completely remove the steady state error by integrating the error signal with time. It also introduces another pole and zero into the closed loop transfer function, which can be seen in Eqn. 5. Later on, it can be seen what this does for the DC gain of the PI controller More sophisticated controllers can be made out of this combination because increasing the system type allows for easier design given modern technology like the $rlocus()$ function in MatLab. Hence, different combinations of K_I and K_P can be swapped to achieve a desired outcome.

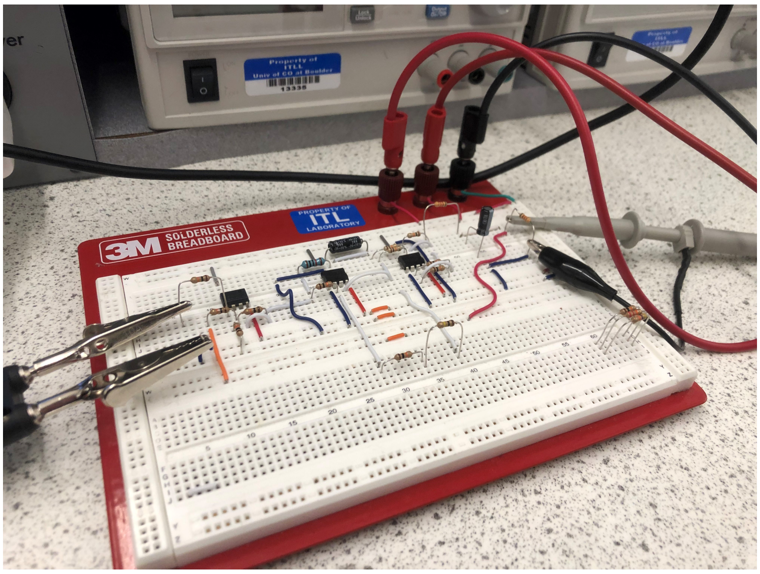

Figure 10: Experimental srt up of both the P and PI controllers on the same breadboard

4.3 Experimental Setup

The two red banana wires at the northeast corner of the breadboard are $\pm V_{cc}$ for the op amps. The black banana wire is ground. All components on the board, as well as measuring and signal generating equipment share the same ground. The two alligator clips on left serve as the V_{in} and ground from the function generator. The grey and black clips on the right feed to the oscilloscope. The order of the op amps in Figure 10. are the same as the ones in Figures 8. and 9., but each chip has two embedded op amps so either side was used to have both the P control and PI control circuits on the same board.

A total of 5 resistors of dissimilar value were set aside to adjust the proportional gain for the P controller only. Although, swapping out the R_1 value in the integrator circuit for resistances of higher values would result in the integrator responding more aggressively to error. Increasing the R_1 value would likely result in overshoot before settling in steady state conditions.

5 Experimental Results

A modified square wave was sent into both the P and PI controllers seen in Figure 10. The output of the system was saved from the oscilloscope then transferred into MatLab to compare theoretical with actual results. A basic Simulink model was also built to simulate the P circuit with a specific proportional gain, but the results from the simulation convey the same information as the figures seen below.

The K_P values used in developing Figure 11. are the true values of the proportional gain. These values were obtained by measuring each resistor set aside for testing the controller then using Eqn. 6.

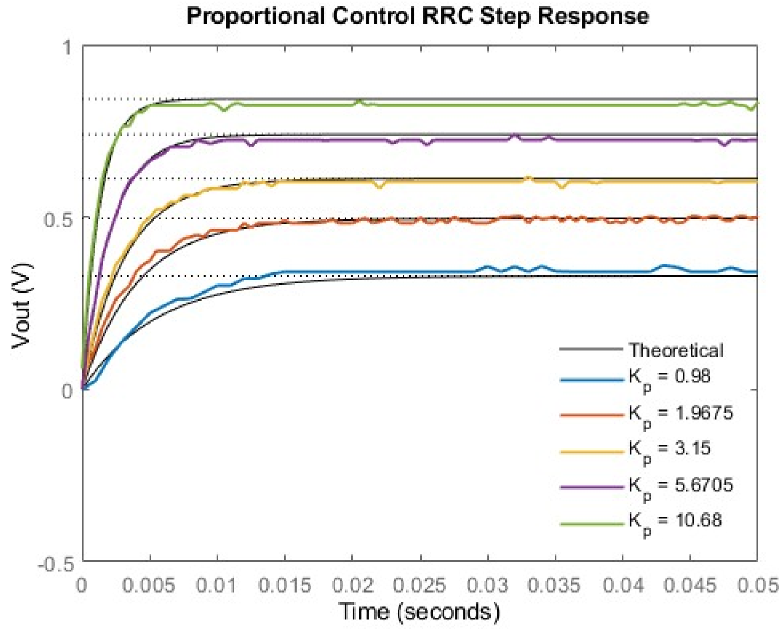

Figure 11: Step response of the RRC circuit with a proportional controller in feedback

The theoretical step responses were calculated by using Eqn. 4 and building an array of transfer functions with the various k_P values in MatLab. Interestingly, both the theoretical and experimental results agree with the FVT. The equation

$$H_{SS} = \frac{k_P}{k_P + 2}$$

accurately predicts where all five of the step responses in Figure 11. settle at.

