ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/317345024

Vehicle Dynamics Modeling & Simulation

Book · September 2016

CITATIONS
4

READS
19,258

3 authors, including:

Mohd Azman Abdullah
Technical University of Malaysia Malacca

82 PUBLICATIONS **218** CITATIONS

SEE PROFILE

Ahmed Esmael Mohan
Al-Furat Al-Awsat Technical University

11 PUBLICATIONS **8** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Optimization of semi-active suspension controller for vehicle application featuring magnetorheological damper View project

Project Performance analysis of brake discs in trains View project

All content following this page was uploaded by Mohd Azman Abdullah on 05 June 2017.

The user has requested enhancement of the downloaded file.

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

VEHICLE DYNAMICS

MODELING & SIMULATION

MOHD AZMAN ABDULLAH
JAZLI FIRDAUS JAMIL
AHMED ESMAEL MOHAN

VEHICLE DYNAMICS
MODELING & SIMULATION

MOHD AZMAN ABDULLAH

JAZLI FIRDAUS JAMIL

AHMED ESMAEL MOHAN

First published 2016

Copyright © 2016 by Centre for Advanced Research on Energy (CARe)

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, electronic, mechanical photocopying, recording or otherwise, without the prior permission of the Publisher.

ISBN: 978-967-0257-78-5

Published and Printed in Malaysia by:

Centre for Advanced Research on Energy,
Faculty of Mechanical Engineering, Universiti Teknikal Malaysia Melaka,
Hang Tuah Jaya 76100, Durian Tunggal, Melaka, MALAYSIA.

Tel: +6062346891 | Fax: +6062346884 | Email: care@utem.edu.my

As part of anti-plagiarism policy, we hereby acknowledge all various sources of facts and figures that we used in this book to give the understanding to the students. Most of the facts and figures we could not find the sources. We would like to thank all legal owners of the facts and figures.

CONTENTS

CHAPTER 6

MECHANICAL SYSTEM MODELING TUTORIAL USING MATLAB

6.1 Introduction

In this chapter, the analysis of 1DOF and 2 DOF quarter car mode is through transfer function coding in MATLAB interface. For higher degree of freedom systems, it is advisable to use MATLAB/Simulink (Chapter 7) or vehicle dynamic software CarSim (Chapter 8).

6.2 Matlab Coding

Fig. 6.1 shows the MATLAB software interface. This software is used for mathematical analysis, graph presentation and basic programming. **Fig. 6.2** shows the editor window of the programming. The coding is saved through this file and executed in the MATLAB interface.

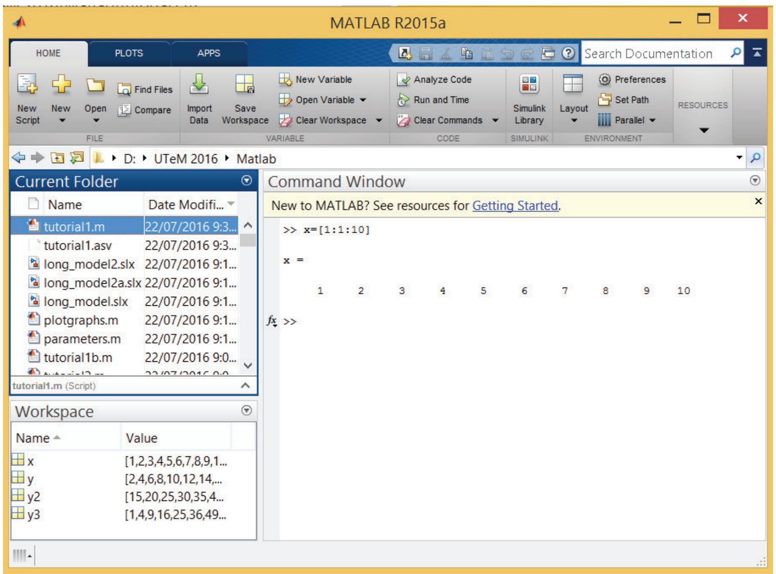


Fig. 6.1: Matlab interface

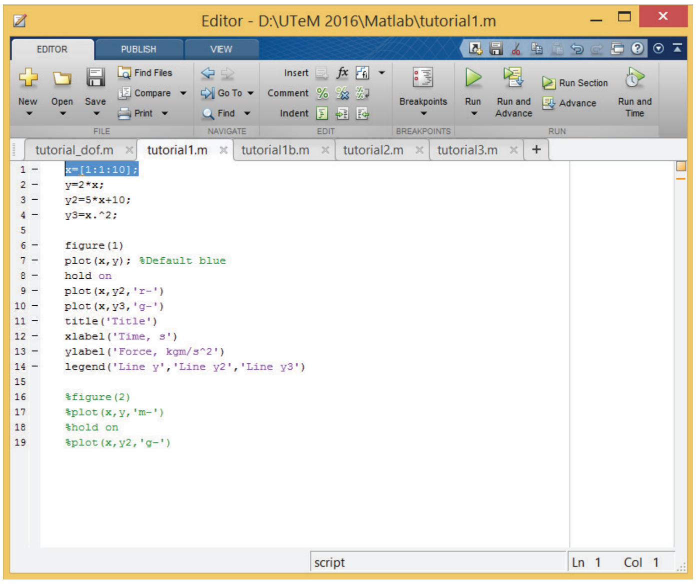


Fig. 6.2: Editor window

Fig. 6.3 shows example of figure produced by MATLAB for this example of coding.

```
clc; %clear command window
close all; %close all figure windows
clear; %clear all parameters

x=[1:1:10]; % ; = value not shown in command window
y=2*x;
y2=5*x+10;
y3=x.^2;

figure(1)
plot(x,y); %default blue
hold on
plot(x,y2,'r--') %r = red line, y = yellow, c = cyan
plot(x,y3,'g-.') %g = green line, k = black, b = blue
title('Title')
xlabel('Time, s')
ylabel('Force, kgm/s^2')
legend('Line y','Line y2','Line y3')
```

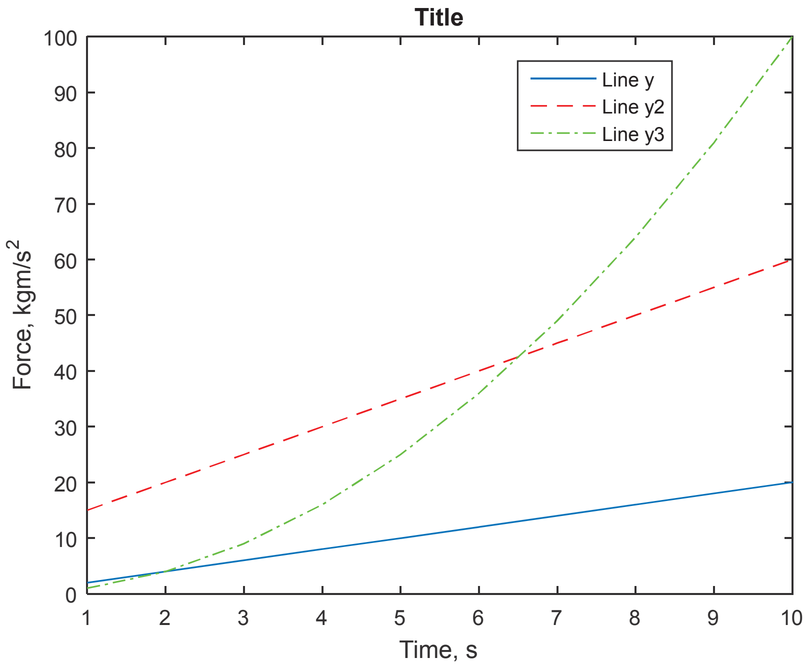


Fig. 6.3: Example plots

6.3 One degree of freedom system (1DOF)

Fig. 6.4 shows a simple 1DOF mechanical system. The derivation of equations is shown in the next page.

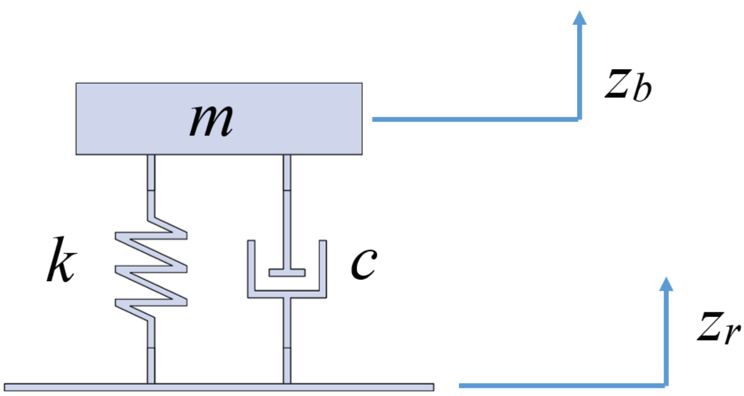


Fig. 6.4: 1DOF system

$$m\ddot{z}_b + c\dot{z}_b + kz_b - c\dot{z}_r - kz_r = 0$$

$$m\ddot{z}_b + c\dot{z}_b + kz_b = c\dot{z}_r + kz_r$$

The parameters for 1DOF system as follows.

```
>>m = 250; %kg
>>c = 1000; %Ns/m
>>k = 16000; %N/m
```

The transfer function coding,

$$Z_b(s)\ [m\ c\ k] = Z_r(s)\ [c\ k]$$

```
>>A = tf([m c k],[1]) %tf = transfer function

>>B = tf([c k],[1])

>>ZbZr = B/A
```

The transfer function is shown by,

```
ZbZr =

        1000 s + 16000
  ------------------------
  250 s^2 + 1000 s + 16000

Continuous-time transfer function.
```

Step input function is used to study the performance of the system (**Fig. 6.5**),

```
>>step(ZbZr)
```

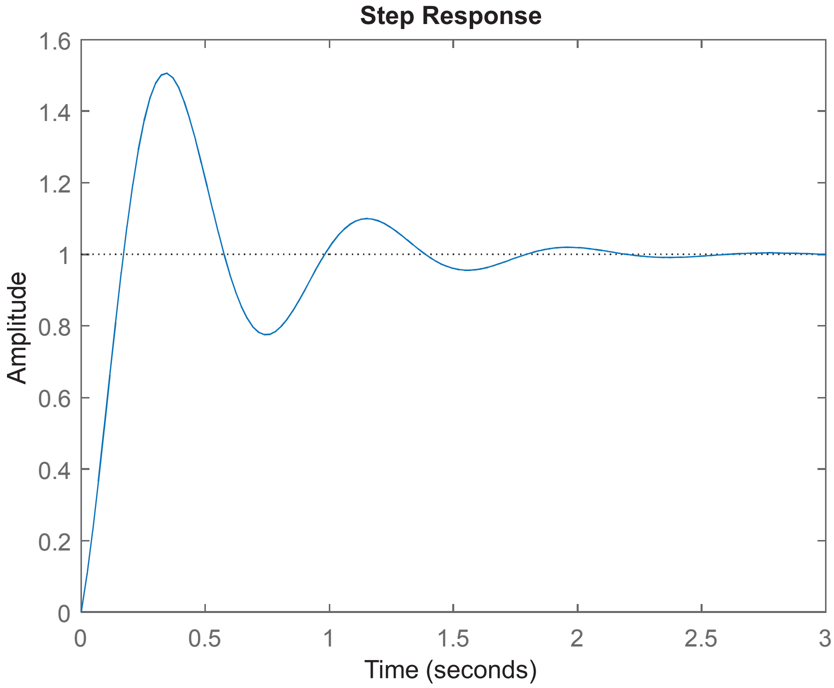


Fig. 6.5: Step response

The bode plot is used to study the frequency of the system (**Fig. 6.6** and **Fig. 6.7**).

```
>>bode(ZbZr)
```

