VEHICLE SUSPENSION SYSTEM
USING SYSTEM IDENTIFICATION

By

ANUM SURAYA BINTI BAHADON

FINAL REPORT

Submitted to the Electrical & Electronics Engineering Programme
in Partial Fulfillment of the Requirements
for the Degree
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Universiti Teknologi Petronas
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

**VEHICLE SUSPENSION SYSTEM
USING SYSTEM IDENTIFICATION**

by

Anum Suraya Binti Bahadon

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Approved:

____

Dr. Vijanth Sagayan Asirvadam
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

December 2009

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

____

Anum Suraya Binti Bahadon

ABSTRACT

This report basically discusses the basic understanding of the chosen topic, which is **Vehicle Suspension System Using System Identification.** It is also discusses the findings on equation of the simplified single spring suspension. Initiation of this project started when physical modeling of the suspension system is costly and too time consuming. Therefore, a half front-rear suspension system simulation is needed to evaluate the optimum parameters to be used. The objective of the project is to find the best possible system identification model structures for the vehicle suspension system with the usage of Matlab System Identification Toolbox. The challenge is to get the possibilities of predicting the suspension model for the simulation of half front-rear suspension. The road surface will be introduced as inputs of the system that will be modeled. The simple model of suspension is used as a reference for further research on the half front-rear suspension system. The basic suspension's equation for quarter-car model is determined to develop a Matlab simulation block diagram. From the quarter-car model, the system is developed to half front-rear model for further analysis. With the half front-rear model simulation, all the parameters which also include the Proportional Integral Derivative (PID) controller gain of the system can be tested. Simulation study shows second order of Auto Regression with Extra Input (ARX) model gives the most promising result in identifying the system.

ACKNOWLEDGMENT

First of all, I would like to express my utmost gratitude and appreciation to Allah because with His blessings and help, my Final Year Project went very smoothly for me. Alhamdulillah, all praises to Him that I have been able to complete this project on time.

My special thanks goes to my supervisor, Dr Vijanth Sagayan Asirvadam for giving me this opportunity and believing that I can contribute and work for this project while guiding, helping and motivating me along the way. Without his help, none of these would be possible. Not forgetting Mr. Afifi bin Md. Akib, post graduate Electrical and Electronics student who have helped me in any matter. Your help is very much appreciated.

Last but not least, I would like to thank my family and friends for always supporting me throughout the completion of this project. Thank you very much to everyone.

TABLE OF CONTENTS

ABSTRACT……………………………………………………………..…….………......iv

ACKNOWLEDGEMENT……………………………………………..………………....…v

LIST OF FIGURES…………………………………………………………………..vii

LIST OF TABLES…………………………………………………………....…...x

CHAPTER 1 INTRODUCTION 1

1.1 Background of Study 1

1.2 Problem Statement 1

1.3 Objective and Scope of Study 2

CHAPTER 2 LITERATURE REVIEW 3

2.1 Theory 3

2.2 Mode of operation 3

2.3 Independent Suspension System 4

CHAPTER 3 METHODOLOGY 5

3.1 Procedure Identification 5

3.1.1 System Identification Toolbox 6

3.1.2 Modelling Approach 6

3.1.3 Model of Suspension 8

CHAPTER 4 RESULTS AND DISCUSSION 14

4.1 Simulation for the Quarter-car Model 14

4.2 Simulation for the Half-car Model 15

4.3 System Identification for Half Model 21

4.3.1 Results for the front suspension 21

4.3.2 Results for the rear suspension 23

CHAPTER 5 CONCLUSION AND RECOMMENDATION 26
5.1 Conclusion 26
5.2 Recommendation.. 26

REFERENCES.. 27
APPENDICES... 29
Appendix A MATLAB COMMAND AND M-FILE CODING........... 30
Appendix B SIMULATION BLOCK DIAGRAM............................... 37
Appendix C FINAL YEAR PROJECT GANTT CHART.................... 41

LIST OF FIGURES

Figure 1: Suspension system in a vehicle 3

Figure 2: Methodology Flow Chart 5

Figure 3: System with input (*u*) and output (*y*) 7

Figure 4: The quarter car model 9

Figure 5: Free-body diagrams for suspension system.. 9

Figure 6: The two DOF schematic of half front-rear model 10

Figure 7: Graph for *x*-plot 14

Figure 8: Graph for *y*-plot 15

Figure 9: Road disturbance as input 16

Figure 10: Bounce motion of the suspension 16

Figure 11: Pitch motion of the suspension...17

Figure 12: Bounce and pitch motion of the suspension after the damper gain is increased by 10 17

Figure 13: Bounce motion using the PID controller..18

Figure 14: Pitch motion using the PID controller..18

Figure 15: Bounce motion using the PID controller when front and rear suspension are connected to each other 19

Figure 16: Step response of bounce motion using the PID controller....................20

Figure 17: Step response of pitch motion using the PID controller 20

Figure 18: First Order AR, ARX and OE model of bounce motion and prediction error 21

Figure 19: Second Order AR, ARX and OE model of bounce motion and prediction error 22

Figure 20: Third Order AR, ARX and OE model of bounce motion and prediction error 22

Figure 21: First Order AR, ARX and OE model of bounce motion and prediction error 23

Figure 22: Second Order AR, ARX and OE model of bounce motion and prediction error 24

Figure 23: Third Order AR, ARX and OE model of bounce motion and prediction error 24

LIST OF TABLES

Table 1: Bounce motion prediction error for AR, ARX and OE models of front suspension……………………………………………......23

Table 2: Bounce motion prediction error for AR, ARX and OE models of rear suspension…………………………..……………......….25

CHAPTER 1
INTRODUCTION

1.1 Background of Study

Suspension is widely used in automotive industry and lots of design has been invented to improve the system. The history of suspension has taken its place since the early Egyptians where by leaf springs are introduced. By the early 19th century, most British horse carriages were equipped with springs where wooden springs is used in the case of light one-horse vehicles to avoid taxation, and steel springs in larger vehicles [1]. These were made of low-carbon steel and usually took the form of multiple layer leaf springs. Basically, suspension system has been used for comfort and convenience of the vehicle's occupants.

1.2 Problem Statement

No road surface is completely even. Apart from rotational movement, the wheels of the vehicle also have to move up and down. During higher speeds these movements are carried out in a short period of time. Therefore, large impact forces act upon the vehicle, which is depending on the move mass of the vehicle. Various researches had been done to produce the most compatible suspension system to over come these problems including independent front and rear suspension. Independent suspension is belief to be more stable than the standard suspension. However, there are few disadvantages for this system.

The disadvantages when using the independent suspension are as the following;

1) Bad handling
2) Space engaging and costly

Therefore, a simulation of suspension model is needed to predict the system output since the physical modelling is costly and too time consuming due to the difficulty to find physical laws that govern its behavior.

1.3 Objective and Scope of Study

The main objectives of this research are:

1) To design a simulation of half front-rear suspension system that is at better handling using Matlab Simulink.
2) To simulate the presence suspension system using Matlab Simulink.
3) To predict the suspension model stability using System Identification Toolbox.

The research will be focusing more on the innovation of suspension system which can give optimum result for occupants comfort. In order to create a properly functioning suspension system, mathematical computations must be calculated. Factors such as spring rate, wheel rate, weight transfer, unsprung weight transfer, travel, and damping must be considered. If even the slightest miscalculation should occur, a vehicle and its occupants could be greatly in danger.

CHAPTER 2
LITERATURE REVIEW

2.1 Theory

Suspension in its simplest explanation is a system of shock absorbers, linkage, and springs that connect the wheels to a vehicle. The main purpose of suspension is to keep the occupants of the vehicle comfortable as they travel over bumps and obstacles on the roadway. Besides, it is also to enhance the handling capabilities. The suspension has a task together with the shock absorber to absorb the impact of the road and transfer the vibrations.

2.2 Mode of operation

The suspension allows the vehicle to become a vibratory form, with the vehicle weight and also it determined the special amount of vibration. Other than the road surface impacts, other forces can act upon the vehicle too, such as driving force, braking forces, wind force and centrifugal forces.

Figure 1**:** Suspension system in a vehicle

The braking and handling of the vehicle is kept safely in check by the suspension systems by giving spring to the entire vehicle which allows the driver to maneuver more efficiently [2]. Likewise, the occupants of the vehicle are protected from the harsh vibrations, road noise, and bumps. The suspension systems also help to protect the vehicle and the cargo inside from wear and damage.

2.3 Independent Suspension System

Independent suspension is a suspension that allows each wheel on the same axle to move vertically independently of each other such as reacting to a bump on the road. With such independent suspension systems, the suspension components win normally possess a geometry which enables each respective suspended wheel to compensate for roll of the chassis structure during cornering and still remain relatively perpendicular to the road surface so maintaining the tyre-to-road contact patch [3]. In order to achieve this result the camber angle of each wheel has to change with respect to the chassis structure as the suspension is compressed or extended during cornering.

CHAPTER 3
METHODOLOGY

3.1 Procedure Identification

This project will be done using the System Identification Toolbox. By using this toolbox, mathematical models of dynamic systems from measured input-output data can be conducted. The following chart shows the methodology flow of this project.

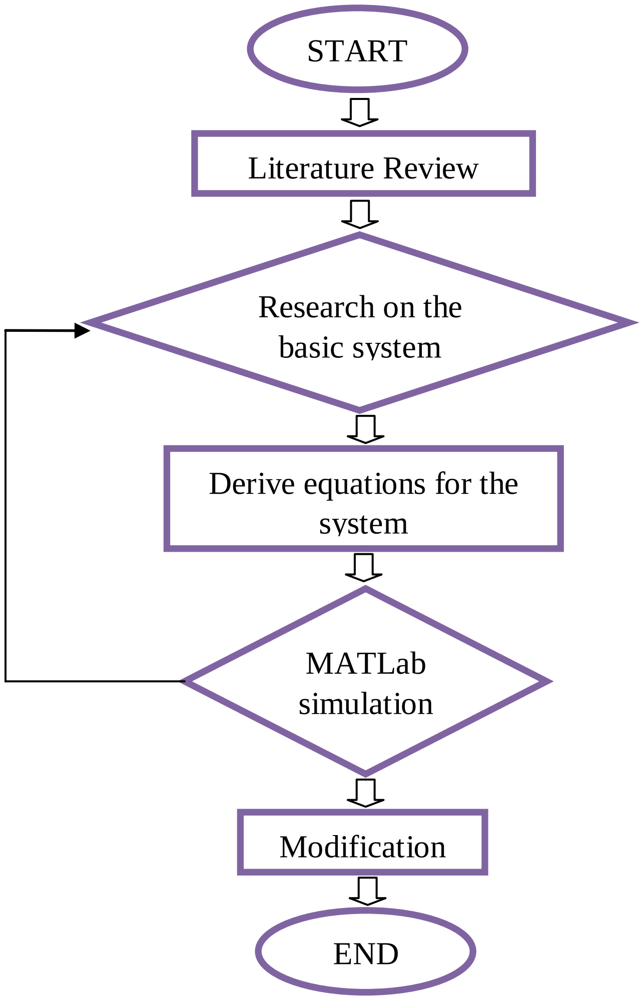


Figure 2: Methodology Flow Chart

3.1.1 System Identification Toolbox

System Identification Toolbox facilitates the multistep process of identifying models from data. The toolbox can:

- Analyze and process data
- Determine suitable model structure and order
- Estimate model parameters and validate model accuracy
- View the model responses and their uncertainties

These tasks can be performed by using either command-line functions or a graphical user interface (GUI).

The models can be converted into linear time-invariant (LTI) objects for use with Control System Toolbox. Most identified models can be incorporated into Simulink models using blocks provided by the toolbox [4].

3.1.2 Modelling Approach

In many engineering circumstances there is a need to model the system dynamic, such circumstances might be the need to achieve deeper knowledge about the system, simulate the system behavior under certain conditions or to predict the system output at the future [5].

The models can be constructed from physical laws and principles, which are known as physical modelling. When the system is complicated and difficult to find the physical laws that reflects its behavior, on the other hand, physical modeling is considered too costly and time consuming. In such cases, the alternative option is to model the system using system identification approaches [6].

System identification deals with the constructing models with dynamical systems. Figure 3 illustrates the observed output signal (y) and the corresponding input signal (u) of a system.

