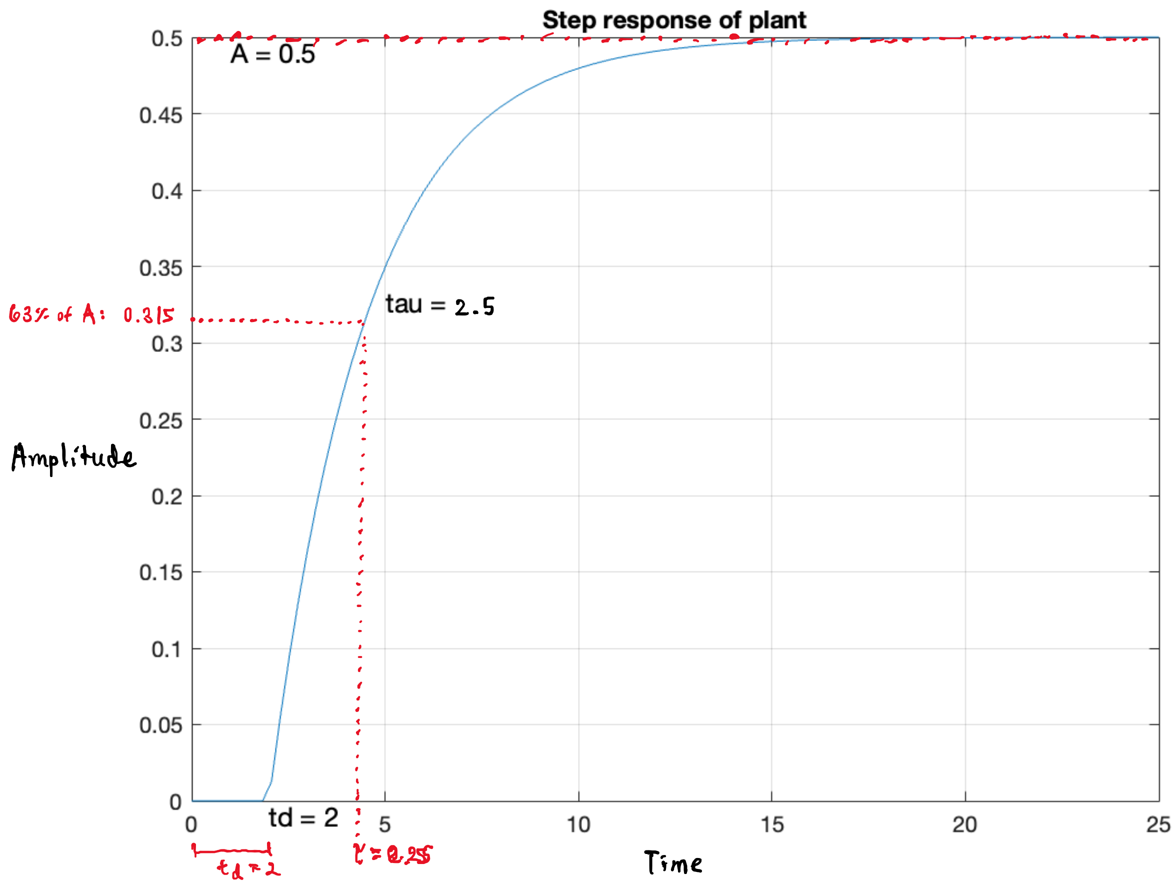
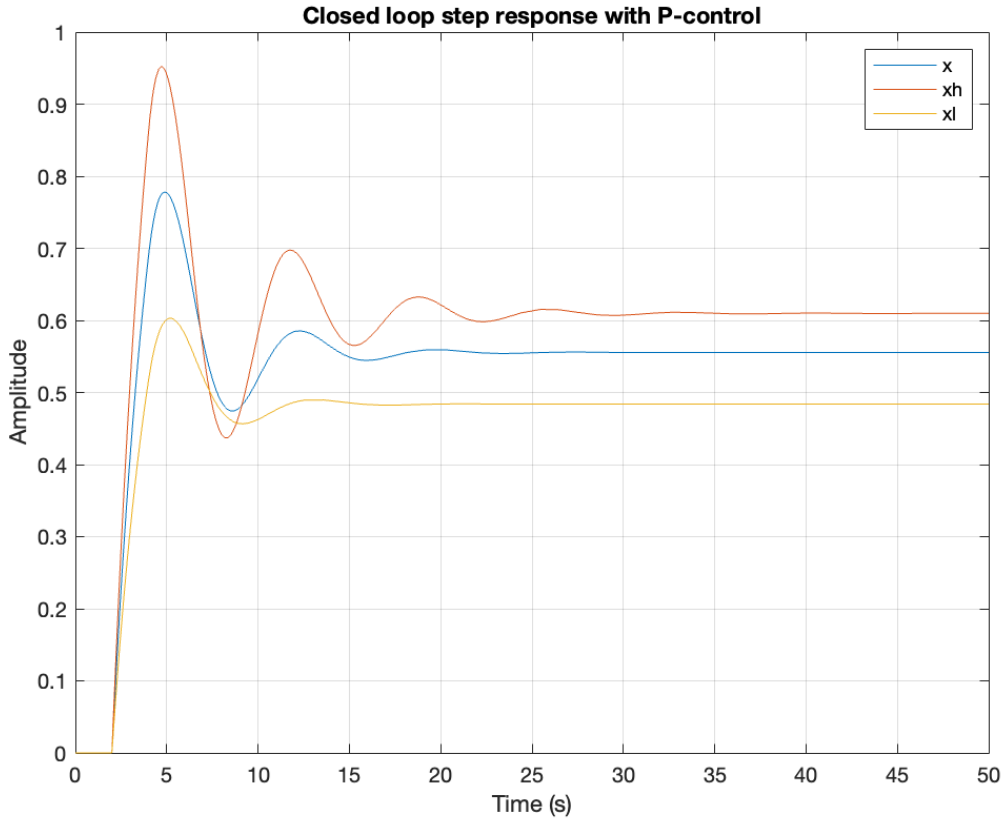
Figure 1




```
% G = 1/(5*s+2) * e^(-2*s)

A = 1;
tau = 5;
td = 2;
Tend = 25;

Gp = tf(A,[tau 2],'InputDelay', td);

[y,t] = step(Gp, Tend);
figure(1), plot(t,y),grid,title('Step response of plant');

% Add labels for A, td, and tau
text(1, 1/2, ['A = ', num2str(1/2)], 'FontSize', 12, 'VerticalAlignment', 'top');
text(td, 0, ['td = ', num2str(td)], 'FontSize', 12, 'VerticalAlignment', 'top');
text(tau, 1/2*(63/100), ['tau = ', num2str(tau)], 'FontSize', 12,
'VerticalAlignment', 'bottom');
```