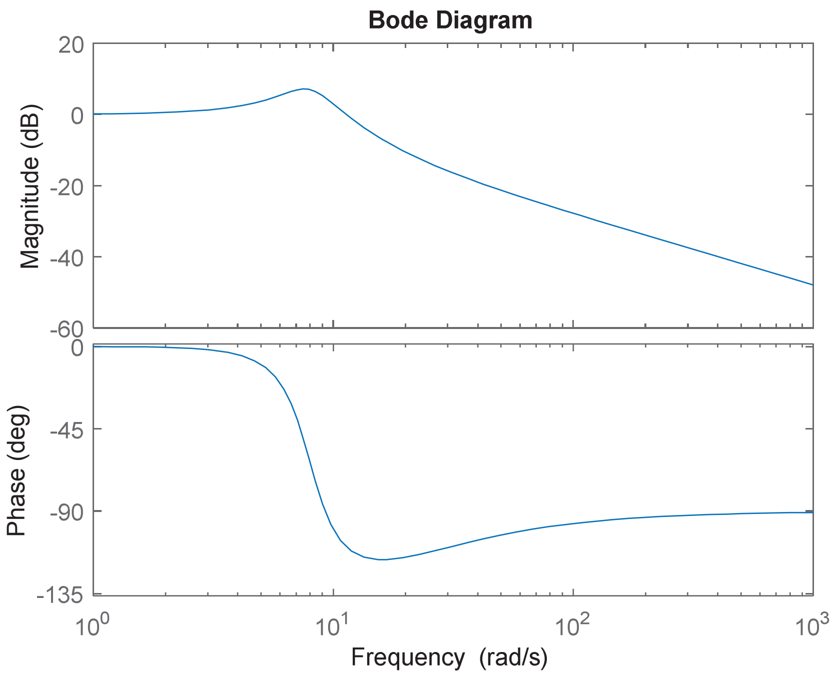


Fig. 6.6: Bode plot

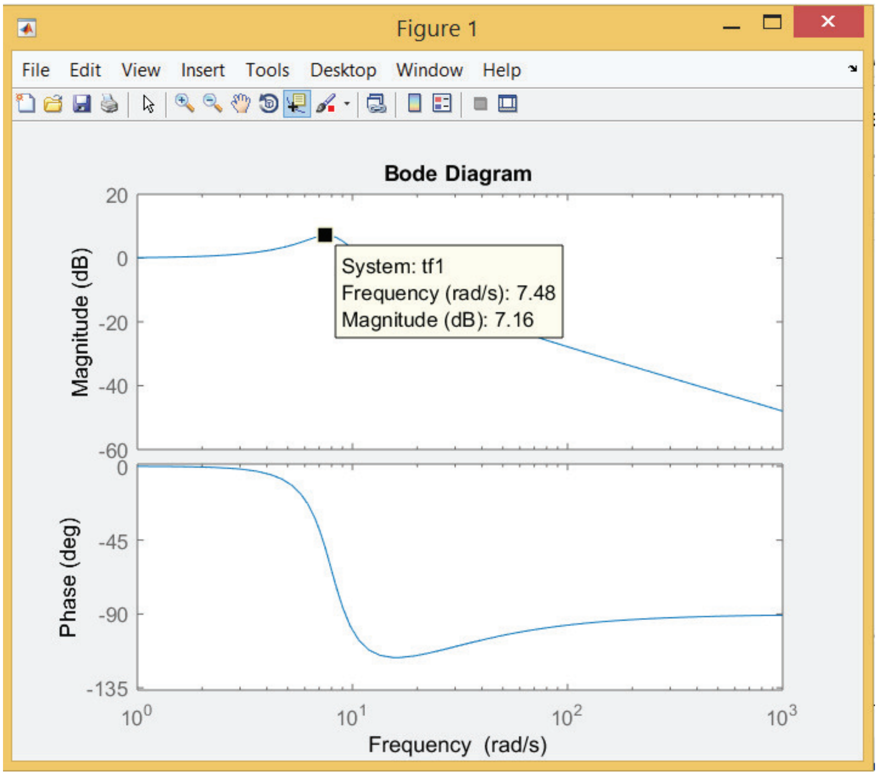


Fig. 6.7: Natural frequency from the plot = 7.48 rad/s (428.57 Hz)

Natural frequency from calculation, wn = sqrt(k/m) = 8 rad/s (458.37 Hz)

Natural frequency from coding,

```
>>[Wn,zeta] = damp(ZbZr) ; %Wn = natural frequency, zeta = damping factor

Wn =

    8.0000

    8.0000


zeta =

    0.2500

    0.2500
```

Analysis at different parameters, analysis at different *k* (**Fig. 6.8**). Write the parameters in the m-file (parameters_1dof.m).

```
clc; %clear command window
close all; %close all figure windows
clear; %clear all parameters

m = 250; %kg
c = 1000; %Ns/m
k = 16000; %N/m
A = tf([m c k],[1]) %tf = transfer function
B = tf([c k],[1])
k0 = 10000; %N/m
k2 = 20000; %N/m
A0 = tf([m c k0],[1])
A2 = tf([m c k2],[1])
B0 = tf([c k0],[1])
B2 = tf([c k2],[1])
ZbZr0 = B0/A0
ZbZr2 = B2/A2

figure(1)
bode(ZbZr0,'r--')
hold on;
bode(ZbZr)
bode(ZbZr2,'g-.')
legend('k = 10000 N/m','k = 16000 N/m','k = 20000 N/m')
```

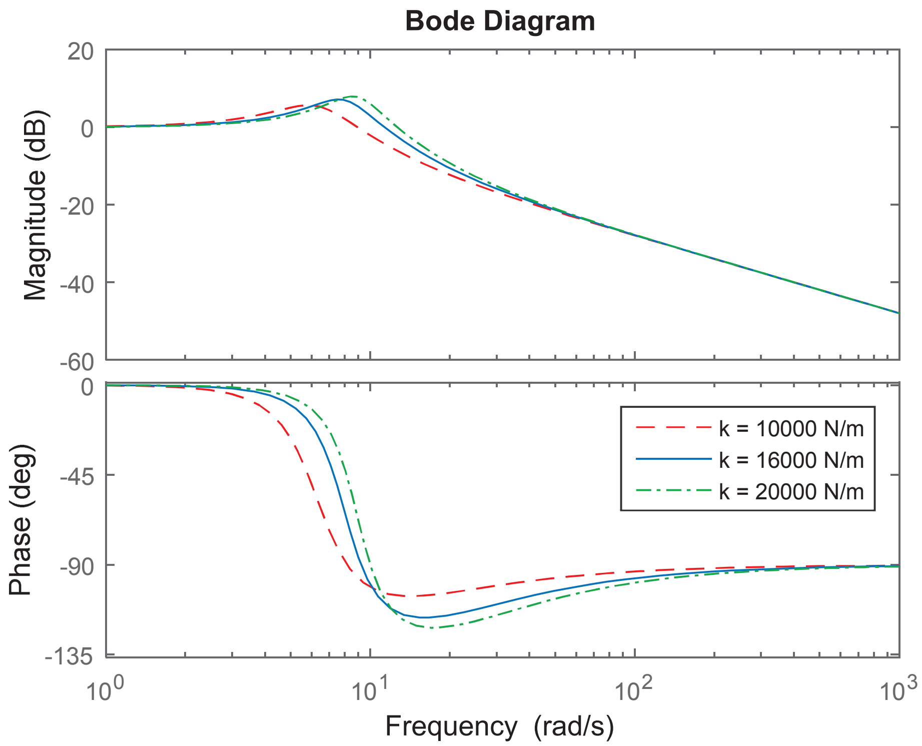


Fig. 6.8: Bode plot at different *k*

Analysis at different *c* (**Fig. 6.9**)

```
clc; %clear command window
close all; %close all figure windows
clear; %clear all parameters

m = 250; %kg
c = 1000; %Ns/m
k = 16000; %N/m
A = tf([m c k],[1]) %tf = transfer function
B = tf([c k],[1])
c0 = 100; %Ns/m
c2 = 10000; %Ns/m
A3 = tf([m c0 k],[1])
A4 = tf([m c2 k],[1])
B3 = tf([c0 k],[1])
B4 = tf([c2 k],[1])
ZbZr3 = B3/A3
ZbZr4 = B4/A4

figure(2)
bode(ZbZr3,'r--')
hold on;
bode(ZbZr)
bode(ZbZr4,'g-.')
legend('c = 100 Ns/m','c = 1000 Ns/m','c = 10000 Ns/m')
```

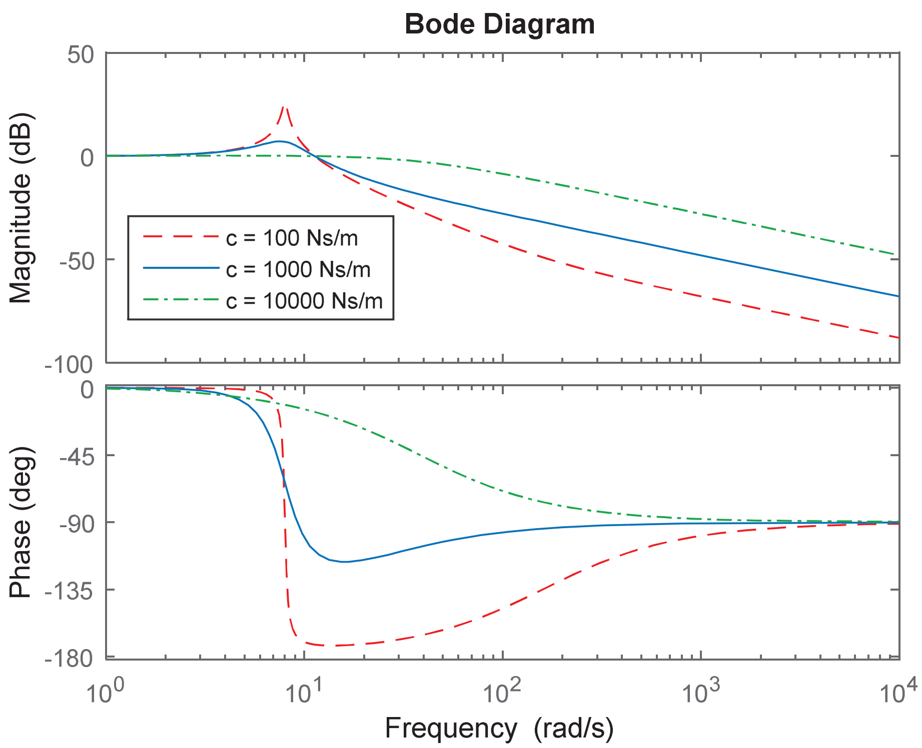


Fig. 6.9: Bode plot at different *c*

Analysis at different *m* (**Fig. 6.10**)

```
clc; %clear command window
close all; %close all figure windows
clear; %clear all parameters

m = 250; %kg
c = 1000; %Ns/m
k = 16000; %N/m
A = tf([m c k],[1]) %tf = transfer function
B = tf([c k],[1])
m0 = 200; %kg
m2 = 300; %kg
A5 = tf([m0 c k],[1])
A6 = tf([m2 c k],[1])
B5 = B % B5 =  B = tf([c k],[1])
B6 = B % B6 =  B = tf([c k],[1])
ZbZr5 = B5/A5
ZbZr6 = B6/A6

figure(3)
bode(ZbZr5,'r--')
hold on;
bode(ZbZr)
bode(ZbZr6,'g-.')
legend('m = 200 kg','m = 250 kg','m = 300 kg')
```

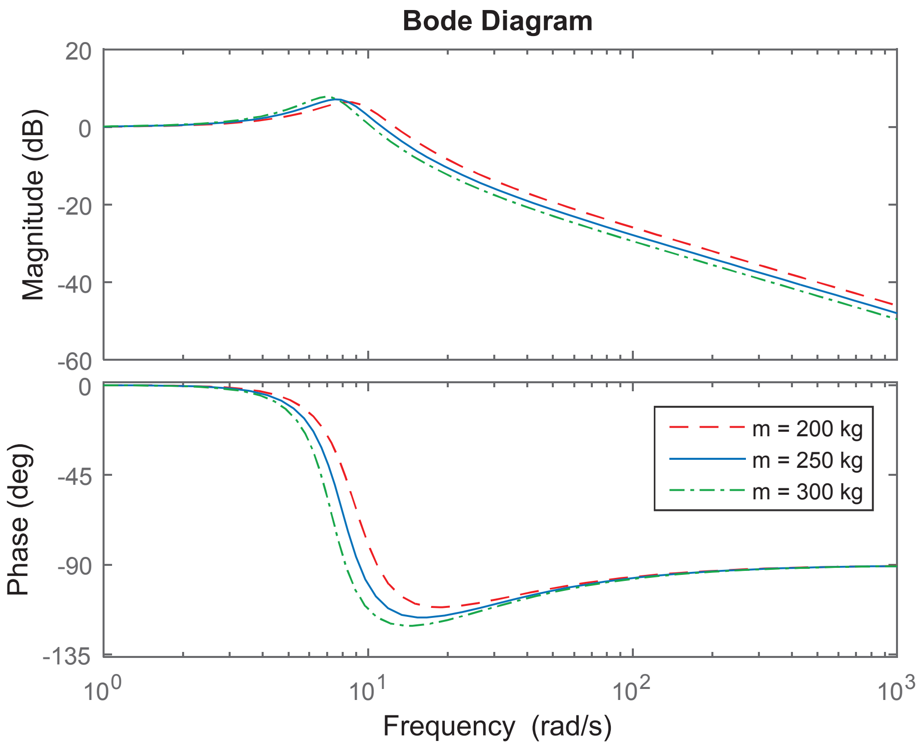


Fig. 6.10: Bode plot at different *m*

6.4 Two degree of freedom system (2DOF)

Fig. 6.11 shows a simple 2DOF mechanical system. The derivation of equations is shown.