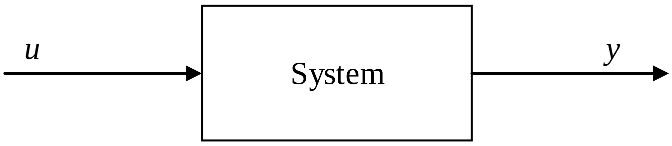


Figure 3: System with input (u) and output (y).

The system used can be any system to be predicted as long as the input and the output of the system are available. First, the input and output data are collected by either applying a certain input sequence directly on the system or its simulation; when it is difficult to apply in the real system; and record both the input and the counterpart output for a certain period of time [5]. It must be noticed during this stage that the input signal must be rich enough to excite the dynamic of the system.

The next stage is to select the model structure, where in this project the model structures that are used are Auto-Regressive (AR), Auto-Regressive with External Input (ARX) and Output Error (OE). The ARX is the most used model structure in system identification. In ARX model the past input and output are used for modeling the system. The general Single Input Single Output (SISO) ARX model can be expressed by the following linear difference equation:

$$y_t = a_1 y_{t-1} + a_2 y_{t-2} + \ldots + a_n y_{t-n} + b_1 x_{t-1} + b_2 x_{t-2} + \ldots + b_n x_{t-n} + e_t \qquad \ldots(1)$$

where x(t) and y(t) are the input and output of the SISO ARX model respectively. The n_y and n_x are the number of past outputs and the number of past inputs used in the model and n_k is the pure time delay or the dead time in the system. The coefficients $a_1 \ldots a_{ny}$ and $b_1 \ldots b_{nx}$ are known as the model parameters.

The Auto-Regressive (AR) model is only using the output for modeling the system. AR model does not consider any relevant inputs in turn; it cannot be expected to yield accurate results. The general SISO AR model can be expressed by the following linear difference equation:

$$y_t = a_1 y_{t-1} + a_2 y_{t-2} + \ldots + a_n y_{t-n} + e_t \qquad \ldots(2)$$

Another model is called Output Error (OE) model which based on the output from other model or prediction from other model. In this case, the OE model is using the ARX prediction output and improved them with a new prediction model. OE model can be expressed with the linear equation same with the SISO ARX equation as the structure is the same. However, the reference data is based on the ARX model's data not from the actual model:

$$y_t = a_1 y_{arxt-1} + a_2 y_{arxt-2} + \ldots + a_n y_{arxt-n} + b_1 x_{t-1} + b_2 x_{t-2} + \ldots + b_n x_{t-n} + e_t \qquad \ldots(3)$$

The final stage is to apply these models to the simulated suspension system and find which structures give the best representation of the system. The structure choosing is depends on the nature of the system, thus a knowledge about it will help in choosing the appropriate structure while the determination of the model order can be done by trial and error.

3.1.3 Model of Suspension

For the model of suspension, first simulation of a quarter-car model is developed and tested. After the performance of the quarter-car model is satisfied then, the simulation of half front-rear model is developed and tested with the system identification.

Quarter-car model

The equations of motion for the automobile and the wheel motion assuming one-dimensional vertical motion of one quarter of the car mass above one wheel. Figure 4 shows the illustration of the simplified single quarter-car suspension.

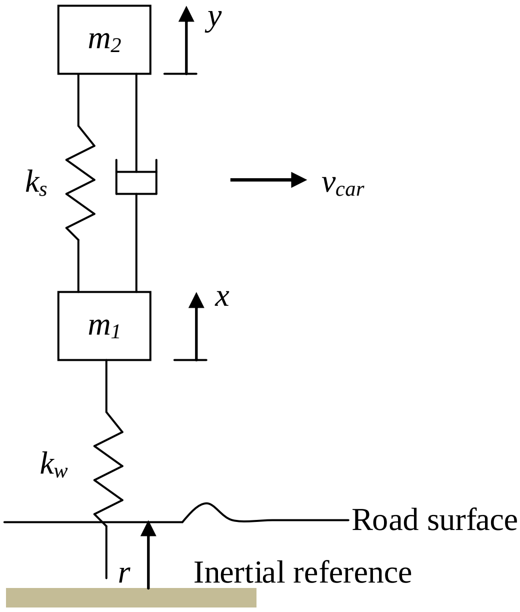


Figure 4: The quarter-car model

The car mass is assume to be 1580 kg and including the wheel's mass of 20 kg each. The spring constant in this model is k_s = 130, 000 N/m and the wheel constant k_w ≈1, 000,000 N/m [7]. The bumpy road is being indicated by r which is not a constant value. Finally, the equation of the simplified single spring suspension is retained from the free body diagrams as shown in Figure 5.

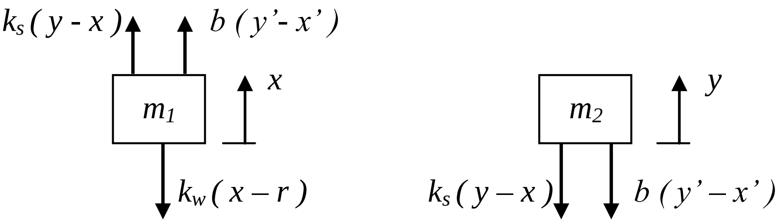


Figure 5: Free-body diagrams for suspension system

The equation obtains are:

$$m_1\, x'' + b\,(x' - y') + k_s\,(x - y) + k_w x = k_w r \qquad \ldots(4)$$

$$m_2\, y'' + b\,(y' - x') + k_s\,(y - x) = 0 \qquad \ldots(5)$$

where x is the deflection of the suspension and y is the deflection of the vehicle's body which will be observed in the Matlab simulation. The input of the system, r, is assume to be a step input signal indicating the road surfaces.

Half front-rear model

For the half model suspension, the front and rear suspension are considered as two independent quarter model attached by a rigid vehicle's body that will be modeled as the two degree-of-freedom (DOF) system. The output of the system will considers the bounce motion as well as the pitch motion of the car. The setup of the suspension will consist of equivalent springs in which the stiffness of the tire and the spring are combined and equivalent dampers that account for the shock absorber and the damping of the tire.

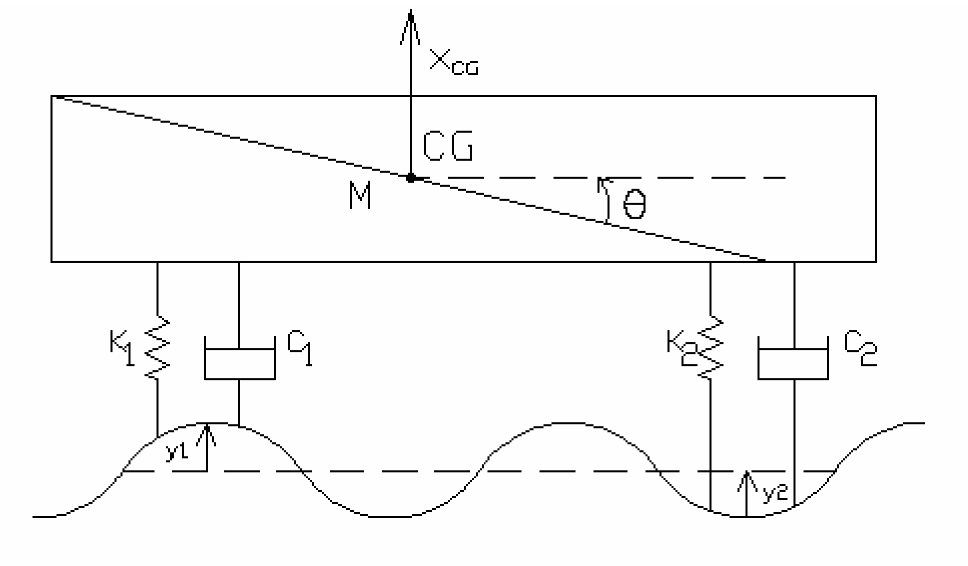


Figure 6: The two DOF schematic of half front-rear model

The equations of motion for the half front-rear suspension are determined by using the Lagrange's equations, also known as the energy method [9]. Equation (6) shows the general form of Lagrange's equations

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = Q_i \qquad \ldots(6)$$

where L is the sum of the kinetic and potential energies, or

$$L = T - U \qquad \ldots(7)$$

where T is the kinetic energy, U is the potential energy of the system. The terms q_i and Qi from equation (6) represents a degree of freedom and the non-conservative work for each DOF; subscript i denoting the first and second degrees of freedom; $\dot{q}_i$ represents the derivative of q_i [9].

By defining the degrees of freedom i, the equations of motion using Lagrange can be determined. This is shown in equation (8) and (9).

$$q_1 = x \qquad \ldots(8)$$

$$q_2 = \theta \qquad \ldots(9)$$

Next, the kinetic energy of the system is shown in equation (10),

$$T = \frac{1}{2}M\dot{x}_{CG}{}^2 + \frac{1}{2}J\dot{\theta}^2 \qquad \ldots(10)$$

where M is the mass of the body, $\dot{x}_{CG}$ is the bounce velocity of the body about its center of gravity, J is the polar moment of inertia, and $\dot{\theta}$ is the angular acceleration of the body [9]. The potential energy of the system is shown in equation (11):

$$U = \frac{1}{2}k_1(x_{CG} - l_1\theta - y_1)^2 + \frac{1}{2}k_2(x_{CG} + l_2\theta - y_2)^2 \qquad \ldots(11)$$

where k_1 and k_2 are the equivalent spring rates of the front and rear suspension, x_{CG} is the displacement of the body's center of gravity, l_1 and l_2 are the distances from the center of gravity to the front suspension and rear suspensions, and y_1 and y_2 are the input functions of the road for the front and rear of the system [9]. By combining equations (10) and (11), the energy equation is produced:

$$L = T - U = \frac{1}{2}M\dot{x}_{CG}{}^2 + \frac{1}{2}J\dot{\theta}^2 - \left[\frac{1}{2}k_1(x_{CG} - l_1\theta - y_1)^2 + \frac{1}{2}k_2(x_{CG} + l_2\theta - y_2)^2\right] \quad \ldots(12)$$

The equations for non-conservative work for both degrees of freedom are shown in equations (13) and (14):

$$Q_1 = -c_1\left(\dot{x}_{CG} - l_1\dot{\theta} - \dot{y}_1\right) - c_2\left(\dot{x}_{CG} + l_2\dot{\theta} - \dot{y}_2\right) \quad \ldots(13)$$

$$Q_2 = -l_1c_1\left(\dot{x}_{CG} - l_1\dot{\theta} - \dot{y}_1\right) - l_2c_2\left(\dot{x}_{CG} + l_2\dot{\theta} - \dot{y}_2\right) \quad \ldots(14)$$

where Q_1 and Q_2 are non-conservative work which is the dissipative force for q_1 and q_2, c_1 and c_2 are the damping coefficients of the system and are the time derivatives of the road input function. Finally, by taking the derivatives of the *q* terms and combining all of the equations into the form of equation (6), the equations of motion for the half front-rear suspension system are:

$$M\ddot{x}_{CG} + k_1(x_{CG} - l_1\theta - y_1) + k_2(x_{CG} + l_2\theta - y_2) =$$
$$-c_1\left(\dot{x}_{CG} - l_1\dot{\theta} - \dot{y}_1\right) - c_2\left(\dot{x}_{CG} + l_2\dot{\theta} - \dot{y}_2\right) \quad \ldots(15)$$

$$J\ddot{\theta} + l_1k_1(x_{CG} - l_1\theta - y_1) - l_2k_2(x_{CG} + l_2\theta - y_2) =$$
$$-l_1c_1\left(\dot{x}_{CG} - l_1\dot{\theta} - \dot{y}_1\right) + l_2c_2\left(\dot{x}_{CG} + l_2\dot{\theta} - \dot{y}_2\right) \quad \ldots(16)$$

The parameters of the system are as follows: $k_1 = k_2 = 30000$ N/m, $c_1 = c_2 = 3000$ N*s/m, $M = 2000$ kg, $J = 2500$ kg*m^2, $l_1 = 1$ m, and $l_2 = 1.5$ m [10]. Substituting these values and expanding equations (15) and (16) yields equations (17) and (18)

$$\ddot{x} + 3\dot{x} + 30x + 0.75\dot{\theta} + 7.5\theta = 1.5\dot{y}_1 + 15y_1 + 1.5\dot{y}_2 + 15y_2 \quad \ldots(17)$$

$$\ddot{\theta} + 3.9\dot{\theta} + 39\theta + 0.6\dot{x} + 6x = 1.2\dot{y}_1 + 12y_1 - 1.8\dot{y}_2 - 18y_2 \qquad \ldots(18)$$

The car is assume to be traveling at 13.88 m/s over road that is assumed to be sinusoidal in cross-section with an amplitude of 10 millimeters and having a wavelength of 5 meters [10]. With this information, the input functions y_1 and y_2 are defined in equations (19) and (20):

Since, (*2*3.14*13.88/5=17.42*)

So,

$$y_1 = 0.01sin(17.42t) \qquad \ldots(19)$$

$$y_2 = 0.01sin(17.42t + \pi) \qquad \ldots(20)$$

Where, t is the time traveled and π is the time shift that accounts for the time that it takes for the rear suspension to experience the "bump" that the front suspension had travelled [10].

