# Control Systems II Assignment – Compensator Design and Simulation

This report presents the analysis and design of lead, lag, and lead–lag compensators for first-order and second-order systems using MATLAB. The objectives are to improve both transient and steady-state performance through dynamic compensation and to analyze the results using step response and Bode plots.

## Part A: Theoretical Questions

### 1. Difference Between Lead and Lag Compensators

|  |  |  |
| --- | --- | --- |
| Aspect | Lead Compensator | Lag Compensator |
| Primary Function | Improves transient response and system stability | Improves steady-state accuracy |
| Phase Effect | Adds positive phase (phase lead) | Adds negative phase (phase lag) |
| Zero–Pole Relation | Zero is closer to origin than pole (z < p) | Pole is closer to origin than zero (p < z) |
| Effect on Speed | Increases speed and bandwidth | Slows response slightly |
| Effect on Error | Little effect on steady-state error | Reduces steady-state error |
| Typical Use | Enhancing rise time, damping, and reducing overshoot | Improving steady-state accuracy |

### 2. Cascade Design for First-Order System

For a first-order system, the lead compensator increases phase margin and bandwidth, while the lag compensator increases low-frequency gain to improve steady-state accuracy. Their combination enhances both transient and steady-state performance.

### 3. Cascade Design for Second-Order System

In a second-order system, the lead compensator improves damping ratio and transient response, while the lag compensator enhances low-frequency gain and steady-state accuracy. The combination results in a faster, stable, and more accurate system.

### 4. Influence of Poles and Zeros on System Behavior

|  |  |  |
| --- | --- | --- |
| System Parameter | Effect of Pole Location | Effect of Zero Location |
| Rise Time | Poles farther left → faster rise; closer to origin → slower | Zeros near origin reduce rise time |
| Settling Time | Poles farther left → shorter settling time | Zeros may slightly affect settling if near poles |
| Overshoot | Complex poles with small damping cause large overshoot | Properly placed zeros can reduce overshoot |
| Steady-State Error | Poles near origin increase steady-state error | Zeros near origin improve accuracy |

### 5. Use of Bode Plots

|  |  |  |  |
| --- | --- | --- | --- |
| Concept | Explanation | Lead Compensator Effect | Lag Compensator Effect |
| Gain Margin (GM) | Extra gain before instability; found where phase = −180° | Increases gain margin | Slightly decreases gain margin |
| Phase Margin (PM) | Extra phase before instability; measured where gain = 0 dB | Adds positive phase → increases PM | Adds negative phase → reduces PM slightly |
| Bandwidth | Frequency range where response is above −3 dB | Widens bandwidth → faster system | Narrows bandwidth slightly |
| Transient Response | Determined by crossover frequency and PM | Faster response, less overshoot | Slower response, more accuracy |
| Steady-State Accuracy | Determined by low-frequency gain | Little improvement | Significantly improves accuracy |

## Part B: MATLAB Simulation and Analysis

The MATLAB simulations were conducted for both first-order and second-order systems using lead, lag, and lead–lag compensators. The goal was to observe improvements in rise time, overshoot, settling time, and steady-state accuracy.

### 1. First-Order System Simulation

Example system: G(s) = 1 / (s + 1)

The lead compensator improves transient response by increasing the phase margin, while the lag compensator enhances low-frequency gain to reduce steady-state error. Their cascade combination achieves a balance between speed and accuracy.

MATLAB Code:

clc;  
clear;  
close all;  
G = tf(1, [1 1]);  
G\_lead = tf([1 1], [1 10]);  
G\_lag = tf([1 0.1], [1 0.01]);  
G\_cascade = G \* G\_lead \* G\_lag;  
G\_closed = feedback(G\_cascade, 1);  
step(G, 'b', G\_closed, 'r');  
legend('Uncompensated', 'Lead-Lag Compensated');  
title('Step Response: First-Order System with Lead-Lag Compensation');

### 2. Second-Order System Simulation

Example system: G(s) = 1 / (s² + 2s + 10)

The lead–lag cascade compensation for the second-order system improved transient performance by increasing the damping ratio and reducing overshoot. It also enhanced steady-state accuracy.

MATLAB Code:

clc;  
clear;  
close all;  
G = tf(1, [1 2 10]);  
G\_lead = tf([1 1], [1 10]);  
G\_lag = tf([1 0.1], [1 0.01]);  
G\_lead\_lag = G\_lead \* G\_lag;  
G\_closed = feedback(G \* G\_lead\_lag, 1);  
step(G, 'b', G\_closed, 'r');  
legend('Uncompensated', 'Lead-Lag Compensated');  
title('Step Response: Second-Order System with Lead-Lag Compensation');

### Simulation Results Summary

|  |  |  |
| --- | --- | --- |
| System | Before Compensation | After Lead–Lag Compensation |
| First-Order | Slow rise time, steady-state error present | Faster response, nearly zero steady-state error |
| Second-Order | Oscillatory with overshoot | Reduced overshoot, faster settling, more stable |
| Phase Margin | Low | Increased, better stability |
| Steady-State Accuracy | Moderate | Improved |