ANALYSIS AND SIMULATION OF AN ACTIVE SUSPENSION SYSTEM

BY

ARASOMWAN AISOSA	ENG1503884
UZOAMAKAH JOSHUA	ENG1503946
OSAGIE SARAH	ENG1503933
OSADOLOR WANPALI DUKE	ENG1503932

SUPERVISED BY

DR. N. ENOMA

DEPARTMENT OF MECHANICAL ENGINEERING

FACULTY OF ENGINEERING

UNIVERISTY OF BENIN, BENIN CITY

IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE AWARD OF BACHELOR OF ENGINEERING (B.ENG)

JULY, 2021

Acknowledgment

A lot of people made this report possible from the company who created matlab ‘mathwork’ to all the members of this group.

We would also like to thank our project supervisor Dr. N. Enoma and other staff of the department of mechanical engineering for guiding us through this process. Special thanks to other researchers whose work where paramount to the completion of this project, all of which were referenced.

I would also like to thank other academics like Kat Kim, Steve Brunton, Zach Star, Brian Douglas, UConn HKN, Neso Academy, Christopher Lum and the Matlab YouTube page for their education content/lectures on control theory and design of control systems on YouTube which was of tremendous help in finishing this report.

Certification

This is to certify that the project titled “Analysis and Simulation of an Active Suspension System” was carried out by AISOSA ARASOMWAN with mat number ENG1503884, UZOAMAKAH JOSHUA with mat number ENG1503946, OSAGIE SARAH with mat number ENG1503933 and OSADOLOR WANPALI DUKE with mat number ENG1503932.

Dr. N. Enoma

Project supervisor

Sign________________

Date:

Dr. S.A Aliu

Head of Department

Sign______________

Date:

Abstract

With the rising need to maximize passenger's comfort, especially with the rising demand for self-driving vehicles, alternatives to the popular passive suspension system have been on the rise. In order to improve the stability of a vehicle and reduce the vibration transferred to the passengers of the vehicle due to different road profiles, it has become necessary to implement a smarter type of suspension system that can respond to different type of road profiles and provide improved damping experience. This type of suspension system is the active suspension system.

In order to analyze and simulate an active suspension system, parameters like the sprung and unsprung mass, spring and tire stiffness, damping constant of the damper were accounted for. Then the mathematical model of the system in the time domain was generated. Mathematical model was transformed from the time domain to frequency domain, using the Laplace transform. Then an appropriate controller for stabilizing the system was obtained.

The simulated results show that the active suspension system performs better than the passive suspension in terms of settling time, rise time and overshoot and had lesser vibrations was transmitted to the passengers.

List of Tables

Table of Figures

Table of Contents

Chapter 1

1 Introduction

1.1 Background to Study

Suspension system is a group of tires, springs, shock absorbers and linkages that connects a vehicle to its wheel and allow relative motion between the two, basically the suspension system is a group of mechanical components that connects the wheel of a vehicle to the main body (Tata, 2012)

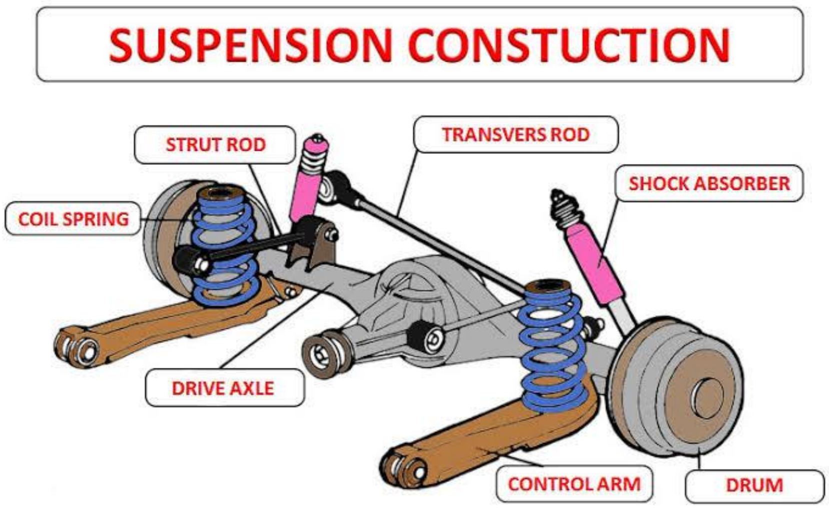


Figure 1-1: schematic of a suspension system

The suspension system keeps the displacement of the vehicle body to a minimum caused by disturbance on the road, it aims to provide a seemingly smooth ride for the driver and passenger of the car.

The suspension system works on the principle of force dissipation, that is, it converts energy into heat thus removing the impact that force would have had. It uses springs, dampers and structs to achieve this. A spring hold/store the energy while a damper converts the energy into heat. The springs stores the energy generated when a car passes through a road disturbance by compressing its size. The springs can't store this energy forever and will definitely have to release the energy sooner or later, this is where the dampers comes in, inside the dampers there is a piston with small holes In it and some pressurized oil, when the damper receives the energy stored by the springs it causes the piston to move through the oil generating heat thus successfully converting the energy stored in the springs to heat.

In summary the springs receives and stores the energy which the wheels receive after going through a disturbance in the road, it then transfers this energy to the dampers which convert it to heat, successfully preventing the initial energy gotten from the bump from getting to the vehicle frame thus allowing for a smoother ride.

There are three major types of suspension system;

- Passive suspension system
- Semi active suspension system
- Active suspension system

For the purpose of this project we would only be focused on the active suspension system and comparing its performance to that of the passive suspension system.

