

**Semester Project**

***Control System Design***

***Supervised By:***

*Dr. Ahsan Ali*

***Submitted By:***

*Asad ur Rehman*

*19-MS-EE-18*

*MuhammadAwais Usmani*

*19-MS-EE-11*

DEPARTMENT OF ELECTRICAL ENGINEERING

UNIVERSITY OF ENGINEERING AND TECHNOLOGY, TAXILA

**Contants**

* 1. Abstract .............................................................. 3

1. **Suspension system................................................4**
   1. Model Chosen.........................................................4
   2. linearization for Transfer Function Equation..................4
2. **Suspension system Model And Dynamic Response....................................7**
   1. Transfer Function...................................................................................7

2.11 For Step

* 1. Bode Plot
  2. Root Locus.
  3. Nyquist Plot
  4. H-Infinity
  5. For PID Controller
  6. Conclusion

0.1 **Abstract:**

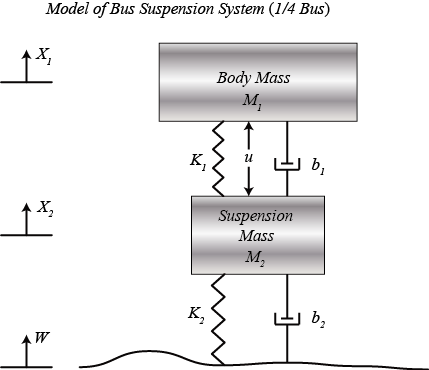
This document include the linearization of suspension system from nonlinear equations. Then it involves equations of motion of suspension system and simplification for transfer function models. Due to transfer function techniques we can make Bode Plot, Root Locus and Nyquist Plot. Controllers like H-infinity and PID are designed to further improve the stability.

**Chapter 1**

**Suspension system**

* 1. **Model Chosen:**

The parametric behavior of suspension system is non-linear but for simplicity most of the researchers have assumed it as linear. The emphasis of this study is to analyze the non-linear behavior of basic components of the suspension system. Designing an automotive suspension system is an interesting and challenging control problem. When the suspension system is designed, a 1/4 model (one of the four wheels) is used to simplify the problem to a 1-D multiple spring-damper system. A diagram of this system is shown below. This model is for an active suspension system where an actuator is included that is able to generate the control force U to control the motion of the bus body.



**Model of suspension system**

* 1. **Linearization for Transfer Function Equation:**

**Equations of motion:**

From the picture above and Newton's law, we can obtain the dynamic equations as the following:

$$ M_1 \ddot{X}_1 = - b_1 (\dot{X}_1 - \dot{X}_2) - K_1 (X_1 - X_2) + U \ $$ (1)

$$ M_2 \ddot{X}_2 = b_1 (\dot{X}_1 - \dot{X}_2) + K_1 (X_1 - X_2) + b_2 (\dot{W} - \dot{X}_2) + K_2 (W - X_2) - U$$ (2)

**Transfer Function Models:**

Assume that all of the initial conditions are zero, so that these equations represent the situation where the vehicle wheel goes up a bump. The dynamic equations above can be expressed in the form of transfer functions by taking the Laplace Transform. The specific derivation from the above equations to the transfer functions G1(s) and G2(s) is shown below where each transfer function has an output of, X1-X2, and inputs of U and W, respectively.

$$ (M_1 s^2 + b_1 s + K_1) X_1(s) - (b_1 s + K_1) X_2(s) = U(s) \ $$ (3)

$$ -(b_1 s + K_1) X_1(s) + (M_2 s^2 + (b_1 + b_2) s + (K_1 + K_2)) X_2(s) = (b_2 s + K_2) W(s) - U(s)$$ (4)

(5)$$ \left[{\begin{array}{cc}(M_1 s^2 + b_1 s + K_1)& -(b_1 s + K_1)\\
-(b_1 s + K_1)& (M_2 s^2 + (b_1 + b_2) s + (K_1 + K_2))\end{array}}\right]
\left[{\begin{array}{c}X_1(s)\\X_2(s)\end{array}}\right] =
\left[{\begin{array}{c}U(s)\\(b_2 s + K_2) W(s) - U(s)\end{array}}\right] $$

$$ A = \left[{\begin{array}{cc}(M_1 s^2 + b_1 s + K_1)& -(b_1 s + K_1)\\
-(b_1 s + K_1)& (M_2 s^2 + (b_1 + b_2) s + (K_1 + K_2))\end{array}}\right]  $$ (6)

$$ \Delta = \mathrm{det} \left[{\begin{array}{cc}(M_1 s^2 + b_1 s + K_1)& -(b_1 s + K_1)\\
-(b_1 s + K_1)& (M_2 s^2 + (b_1 + b_2) s + (K_1 + K_2))\end{array}}\right]  $$ (7)

or

$$ \Delta = (M_1 s^2 + b_1 s + K_1) \cdot (M_2 s^2 + (b_1 + b_2) s + (K_1 + K_2)) - (b_1 s + K_1) \cdot (b_1 s + K_1) $$ (8)