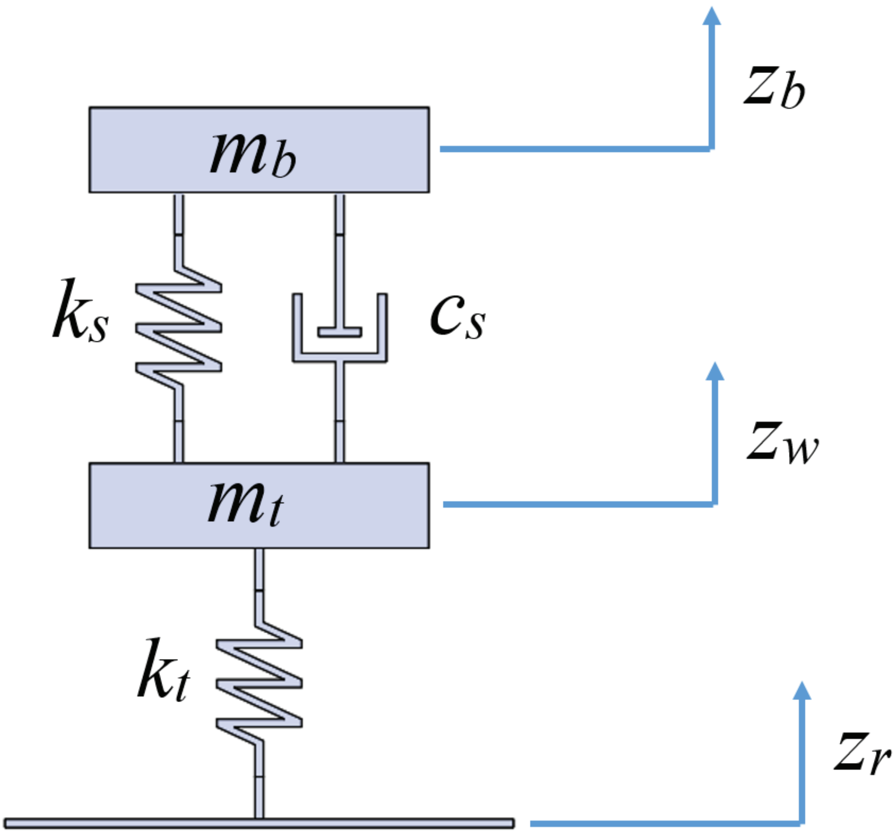


Fig. 11: 2DOF system

$$m_b\ddot{z}_b + c_s\dot{z}_b + k_s z_b = c_s\dot{z}_w + k_s z_w$$

$$m_t\ddot{z}_w + c_s\dot{z}_w + (k_s + k_t)z_w = k_t z_r$$

Transfer function,

$$Z_b(s)\ [m_b\ c_s\ k_s] = Z_w(s)\ [c_s\ k_s]$$

$$Z_w(s)\ [m_t\ c_s\ k_s{+}k_r] = Z_r(s)\ [k_t]$$

```
A = tf([mb cs ks],[1])
B = tf([cs ks],[1])
C = tf([mt cs ks+kt],[1])
D = tf([kt],[1])
```

The parameters for 2DOF system is shown in the next coding.

```
mb = 454.5; %kg
mt = 45.45; %kg
ks = 22000; %N/m
cs = 2400; %Ns/m
kt = 176000; %N/m

A = tf([mb cs ks],[1])
B = tf([cs ks],[1])
C = tf([mt cs ks+kt],[1])
D = tf([kt],[1])

% vibration isolation
ZbZr = D/(A*C/B-B); %Zb output Zr input, Zb/Zr = vibration isolation
% ZbZr = (D*B/C)/(A-B*B/C)

% susp travel response and road input
ZwZbZr = ZwZr - ZbZr % ( Zw - Zb ) / Zr

% tire deflect response and road input
ZrZwZr = 1 - ZwZr; % = ( Zr - Zw ) / Zr

figure(1)
step(ZbZr);
```

Fig 6.12 shows the step response for ZbZr.

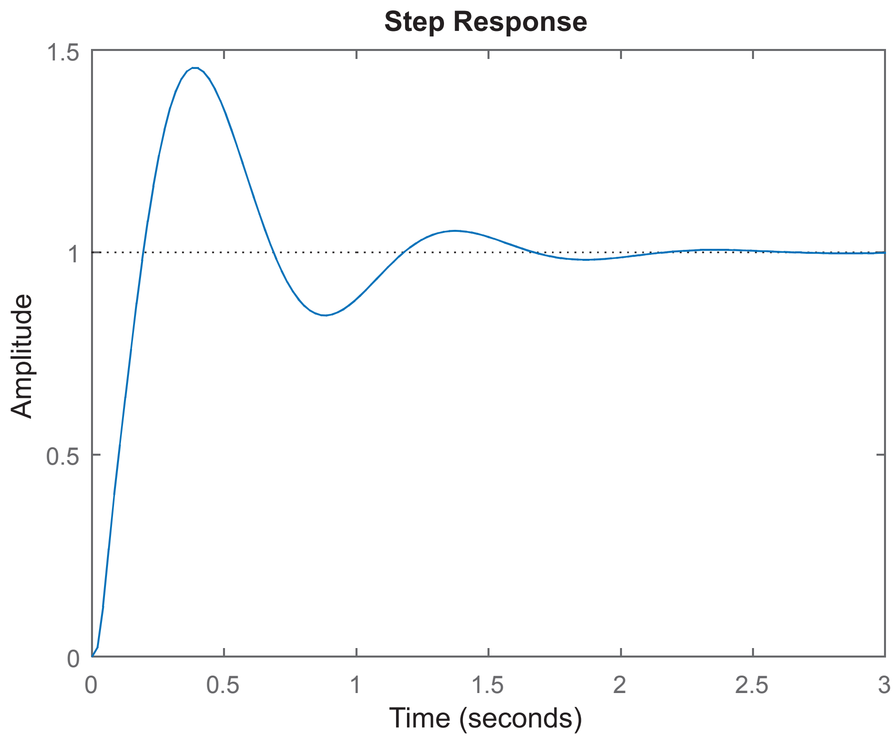


Fig. 6.12: Step response for ZbZr

```
wnb = sqrt(ks/mb) % nat freq body rad/s

wnt = sqrt(kt/mt) % nat freq tire rad/s

wnb =

    6.9574

wnt =

   62.2285


figure(2)
bode(ZbZr)
```

Fig. 6.13 shows the frequency response for ZbZr

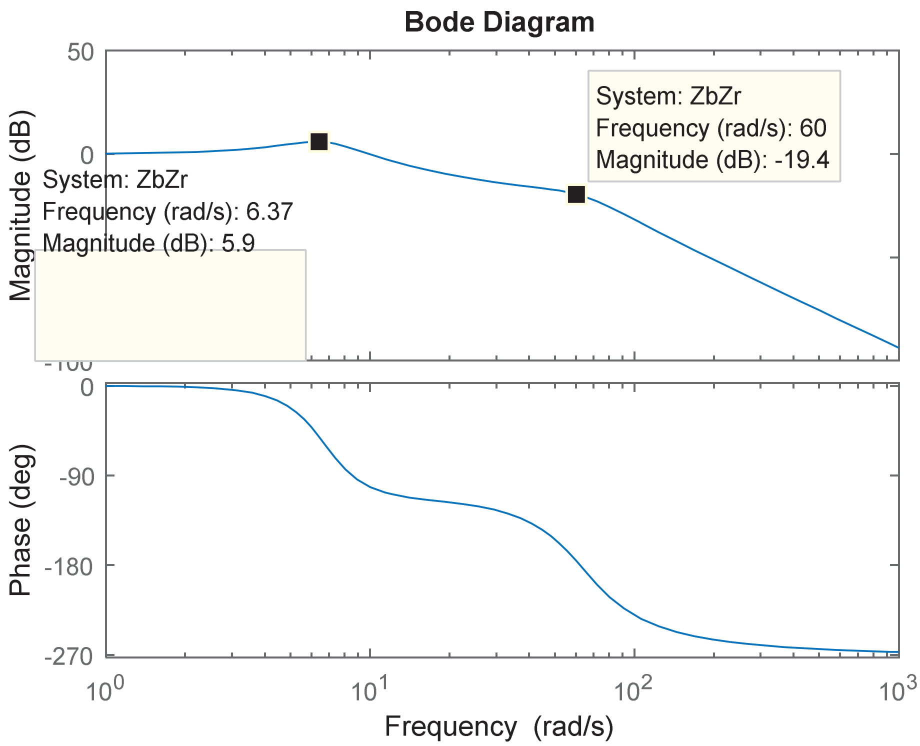


Fig. 6.13: Frequency response for ZbZr

```
% constant mb, increase mt

mt0 = mb*0.05;
mt2 = mb*0.20;
mt3 = mb*0.75;

wnt0 = sqrt(kt/mt0) % nat freq tire rad/s
wnt2 = sqrt(kt/mt2) % nat freq tire rad/s
wnt3 = sqrt(kt/mt3) % nat freq tire rad/s

% 1) mt affects C
C0 = tf([mt0 cs ks+kt],[1]);
```

```
C2 = tf([mt2 cs ks+kt],[1]);
C3 = tf([mt3 cs ks+kt],[1]);

ZbZr0 = D/(A*C0/B-B);
ZbZr2 = D/(A*C2/B-B);
ZbZr3 = D/(A*C3/B-B);

figure(2)
bode(ZbZr0,'r--');
hold on;
bode(ZbZr);
bode(ZbZr2,'g-.');
bode(ZbZr3,'k:');
legend('mt/mb = 0.05','mt/mb = 0.10','mt/mb = 0.20','mt/mb = 0.75')
title('Zb/Zr')
```

Fig. 6.14 shows the analysis at different mass ratio.

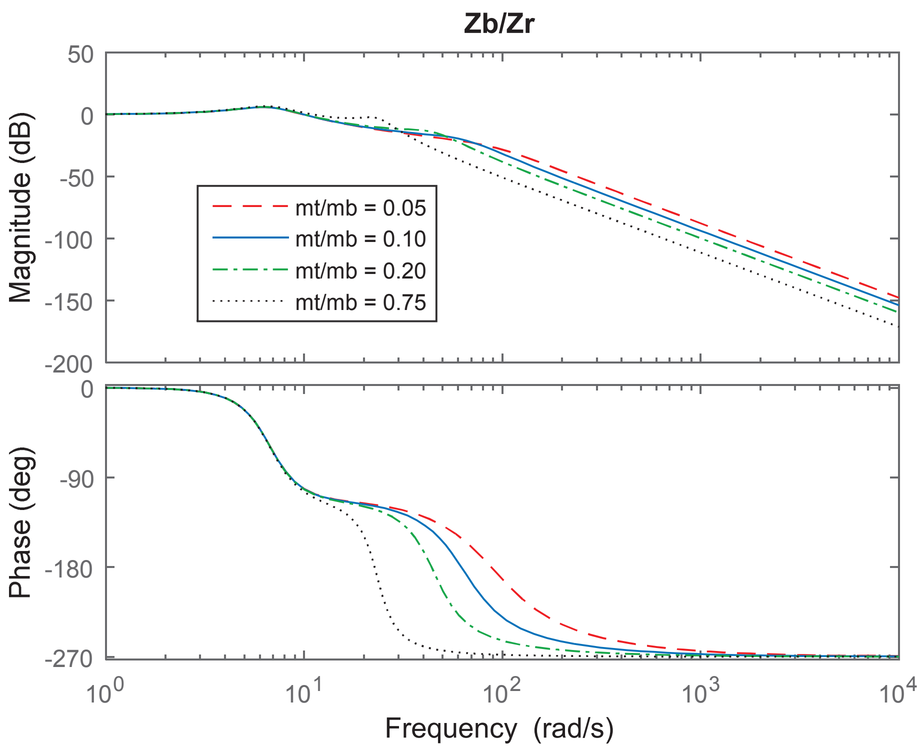


Fig. 6.14: Analysis at different mass ratio

```
% 2) change ks increase ks constant kt
ks0 = kt/5
ks2 = kt/8
ks3 = kt/10

% ks affects A, B, C
A0 = tf([mb cs ks0],[1]);
A2 = tf([mb cs ks2],[1]);
A3 = tf([mb cs ks3],[1]);

B0 = tf([cs ks0],[1]);
B2 = tf([cs ks2],[1]);
```

```
B3 = tf([cs ks3],[1]);

C0 = tf([mt cs ks0+kt],[1]);
C2 = tf([mt cs ks2+kt],[1]);
C3 = tf([mt cs ks3+kt],[1]);

ZbZr00 = D/(A0*C0/B0-B0);
ZbZr02 = D/(A2*C2/B2-B2);
ZbZr03 = D/(A3*C3/B3-B3);

figure(3)
bode(ZbZr00,'r--');
hold on;
bode(ZbZr02,'g-.');
bode(ZbZr03,'k:');
legend('kt/ks = 5','kt/ks = 8','kt/ks = 10')
title('Zb/Zr')
```

Fig. 6.15 shows the analysis a different spring stiffness ratio.