In the passive suspension system, the shock absorbers cannot provide energy to the suspension system and can only limit the sprung and sprung mass velocities and displacements by an amount pre-determined by the manufacturer, hence the quality of a passive suspension system varies with different road profiles (Bhise, et al., 2016) , the active suspension system on the other hand includes an actuators which supplies an active force to the suspension system regulated by controlling algorithms which uses information gathered by attached vehicle sensors. This type of suspension has much better reacting capabilities against generated vertical forces caused by unpredictable road input changes since the dampers as well as the springs are regulated through an actuator force. (Riduan, et al., 2018). The semi-active or adaptive suspension work by varying the shock absorber firmness to match the change in road conditions

An active suspension system would make it possible to design vehicles which are suitable for a large variety of road profiles.

Matlab would be used extensively within this project to analyze our system, design our controller and simulate our results.

Requirement of suspension system

1. Independent movement of each of the wheel on the axle
2. The introduction of wheel force into the body in a manner favorable to the flow of forces
3. Behavior with regards to the passive safety of passengers and other road users
4. To preserve stability of the vehicle in pitching and rolling while in motion
5. Small unsparing masses of the suspension in order to keep wheel load fluctuation as low as possible

1.2 Statement of Problem

To derive the necessary parameters of a suspension system, develop a mathematical model of the suspension system and from that develop the detailed design of an active suspension system and simulate our result.

1.3 Aim

To analyze and simulate an active suspension system using matlab with better system performance than that of the passive suspension system.

1.4 Research Objective

- To derive the mathematical model of both the passive and active suspension
- Analyze the active suspension system
- Design the control system for the active suspension system
- Compare the system performance of the active and passive suspension system.

1.5 Scope of Work

In this project we intend to analyze and simulated an active suspension system using matlab and specifically the control system designer app within it to design the control system for the active suspension and finally compare and see how much better our active suspension model performed than a passive suspension.

1.6 Limitations of the Study

As a result of the complexity of the mathematical and Simulink model that would arise if we work with the complete model of the vehicle, we were only able to work with the quarter mass model of the vehicle. We used a quarter mass model which did not take into consideration the difference in deflection between the rear and front part of the car, our model assumed the vehicle had the same level of suspension throughout and that the front, rear and side of the car were all subjected to the same force. We also didn't include disturbance that may occur in real life applications like friction or noise and disturbance that may enter the system either from and external source or from the sensors.

Chapter 2
Literature review

2 Literature Review

Similar analysis to this project were studied and the various method used in carrying out their simulation and analysis were compared to the ones used in this project, from the way we arrived at the sprung and unsprung mass to the methods of simulation used.

2.1 Sprung and Unsprung Mass

The sprung mass refers to the sum of all the mass above the suspension system, it includes the mass of the passengers in the car and the mass of the all the vehicle part above the suspension system. The unsprung mass is the reverse case and refers to the sum of all the mass below the suspension system, it includes the mass of the rear axle assembly, steering knuckles, front axle and wheels. Some studies considered the sprung mass as the sum of the vehicle components above the suspension system and a biodynamic model of the human body. A lumped mass-spring system was used as the biodynamic model the human body, in this model the human body was thought of as a spring and damper system with degree of freedom (DOF) ranging from 1 to 4. In the lumped mass–spring–damper system models the human body was represented by degree of freedom starting from 1 DOF into 15 DOF, which typically depends on the number of bodies lumped masses connected through the springs and dampers

For simplicity, the majority of studies have considered the seat suspension system model with a seated driver body as 1 DOF, which is mainly composed of a single stiffness, mass and damper, while some researchers like (Muksiar, 1976) have used 2 DOF to model the seat and human body. The 2 DOF system was widely used in the vibration field due to its simple structure and it can offer more information about the model dynamic parameters than the 1DOF system. Some studies have presented the human body system of neck, pelvis, head and torso as 4 DOF in addition to the 1 DOF seat structure. The 4 DOF is the best representation of the seated human body among all the DOF models because it can provide the most accurate parameter estimation of the human body system model, in a report by (Boileau, 1998) he used a 4DOF model of the human body to study how the human being responded to vertical displacement in the vehicle, in his report he studied how the amount of vibration/ acceleration experienced by the passengers in a vehicle varied based on their position in the vehicle, whether or not they are standing or if they are making use of a head rest.

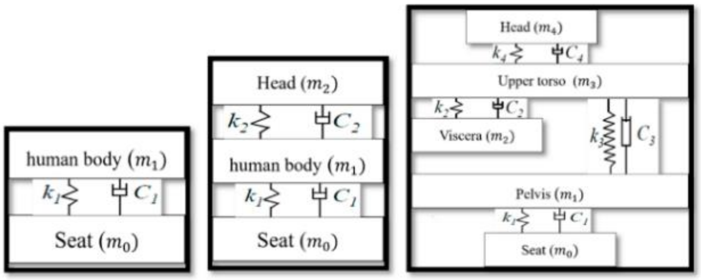


Figure 2-1:various degree of freedom of the sprung mass of a suspension system

1DOF 2DOF 4DOF

For the purpose of our analysis we computed the human mass as the average mass the specific vehicle is designed to carry, thus our sprung mass was calculated as the sum of this mass and the total mass of all the vehicle parts above the suspension system. We used this method because we are not designing a suspension system from scratch but rather, we were analyzing how the active suspension works and comparing it to the passive suspension, so this simplified method was both accurate to the extend we needed it to be and also time saving.

2.2 Transmission of Vibration

Some studies on suspension system (both passive and active) investigated the level of vibrations that was transmitted to the passenger in the car and how it affected their comfort using different metrics. However, for the purpose of this research we did not investigate this, we instead used our simulation to investigate the amount of deflection observed in the suspension during simulation and how they were damped out and draw from that the conclusion that the higher the deflection the higher the vibration that would be felt by the passenger of the car. This approach saved us critical time and reduced the complexity of our Simulink block model thereby reducing the risk of errors coming up.