A Proportional, Integral and Derivative (PID) controller are used to get the optimum result for the half front-rear suspension model performances. Therefore, the system's parameters can be determined

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Simulation for the Quarter-car Model

A block diagram is designed from the basic equations (4) and (5) obtained earlier to monitor the deflection of the suspension as well as the deflection of the vehicle's body towards the changes in the road surface. For this simulation, first, the changes in the road surface, r will be set to a step input signal. Hence, x and y can be treated as output to observe the suspension damping ratio. Figure 7 below shows the result of the Matlab simulation which indicates the signals are damped over time.

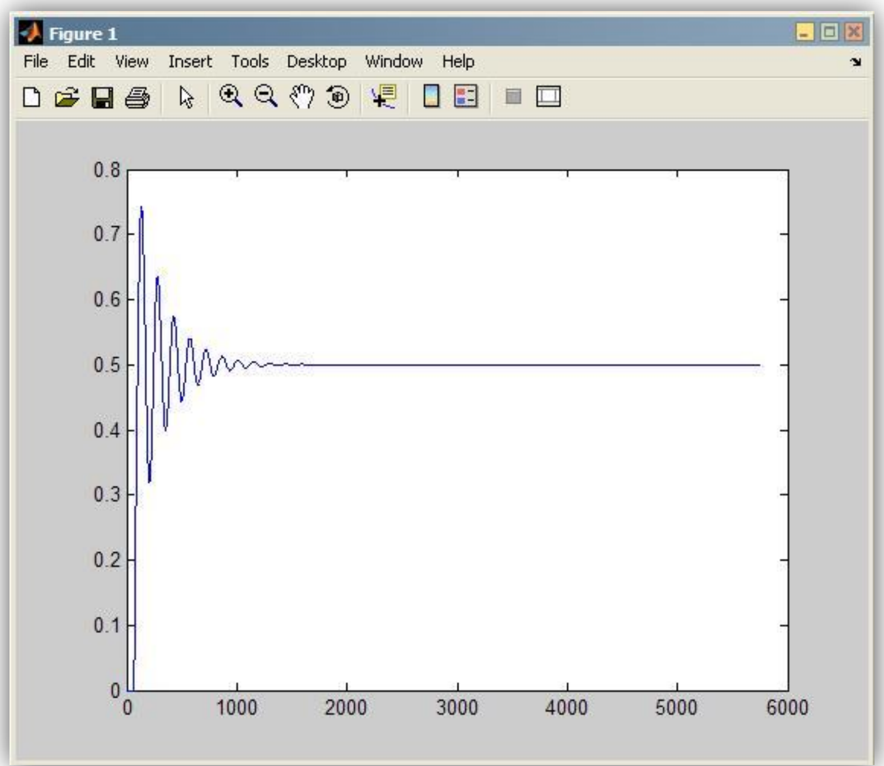


Figure 7: Graph for x-plot

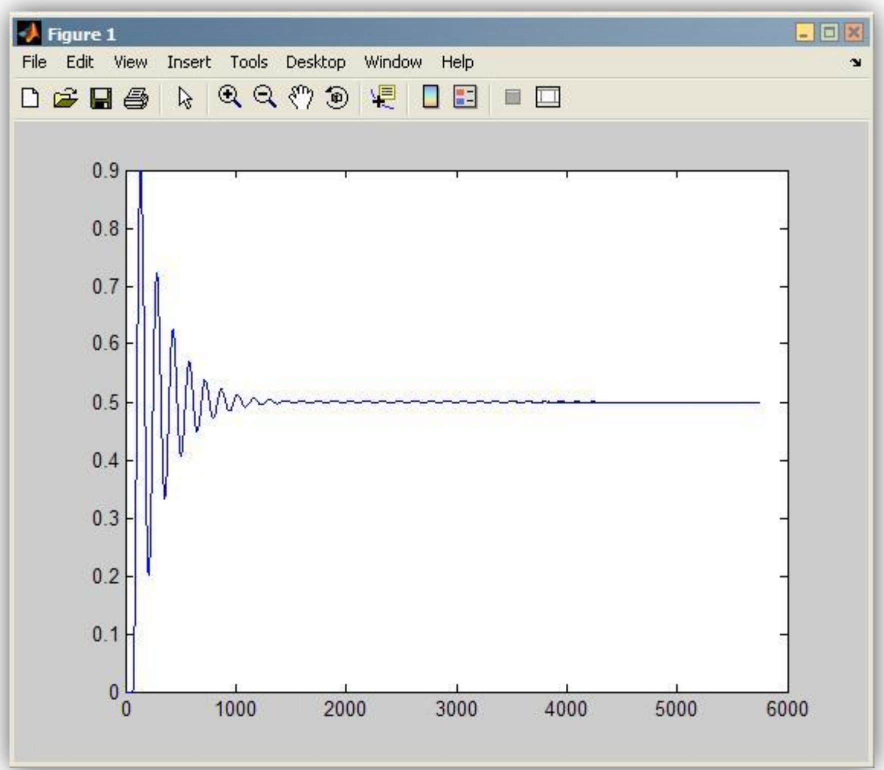


Figure 8: Graph for *y*-plot

From the simulation, the oscillation of the deflection signal for the vehicle's body found to be taking a longer period of time to become stabilize compared to the suspension's deflection. This is due to the huge difference in mass between the vehicle's body and the wheels.

4.2 Simulation for the Half-car Model

Simulation for the half front-rear suspension is modeled based on the equations (17) and (18) while the input of road disturbance is assumed to be a sine wave from equations (19) and (20) for front and rear suspension's input respectively. The simulation time is set to 50 seconds so that the data can be observed clearly. Besides the bounce motion, pitch motion also will be studied in the half front-rear suspension. Data 1, blue line, in Figure 9 indicates the rear suspension's input while data 2, red line, indicates front suspension's input. The input of the rear suspension experienced a bit of delay compared to the front suspension.

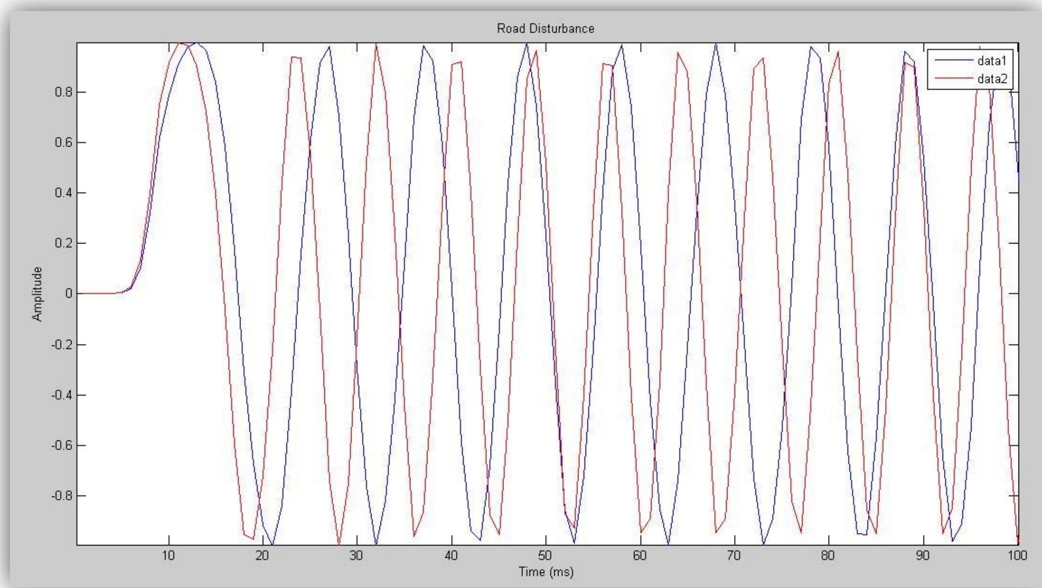


Figure 9: Road disturbance as input

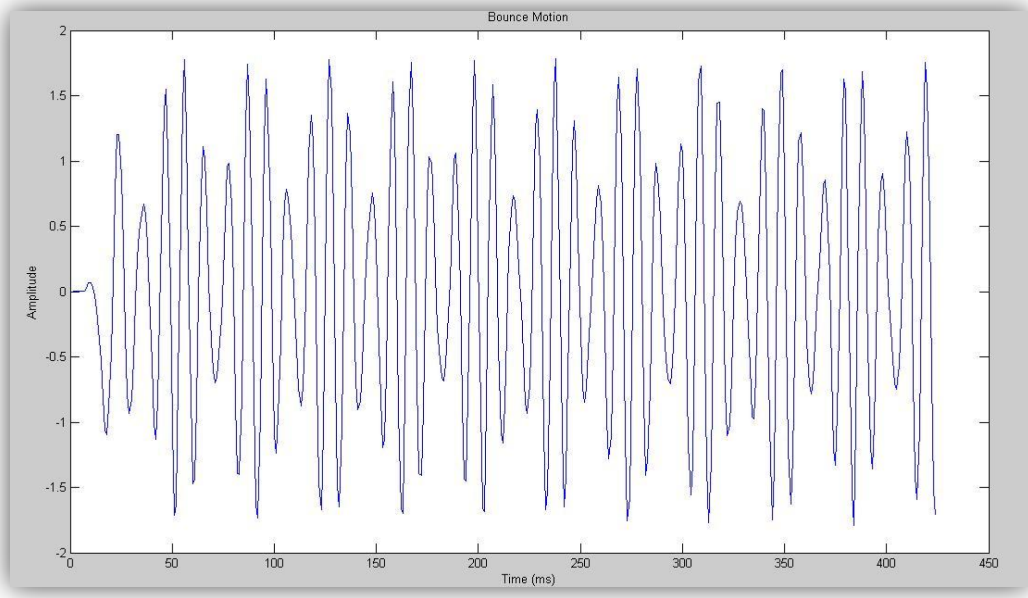


Figure 10: Bounce motion of the suspension

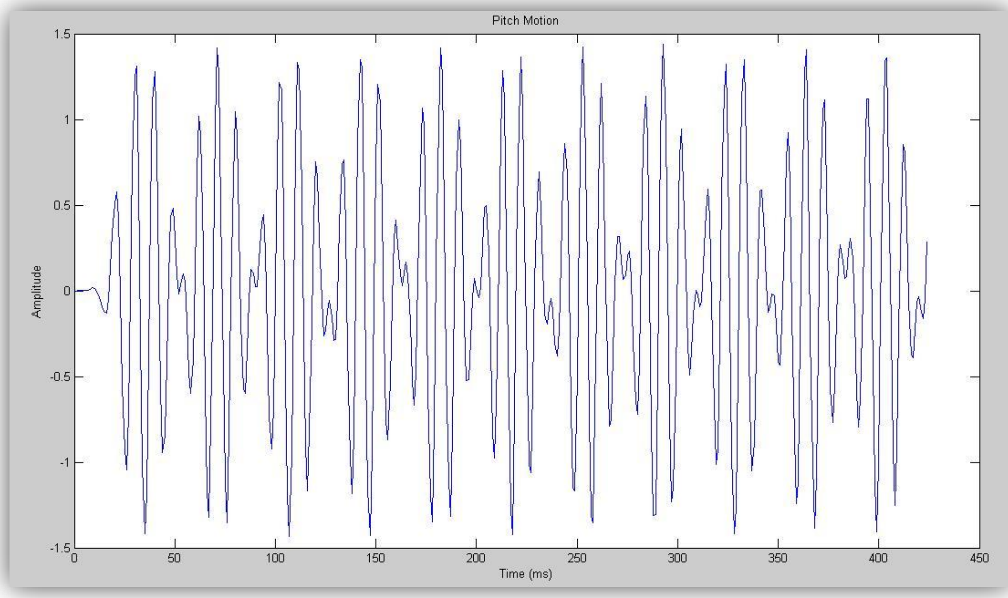
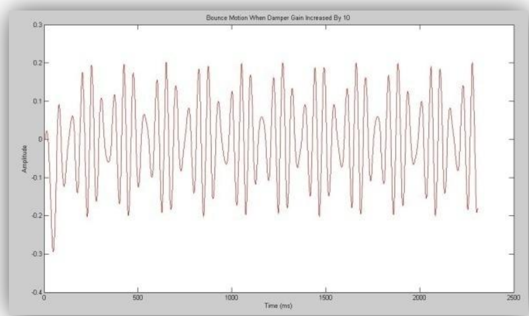


Figure 11: Pitch motion of the suspension

The bounce motion of the suspension is in a uniform pattern as the inputs of the system are sine waves. It shows that the system have a good response towards the input. This is same goes to the pitch motion of the car which shows smaller amplitude than the bounce motion. With the same model, the gains on dampers were modified by ten times larger to achieve a desirable performance.