Figure 2



e_{ss} @ x = r - y_{ss} = 1 - 0.5554 = **0.4446**
e_{ss} @ xh = 1 - 0.6067 = **0.3933**
e_{ss} @ xl = 1 - 0.4838 = **0.5162**

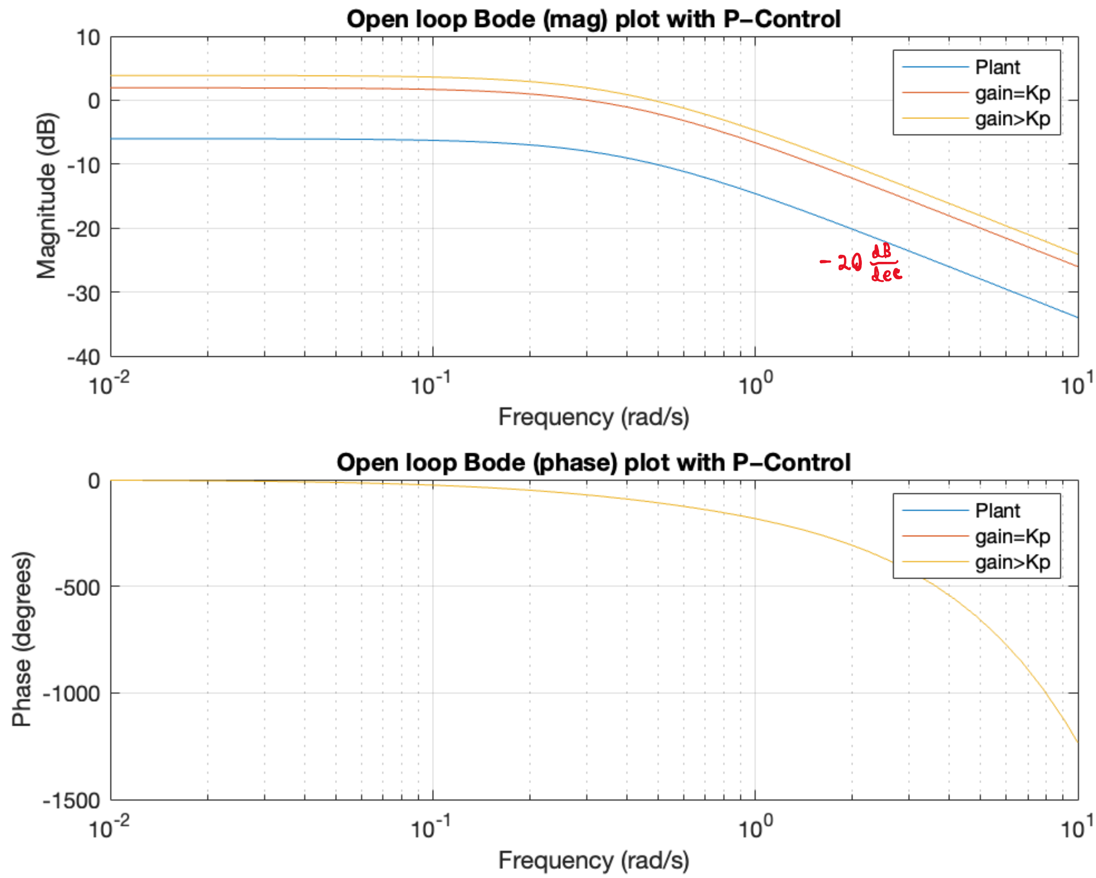
As gain increases, the steady state error decreases. This aligns with our understanding because a larger gain when there is a closed loop means that the error should decrease.

However, increasing the gain (Kp), should also lead to a larger Tr and Ts, as the signal is being amplified.

```
x = 1/((A/tau)*td);
h = 1.25;
l = 0.75;

Kp = x;
Gol = series(Kp, Gp);
Gcl = feedback(Gol, 1);
[y1, t1] = step(Gcl, Tend);

Kp = x*h;
```

```
Gol = series(Kp, Gp);
Gcl = feedback(Gol, 1);