Various methods have been developed to evaluate the ride comfort of the passengers (Padden, 2002). The most popular method is the ISO 2631-1: 1997 which is proposed by the international organization for standardization (ISO) (ISO, 1997). This method is the most widely used to evaluate the WBV using the calculation of the weighted RMS acceleration as defined in Equation (1).

$$a_{rms,weighted} = \left[\Sigma \left(W_{(\acute{\omega})} a_{rms}(\acute{\omega})\right)^2\right]^{0.5}$$

Where a_{rms} present the RMS acceleration at the frequency ώ, and W(ώ) is the weighting factor which is plotted in a diagram stated in the ISO 2631:1997. The value of $a_{rms,weighted}$ is compared with the level of comfort in the table which is specified in ISO 2631:1997. The table below presents the comfort level scales suggested by the ISO 2631standard.

Table 2-1:ISO standards for comfort levels

Weighted Acceleration(m/s^2)	ISO Comfort Level
<0.315	Not uncomfortable
0.315-0.63	A little uncomfortable
0.5-1	Fairly uncomfortable
0.8-1.6	Uncomfortable
1.25-2.5	Very uncomfortable
>2	Extremely uncomfortable

2.3 Mass Model Used to Generate Mathematical Model

When it comes to the analysis of a vehicle suspension system there are three major type used;

- Half mass models
- Quarter mass model
- Full mass model

The first two types work on the principle that the overall weight of the vehicle is distributed uniformly throughout the car and thus for the purpose of generating a mathematical model we could consider on or two of the vehicle wheels and assume that wheel carries a quarter or half of the vehicle mass. The full mass model however considers all wheels in their analysis. Of all three models the full mass model was scarcely used most likely because of its complexity and how cumbersome it mathematical and Simulink model would be; however, this model was used by A. Bala Raju and R. Venkatachalam in their analysis of vibration of an automobile suspension system. With the full mass model, the deflection on the front and back of the vehicle as well as the sides can be measured. A couple of other studies carried out their analysis using the half mass model. This model has the advantage of being simpler than the full mass model and at the same time provide more details than the quarter mass model. This model had the advantage of being able account for the vehicle deflection in the front and rear. However for the purpose of our analysis we made use of the quarter mass model, a drawback to this model is that it only considered the deflection on a quarter part of the vehicle and assumes it was the same all through the vehicle even though the suspension in the rear and front of most vehicles are made different and the deflection from the road may be different in the front and rear of the vehicle. These drawbacks were of little consequences to us because we were not building an actual suspension system but rather analyzing and simulating the behavior of the active suspension system and compare it to the passive suspension, so whist it may not have accounted for the deflection in the front and rear part of the vehicle it still gave us the general behavior of the suspension system while having the added advantage of being simpler to analyze and build a Simulink model.

2.4 Simulating Software

The majority of other literatures done on this subject or on subject related to it where done with Simulink. However, they are other alternatives to Simulink, the major alternatives are;

- Simscape
- Dymola

Simscape is a also a part of MATLAB, but unlike Simulink which is a block based modelling tool, simscape is a component and block based modelling tool, simscape has the advantage of providing more information about the system and the output of each component. However, Simulink is way easier to design with and require less user information/input about the components of the system. Simulink is capable of generating the necessary information we need for this project without the added complexities.

Dymola on the other hand is a tool in modelica and like Simulink it is a block and component-based modelling tool which introduce other complexities and extra information which we do not need for the purpose of our design.

2.5 Controller Used

Like every other automatic system, the active suspension requires a controller to make sure the output (in this case the vehicle deflection/displacement) is within a certain limit. The controller works by measuring the current output via a sensor, taking that value into a controller unit where the deviation from the desired output is measured and a gain matrix used to bring it back to the desired value which is then fed back into the system.

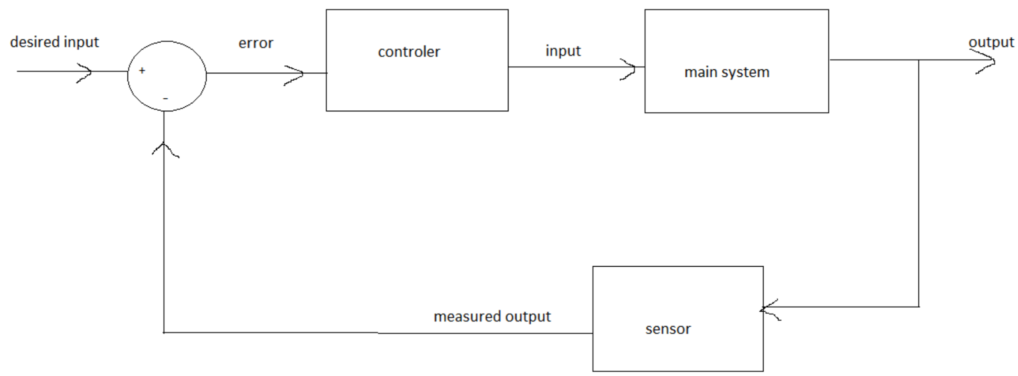


Figure 2-2:block diagram representation of negative feedback control system

We want the controller to minimize the error signal as small as possible. The error signals of an active suspension system are;

- Acceleration of the sprung and unsprung mass
- Displacement of the sprung and unsprung mass

The controller can modify how the road disturbance affect the vertical acceleration and displacement by driving the actuator which raise and lower the suspension. The goal is to drive the actuator such that the effect of the road disturbance on the vehicle is minimized while using the least amount of actuator force as possible.

There are various methods of controlling a system, some of which are;

- PID controller
- LQR controller
- Fuzzy logic controller etc.