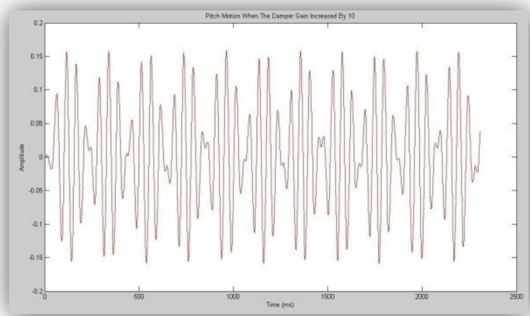


Figure 12: Bounce and pitch motion of the suspension after the damper gain is increased by 10.

Based on Figure 12, the bounce motion decreases from the amplitude of 1.8 to 0.2 after the damper gain increased by ten. The pitch motion as well decreases from the amplitude of 1.4 to about 0.15 in Figure 13.

From the simulation, the matrices of the system as well as the transfer function can be determined. Using the transfer function a PID controller is designed to study the parameters of the system before proceed with the system identification.

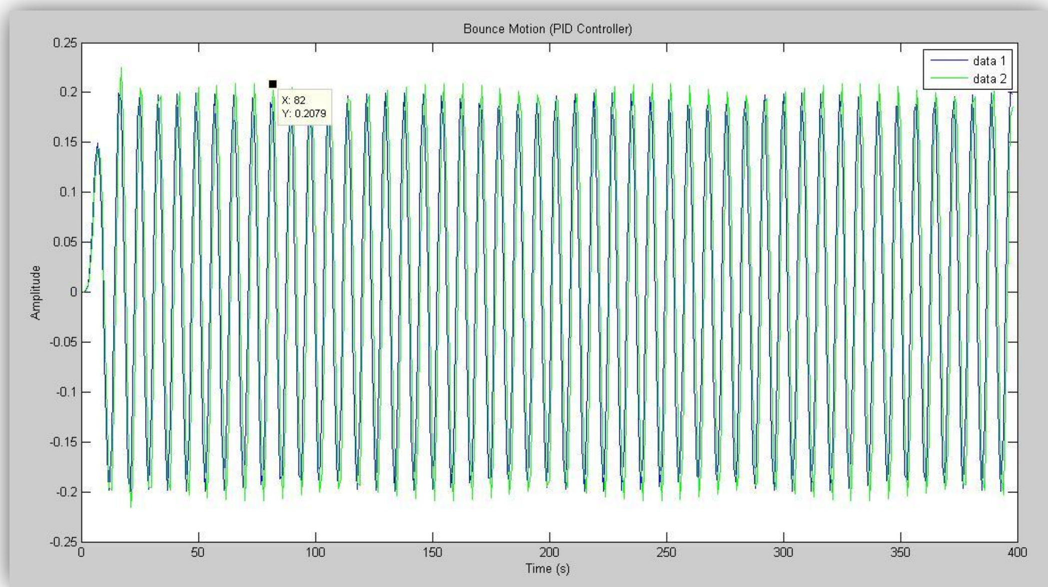


Figure 13: Bounce motion using the PID controller

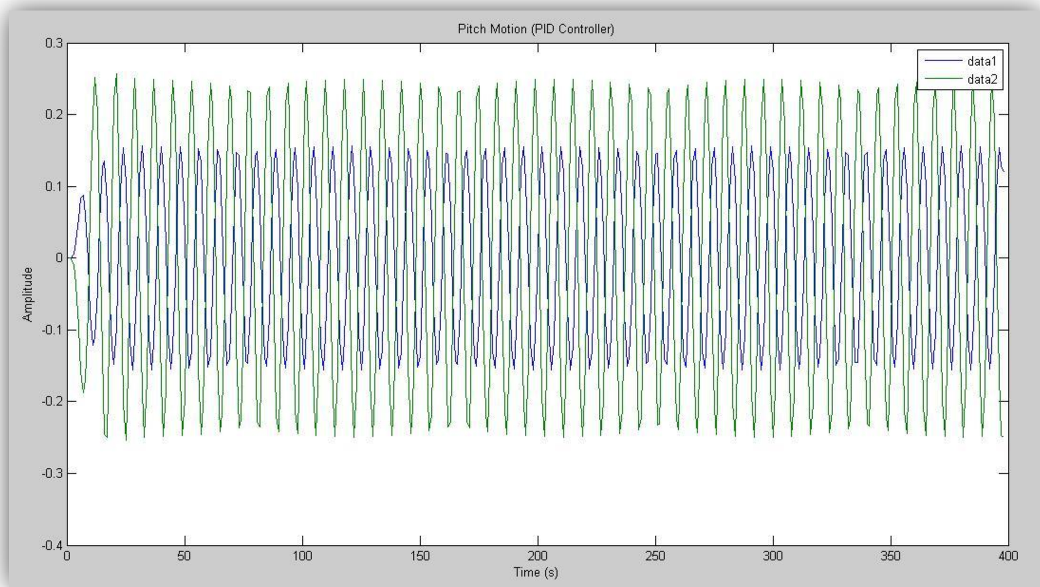


Figure 14: Pitch motion using the PID controller

Data 1 for both Figure 13 and 14 represent the front suspension while data 2 is representing the rear suspension. It is shown that besides increasing the damper gains, the amplitude of bounce motion and pitch motion can also be decreases by changing the parameters of PID controller.

Next figure shows the result if the front suspension output becomes the rear suspension input. From the graph, data 2 which is representing the rear suspension's bounce motion have much smaller amplitude. It is because when the front suspension is experiencing a bump, it will send signals to the rear suspension about the bump. Thus, the rear suspension will increase the parameters that suitable for the bump to experience less oscillation.

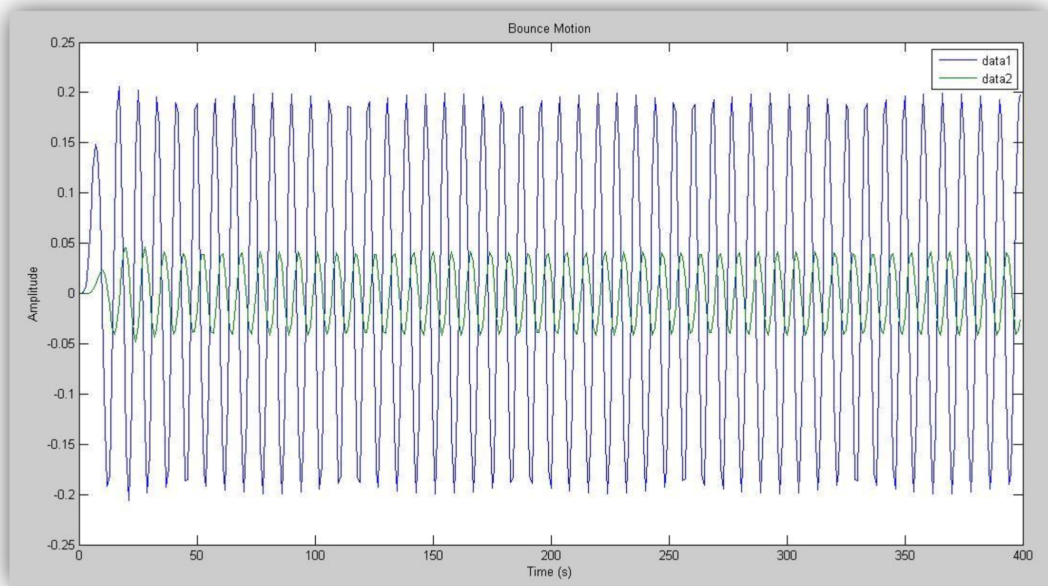


Figure 15: Bounce motion using the PID controller when front and rear suspension are connected to each other

The next stage is to study the step response towards the PID controller. The step signal is assume to be a single bumper as the road condition. The results are shown in Figure 16 and 17.

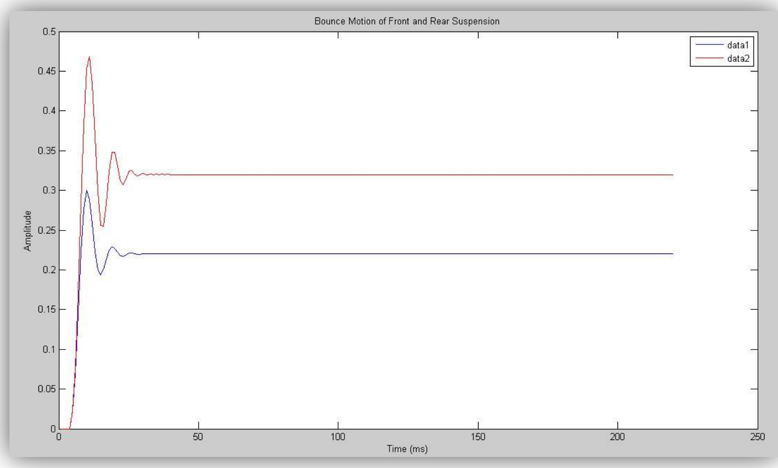


Figure 16: Step response of bounce motion using the PID controller

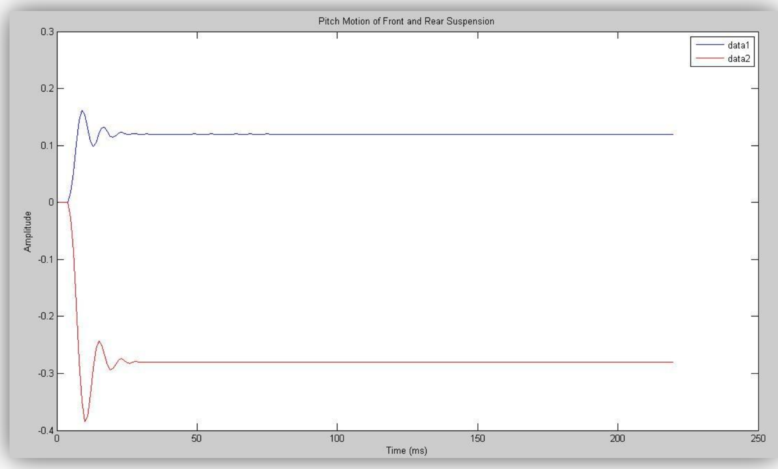


Figure 17: Step response of pitch motion using the PID controller

From the graph in Figure 13, the bounce motion of the rear suspension, data 2 is higher than the front suspension, data 1. This is because when the front suspension experience a bump, the rear suspension will also got affected with the motion, thus, the force will sum up causing the amplitude of the rear suspension bounce motion to be slightly higher.

For the pitch motion, the positive and the negative value of the amplitude indicates the direction of the suspension vertical movement. When the front suspension experience a bump the magnitude of the pitch motion is lesser compared to the magnitude of the rear suspension's pitch motion.