[y2, t2] = step(Gcl, Tend);

Kp = x*l;
Gol = series(Kp, Gp);
Gcl = feedback(Gol, 1);
[y3, t3] = step(Gcl, Tend);

figure(2),plot(t1,y1,t2,y2,t3,y3), grid;
title('Closed loop step response with P-control');
xlabel('Time (s)');
ylabel('Amplitude');
legend('x', 'xh', 'xl');
```

Figure 3



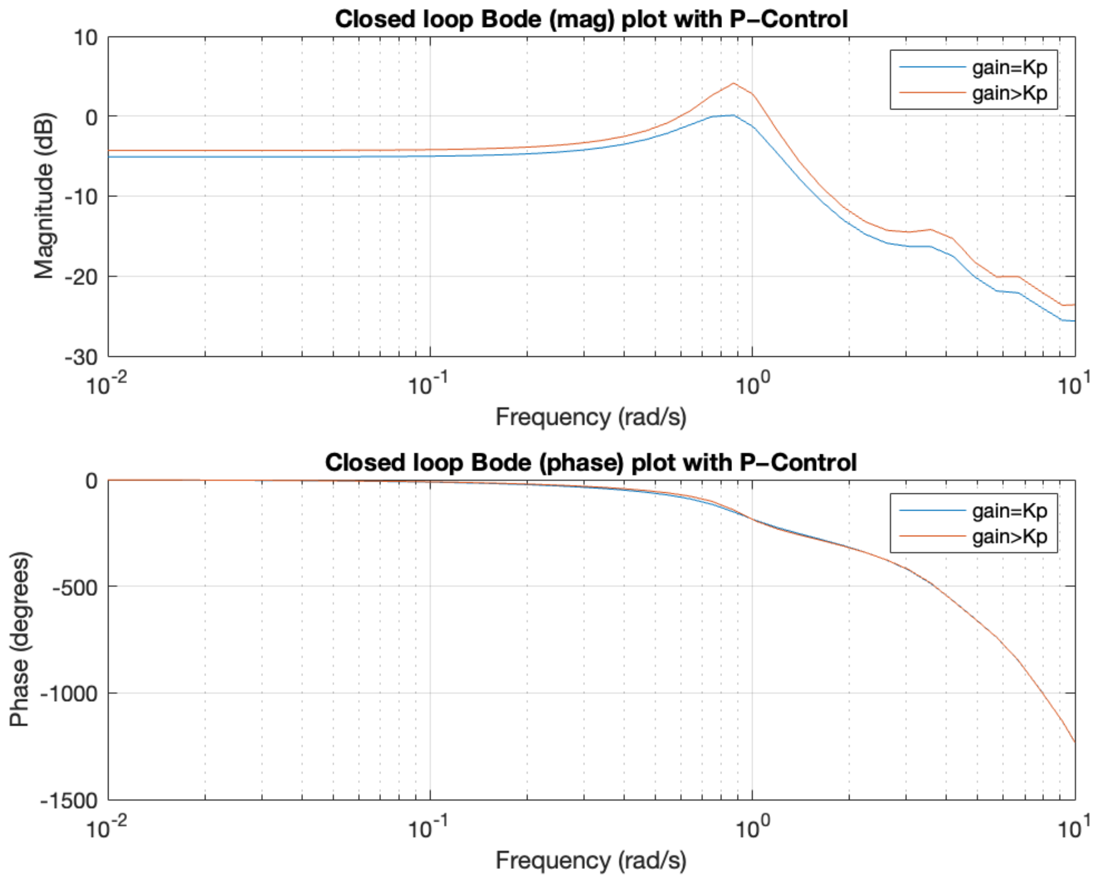
Kp, or the gain of the transfer function, will have an obvious impact on the magnitude-frequency response bode plot. The open loop transfer function is G(s) = Kp*Gp, so as you increase Kp, it will increase the magnitude of the transfer function. This is seen above, as the larger the gain (Kp), the larger the magnitude of the response.