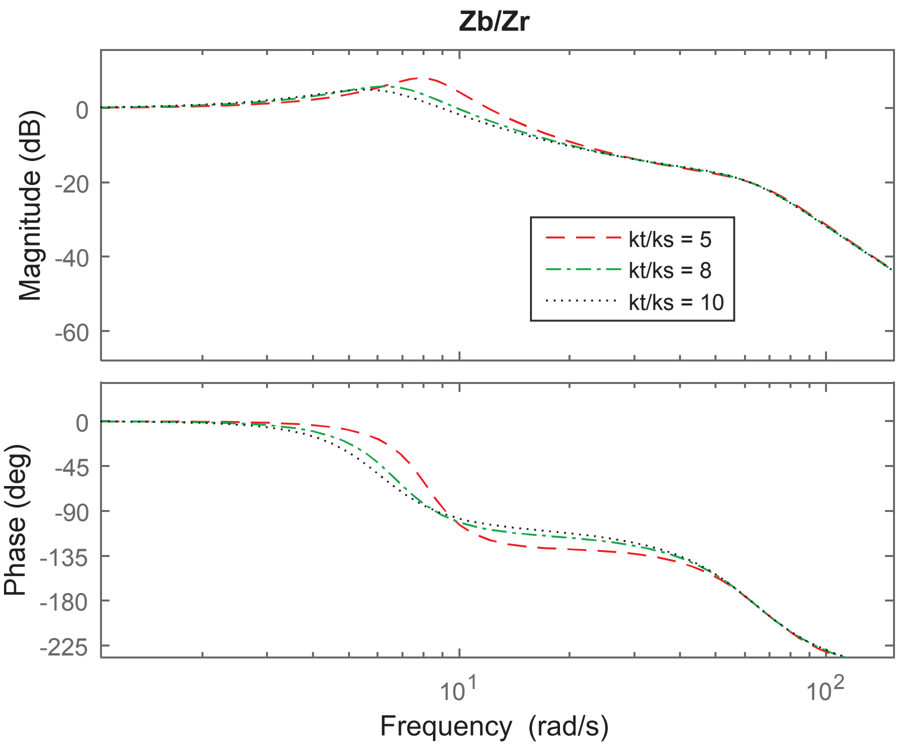


Fig. 6.15: Analysis at different spring stiffness ratio

```
% 3) change zeta increase zeta increase cs
% cs = 2*zeta*mb*wnb
cs0 = 2*0.1*mb*sqrt(ks/mb)
cs2 = 2*0.3*mb*sqrt(ks/mb)
cs3 = 2*0.707*mb*sqrt(ks/mb)

% cs affects A, B, C
A0 = tf([mb cs0 ks],[1]);
A2 = tf([mb cs2 ks],[1]);
A3 = tf([mb cs3 ks],[1]);
```

```
B0 = tf([cs0 ks],[1]);
B2 = tf([cs2 ks],[1]);
B3 = tf([cs3 ks],[1]);

C0 = tf([mt cs0 ks+kt],[1]);
C2 = tf([mt cs2 ks+kt],[1]);
C3 = tf([mt cs3 ks+kt],[1]);

ZbZr00 = D/(A0*C0/B0-B0);
ZbZr02 = D/(A2*C2/B2-B2);
ZbZr03 = D/(A3*C3/B3-B3);

figure(4)
bode(ZbZr00,'r--');
hold on;
% bode(ZbZr);
bode(ZbZr02,'g-.');
bode(ZbZr03,'k:');
legend('zeta = 0.1','zeta = 0.3','zeta = 0.707')
title('Zb/Zr')
```

Fig. 6.16 shows the analysis at different damping ratio.

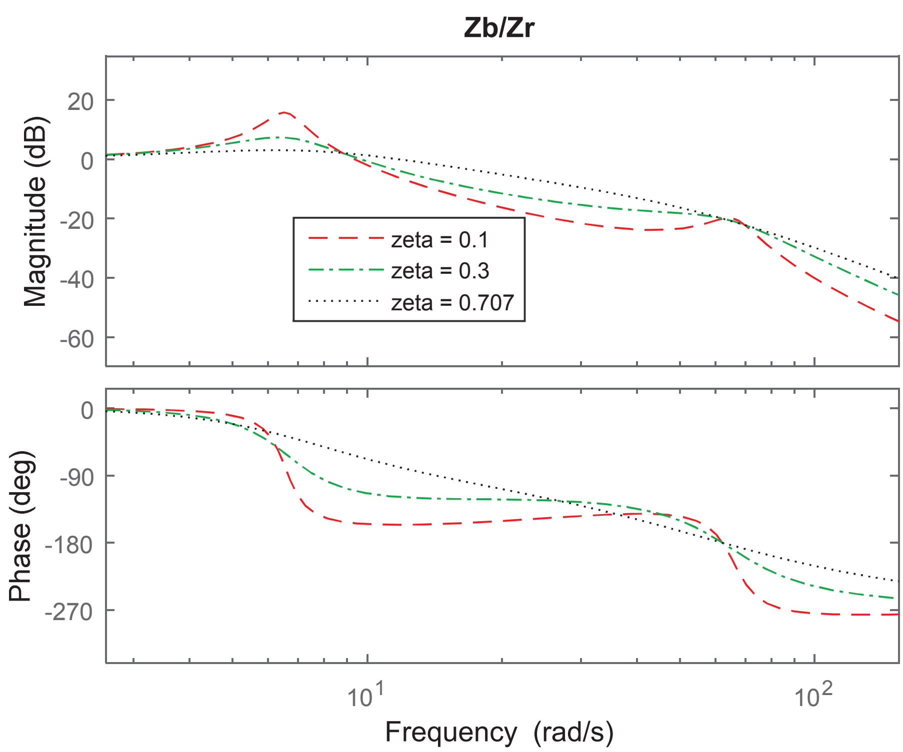


Fig. 6.16: Analysis at different damping ratio

Exercise

Find the responses for (Zw-Zb)/Zr (coding ZwZbZr) and (Zr-Zw)/Zr (coding ZrZwZr)
Analysis at different parameters (Zw-Zb)/Zr and (Zr-Zw)/Zr

CHAPTER 7

VEHICLE DYNAMIC MODELING TUTORIAL USING MATLAB/SIMULINK

7.1 Introduction

This chapter provides step by step tutorial of modelling the vehicle dynamic using MATLAB/Simulink coding. At the end of this chapter, students should be able to model the dynamic of a vehicle, in this case the 1DOF and 2 DOF quarter car model using Matlab/Simulink.

7.2 One degree of freedom system (1DOF)

Fig. 7.1 shows the 1DOF system.

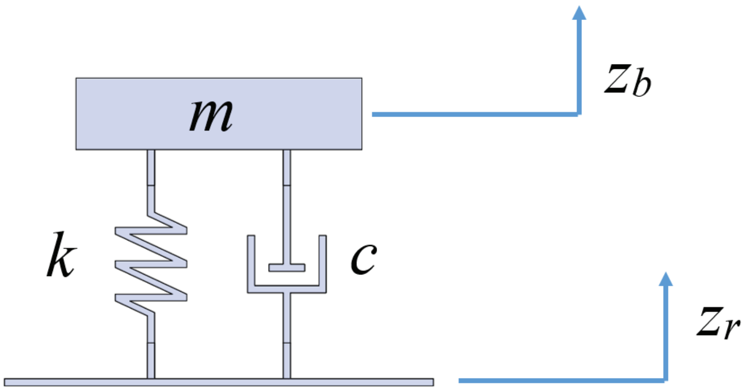


Fig. 7.1: 1DOF system

$$m\ddot{z}_b + c\dot{z}_b + kz_b - c\dot{z}_r - kz_r = 0$$

$$m\ddot{z}_b = -c\dot{z}_b - kz_b + c\dot{z}_r + kz_r$$

$$\ddot{z}_b = \frac{1}{m}(-c\dot{z}_b - kz_b + c\dot{z}_r + kz_r)$$

Modeling in Matlab/Simulink

Click Simulink Library button to start Simulink (**Fig. 7.2**)

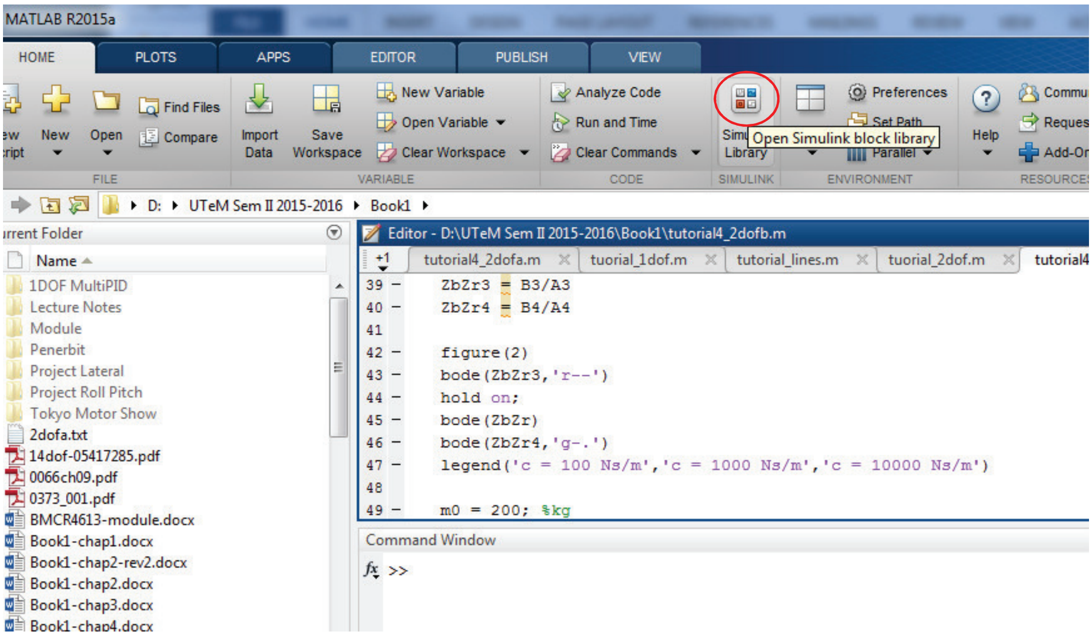


Fig. 7.2: Simulink block library icon

Click New Model icon → New Model to create a new Simulink model (**Fig. 7.3**)