For this project we made use of the PID controller, however various researchers have made use of different type of controllers like (Yatak & Sahin, 2020) who made use of a hybrid type-2 fuzzy logic controller to study the tradeoff between the ride comfort and road handling of an active suspension, they made use of full vehicle model with two controllers, one road comfort controller and another road handling controller. The total output of the hybrid controller was the sum of the two individual controllers weighted with a weighting coefficient which was between 0-1. (Ibrahim, et al., 2018) made use of both fuzzy logic controller and linear quadratic regulator to see which one had a better performance. In his research he concluded that fuzzy logic controller had an overshoot of 3.3% and settling time of 0.12 seconds while the linear quadratic regulator had an overshoot of 1.5% and a settling time of 0.24seconds, the fuzzy logic had a quicker rise time but a higher overshoot while the linear quadratic regulator had a slower settling time but lower overshoot, that is with fuzzy logic we get a higher vibrational amplitude but it dampens out quicker while with linear quadratic regulator we get a lower vibrational amplitude but it dampens out slower.

Chapter 3

3 Methodology

In order to analyze and simulate an active suspension system, the following steps was adopted:

- **System parameters**: Before proceeding we have to first to derive the necessary parameters for any suspension system like the sprung and unsprung mass, spring stiffness, tire stiffness and damping constant.
- **Mathematical model**: This is where we use first principle to derive the mathematical model that governs how system work.
- **Simulink model**: Using MATLAB's Simulink we can now develop a Simulink model from our mathematical model using block diagrams.
- **Feedback control system and the PID controller**: With the use of the control system designer app and Simulink in matlab we can design our feedback control system and the PID control system needed to stabilize it.

3.1 Stiffness of Spring and Damping Constant of Damper

Model: Audi A8 2015 3.0TFSi (because it utilizes an active suspension system)

Specification (Auto123, 2016)

Gross mass: 2675kg

Sprung mass: 2045kg

Unsprung mass: 630kg

3.2 Spring Stiffness

To maintain a proper balance between passenger comfort and proper road handling suspension frequency of the sprung mass should be between 1Hz and 3Hz (Barak, 1991) (Felipe Mrad, 2018), for the purpose of our analysis we have choose a frequency of 2.5Hz

$$f = \frac{1}{2\pi}\sqrt{\frac{g}{\delta}} = 2.5$$

Where δ = maximum deflection of spring

Making δ subject formula we get

$$\delta = 0.03976\text{m}$$

the stiffness of the spring is now given by

$$k = \frac{F}{\delta}$$

Where;

k= stiffness

F= force on spring = mg

m = quarter mass of car

$$m = \frac{\text{sprung mass}}{4} = \frac{2045}{4} = 511.25\text{kg}$$

$$k = \frac{511.25 \times 9.81}{0.03976} = 126.140\text{KN/m}$$

3.3 Damping Constant

Quarter mass m = 511.25kg

Spring stiffness = 126.140KN/m

Assuming that the amplitude is to be reduced to one-hundredth in one cycle

$$x_2 = \frac{x_1}{100}$$

Damping ratio ζ is given by

$$\ln\frac{x_1}{x_2} = \ln 100 = \frac{2\pi\zeta}{\sqrt{1-\zeta^2}}$$

$$\zeta = 0.5912$$

The damping constant is given by;

$$b = \zeta \times c_c$$

Where c_c is the critical damping constant, and is given by;

$$c_c = 2\sqrt{m \times k} = 2\sqrt{511.25 \times 126140} = 16061 Ns/m$$

The damping constant is thus given by;

$$b = 0.5912 \times 16061 = 9495 Ns/m$$

Spring stiffness	126140N/m
Damping constant	9495Ns/m
Sprung mass	2045kg
Unsprung mass	630kg

3.4 Tire Stiffness and Damping Constant

For the purpose of this analysis we have decided to give the tire a negligible damping coefficient and only work on its stiffness.

Tire stiffness

The only accurate way to establish the actual stiffness of a tire is to test the tire, which is an expensive procedure, from experimental work carried out by Liunan Yang on the relationship between tire size and stiffness (Yang, et al., 2019), we can safely estimate our tire stiffness as 350KN/m

Tire stiffness: 350KN/m

3.5 Mathematical Model of the Suspension System

The active suspension system is basically a passive suspension system with a controller and actuator attached to it, so our mathematical model is going to be almost the same as that of the passive suspension system.

Assumptions

- Mass are constant, that is no one is getting in or off the vehicle
- Coulomb damping

- Linear spring stiffness
- Tire is not rigid, that is, it has a stiffness but negligible damping

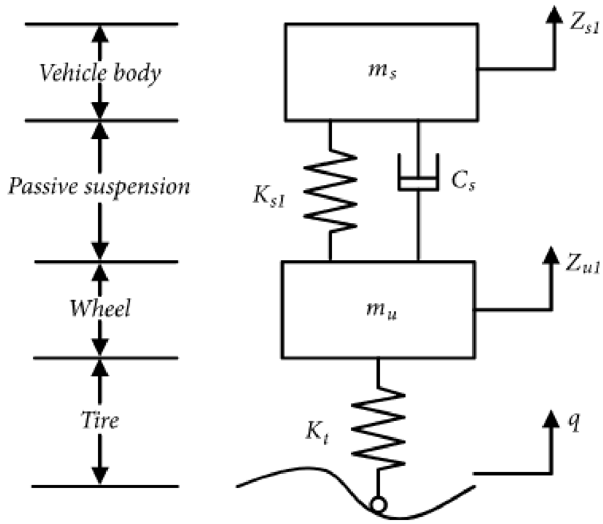
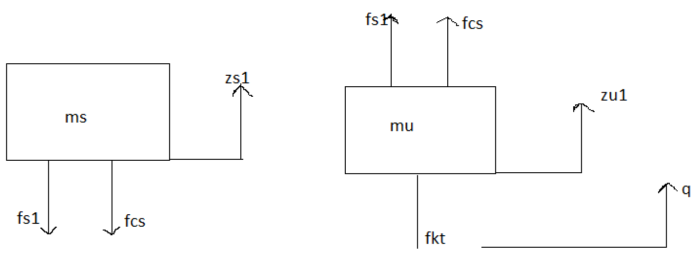