However, since increasing the gain does not introduce any new phase shift, the phase-frequency response bode plot will not change with changing gain. This is seen above, as all three functions have the same phase response.

```
[mag,php,w] = bode(Gp);
dbp=20*log10(mag);

Kp = x;
[mag,phol1] = bode(series(Kp,Gp),w);
dbol1=20*log10(mag);

Kp = x*h;
[mag,phol2] = bode(series(Kp,Gp),w);
dbol2 = 20*log10(mag);
```

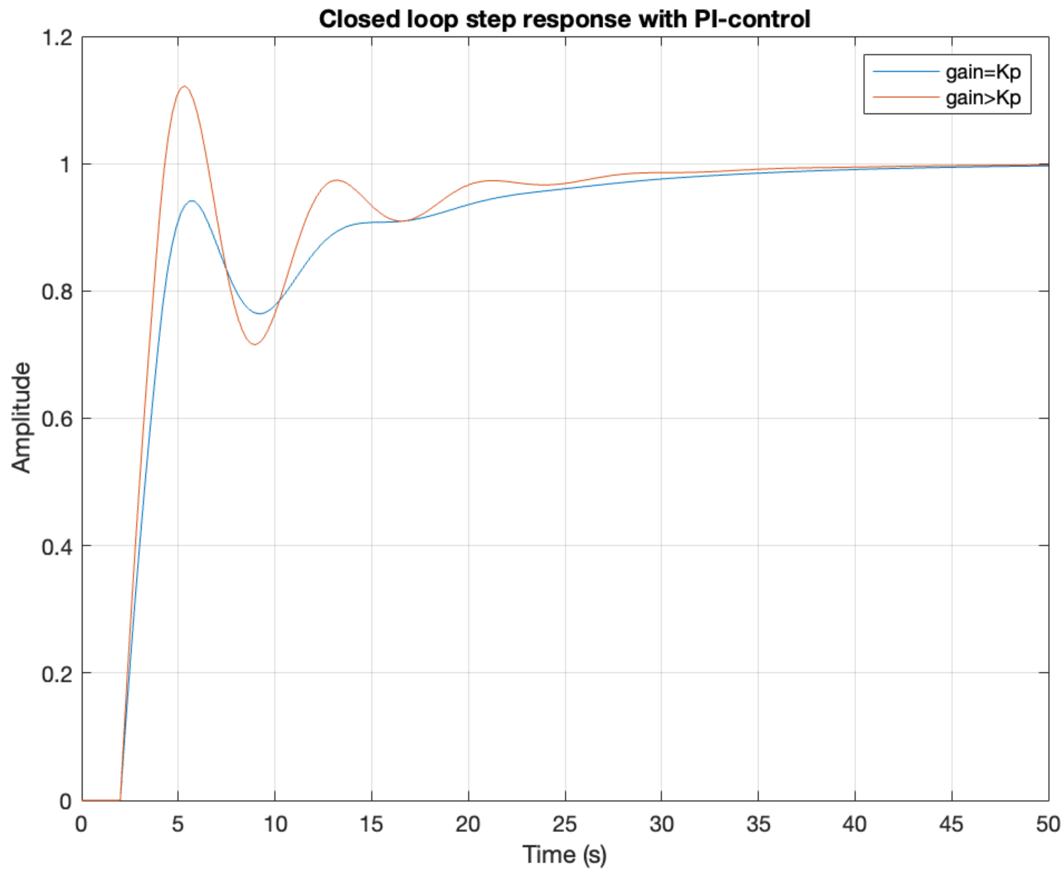
```
figure(3);
subplot(211), semilogx(w, dbp(:), w, dbol1(:), w, dbol2(:)), grid;
title('Open loop Bode (mag) plot with P-Control');
legend('Plant','gain=Kp','gain>Kp');
xlabel('Frequency (rad/s)');
ylabel('Magnitude (dB)');
subplot(212), semilogx(w, php(:), w, phol1(:), w, phol2(:)), grid;
title('Open loop Bode (phase) plot with P-Control');
legend('Plant','gain=Kp','gain>Kp');
xlabel('Frequency (rad/s)');
ylabel('Phase (degrees)');
```