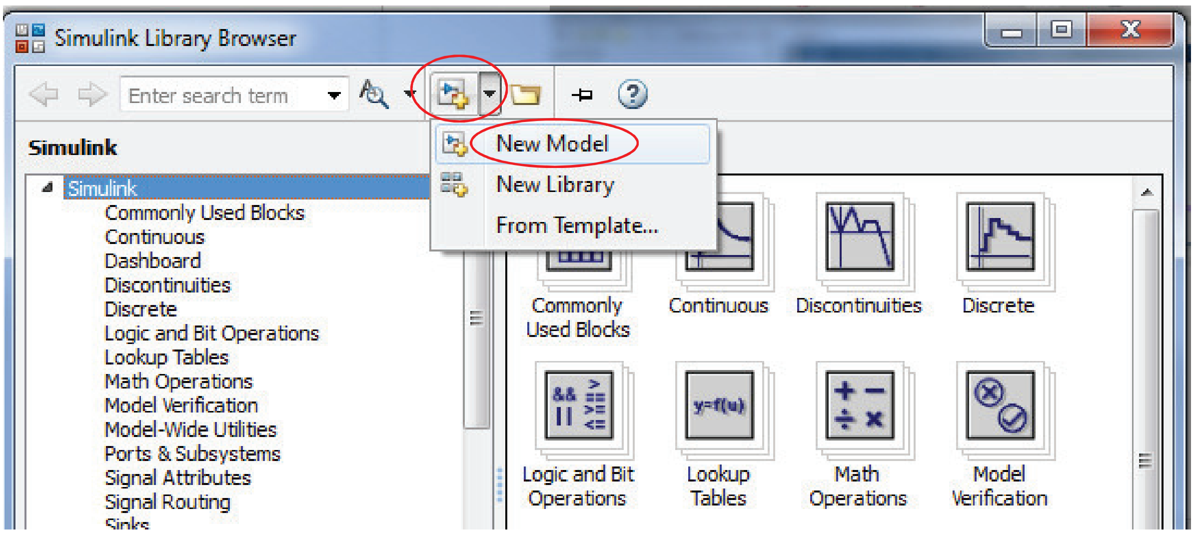


Fig. 7.3: New model for Simulink

New untitled Simulink model window (**Fig. 7.4**)

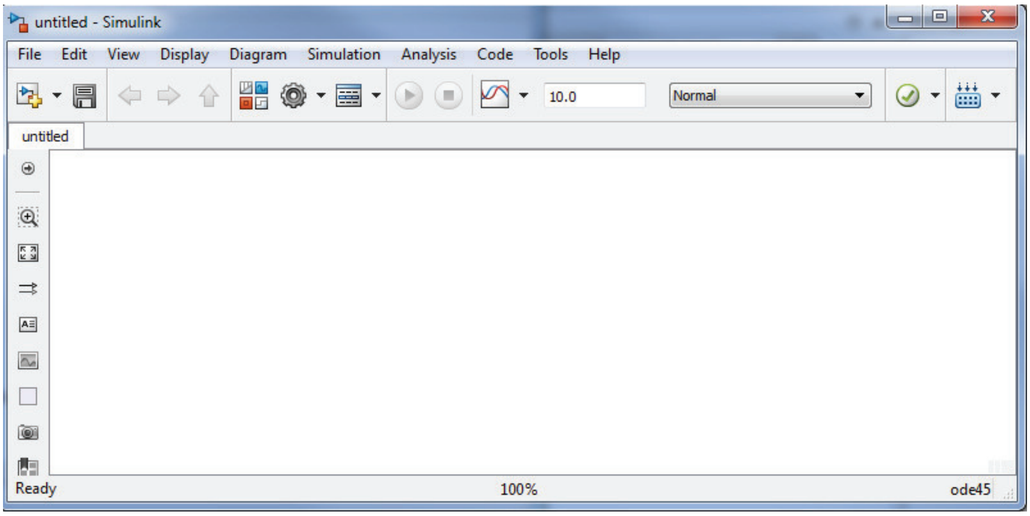


Fig. 7.4: New untitled window

Modeling the equation.

$$\ddot{z}_b = \frac{1}{m}(-c\dot{z}_b - kz_b + c\dot{z}_r + kz_r)$$

From the Simulink Library Browser window click Commonly Used Blocks (**Fig. 7.5**), click, hold and drag the Gain block to the new model window (**Fig. 7.6**).

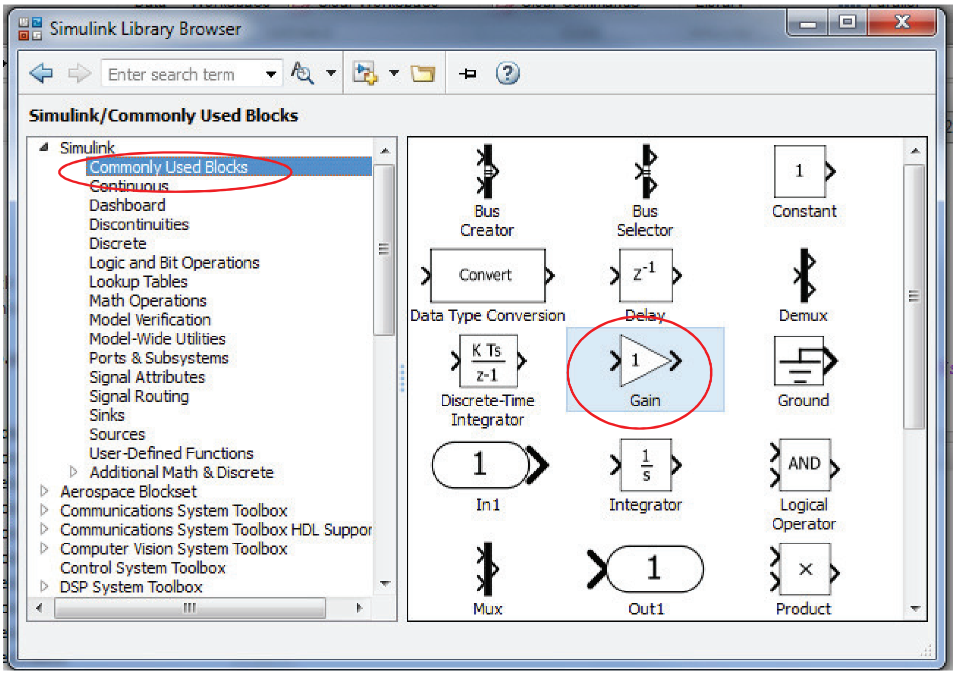


Fig. 7.5: Commonly used blocks

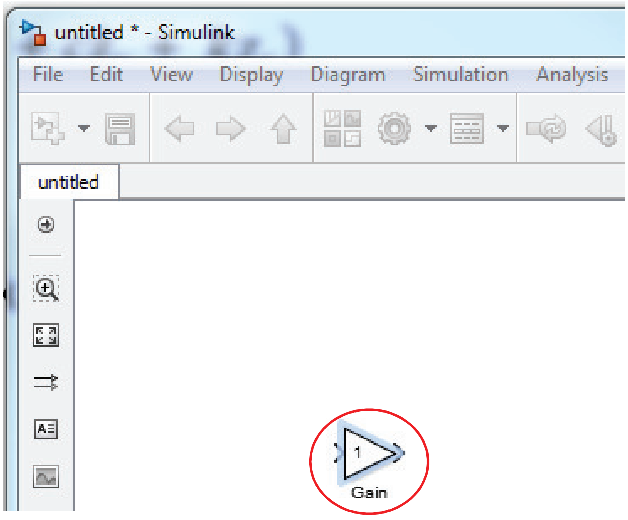


Fig. 7.6: New model window

Draw a line from the left point and right point of the Gain block. The line on the right represents $\ddot{z}_b$ and label the line as *zbdotdot* (**Fig. 7.7**).

$$\ddot{z}_b = \frac{1}{m}(-c\dot{z}_b - kz_b + c\dot{z}_r + kz_r)$$

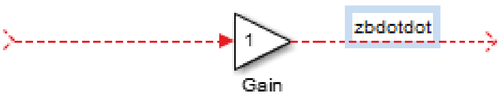


Fig. 7.7: Line zbdotdot

Double click the Gain block (**Fig. 7.8**) and change the Gain to 1/*m* and click Apply

$$\ddot{z}_b = \frac{1}{m}(-c\dot{z}_b - kz_b + c\dot{z}_r + kz_r)$$

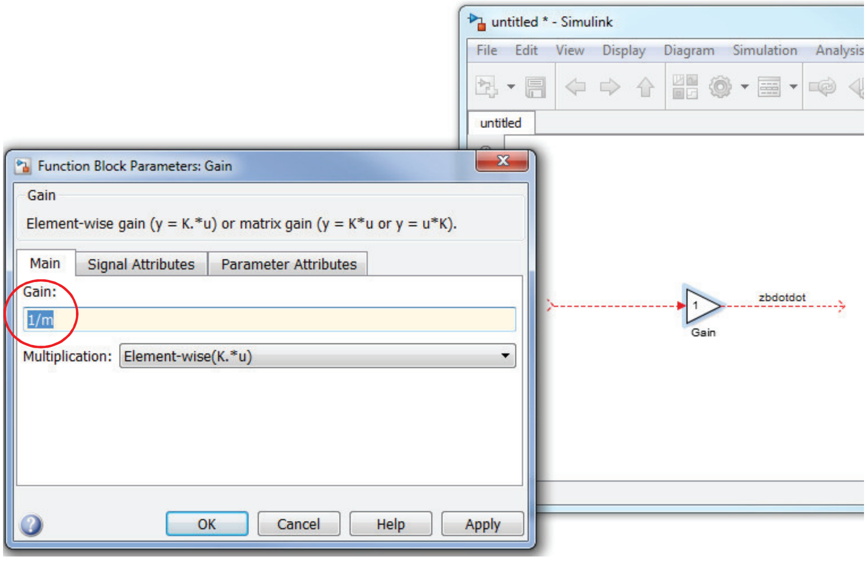


Fig. 7.8: Gain block parameters

Resize the Gain block to see the Gain parameter (**Fig. 7.9**)

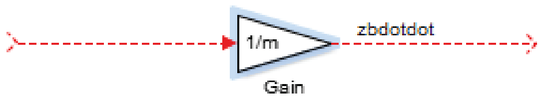


Fig. 7.9: Gain block

On the Simulink Library Browser, click Continuous and click, hold and drag the Integrator block to the model window.

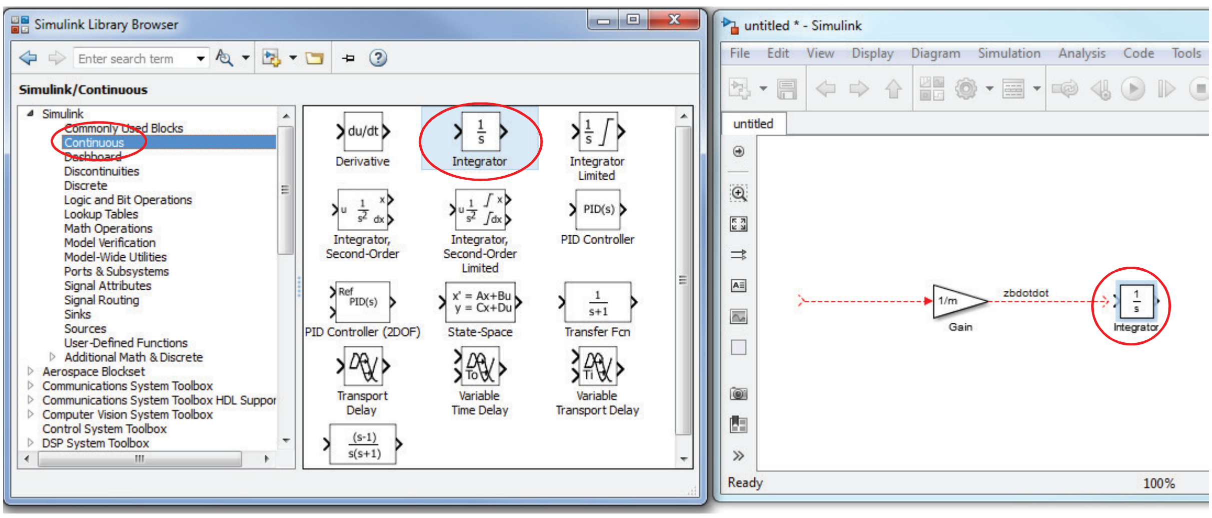


Fig. 7.10: Continuous

Connect the line zbdotdot to the Integrator block.

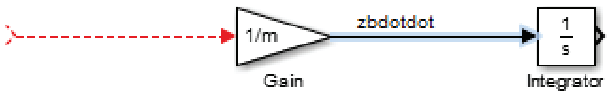


Fig. 7.11: Integrator

Click the Integrator block, copy and paste the block (control c and control v).