Find the inverse of matrix A and then multiply with inputs U(s)and W(s) on the righthand side as follows:

(9)$$ \left[{\begin{array}{c}X_1(s)\\X_2(s)\end{array}}\right] =
\frac{1}{\Delta} \left[{\begin{array}{cc}(M_2 s^2 + (b_1 + b_2) s + (K_1 + K_2))&(b_1 s + K_1)\\
(b_1 s + K_1)& (M_1 s^2 + b_1 s + K_1)\end{array}}\right]
\left[{\begin{array}{c}U(s)\\(b_2 s + K_2) W(s) - U(s)\end{array}}\right] $$

(10)$$ \left[{\begin{array}{c}X_1(s)\\X_2(s)\end{array}}\right] =
\frac{1}{\Delta} \left[{\begin{array}{cc}(M_2 s^2 + b_2 s + K_2)& (b_1 b_2 s^2 + (b_1 K_2 + b_2 K_1) s + K_1 K_2) \\
-M_1 s^2& (M_1 b_2 s^3 + (M_1 K_2 + b_1 b_2) s^2 +(b_1 K_2 + b_2 K_1) s + K_1 K_2) \end{array}}\right]
\left[{\begin{array}{c}U(s)\\W(s)\end{array}}\right] $$

When we want to consider the control input U(s) only, we set W(s) = 0. Thus we get the transfer function G1(s) as in the following:

$$ G_1(s) = \frac{X_1(s) - X_2(s)}{U(s)} = \frac{(M_1+M_2) s^2 + b_2 s + K_2}{\Delta} $$ (11)

When we want to consider the disturbance input W(s) only, we set U(s) = 0. Thus we get the transfer function G2(s) as in the following:

$$ G_2(s) = \frac{X_1(s) - X_2(s)}{W(s)} = \frac{-M_1 b_2 s^3 -M_1 K_2 s^2}{\Delta} $$ (12)

**Chapter 02**

**Suspension system Model and Dynamic Response**

* 1. **For Transfer Function**

In engineering, a transfer function (also known as system function[\](https://en.wikipedia.org/wiki/Transfer_function#cite_note-1) or network function) of an electronic or [control system](https://en.wikipedia.org/wiki/Control_system) [component](https://en.wikipedia.org/wiki/Electronic_component) is a [mathematical function](https://en.wikipedia.org/wiki/Function_(mathematics)) which [theoretically models](https://en.wikipedia.org/wiki/Mathematical_model) the device's output for each possible input.

**Code:**

M1 = 2500;

M2 = 320;

K1 = 80000;

K2 = 500000;

b1 = 350;

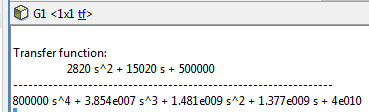
b2 = 15020;

s = tf('s');

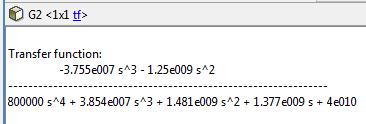
G1 = ((M1+M2)\*s^2+b2\*s+K2)/((M1\*s^2+b1\*s+K1)\*(M2\*s^2+(b1+b2)\*s+(K1+K2))-(b1\*s+K1)\*(b1\*s+K1));

G2 = (-M1\*b2\*s^3-M1\*K2\*s^2)/((M1\*s^2+b1\*s+K1)\*(M2\*s^2+(b1+b2)\*s+(K1+K2))-(b1\*s+K1)\*(b1\*s+K1));

**T.f of G1:**



**T.f of G2:**



**2.11 For Step Response:**

In [electronic engineering](https://en.wikipedia.org/wiki/Electronic_engineering) and [control theory](https://en.wikipedia.org/wiki/Control_theory), step response is the time behaviour of the outputs of a general [system](https://en.wikipedia.org/wiki/System) when its inputs change from zero to one in a very short time.