4.3 System Identification for Half Model

The half front-rear suspension model is identified with AR (Auto Regression), ARX (Auto Regression with External Input) and OE (Output Error) models. Simulation is carried out using first, second and third order linear model structure for linear network. For the front and rear suspension, the prediction is done separately to see a clearer prediction on the output of the model.

4.3.1 Results for the front suspension

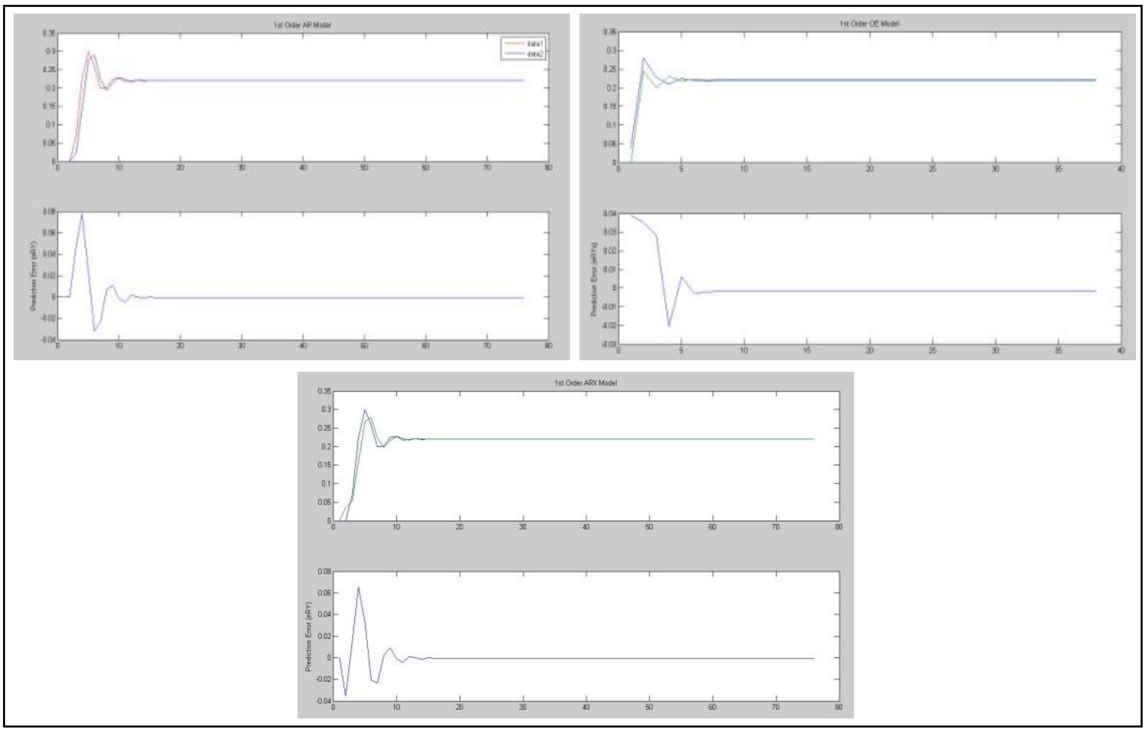


Figure 18: First Order AR, ARX and OE model of bounce motion and prediction error

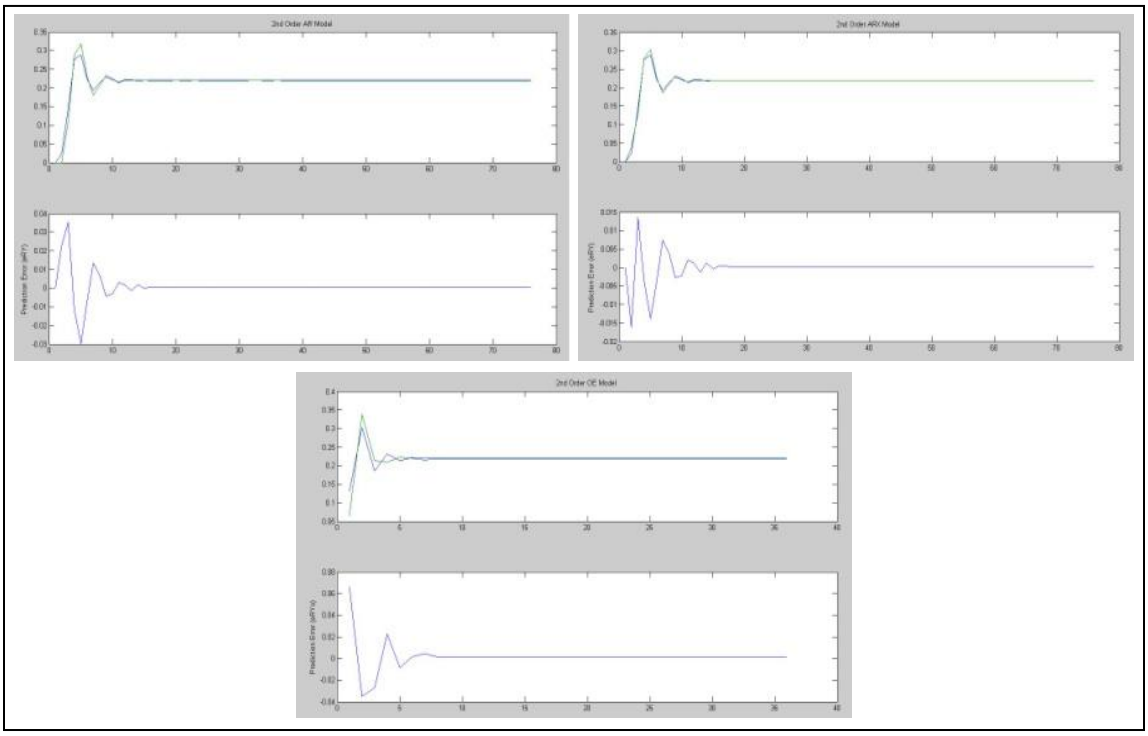


Figure 19: Second Order AR, ARX and OE model of bounce motion and prediction error

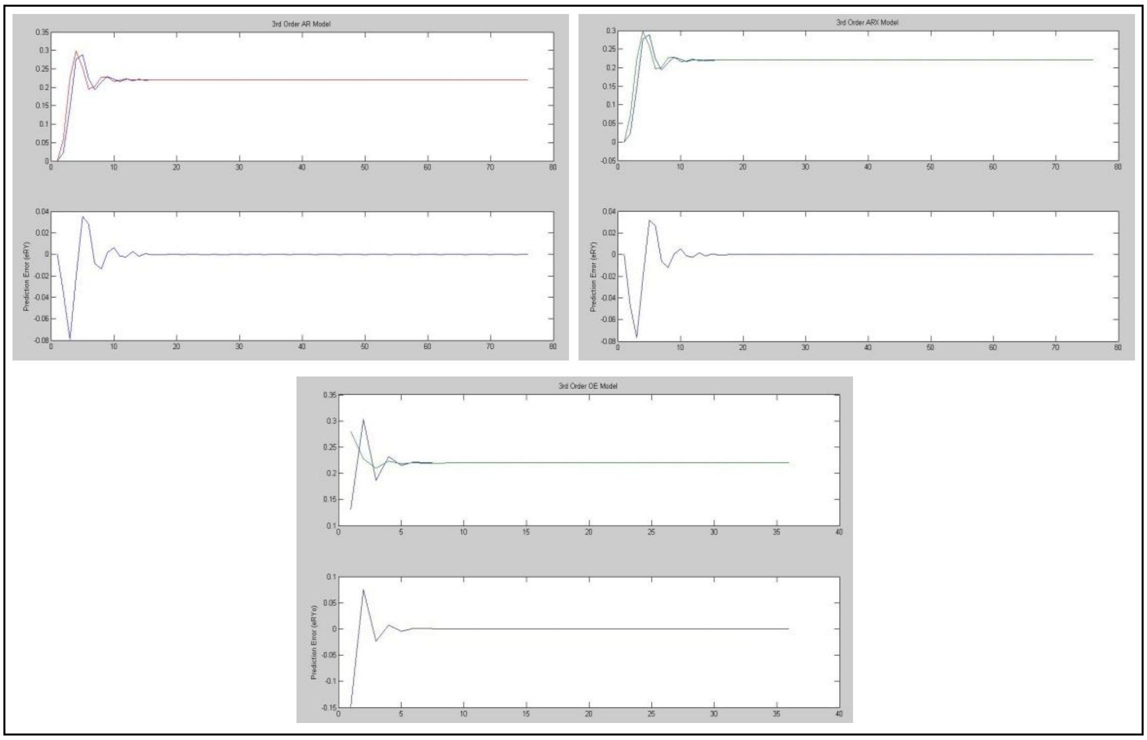


Figure 20: Third Order AR, ARX and OE model of bounce motion and prediction error

Table 1: Bounce motion prediction error for AR, ARX and OE models of front suspension.

Model	1st Order	2nd Order	3rd Order
AR	0.0053	0.0016	0.0050
ARX	0.0040	3.7938e-004	0.0053
OE	0.0020	0.0035	0.0142

Graphs shown in Figure 18 are a set of graphs for modeling the front and rear suspension's bounce motion and the error prediction of the system. It is noticed that for first order AR, ARX and OE model, the prediction error for OE model is the best. However, the best prediction error for the second order is ARX model and for third order is AR model. Since the smallest prediction error shows the best model structure, ARX of second order is the most suitable model for this system as it gives an error of 3.7938e-004 in the amplitude of the bounce motion.

4.3.2 Results for the rear suspension

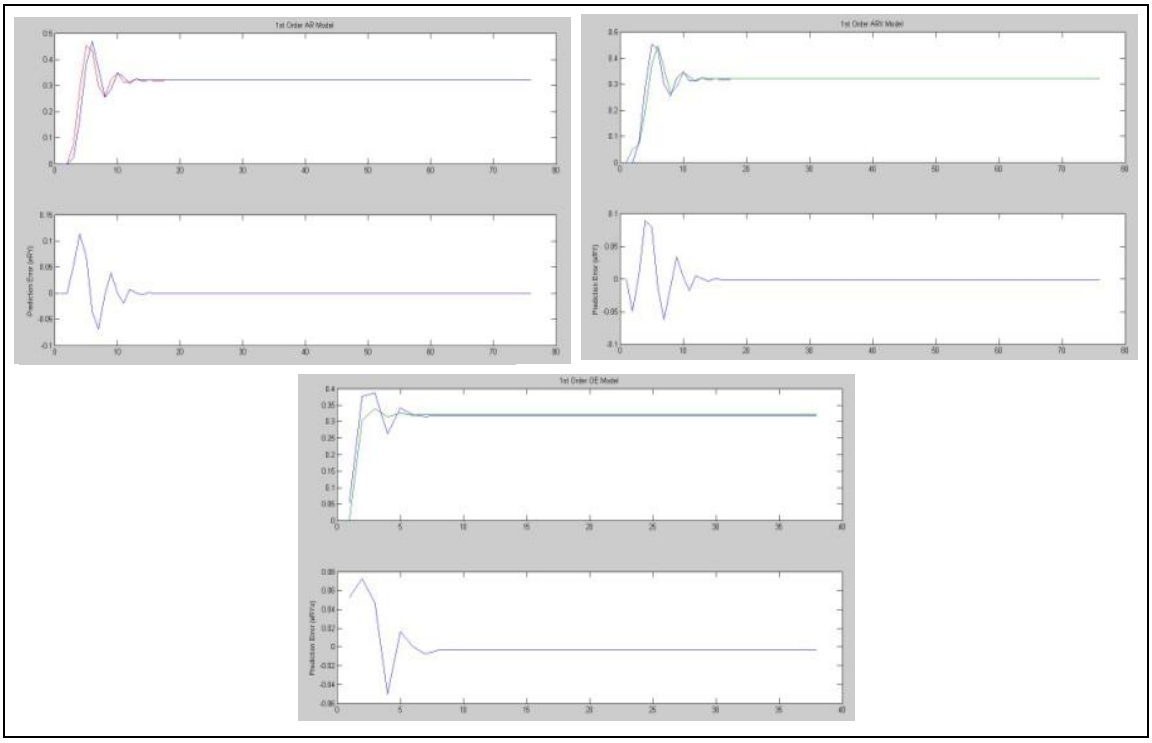


Figure 21: First Order AR, ARX and OE model of bounce motion and prediction error