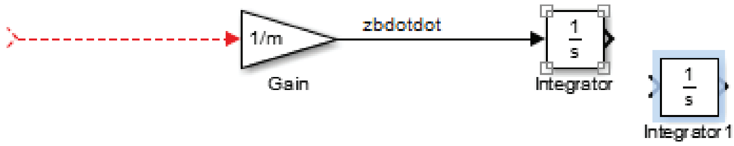


Fig. 7.12: Integrator blocks

Rearrange the block and connect a line between the Integrator blocks and draw a line on the right point of the second Integrator block. The line between the Integrator blocks represents $\dot{z}_b$ and label the line as zbdot. The line on the right point of the second Integrator block represent z_b and label the line as zb. Save the model as model_1dof.slx.

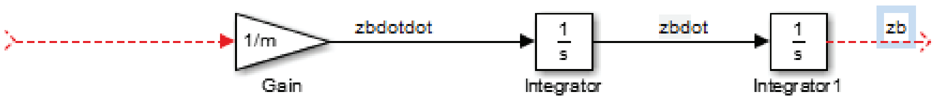


Fig. 7.13: Lines of zbdot and zb

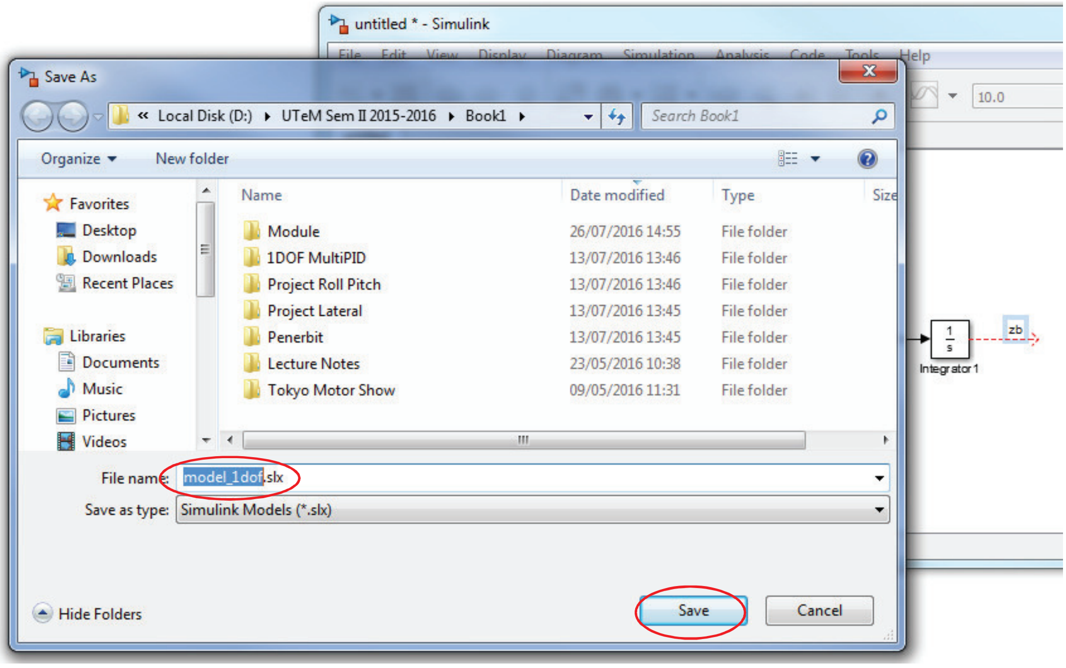


Fig. 7.14: Saving the model

Notice that the window title is changed from untitled* to mode_1dof and in the Model Browser the model name untitled is changed to model_1dof. The * on the window title means the file is not being saved yet. Please save the file as often as possible.

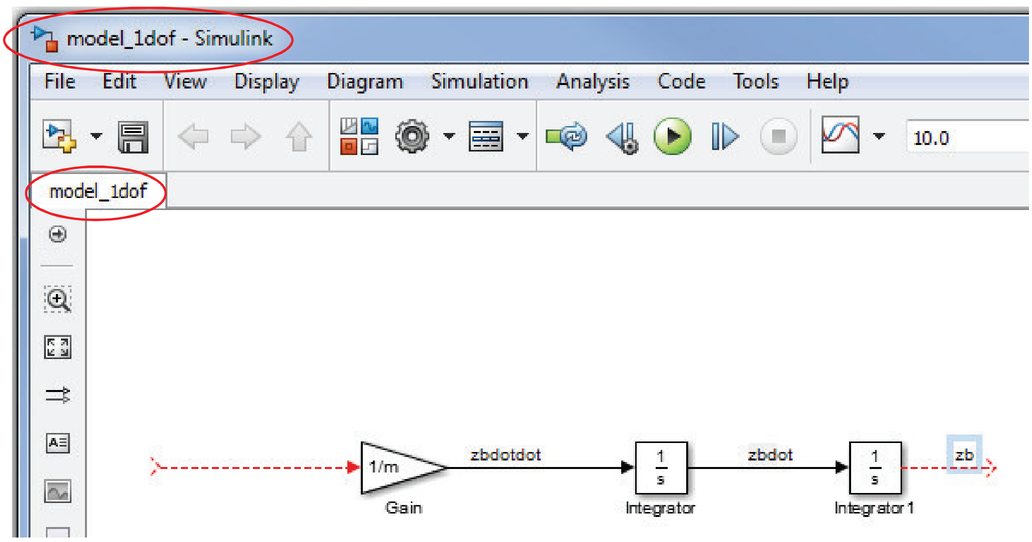


Fig. 7.15: Model file saved

From the Equation 1, there are 4 signs (terms) in a box bracket, 2 negatives and 2 positives.

$$\ddot{z}_b = \frac{1}{m}\boxed{(-c\dot{z}_b - kz_b + c\dot{z}_r + kz_r)}$$

In the Simulink Library Browser, click Math Operations, click, hold and drag Add block to the model_1dof window. Add block is used for submission (and subtraction) functions in an equation.

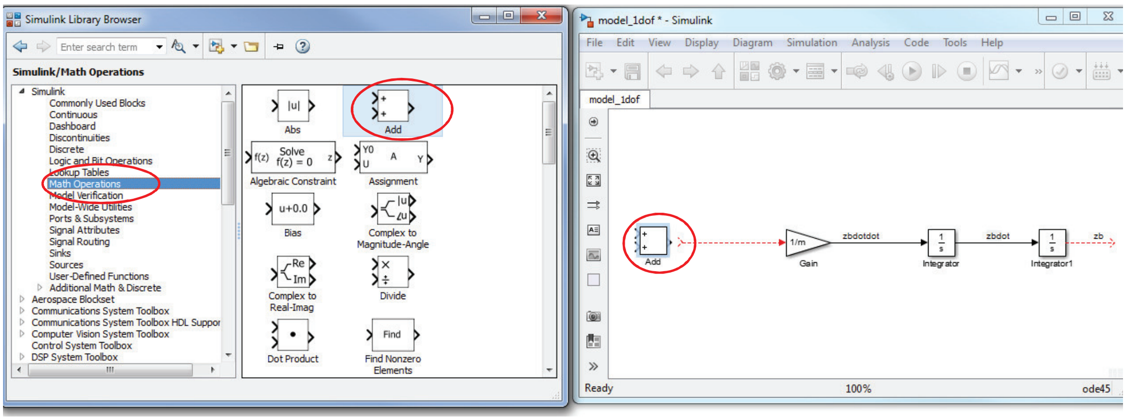


Fig. 7.16: Math operation blocks

Double click the Add block and add another two '-' signs in the List of signs and click Apply (**Fig. 7.17**).

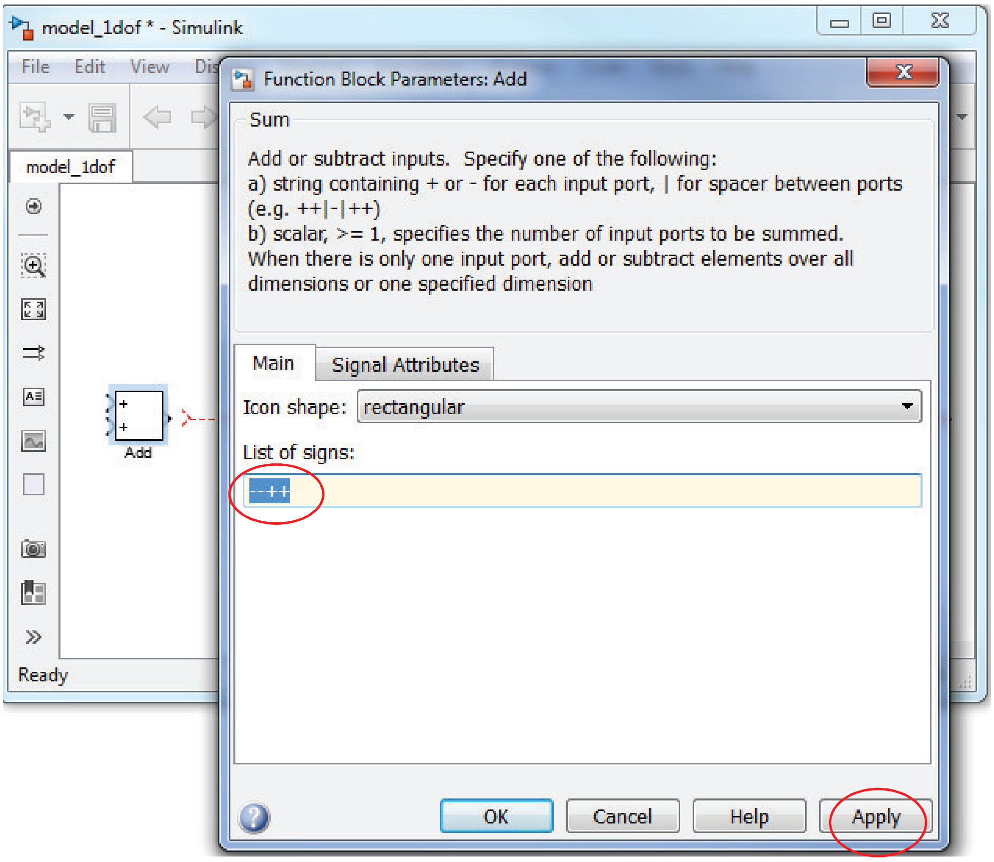


Fig. 7.17: Add block parameters

Enlarge the Add block and connect it to the Gain block (**Fig. 7.18**).

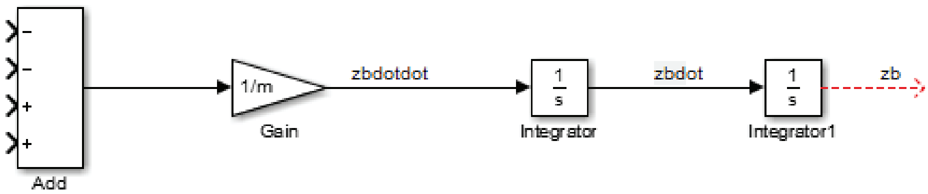


Fig. 7.18: Connections

The first – sign is for the first – term in the bracket of Equation 1 and so on and so forth. The constants *c* and *k* will be represented by a Gain blocks.

$$\ddot{z}_b = \frac{1}{m}(-c\dot{z}_b - kz_b + c\dot{z}_r + \, kz_r)$$

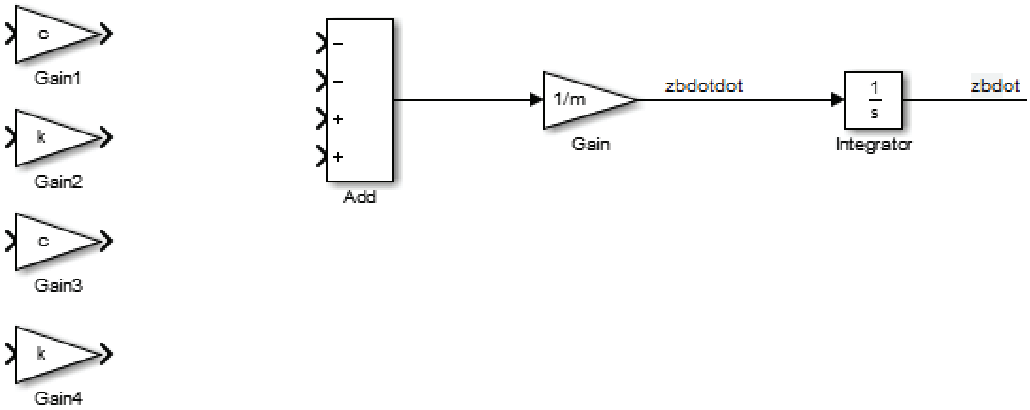


Fig. 7.19: Gain blocks

Connect the Gain blocks to the Add block. The line connected on the left side of *c* is zbdot and on the left side of *k* is zb.