**Code:**

M1 = 2500;

M2 = 320;

K1 = 80000;

K2 = 500000;

b1 = 350;

b2 = 15020;

s = tf('s');

G1 = ((M1+M2)\*s^2+b2\*s+K2)/((M1\*s^2+b1\*s+K1)\*(M2\*s^2+(b1+b2)\*s+(K1+K2))-(b1\*s+K1)\*(b1\*s+K1));

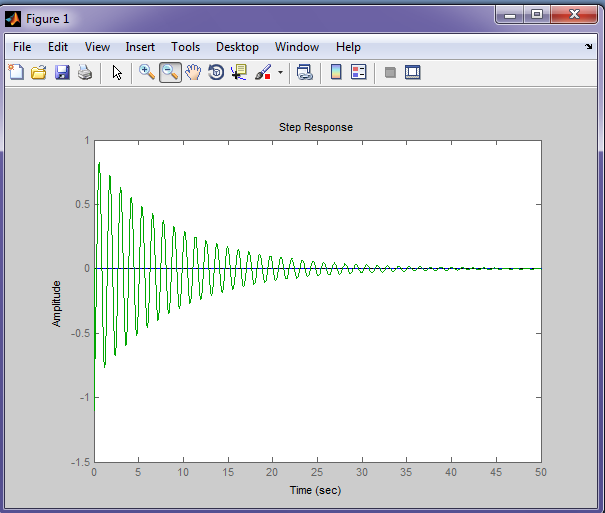
G2 = (-M1\*b2\*s^3-M1\*K2\*s^2)/((M1\*s^2+b1\*s+K1)\*(M2\*s^2+(b1+b2)\*s+(K1+K2))-(b1\*s+K1)\*(b1\*s+K1));

step(G1)

hold on;

step(G2)

**Step Response:**



* 1. **For Bode Plot:**

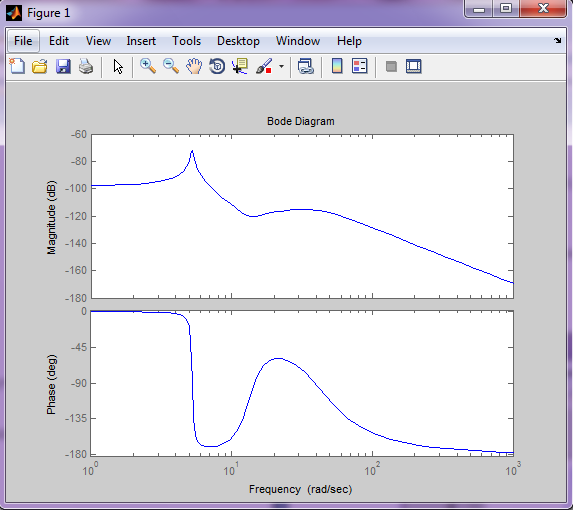
In electrical engineering and control theory, a Bode plot is a graph of the frequency response of a system. It is usually a combination of a Bode magnitude plot, expressing the magnitude (usually in decibels) of the frequency response, and a Bode phase plot, expressing the phase shift.

**For TF G1:**

G=tf([0 0 2820 15020 500000],[800000 3.854e007 1.481e009 1.377e009 4e010]);

bode(G)

**Bode Plot:**

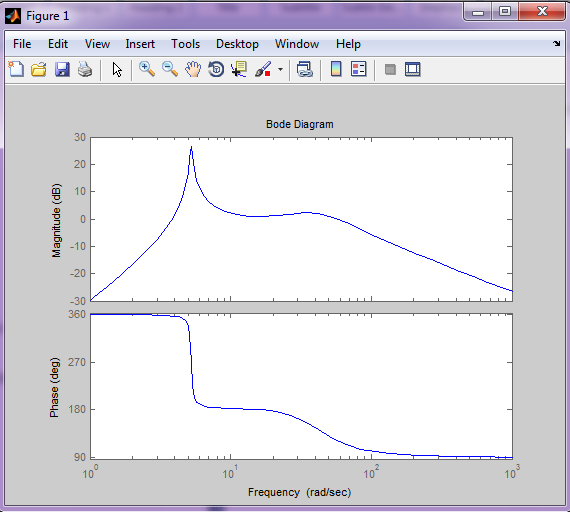


**For TF G2:**

G=tf([0 -3.755e007 -1.25e009 0 0],[800000 3.854e007 1.481e009 1.377e009 4e010]);

bode(G)

**Bode Plot:**



* 1. **For Root Locus**

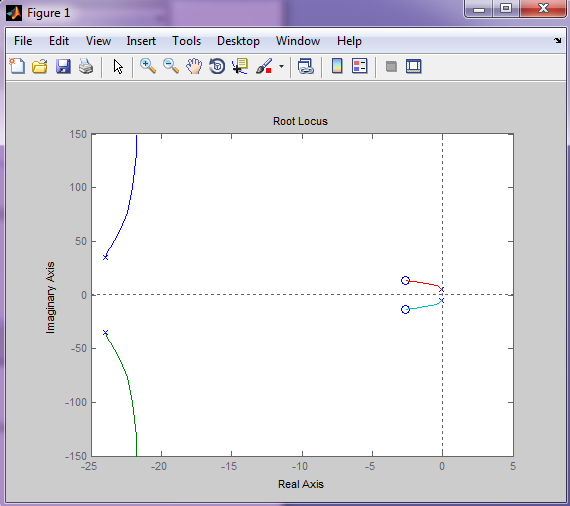
In control theory and stability theory, root locus analysis is a graphical method for examining how the roots of a system change with variation of a certain system parameter, commonly a gain within a feedback system.