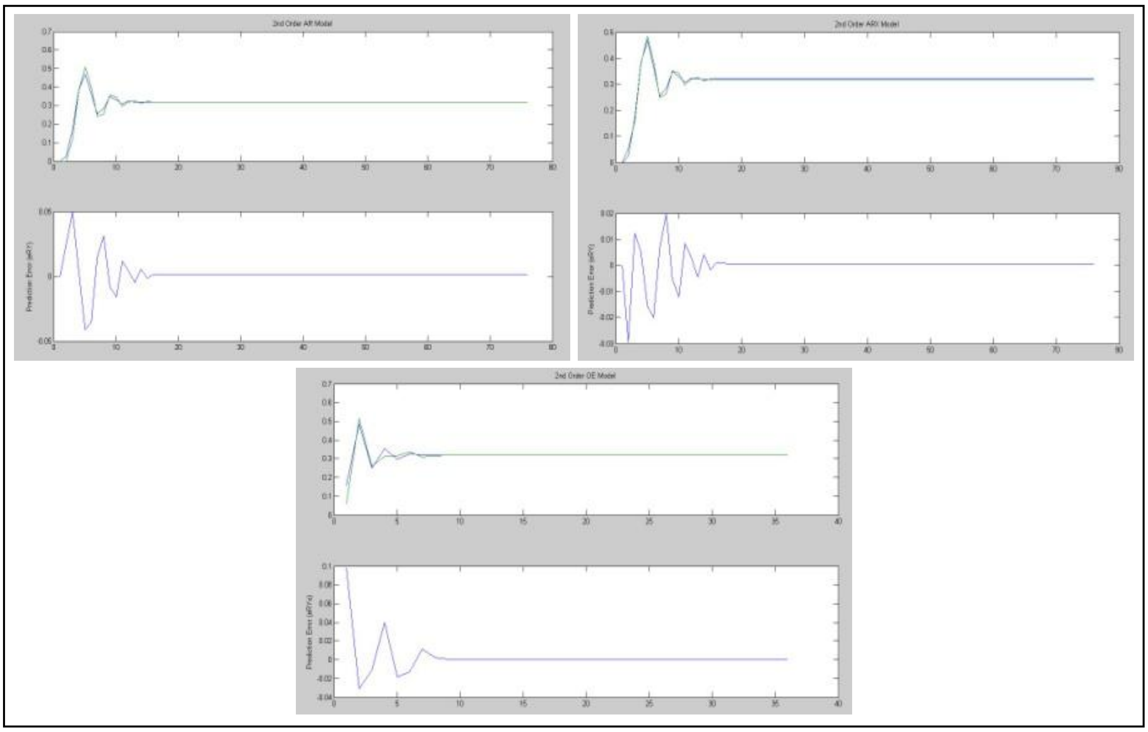


Figure 22: Second Order AR, ARX and OE model of bounce motion and prediction error

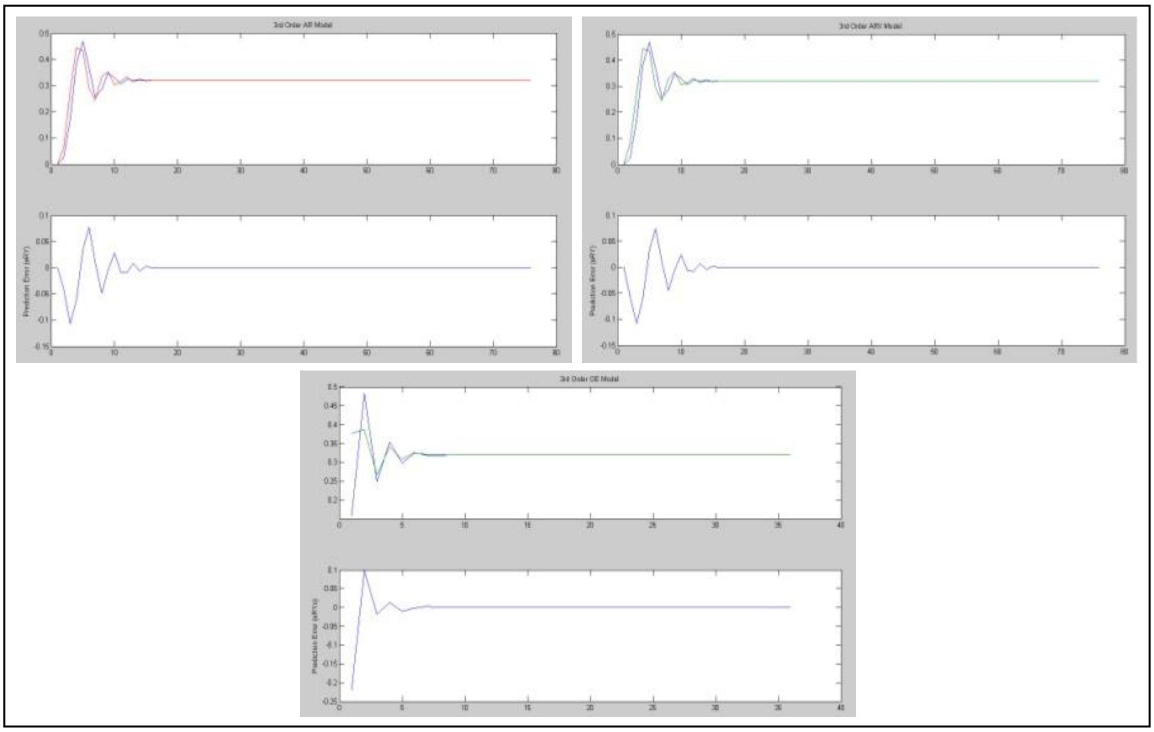


Figure 23: Third Order AR, ARX and OE model of bounce motion and prediction error

Table 2: Bounce motion prediction error for AR, ARX and OE models of rear suspension .

Model	1st Order	2nd Order	3rd Order
AR	0.0143	0.0039	0.0139
ARX	0.0113	0.0012	0.0141
OE	0.0067	0.0065	0.0291

In Figure 21, a set of graphs for modeling the rear suspension's bounce motion and the error prediction of the system is shown. The characteristic of the graphs towards each system identification model structure is the same as given by the front suspension. The rear suspension also shows that the second order of ARX model is giving the best prediction. The difference between front and rear suspension is just the amplitude of the bounce motion where the rear suspension shows slightly higher amplitude than the front suspension.

CHAPTER 5
CONCLUSION AND RECOMMENDATION

5.1 Conclusion

As a conclusion, the derivation of the equation for the systems needs to take account all factors such as the gravity acceleration, the velocity of the car, the car mass and also the effects of bumpy road to the suspension system. The correct equation will make the simulation in the Matlab is possible to get the correct data and observation on the suspension system.

From the simulation for the quarter-car model suspension, the oscillation of the deflection signal of the vehicle's body needs to be reduced as it takes a longer time to become stabilize compared to the suspension's deflection. A good suspension system will stabilize the oscillations in a short period of time thus, giving comfort to the occupants. These can be achieved by varying the spring constant.

In the half-car model suspension, the bounce motion and the pitch motion of the car can be reduced by increasing the damper gains and modifying the PID controller gains. From the system identification models, the second order ARX model is given the least error prediction in the bounce motion of the suspension; thus, it is the best model for the suspension system.

5.2 Recommendation

For the improvement, system identification for the pitch motion can be studied and identified whether the same model for bounce motion can be used. The suspension system also can be improved with a more optimum result using other design and equations.

REFERENCES

[1] “Suspension (vehicle)” Wikipedia: The Free Encyclopedia <http://en.wikipedia.org/wiki/Suspension_(vehicle)>

[2] Harris, William. "How Car Suspensions Work." 11 May 2005. HowStuffWorks.com. <http://auto.howstuffworks.com/car-suspension.htm>

[3] “Independent Suspension” Wikipedia: The Free Encyclopedia <http://en.wikipedia.org/wiki/Independent_suspension> 09 March 2009

[4] “System Identification Toolbox 7.3.1” TheMathWorks: Accelerating the pace of Engineering and Science. <http://www.mathworks.com/products/sysid/description1.html> 16 March 2009.

[5] Afifi bin Md Akib, Nordin Saad, Vijanth S. Asirvadam. “System Identification for Modelling Gas Flow With a Leakage”, International Conference on Soft Computing and Pattern Recognition, December 2009.

[6] Mousab J. O. Elamin, Vijanth S. Asirvadam, Nordin Saad. “Modelling Control Systems With lost Information Packets using ARMAX Model”, International Conference in Advanced System and Technology (ICIAS 2007), November 2007.

[7] Gene F. Franklin, J. David Powell, Abbas Emami-Naemi, Feedback Control of Dynamic Systems 5th Edition, Pearson International Edition, 2006.

[8] “Bus Suspension Modelling in Simulink” Control Tutorials for Matlab and Simulink. <http://www.engin.umich.edu/class/ctms/simulink/examples/susp/suspsim.htm>

[9] “Lagrangian mechanics” Wikipedia: The Free Encyclopedia <http://en.wikipedia.org/wiki/Lagrangian_mechanics>

[10] Vu Trieu Minh. “Analysis of Automobile Suspension with Half Model” Engine Design Project, 02 August 2008.

[11] Giancarlo Genta, Lorenzo Morello, The Automotive Chassis, Volume 2, Mechanical Engineering Series, Springer, 2009.

APPENDICES

APPENDIX A

MATLAB COMMAND AND M-FILE CODING

Matlab Command

```
>>[A,B,C,D]=linmod('susp202c')
A =

         0         0         0    1.0000
         0         0    1.0000         0
   -6.0000  -39.0000   -3.9000   -0.6000
  -30.0000   -7.5000   -0.7500   -3.0000


B =

     0     0
     0     0
    12   -18
    15    15


C =

     1     0     0     0
     0     1     0     0


D =

     0     0
     0     0

>> [num1,den1]=ss2tf(A,B,C,D,1)
num1 =

         0   -0.0000   15.0000   49.5000  495.0000
         0   -0.0000   12.0000   27.0000  270.0000


den1 =

  1.0e+003 *

    0.0010    0.0069    0.0803    0.2250    1.1250

>> [num2,den2]=ss2tf(A,B,C,D,2)
num2 =

         0   -0.0000   15.0000   72.0000  720.0000
         0   -0.0000  -18.0000  -63.0000 -630.0000


den2 =

  1.0e+003 * 0.0010    0.0069    0.0803    0.2250    1.1250
```

Auto Regression (AR) Model

1st Order

```
Y2 = Boutr(2:2:152);
Y1 = Boutr(1:2:151);
A = [Y1];
b = [Y2];
X = pinv(A)*b
YpAR = X(1)*Y1;
erY = Y2-YpAR;
subplot(2,1,1);
plot (Y2,'r');
hold
subplot(2,1,1);
plot(YpAR,'b');
subplot(2,1,2)
plot(erY);
0.5*(sum(erY.^2))
```

2nd Order

```
Y3 = Boutf(3:2:153);
Y2 = Boutf(2:2:152);
Y1 = Boutf(1:2:151);
A = [Y2 Y1];
b = [Y3];
X = pinv(A)*b
YpAR = X(1)*Y2 + X(2)*Y1;
erY = Y3-YpAR;
subplot(2,1,1);
plot ([Y3 YpAR])
hold
subplot(2,1,2)
plot(erY);
0.5*(sum(erY.^2))
```

3rd Order

```
Y4 = Boutr(4:2:154);
Y3 = Boutr(3:2:153);
Y2 = Boutr(2:2:152);
Y1 = Boutr(1:2:151);
A = [Y3 Y2 Y1];
b = [Y4];
X = pinv(A)*b;
YpAR = X(1)*Y3 + X(2)*Y2 + X(3)*Y1;
erY = Y3 - YpAR;
subplot(2,1,1);
plot(Y3)
hold
plot(YpAR,'r')
hold
subplot(2,1,2)
plot(erY)
0.5*(sum(erY.^2))
```

4th Order

```
Y5 = Boutf(5:2:155);
Y4 = Boutf(4:2:154);
```

```
Y3 = Boutf(3:2:153);
Y2 = Boutf(2:2:152);
Y1 = Boutf(1:2:151);
A = [Y4 Y3 Y2 Y1];
b = [Y5];
X = pinv(A)*b;
YpAR = X(1)*Y4 + X(2)*Y3 + X(3)*Y2 + X(4)*Y1;
erY = Y4 - YpAR;
subplot(2,1,1);
plot(Y4)
hold
plot(YpAR,'r')
hold
subplot(2,1,2)
plot(erY)
0.5*(sum(erY.^2))
```