Figure 3-1: schematic representation of a passive suspension system

The free body diagram of the sprung and unsprung mass is given below;



For mass m_s

ΣF=Ma

$-F_{s2} - F_d = m_s \frac{d^2 z_{s1}}{dt^2}$

$$-K_{s1}(z_{s1} - z_{u1}) - b\left(\frac{dz_{s1}}{dt} - \frac{dz_{u1}}{dt}\right) = m_s \frac{d^2 z_{s1}}{dt^2} \quad (1)$$

For mass m_u

ΣF=Ma

$F_{s2} + F_d - F_{kt} = m_u \frac{d^2 z_{u1}}{dt^2}$

$$K_{s1}(z_{s1} - z_{u1}) + b\left(\frac{dz_{s1}}{dt} - \frac{dz_{u1}}{dt}\right) - k_t(z_{u1} - q) = m_u \frac{d^2 z_{u1}}{dt^2} \quad (2)$$

Equation 1 and 2 can be rearranged as;

$$\frac{d^2 z_{s1}}{dt^2} = \frac{1}{m_s}\left[-K_{s1}(z_{s1} - z_{u1}) - b\left(\frac{dz_{s1}}{dt} - \frac{dz_{u1}}{dt}\right)\right] \quad (3)$$

$$\frac{d^2 z_{u1}}{dt^2} = \frac{1}{m_u}\left[K_{s1}(z_{s1} - z_{u1}) + b\left(\frac{dz_{s1}}{dt} - \frac{dz_{u1}}{dt}\right) - k_t(z_{u1} - q)\right] \quad (4)$$

From equation 3 and 4 we can how proceed to build a Simulink model of the system where;

q is the input from road

K_{s1} is the stiffness of the suspension spring

k_t is the stiffness of the tire

b is the damping constant of the damper

z_{s1} is the displacement of the sprung mass

z_{u1} is the displacement of the unsprung mass

m_s is the sprung mass

m_u is the unsprung mass

Table 3-1: parameters for the suspension system

Q	**-**
K_{s1}	126140N/m
k_t	350000N/m
b	9495Ns/m
z_{s1}	-
z_{u1}	-
m_s	2045kg
m_u	630kg

The Simulink model (passive suspension)

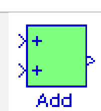
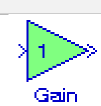
We are going to simulate 2 different models each with a different road profile but with all other parameter remaining constant. The 5 different road profiles we would be working with are;

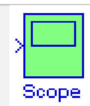
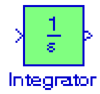
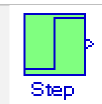
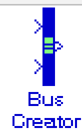
- Bumpy roads
- Pot holes

Simulink block used

The major Simulink block used are; (Karris, 2006)

Table 3-2: blocks used for the passive suspension Simulink model

Block name	Symbol	Function
Add block		To add two quantities together, it can carry out subtraction by changing one of the plus sign to a minus
Gain block		Multiplies the input by a constant value

Scope		Displays the output as a graph
Integrator		Integrates the input
Step function		Generate a step between two defined values
Bus creator		Displays the graph of multiple outputs in a single graph

Note: A step function is usually used as the road input to show a proof of concept. Although other functions can also be used. A step function just shows an exponential and sudden increase in the displacement of the road and back to levelness.

Below is a Simulink model of a passive suspension system with a step function taken as the input