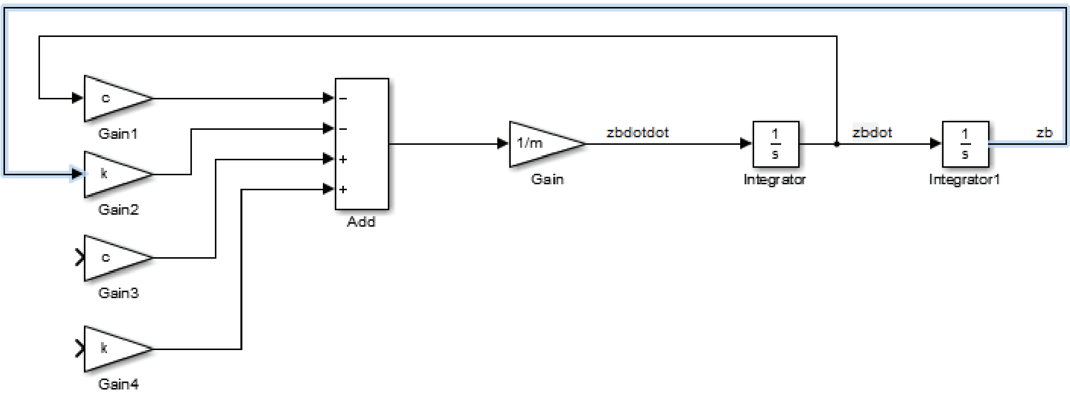


Fig. 7.20: Gain blocks connections

In this analysis, zr is the input to the system.

$$\ddot{z}_b = \frac{1}{m}(-c\dot{z}_b - kz_b + c\dot{z}_r + kz_r)$$

In the Simulink Library, click Sources. Click hold and drag the Step block into the model_1dof window.

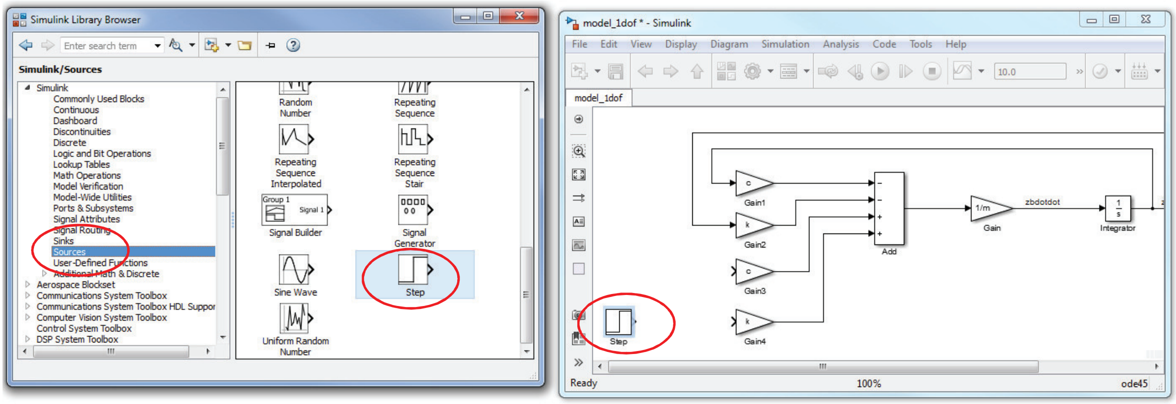


Fig. 7.21: Step input block

Connect the Step block to the 2nd Gain *k* block. Name the line as zr.

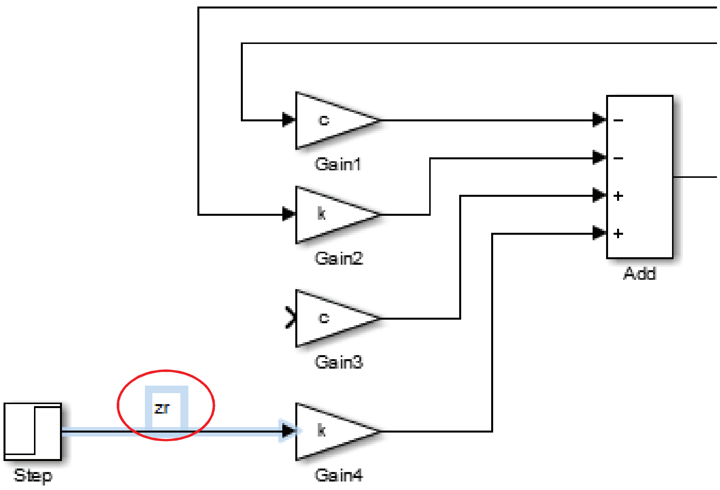


Fig. 7.22: Gain block connection

Double click the Step block and change the Final value to 0.1 and click OK or Apply. This means the Step input to the system is 0.1 m.

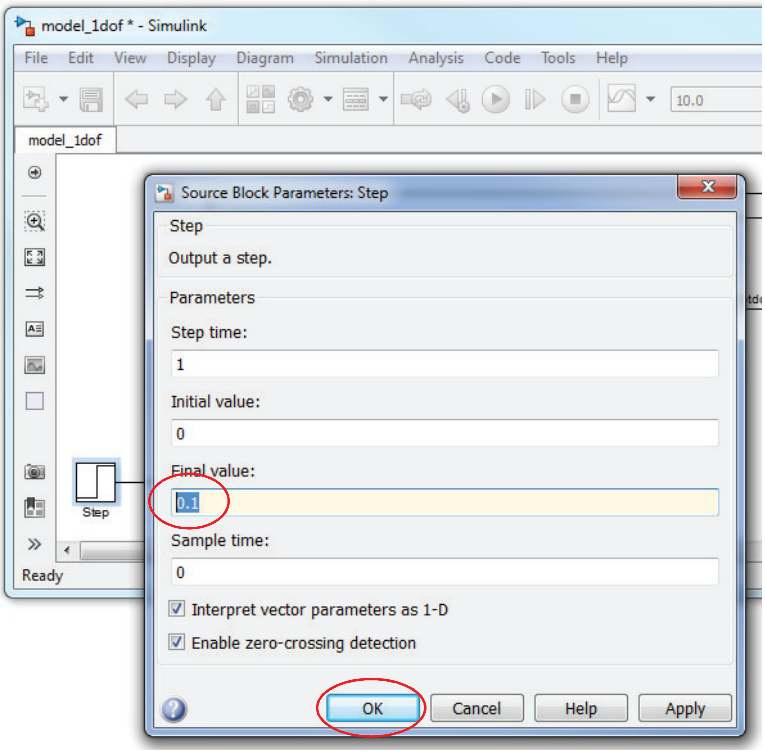


Fig. 7.23: Step input parameters

In the Simulink Library, click Continous. Click, hold and drag the Derivative block into the model_1dof window.

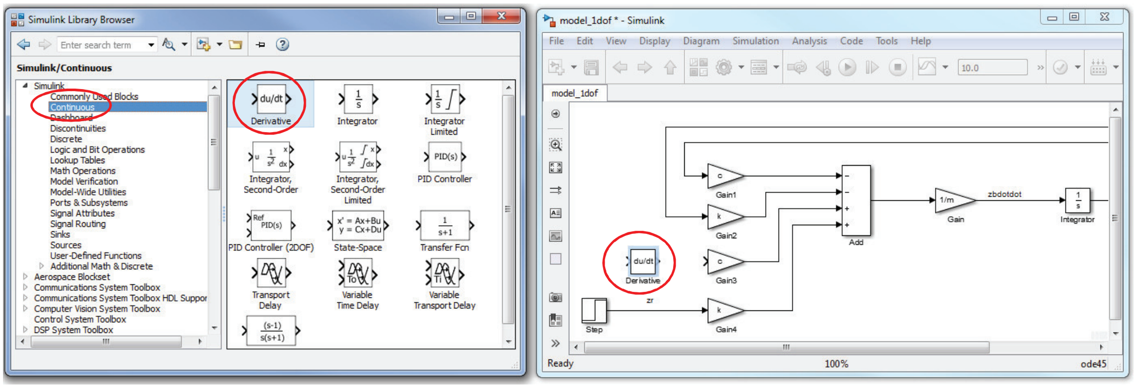


Fig. 7.24: Derivative block

Connect the line from the Step block to the Derivative block and proceed to 2nd Gain *c* block. Rename the line as zrdot.

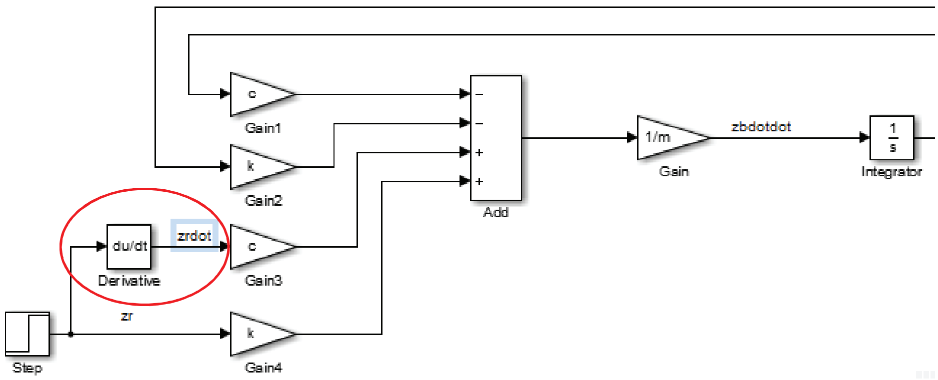


Fig. 7.25: Derivative block connection

In the Simulink Library, click Sinks. Click, hold and drag the Scope block to the model_1dof to observe the response.

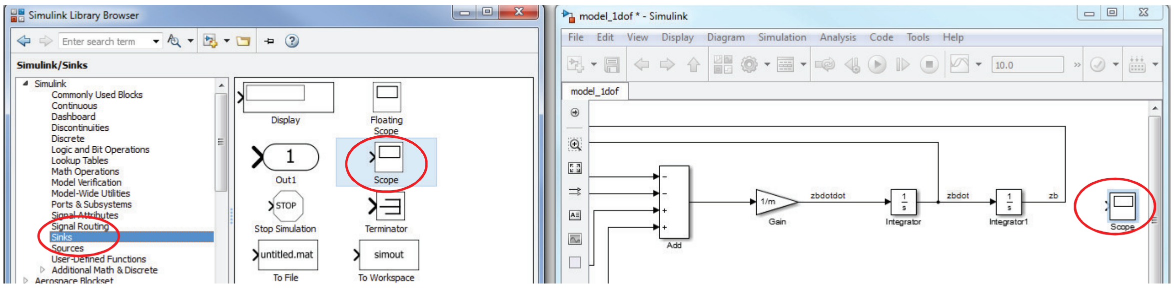


Fig. 7.26: Scope

Double click the Scope block, click Parameters icon and change the Number of axes to 2. Click Apply.

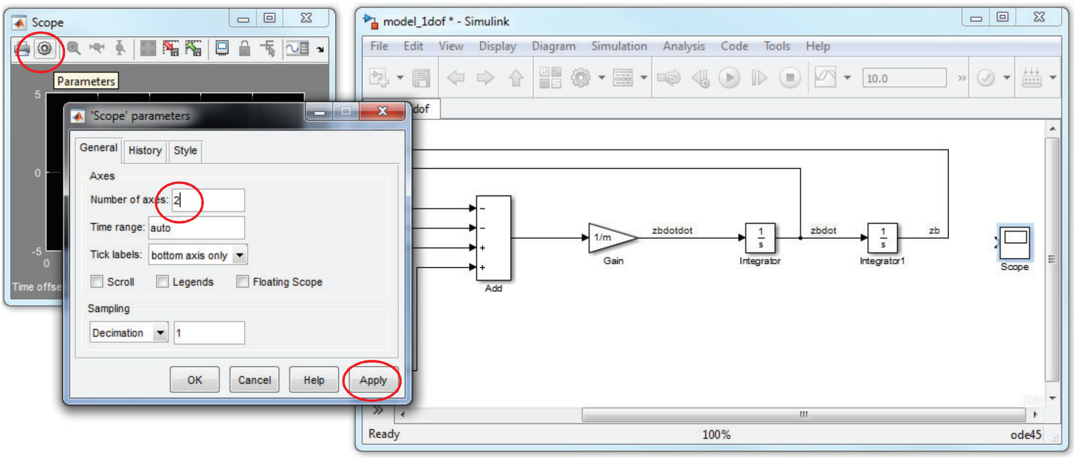


Fig. 7.27: Scope parameters

Connect the Scope to line zb as the output and zr as the input.