**For TF G1:**

G=tf([0 0 2820 15020 500000],[800000 3.854e007 1.481e009 1.377e009 4e010]);

rlocus(G)

**Root Locus:**

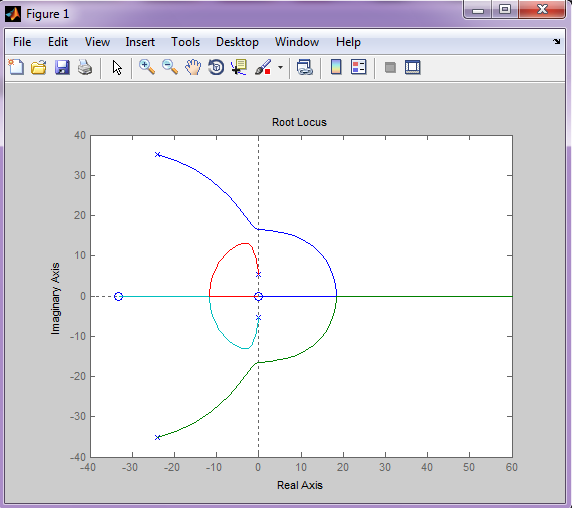


**For TF G2:**

G=tf([0 -3.755e007 -1.25e009 0 0],[800000 3.854e007 1.481e009 1.377e009 4e010]);

rlocus(G)

**Root Locus:**



* 1. **For Nyquist Plot:**

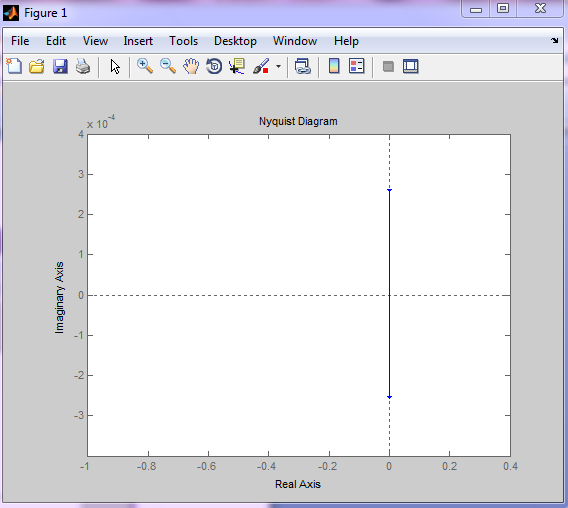
In [control theory](https://en.wikipedia.org/wiki/Control_theory) and [stability theory](https://en.wikipedia.org/wiki/Stability_theory), the Nyquist stability criterion or Strecker–Nyquist stability criterion, independently discovered by the German electrical engineer [Felix Strecker](https://en.wikipedia.org/w/index.php?title=Felix_Strecker&action=edit&redlink=1)  at [Siemens](https://en.wikipedia.org/wiki/Siemens) in 1930 is a graphical technique for determining the [stability](https://en.wikipedia.org/wiki/Stability_criterion) of a [dynamical system](https://en.wikipedia.org/wiki/Dynamical_system).

**For TF G1:**

G=tf([0 0 2820 15020 500000],[800000 3.854e007 1.481e009 1.377e009 4e010]);

nyquist(G)

**Nyquist Diagram:**

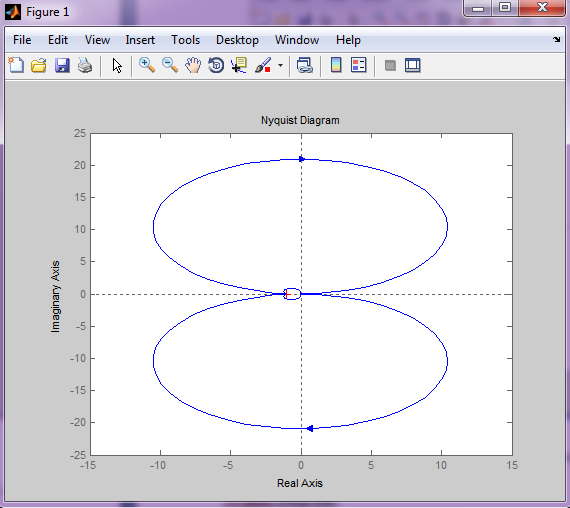


**For TF G2:**

G=tf([0 -3.755e007 -1.25e009 0 0],[800000 3.854e007 1.481e009 1.377e009 4e010]);

nyquist(G)