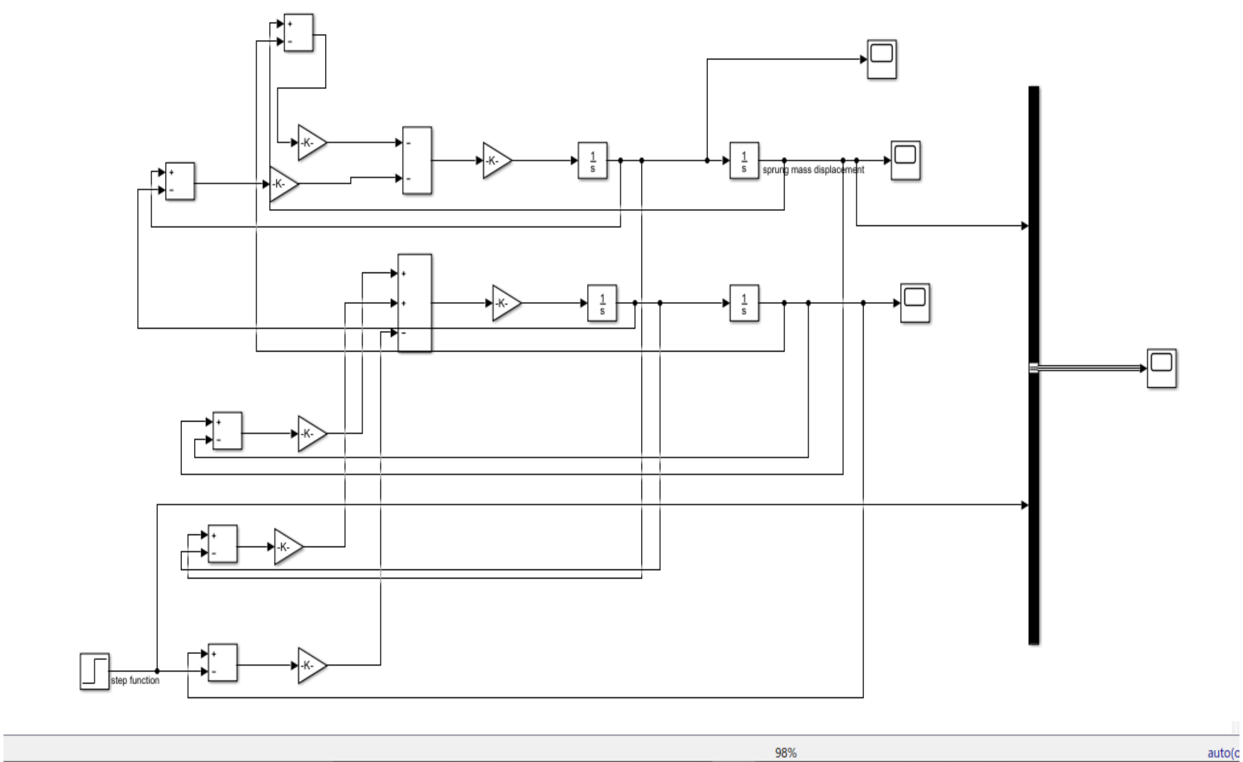


Figure 3-2:Simulink model of the passive suspension system

3.6 The Active Suspension Model

In the active suspension model an actuator is connected in parallel to the damper and springs. The job of the actuator is to add an external force to the system when needed to stabilize the system faster. The amount of force generated by the actuator is not a fixed quantity but varies depending in the type of input to the system. In order to control the amount of force the actuator generates; a feedback controller is needed. The general schematic of an active suspension system is shown below;

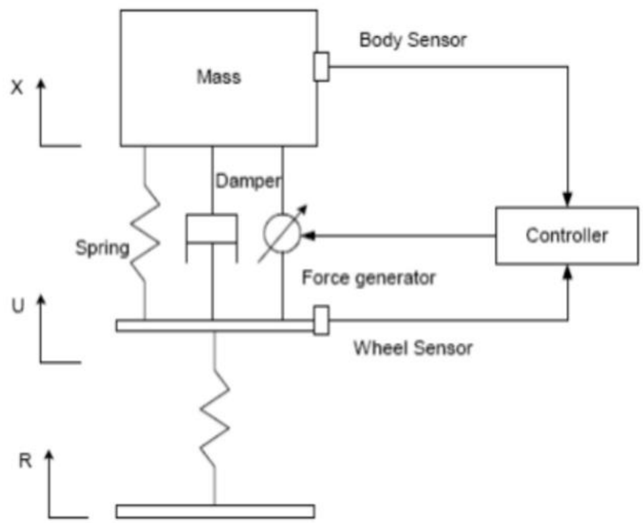


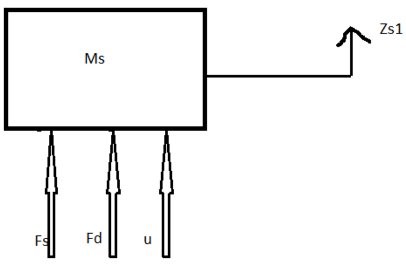
Figure 3-3: schematic representation of the active suspension system

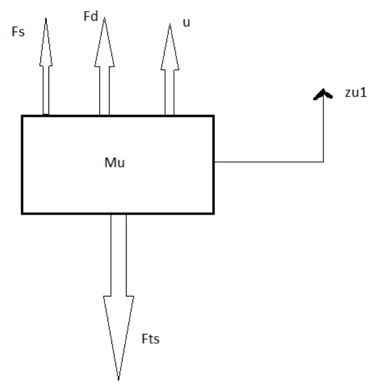
from the schematic above, the active suspension systems make use of three new components;

- A sensor
- A controller
- Actuator

The sensor measures the disturbance in the road and sends the information gathered to the controller which compares the disturbance in the road to a reference signal and calculates the amount of force needed to be inputted in the system by the actuator. After receiving its command form the controller, the actuator then add the required force the system needs to dissipate the vibrations faster and steadily to ensure passenger comfort.

Free body diagram and mathematical model**:** The figure below shows the figure body diagram of the active suspension system





Where;

F_s is the spring force on the sprung and unsprung mass

F_d is the damping force on the sprung and unsprung mass

F_{ts} is the tire damping force

u is the actuating force on the sprung and unsprung mass

m_u is the unsprung mass

m_s is the sprung mass

from the free body diagram, we can now proceed to generate the mathematical model of the active suspension model

For mass m_s

ΣF=Ma

$-F_{s2} - F_d = m_s \frac{d^2 z_{s1}}{dt^2}$

$$-K_{s1}(z_{s1} - z_{u1}) - b\left(\frac{dz_{s1}}{dt} - \frac{dz_{u1}}{dt}\right) + u = m_s \frac{d^2 z_{s1}}{dt^2} \quad (5)$$

For mass m_u

ΣF=Ma

$F_{s2} + F_d - F_{kt} = m_u \frac{d^2 z_{u1}}{dt^2}$

$$K_{s1}(z_{s1} - z_{u1}) + b\left(\frac{dz_{s1}}{dt} - \frac{dz_{u1}}{dt}\right) - k_t(z_{u1} - q) - u = m_u \frac{d^2 z_{u1}}{dt^2} \quad (6)$$