5th Order

```
Y6 = Boutr(6:2:156);
Y5 = Boutr(5:2:155);
Y4 = Boutr(4:2:154);
Y3 = Boutr(3:2:153);
Y2 = Boutr(2:2:152);
Y1 = Boutr(1:2:151);
A = [Y5 Y4 Y3 Y2 Y1];
b = [Y6];
X = pinv(A)*b;
YpAR = X(1)*Y5 + X(2)*Y4 + X(3)*Y3 + X(4)*Y2 + X(5)*Y1;
erY = Y5 - YpAR;
subplot(2,1,1);
plot(Y5)
hold
plot(YpAR,'r')
hold
subplot(2,1,2)
plot(erY)
0.5*(sum(erY.^2))
```

Auto Regression With External Input (ARX) Model

1st Order

```
Y2 = Boutr(2:2:152);
Y1 = Boutr(1:2:151);
X1 = Rinp(1:2:151);
A = [Y1 X1];
b = [Y2];
X = pinv(A)*b;
YpARX = X(1)*Y1 + X(2)*X1;
erYx = Y2-YpARX;
subplot(2,1,1);
plot ([Y2 YpARX])
hold
subplot(2,1,2)
plot(erYx);
0.5*(sum(erYx.^2))
```

2^{nd} Order

```
Y3 = Boutr(3:2:153);
Y2 = Boutr(2:2:152);
Y1 = Boutr(1:2:151);
X2 = Rinp(2:2:152);
X1 = Rinp(1:2:151);
A = [Y2 Y1 X2 X1];
b = [Y3];
X = pinv(A)*b
YpARX = X(1)*Y2 + X(2)*Y1 + X(3)*X2 + X(4)*X1;
erYx = Y3-YpARX;
subplot(2,1,1);
plot ([Y3 YpARX])
hold
subplot(2,1,2)
plot(erYx);
0.5*(sum(erYx.^2))
```

3^{rd} Order

```
Y4 = Bmot(304:2:804);
Y3 = Bmot(303:2:803);
Y2 = Bmot(302:2:802);
Y1 = Bmot(301:2:801);
X3 = inp(303:2:803);
X2 = inp(302:2:802);
X1 = inp(301:2:801);
A = [Y3 Y2 Y1 X3 X2 X1];
b = [Y4];
X = pinv(A)*b
YpARX = X(1)*Y3 + X(2)*Y2 + X(3)*Y1 + X(4)*X3 + X(5)*X2 + X(6)*X1;
erY = Y3-YpARX;
subplot(2,1,1);
plot ([Y3 YpARX])
hold
subplot(2,1,2)
plot(erY);
0.5*(sum(erY.^2))
```

4^{th} Order

```
Y5 = Bmot(305:2:805);
Y4 = Bmot(304:2:804);
Y3 = Bmot(303:2:803);
Y2 = Bmot(302:2:802);
Y1 = Bmot(301:2:801);
X4 = inp(304:2:804);
X3 = inp(303:2:803);
X2 = inp(302:2:802);
X1 = inp(301:2:801);
A = [Y4 Y3 Y2 Y1 X4 X3 X2 X1];
b = [Y5];
X = pinv(A)*b
YpARX = X(1)*Y4 + X(2)*Y3 + X(3)*Y2 + X(4)*Y1 + X(5)*X4 + X(6)*X3 +
X(7)*X2 + X(8)*X1;
erY = Y4-YpARX;
subplot(2,1,1);
plot ([Y4 YpARX])
hold
subplot(2,1,2)
plot(erY);
```

```
0.5*(sum(erY.^2))
```

5th Order

```
Y6 = Bmot(306:2:806);
Y5 = Bmot(305:2:805);
Y4 = Bmot(304:2:804);
Y3 = Bmot(303:2:803);
Y2 = Bmot(302:2:802);
Y1 = Bmot(301:2:801);
X5 = inp(305:2:805);
X4 = inp(304:2:804);
X3 = inp(303:2:803);
X2 = inp(302:2:802);
X1 = inp(301:2:801);
A = [Y5 Y4 Y3 Y2 Y1 X5 X4 X3 X2 X1];
b = [Y6];
X = pinv(A)*b
YpARX = X(1)*Y5 + X(2)*Y4 + X(3)*Y3 + X(4)*Y2 + X(5)*Y1 + X(6)*X5 +
X(7)*X4 + X(8)*X3 + X(9)*X2 + X(10)*X1;
erY = Y5-YpARX;
subplot(2,1,1);
plot ([Y5 YpARX])
hold
subplot(2,1,2)
plot(erY);
0.5*(sum(erY.^2))
```

Output Error (OE) Model

1st Order

```
Y2 = YpARX(2:2:76);
Y1 = YpARX(1:2:75);
X2 = Rinp(2:2:76);
X1 = Rinp(1:2:75);
A = [Y1 X1];
b = [Y2];
X = pinv(A)*b
YpOE = X(1)*Y1 + X(2)*X1
erYo = Y2-YpOE;
subplot(2,1,1);
plot ([Y2 YpOE])
hold
subplot(2,1,2)
plot(erYo);
0.5*(sum(erYo.^2))
```

2nd Order

```
Y3 = YpARX(3:2:73);
Y2 = YpARX(2:2:72);
Y1 = YpARX(1:2:71);
X2 = Rinp(2:2:72);
X1 = Rinp(1:2:71);
A = [Y2 Y1 X2 X1];
b = [Y3];
X = pinv(A)*b
YpOE = X(1)*Y2 + X(2)*Y1 + X(3)*X2 + X(4)*X1;
erYo = Y3-YpOE;
subplot(2,1,1);
```

```
plot ([Y3 YpOE])
hold
subplot(2,1,2)
plot(erYo);
0.5*(sum(erYo.^2))
```

3rd Order

```
Y4 = YpARX(4:2:74);
Y3 = YpARX(3:2:73);
Y2 = YpARX(2:2:72);
Y1 = YpARX(1:2:71);
X3 = Rinp(3:2:73);
X2 = Rinp(2:2:72);
X1 = Rinp(1:2:71);
A = [Y3 Y2 Y1 X3 X2 X1];
b = [Y4];
X = pinv(A)*b
YpOE = X(1)*Y3 + X(2)*Y2 + X(3)*Y1 + X(4)*X3 + X(5)*X2 + X(6)*X1;
erYo = Y3-YpOE;
subplot(2,1,1);
plot ([Y3 YpOE])
hold
subplot(2,1,2)
plot(erYo);
0.5*(sum(erYo.^2))
```

4th Order

```
Y5 = YpARX(5:2:75);
Y4 = YpARX(4:2:74);
Y3 = YpARX(3:2:73);
Y2 = YpARX(2:2:72);
Y1 = YpARX(1:2:71);
X4 = Rinp(4:2:74);
X3 = Rinp(3:2:73);
X2 = Rinp(2:2:72);
X1 = Rinp(1:2:71);
A = [Y4 Y3 Y2 Y1 X4 X3 X2 X1];
b = [Y5];
X = pinv(A)*b
YpOE = X(1)*Y4 + X(2)*Y3 + X(3)*Y2 + X(4)*Y1 + X(5)*X4 + X(6)*X3 +
X(7)*X2 + X(8)*X1;
erYo = Y4-YpOE;
subplot(2,1,1);
plot ([Y4 YpOE])
hold
subplot(2,1,2)
plot(erYo);
0.5*(sum(erYo.^2))
```

5th Order

```
Y6 = YpARX(6:2:76);
Y5 = YpARX(5:2:75);
Y4 = YpARX(4:2:74);
Y3 = YpARX(3:2:73);
Y2 = YpARX(2:2:72);
Y1 = YpARX(1:2:71);
X5 = Rinp(5:2:75);
X4 = Rinp(4:2:74);
X3 = Rinp(3:2:73);
```

```
X2 = Rinp(2:2:72);
X1 = Rinp(1:2:71);
A = [Y5 Y4 Y3 Y2 Y1 X5 X4 X3 X2 X1];
b = [Y6];
X = pinv(A)*b
YpOE = X(1)*Y5 + X(2)*Y4 + X(3)*Y3 + X(4)*Y2 + X(5)*Y1 + X(6)*X5 +
X(7)*X4 + X(8)*X3 + X(9)*X2 + X(10)*X1;
erYo = Y5-YpOE;
subplot(2,1,1);
plot ([Y5 YpOE])
hold
subplot(2,1,2)
plot(erYo);
0.5*(sum(erYo.^2))
```