**Nyquist Diagram:**



* 1. **For H Infinity:**

H-infinity methods are used in [control theory](https://en.wikipedia.org/wiki/Control_theory) to synthesize controllers to achieve stabilization with guaranteed performance. To use H∞ methods, a control designer expresses the control problem as a [mathematical optimization](https://en.wikipedia.org/wiki/Mathematical_optimization) problem and then finds the controller that solves this optimization.

**For TF G1:**

%asad

%uses the robust control tool box

clc

close all

G=tf([0 0 2820 15020 500000],[800000 3.854e007 1.481e009 1.377e009 4e010]);

M=1.5; wb=10; A=1.e-4;

Wp=tf([1/M wb], [1 wb\*A]); Wu=1;

%Find H-infinity optimal controller;

[khinf,ghinf,gopt]=mixsyn(G,Wp,Wu,[]);

Marg= allmargin(G\*khinf)

khinf

K=tf(khinf)

L=K\*G

subplot(2,1,1)

step(feedback(L,1),3)

hold on

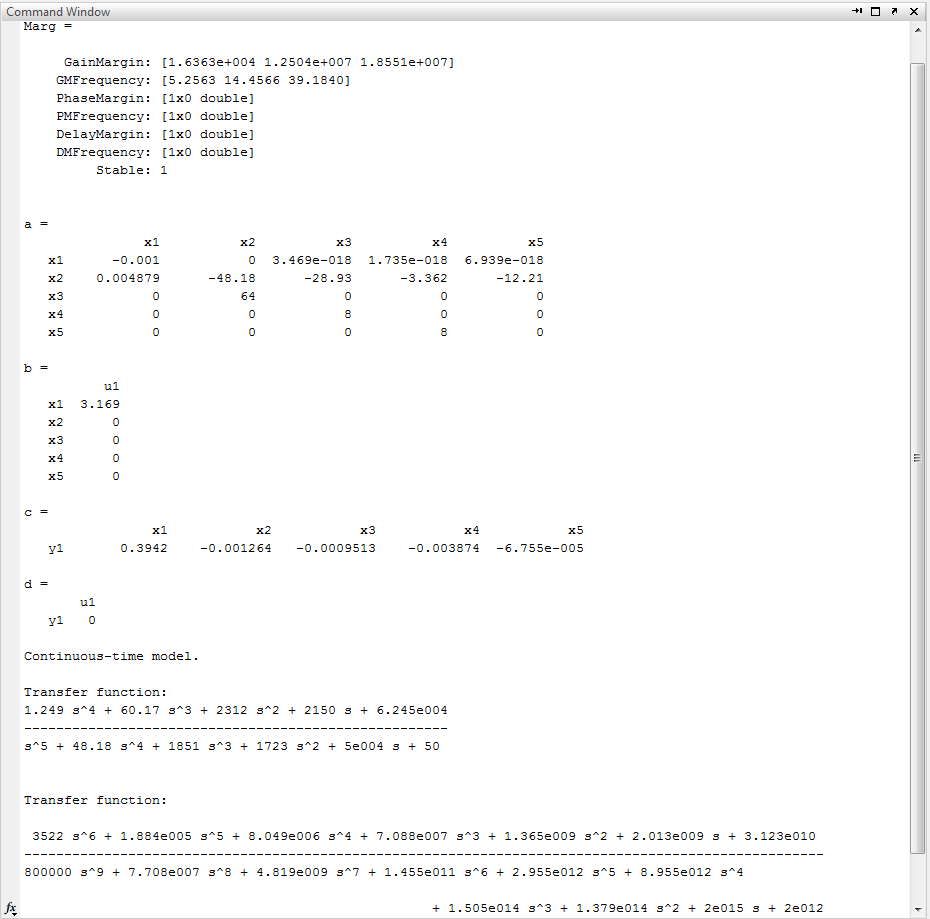
subplot(2,1,2)

step(feedback(G,K),3)