Equation 1 and 2 can be rearranged as;

$$\frac{d^2 z_{s1}}{dt^2} = \frac{1}{m_s}\left[-K_{s1}(z_{s1} - z_{u1}) - b\left(\frac{dz_{s1}}{dt} - \frac{dz_{u1}}{dt}\right) + u\right] \quad (7)$$

$$\frac{d^2 z_{u1}}{dt^2} = \frac{1}{m_u}\left[K_{s1}(z_{s1} - z_{u1}) + b\left(\frac{dz_{s1}}{dt} - \frac{dz_{u1}}{dt}\right) - k_t(z_{u1} - q) - u\right] \quad (8)$$

q is the input from road

K_{s1} is the stiffness of the suspension spring

k_t is the stiffness of the tire

b is the damping constant of the damper

z_{s1} is the displacement of the sprung mass

z_{u1} is the displacement of the unsprung mass

m_s is the sprung mass

m_u is the unsprung mass

u is the actuating force

Table 3-3: parameters for the active suspension system

q	Road profile
K_{s1}	126140N/m
k_t	350000N/m
b	9495Ns/m
z_{s1}	To be computed
z_{u1}	To be computed
m_s	2045kg
m_u	630kg
u	Depends on controller

3.7 The Feedback Control System

Feedback is the property of a closed loop system which permits the output or some other control variable to be compared with the input of the system so that an appropriate control action may be formed as some function of the output and input. (Distefano, et al., 1996)

A closed loop system is one in which the control action is somehow dependent on the output of the plant. (Distefano, et al., 1996)

The controller is most likely the most important component in the active suspension system. Its job is to determine the amount of force the system needs to achieve stability faster and comfortably. It does this my measure the taking in an error signal as its input and calculates the amount of force it needs to send into the system via the actuating. The way it does this calculation depends on the type of controller used.

The error signal is difference between the current output signal and a reference signal.

The reference signal is an elemental signal applied to the feedback summing point in order to compare a specified action of the plant. It usually represents ideal or desired plant output behavior.

The plant is the system, subsystem process or object controlled by the feedback control system.

Below is a general schematic of a feedback controller for an active suspension system;

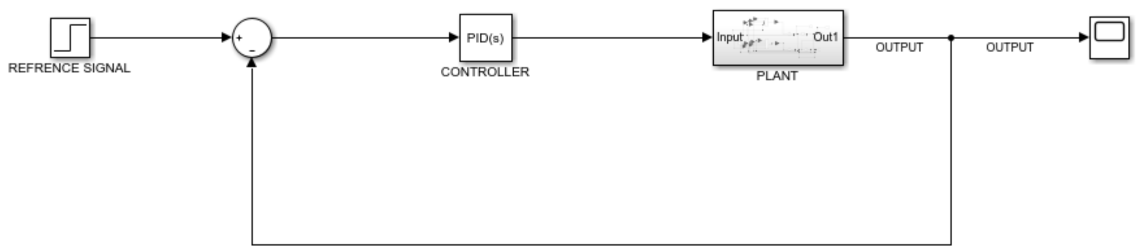


Figure 3-4: block diagram representation of the active suspension system

3.8 The PID Controller

In control design there are numerous types of controller that could be used, but for the purpose of this design we opted to use the PID controller because;

- It is the most widely used in the industry
- It accurate enough for the purpose of our design
- It is easier to design and tune.

A PID controller calculates an error value as the difference between a measured process value and a desired set point. The controller attempts to minimize the error by adjusting the process through the use of a manipulated variable.

The PID controller algorithm involves 3 separate constant parameters; the proportional, integral and derivative values. The proportional parameters depend on the current error, the integral on the accumulation of past errors while the derivative on the prediction of future errors (MATHWork, n.d.). The weighted sum of the three gains is used to adjust the process by a control element like the position of a valve or in our case the amount of force supplied by an actuator.

The response of a controller can be described in terms of the responsiveness of the controller to an error, the degree to which it overshoots the setpoint and the degree of system oscillation.

The proportional, integral and derivative term are summed to calculate the output u(t) of the controller via the equation;

$$u(t) = k_c \left[e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau \right] \times \left[1 + T_d \frac{d}{dt} e(t) \right]$$

Where;

$k_p = k_c$

$k_i = \frac{k_c}{T_i}$

$k_d = k_c T_i$

e(t) = r(t)-y(t)

3.8.1 The Proportional Term

The proportional term produces an output value that is proportional to the current error value

$$p_{out} = k_p e(t)$$

A large proportional gain k_p will result in a large in output for a given error signal input which could result in system instability, on the other hand a small proportional gain will result in a small output response for a given error signal input, this could result in a controller with very low sensitivity and responsiveness.

3.8.2 The Integral Term

The contribution of the integral term is proportional to both the magnitude of the error and the duration of the error. The integral gain is the sum of the instantaneous error over time:

$$I_{out} = k_i \int_0^t e(\tau)d\tau$$

3.8.3 Derivative Term

The derivative of the process error is calculated by determining the slope of the error over time and multiplying the rate of change by the derivative gain

$$D_{out} = k_d \times \frac{d}{dt} e(\tau)$$

The derivative term predicts system behavior to improve settling and stability time of the system. Derivative action is seldom used in practice though as the proportional and integral term usually suffice (Nise, 2015)

3.8.4 Tuning the PID Controller

Tuning the PID controller is the process of choosing the appropriate gains or values for the proportional, integral and derivative term i.e. choosing the value of $k_p, k_i \; and \; k_d$ needed to stabilize the system to a required level.

There are several methods used to tune the PID controller, some of which include;

- Manual tuning
- Ziegler Nichols method
- Cohen coon method
- Root locus method
- Using the PID tuning app in MATLAB

For this project we would be making use of the root locus method in which we use the control system designer app in Matlab to move around the roots of closed system transfer function until we get the desired system requirement we are after. From there we can take down the appropriate value of Kp, Ki and Kd needed for our controller which is automatically generate to a Simulink model from the control system app directly.

System Requirement for an Active Suspension System (Ibrahim, et al., 2018)

Table 3-4: system requirement for an active suspension system

Overshoot	35% or better
Settling time	2.3 seconds or better
Rise time	0.25 seconds or better

3.9 Simulating the Road Profile

Another important aspect of this analysis is the input to the system which in this case is the road profile.