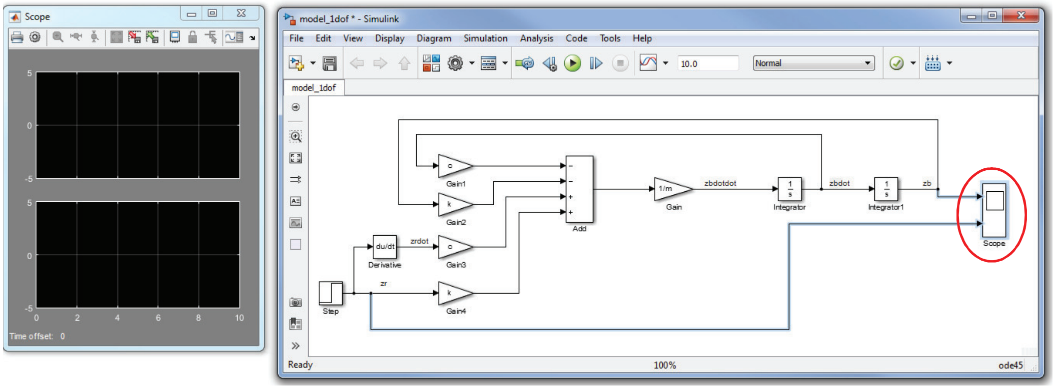


Fig. 7.28: Scope connections

Create an m-file in the Editor window with the 1DOF system parameters. Run the file before we run the model.

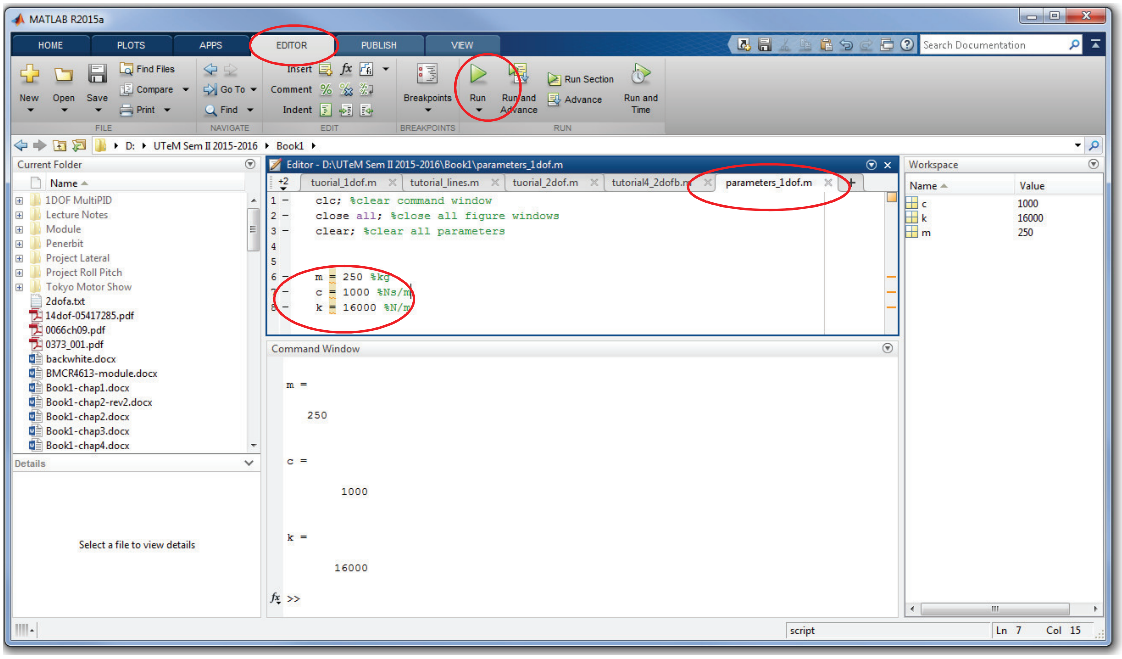


Fig. 7.29: File parameters_1dof

Run the model. Double click the Scope to see the result. Click Autoscale to enlarge the graph.

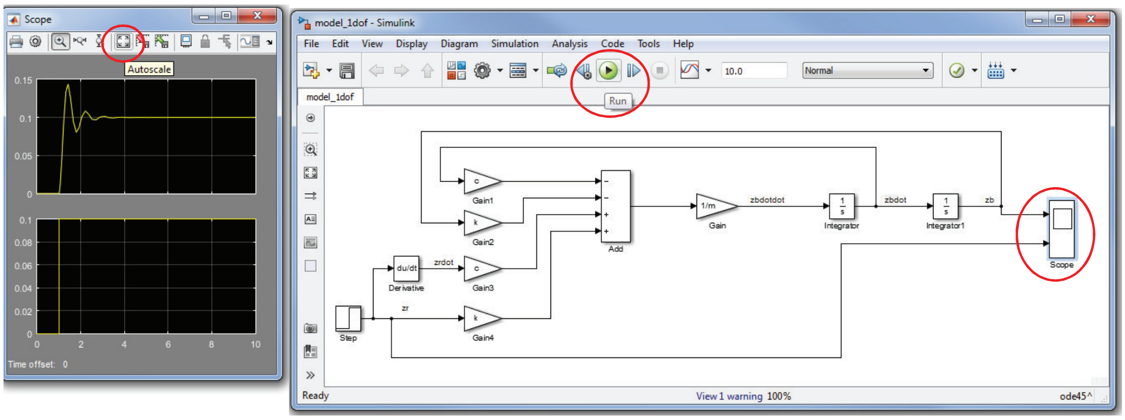


Fig. 7.30: Scope autoscale

The top graph is the output zb of the system and the bottom graph is the input zr of the system. Compare the result to step response from Chapter 6.

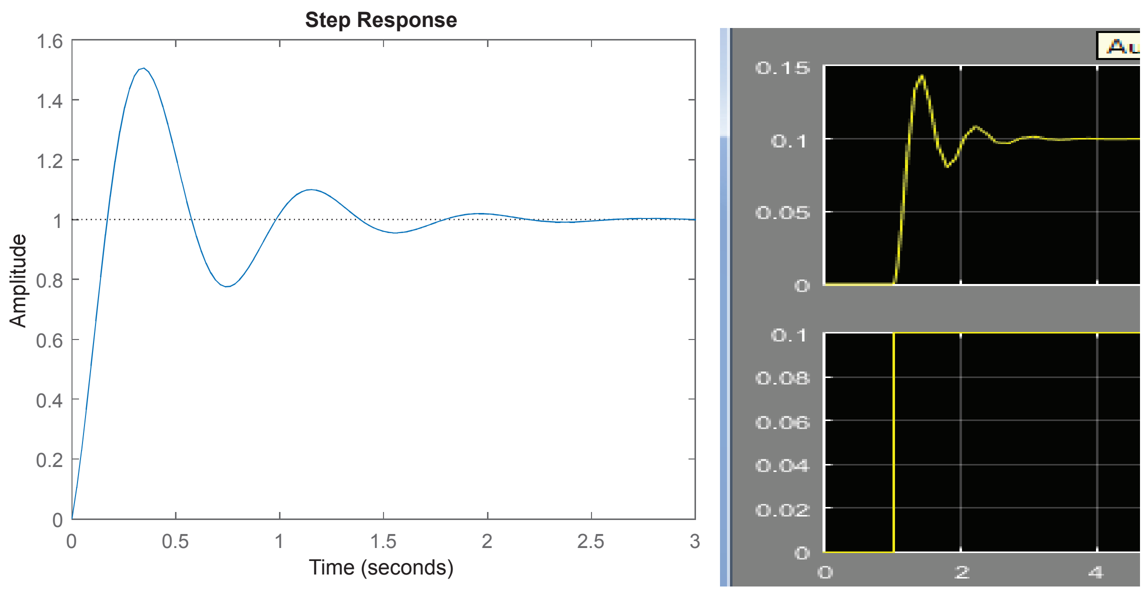
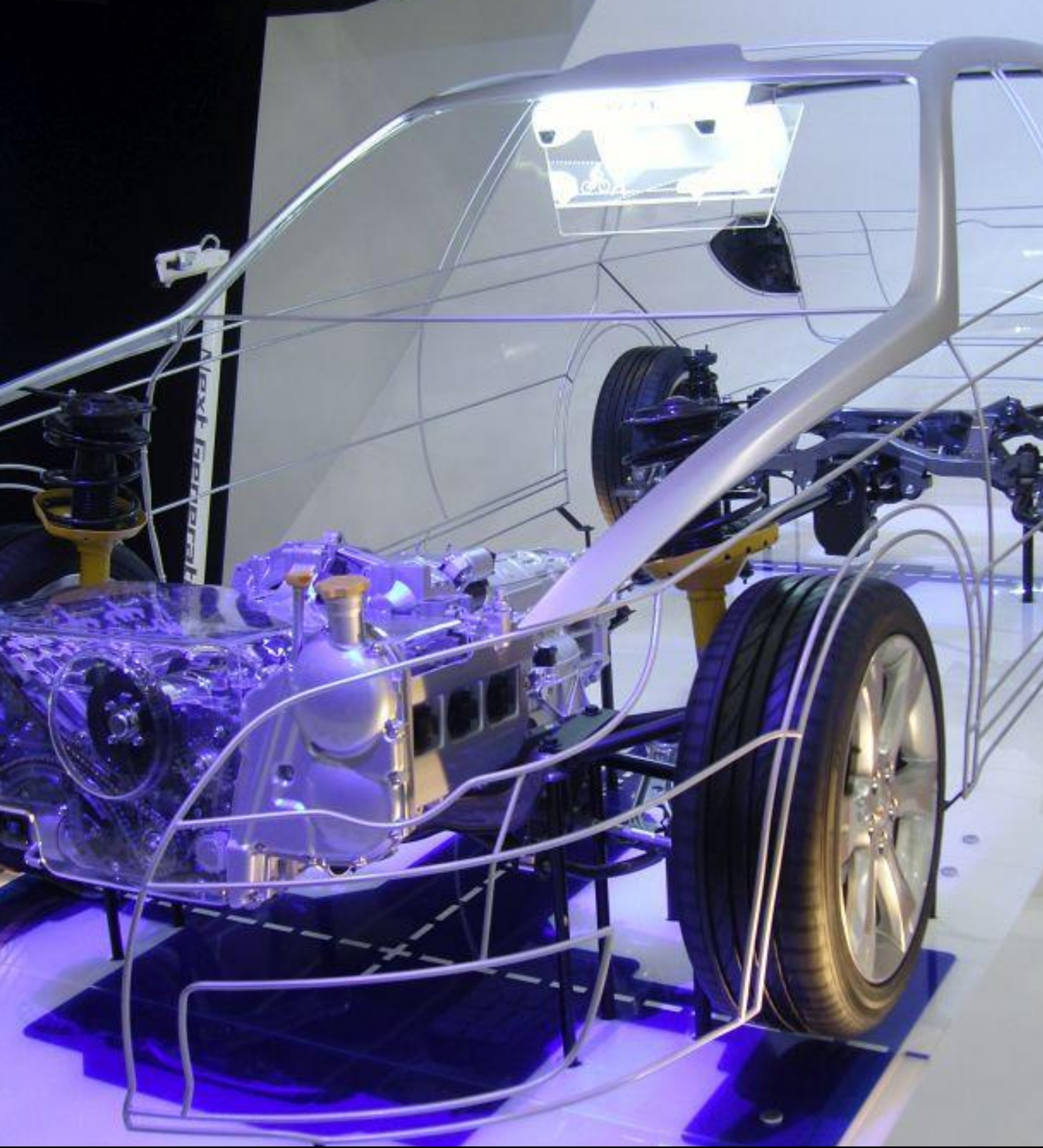


Fig. 7.31: Step response from Chapter 6 and Simulink

Exercise

1) 2DOF Simulink model.
2) 4DOF Simulink model.



UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CARe
CENTRE FOR ADVANCED RESEARCH ON ENERGY
Universiti Teknikal Malaysia Melaka

AcTiVe
Advanced Vehicle Technology

ISBN 978-967-0257-78-5

27th September 2016

View publication stats