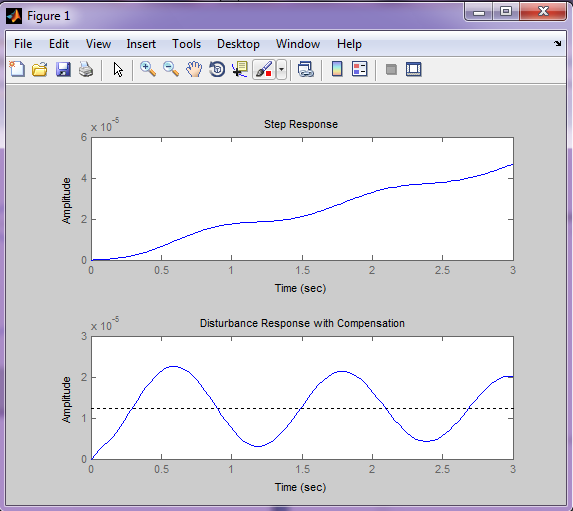
title(' Disturbance Response with Compensation')

hold on

**In Command window:**



**Step And Disturbance response:**

****

**For TF G2:**

%asad

%uses the robust control tool box

clc

close all

G=tf([0 -3.755e007 -1.25e009 0 0],[800000 3.854e007 1.481e009 1.377e009 4e010]);

M=1.5; wb=10; A=1.e-4;

Wp=tf([1/M wb], [1 wb\*A]); Wu=1;

%Find H-infinity optimal controller;

[khinf,ghinf,gopt]=mixsyn(G,Wp,Wu,[]);

Marg= allmargin(G\*khinf)

khinf

K=tf(khinf)

L=K\*G

subplot(2,1,1)

step(feedback(L,1),3)

hold on

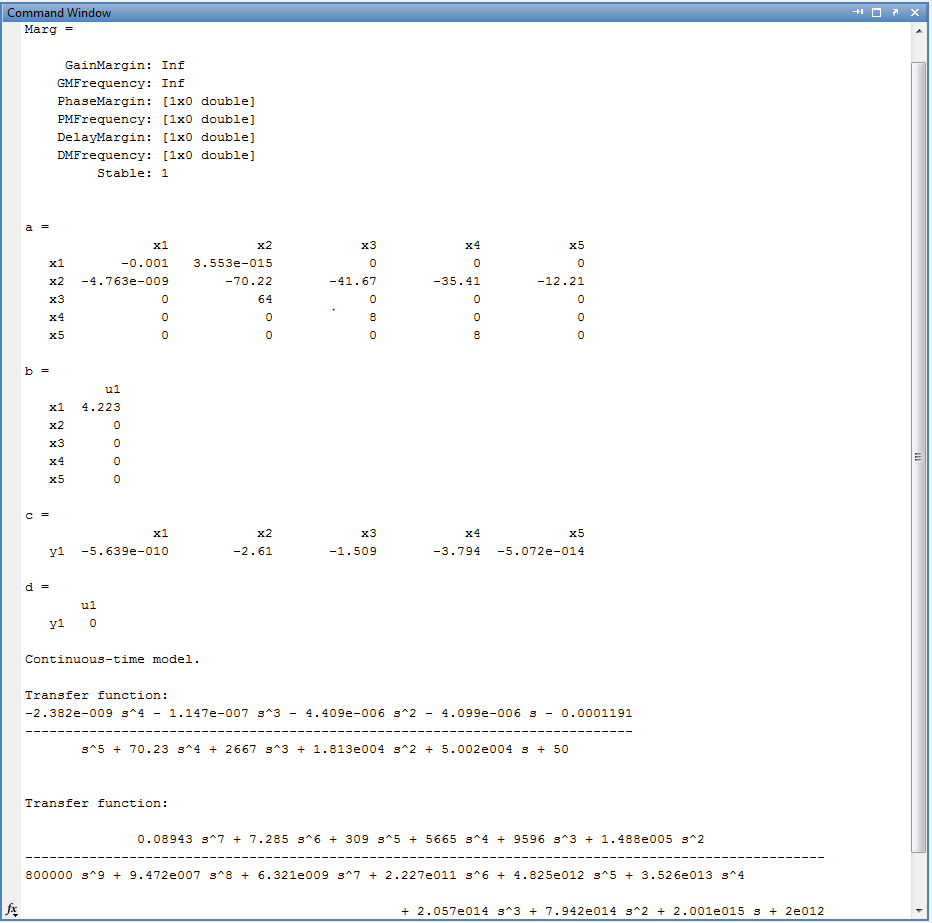
subplot(2,1,2)

step(feedback(G,K),3)