Figure 4




```
Kp = x;
[mag,phcl1] = bode(feedback(series(Kp,Gp),1),w);
dbcl1=20*log10(mag);

Kp = x*h;
[mag,phcl2] = bode(feedback(series(Kp,Gp),1),w);
dbcl2=20*log10(mag);

figure(4);
subplot(211), semilogx(w, dbcl1(:), w, dbcl2(:)), grid;
title('Closed loop Bode (mag) plot with P-Control');
legend('gain=Kp','gain>Kp');
xlabel('Frequency (rad/s)');
ylabel('Magnitude (dB)');
subplot(212), semilogx(w, phcl1(:), w, phcl2(:)), grid;
title('Closed loop Bode (phase) plot with P-Control');
legend('gain=Kp','gain>Kp');
xlabel('Frequency (rad/s)');
ylabel('Phase (degrees)');
```

Figure 5



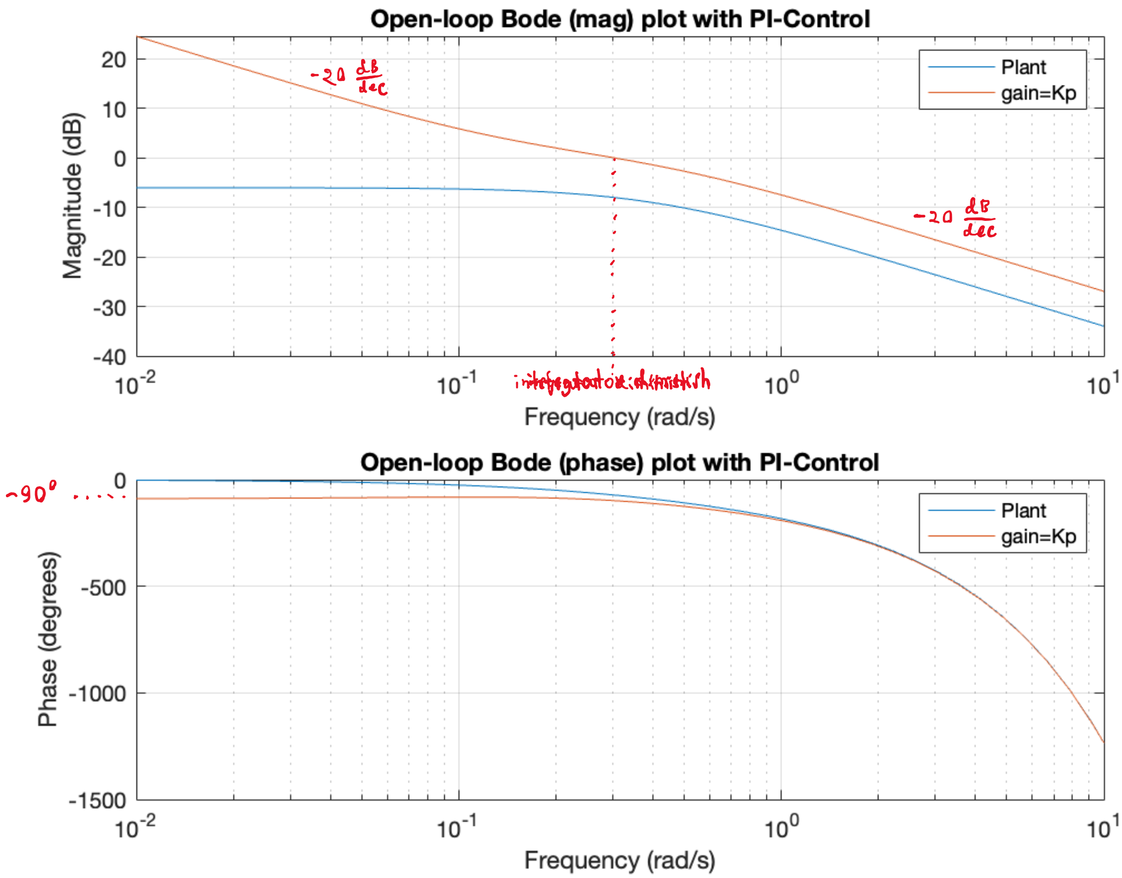
The effect of the integral on the e_{ss} is very good. Now, instead of there being an obviously present steady state error, it is essentially 0 no matter the gain, as shown in the figure above. It reaches r, with the expected and actual gain being nearly equivalent. This means it is better at handling noise, which is what the integrator is for. This is not the case in Figure 2, where e_{ss} will not reach zero unless the gain is increased a lot more to compensate through feedback.