Figure 12. shows the step response of the RRC circuit with the same square wave as before, except the frequency of the input signal was changed to 500mHz to capture most of the settling time. The FVT also applies and the DC gain can be written as

$$H_{ss} = \frac{k_I}{k_I} = 1.$$

. For all other signals not seen in Figure 12, the final value was one. Although the settling time with the PI controller was considerably longer than the settling time with just the proportional controller, the steady state error in the PI feedback circuit is zero such that a long enough period is sent with the desired signal.

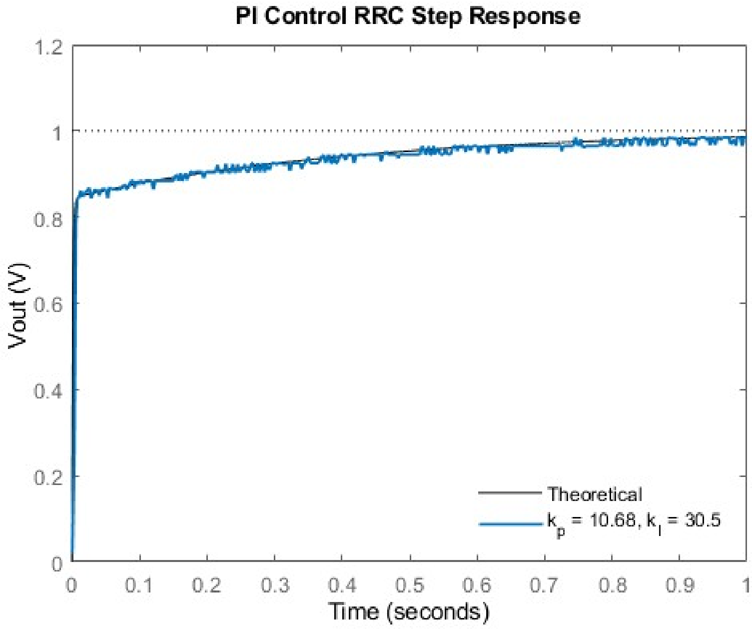

Figure 12: Step response of the RRC circuit with a PI controller in feedback

6 Conclusion

The experimental results matched closely with theory which is reassuring. Both of the controllers accurately adjusted for error, although the PI control was much more accurate. This is due to the integrator compensating for error in junction with the proportional gain. The steady state error in the P control feedback loop is due to the proportional gain constantly correcting the incoming error signal. Because of this, the P controller has benefits like short rise times but it is not as precise as the PI controller. Both of these controllers have different applications, however, since the steady state value for the P control is not always 1.

7 Appendix

```
clc, clear

P = [];
PI = [];
for i = 1:5
    P = [P readmatrix(sprintf("KP%i.csv",i))];
    PI = [PI readmatrix(sprintf("kpi%i.csv",i))];
end

R = 5002; % ohms
C = 3.3e-6; % uF
tau = R*C;

R_kp = 9800; % ohms
kp = (R_kp + [0 9875 21700 46905 97000])./10000; % ohms

% open loop tf
G = tf(1,[tau,2]);

ki = 1/(9941*3.3e-6);
% sys = tf([kp(5),ki],[]);

% closed loop tfs
Hp = [];
Hpi = [];
for i = 1:5
    Hp = [Hp tf(kp(i),[tau,(2+kp(i))])];
    Hpi = [Hpi tf([kp(i),ki],[tau,(kp(i)+2),ki])];
end

figure;
bode(G)
```

```
title("Open Loop Frequecy Response RRC")

figure;
hold on
xlim([0 0.1])
hp1 = stepplot(Hp(1),'k');
hp2 = stepplot(Hp(2),'k');
hp3 = stepplot(Hp(3),'k');
hp4 = stepplot(Hp(4),'k');
hp5 = stepplot(Hp(5),'k');
plot(P(:,1) - .009-.15, P(:,2),'linewidth',1.4) % Kp1
plot(P(:,3) + .032-.15, P(:,4),'linewidth',1.4) % Kp2
plot(P(:,5) + .0355-.15, P(:,6),'linewidth',1.4) % Kp3
plot(P(:,7) - .014-.15, P(:,8),'linewidth',1.4) % Kp4
plot(P(:,9) + .042-.15, P(:,10),'linewidth',1.4) % Kp5
title("Proportional Control RRC Step Response")
ylim([-.5 1])
ylabel("Vout (V)")
legend("","","","","Theoretical","K_p = 0.98","K_p = 1.9675","K_p =
   3.15", ...
    "K_p = 5.6705","K_p = 10.68",'location','southeast')
legend boxoff
hold off

figure;
hold on
hpi5 = stepplot(Hpi(5),'k');
plot(PI(:,9)+.19, PI(:,10),'linewidth',1.4) % Kp5
ylim([0 1.2])
xlim([0 1])
title("PI Control RRC Step Response")
ylabel("Vout (V)")
legend("Theoretical","k_p = 10.68, k_I = 30.5","location","southeast")
```

```
legend boxoff
hold off

RC = readmatrix("RRC Step Response.csv");
figure;
hold on
g = stepplot(G,'k');
plot(RC(:,1)+.402,RC(:,2),'linewidth',1.4)
title("Step Response of Open Loop RRC")
ylabel("Volts (V)")
legend("Theoretical", "Actual",'location','southeast')
legend boxoff
xlim([0 .25])
ylim([0 .6])
hold off
```