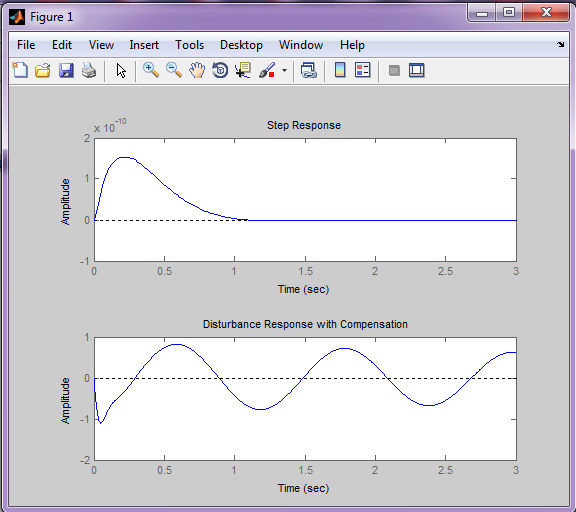
title(' Disturbance Response with Compensation')

hold on

**In Command window:**

****

**Step And Disturbance response:**

****

* 1. **For PID Controller:**

From the main problem, the dynamic equations in transfer function form are the following:

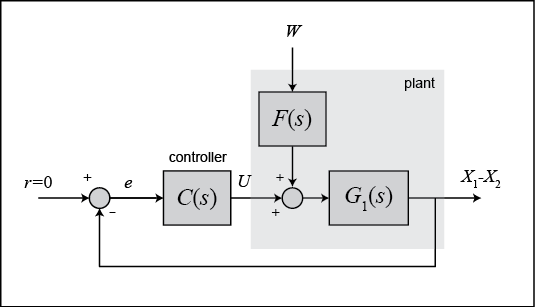
$$
G_1(s) = \frac{X_1(s)-X_2(s)}{U(s)}=\frac{(M_1+M_2)s^2+b_2s+K_2}{\Delta}
$$ (1)

$$
G_2(s) = \frac{X_1(s)-X_2(s)}{W(s)}=\frac{-M_1b_2s^3-M_1K_2s^2}{\Delta}
$$ (2)

where,

$$
\Delta = (M_1s^2+b_1s+K_1)(M_2s^2+(b_1+b_2)s+(K_1+K_2))-(b_1s+K_1)(b_1s+K_1)
$$ (3)

and the system schematic is the following where F(s)G1(s) = G2(s).



**For Closed-loop response with PID Controller:**

Kd = 208025;

Kp = 832100;

Ki = 624075;

C\_PID=tf([208025 832100 624075],[1 0]);

G=tf([0 0 2820 15020 500000],[800000 3.854e007 1.481e009 1.377e009 4e010]);

F=tf([-3.755e007 -1.25e009 0 0],[0 2820 15020 500000]);

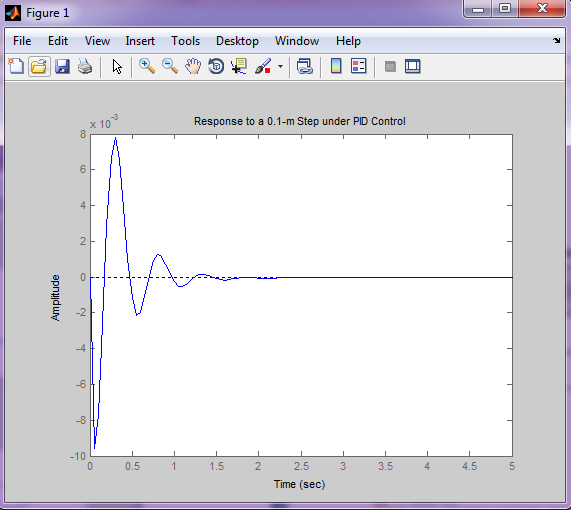
sys\_cl=F\*feedback(G,C\_PID);

t=0:0.05:5;

step(0.1\*sys\_cl,t)

title('Response to a 0.1-m Step under PID Control')

**Step Response under PID:**

****

**For Nyquist Plot of Plant with PID Controller:**

Kd = 208025;

Kp = 832100;

Ki = 624075;

C\_PID=tf([208025 832100 624075],[1 0]);

G=tf([0 0 2820 15020 500000],[800000 3.854e007 1.481e009 1.377e009 4e010]);

F=tf([-3.755e007 -1.25e009 0 0],[0 2820 15020 500000]);

sys\_cl=F\*feedback(G,C\_PID);

t=0:0.05:5;

step(0.1\*sys\_cl,t)

title('Response to a 0.1-m Step under PID Control')

z1=1;

z2=3;

p1=0;