Moving to Kp, it no longer affects the steady state error, increasing Kp, which is the gain, increases the rise time and overshoot. It does not appear to have an effect on the time it takes to steady state, as they are about the same at Kp and 1.25*Kp.

```
alpha = 0.9/((A/tau)*td);
beta = td/0.3;

Kp = alpha;
Ti = beta;
C = Kp*tf([Ti 1],[Ti 0]);
Gol = series(C,Gp);
Gcl = feedback(Gol,1);
[y4,t4] = step(Gcl, Tend);
```

```
Kp = alpha*h;
Ti = beta;
C = Kp*tf([Ti 1],[Ti 0]);
Gol = series(C,Gp);
Gcl = feedback(Gol,1);
[y5,t5] = step(Gcl, Tend);

figure(5),plot(t4,y4,t5,y5),grid;
title('Closed loop step response with PI-control');
legend('gain=Kp','gain>Kp');
xlabel('Time (s)');
ylabel('Amplitude');
```

Figure 6



When the integral term is introduced, the magnitude response will look different than without it because the integral term introduces a gain due to its phase shift of - 90 degrees. At low frequencies, this is quite obvious, as it is -20 db/dec, but as the frequency increases, the impact of the integral term will diminish. Integrators act similar to a low-pass filter, because its low-frequency response is bigger than its high-frequency response, and this is seen in the figure above.

As briefly discussed, the integral adds a phase shift of -90 degrees, called a phase lag at low frequencies and which approaches 0 degrees at high frequencies. This explains why the phase of the PI control plot starts at a different value as the P control, but then ends up looking similar at higher frequency.

```
[mag,php,w] = bode(Gp);
dbp=20*log10(mag);

Kp = alpha;
Ti = beta;
C = Kp*tf([Ti 1],[Ti 0]);
[mag, phol1] = bode(series(C, Gp), w);
```

```
dbol1 = 20 * log10(mag);

Kp = alpha * h;
Ti = beta;
C = Kp*tf([Ti 1],[Ti 0]);
[mag, phol2] = bode(series(C, Gp), w);
dbol2 = 20 * log10(mag);

figure(6);
subplot(211), semilogx(w, dbp(:), w, dbol1(:)), grid;
title('Open-loop Bode (mag) plot with PI-Control');
legend('Plant', 'gain=Kp');
xlabel('Frequency (rad/s)');
ylabel('Magnitude (dB)');
subplot(212), semilogx(w, php(:), w, phol1(:)), grid;
title('Open-loop Bode (phase) plot with PI-Control');
legend('Plant', 'gain=Kp');
xlabel('Frequency (rad/s)');
ylabel('Phase (degrees)');
```