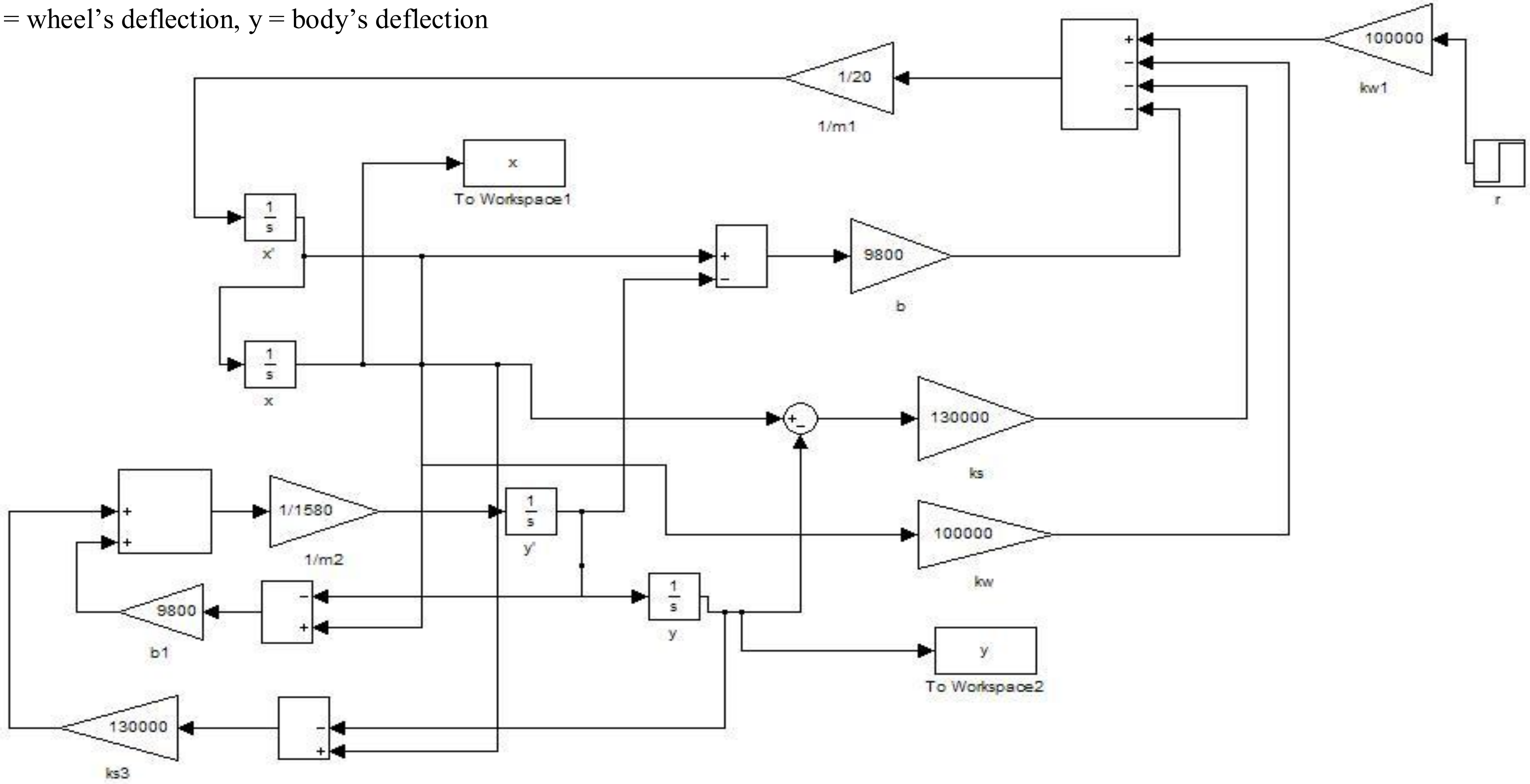
APPENDIX B

SIMULATION BLOCK DIAGRAM

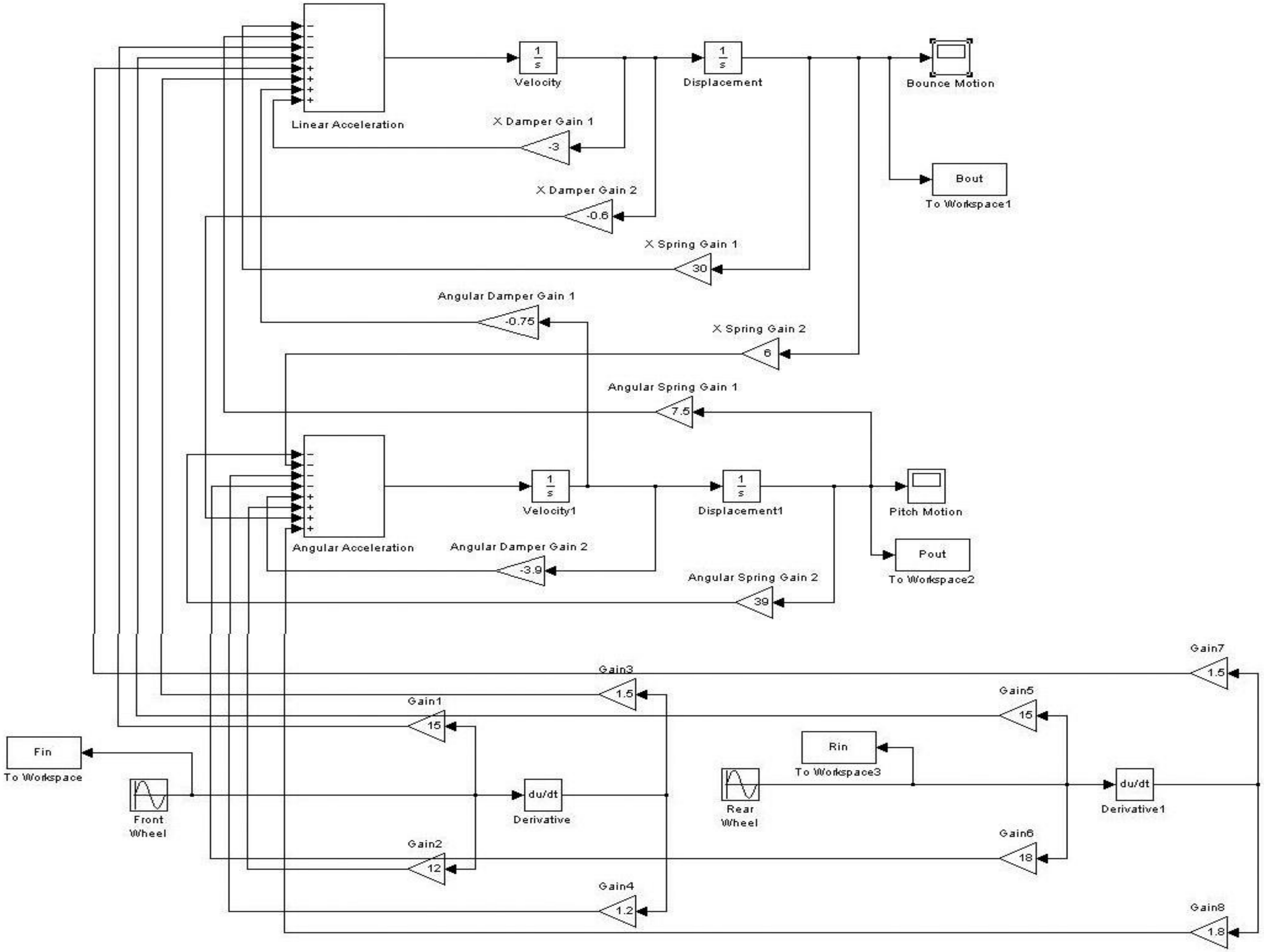
Quarter-car Model

r = road disturbance, b = b1 = damping coefficients, m1= m2 = mass, ks = spring constant, kw = wheel constant

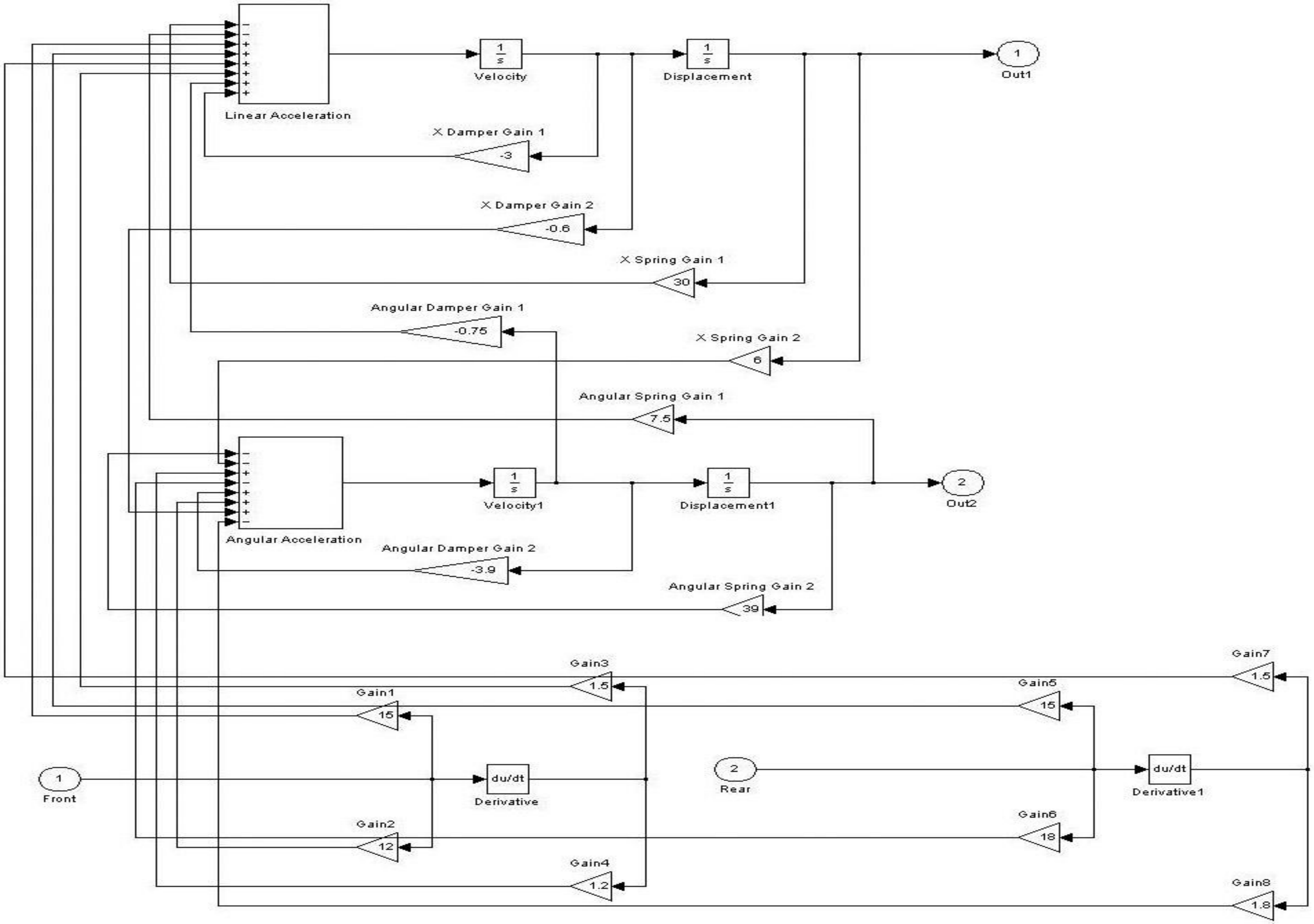
x = wheel's deflection, y = body's deflection



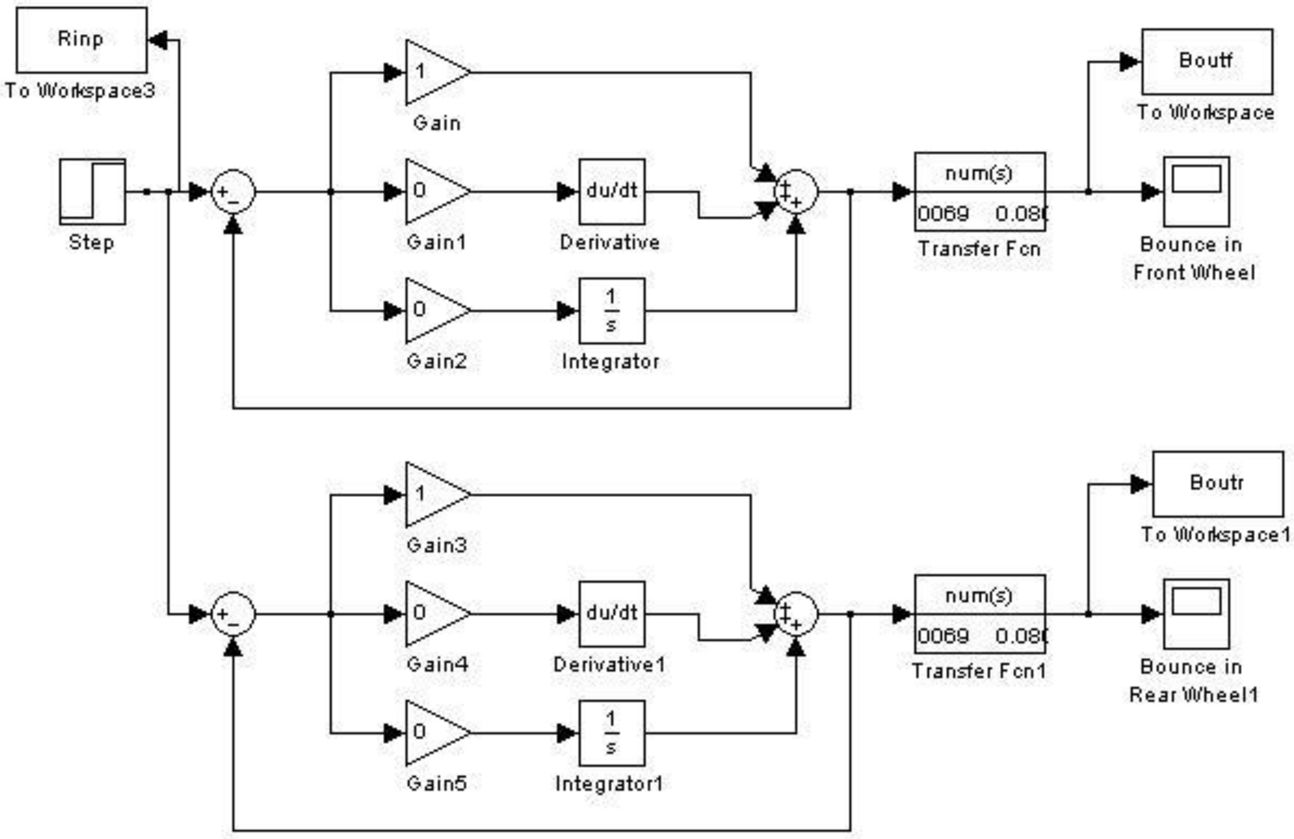
Half Front-Rear Model



Half Front-Rear Model (*Determining the system matrices and transfer function*)



PID Controller for the Bounce Motion of the Suspension



APPENDIX C

FINAL YEAR PROJECT GANTT CHART

	Task Name	Jan		Feb				March				April				May	July		August				September				October				Nov
		1	2	3	4	5	6	7	8	9		10	11	12	13	14	1	2	3	4	5		6	7	8	9	10	11	12	13	14
	Planning Phase																														
1	Briefing																														
2	Define Project Title																														
3	Research																														
	Analysis Phase																														
4	Matlab Simulation																														
5	Analysis of Results																														
6	Improvement of Results																														
	Presentation																														
7	Submission of Report																														
8	Oral Presentation																														