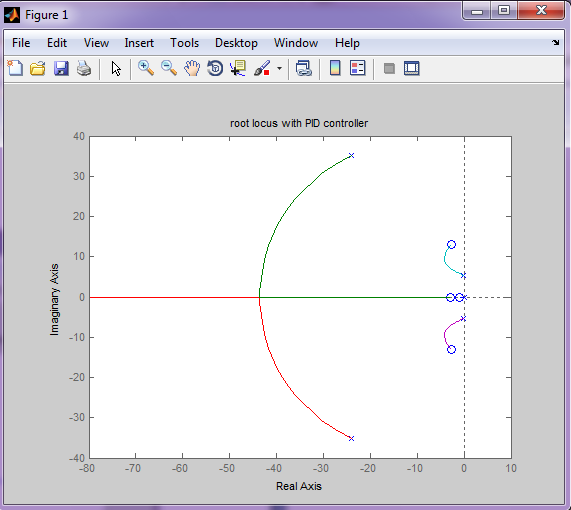
s = tf('s');

C = ((s+z1)\*(s+z2))/(s+p1);

rlocus(C\*G)

title('root locus with PID controller')

**Root Locus with PID Controller:**

****

**For Bode Plot of Plant with PID Controller:**

Kd = 208025;

Kp = 832100;

Ki = 624075;

C\_PID=tf([208025 832100 624075],[1 0]);

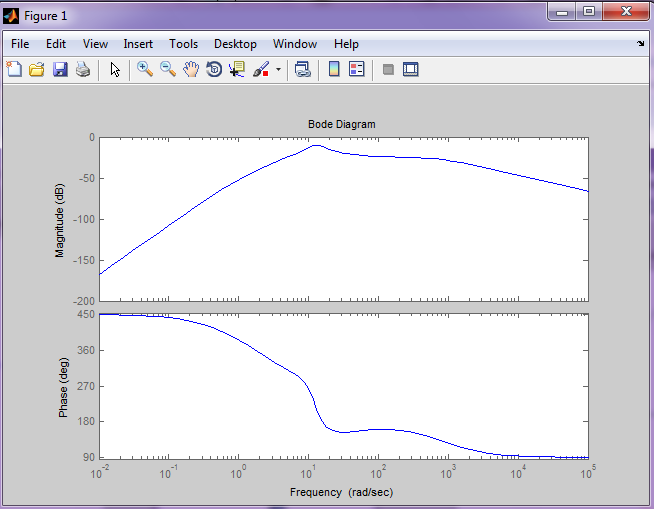
G=tf([0 0 2820 15020 500000],[800000 3.854e007 1.481e009 1.377e009 4e010]);

F=tf([-3.755e007 -1.25e009 0 0],[0 2820 15020 500000]);

sys\_cl=F\*feedback(G,C\_PID);

bode(sys\_cl)

**Bode Diagram:**

****

**For Root Locus Plot of Plant with PID Controller:**

Kd = 208025;

Kp = 832100;

Ki = 624075;

C\_PID=tf([208025 832100 624075],[1 0]);

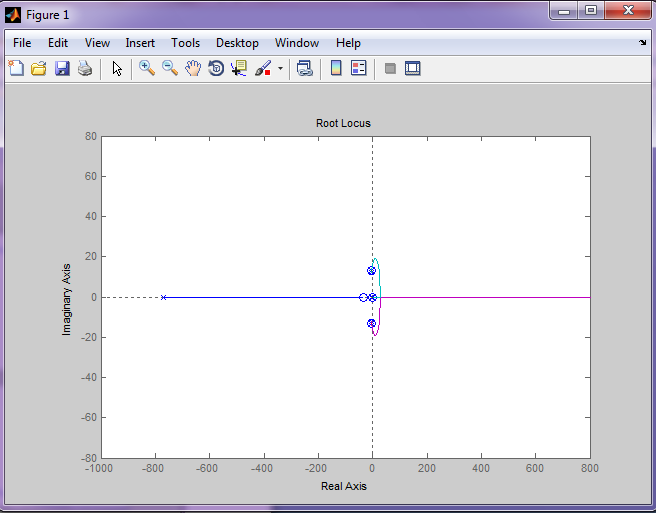
G=tf([0 0 2820 15020 500000],[800000 3.854e007 1.481e009 1.377e009 4e010]);

F=tf([-3.755e007 -1.25e009 0 0],[0 2820 15020 500000]);

sys\_cl=F\*feedback(G,C\_PID);

rlocus(sys\_cl)

**Root Locus Diagram:**

****

* 1. **Conclusion**

In this project we work on linearization of suspension system from non-linear equations. Then we start with equations of motion of suspension system and simplification for transfer function models. Due to transfer function techniques we made Bode Plot, Root Locus and Nyquist Plot. Controllers like H-infinity and PID are designed to further improve the stability of the plant.