Two road profiles were used;

- **Speed bumps:**

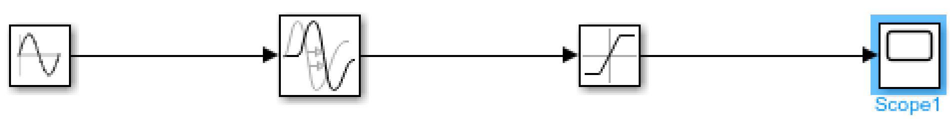


Figure 3-5:Simulink model of a speed bump

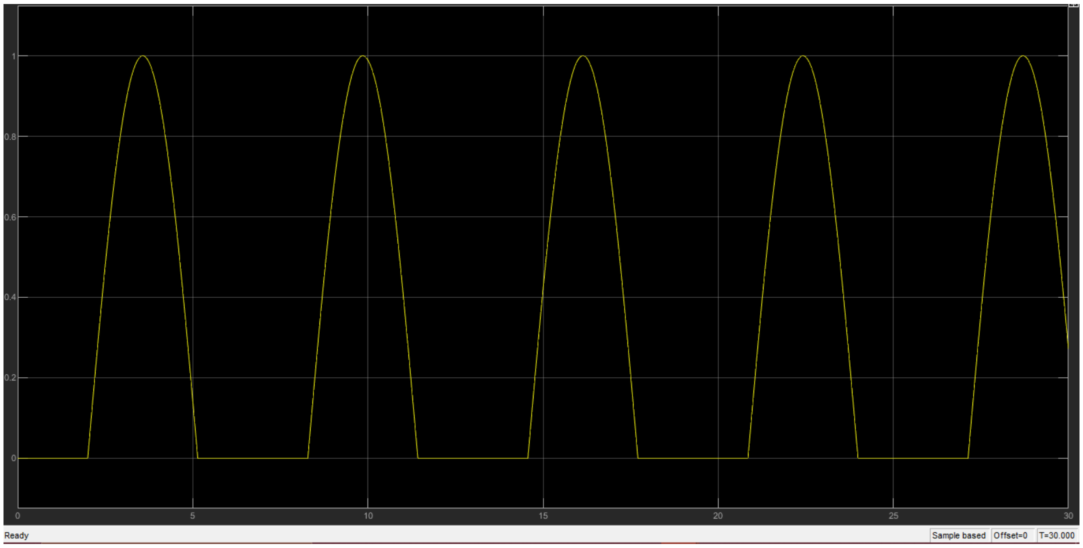


Figure 3-6: graphical representation of a speed bump

- **Pot holes:**

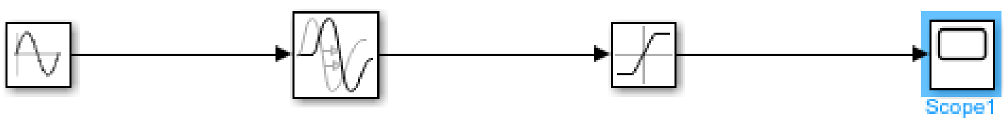


Figure 3-7: Simulink model of a pothole

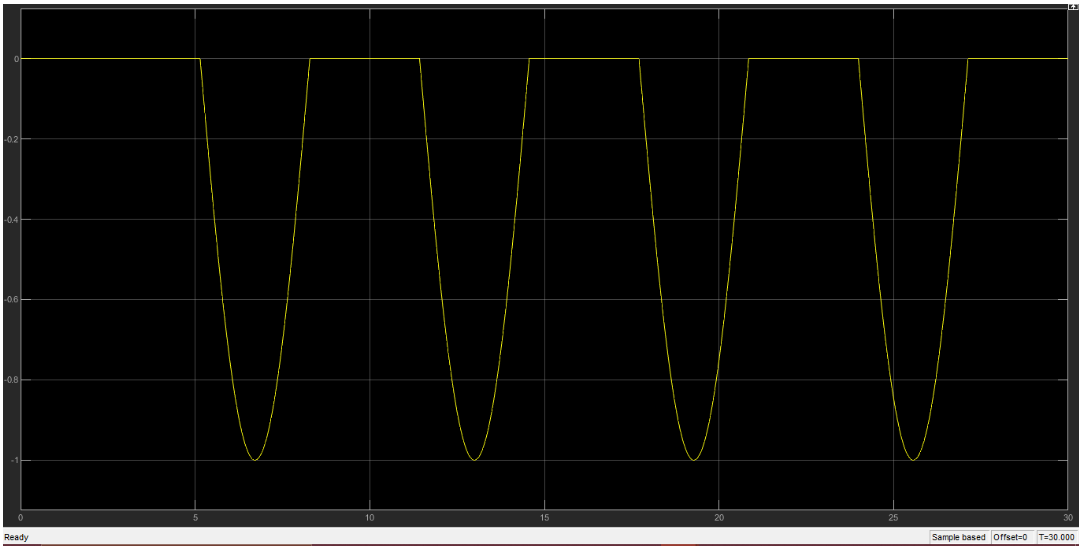


Figure 3-8: graphical representation of a pothole

Table 6: blocks used in the Simulink model of the road profile

Name of block	**function**
Sine wave	This is the first block; it was used to generate a sinusoidal oscillation
Transport delay	This is the block right after the sine wave block. Its purpose was to delay the initial output of the sine wave by a specific time. In order words it starts oscillation at a point other than zero
Saturation	This block came after the transport delay block and its purpose was to set an upper and lower fix point of our output. For the speed bump the lower fixpoint was set to zero, i.e. any output less than zero would automatically be round up to zero, while the

	upper fix point was the same as the amplitude of the sin wave so no action was taken. The reverse was done for the pot holes
Scope	This is the final block and is used to display our simulated result.

3.10 Simulink Model of the Passive Suspension Model to Both Road Input

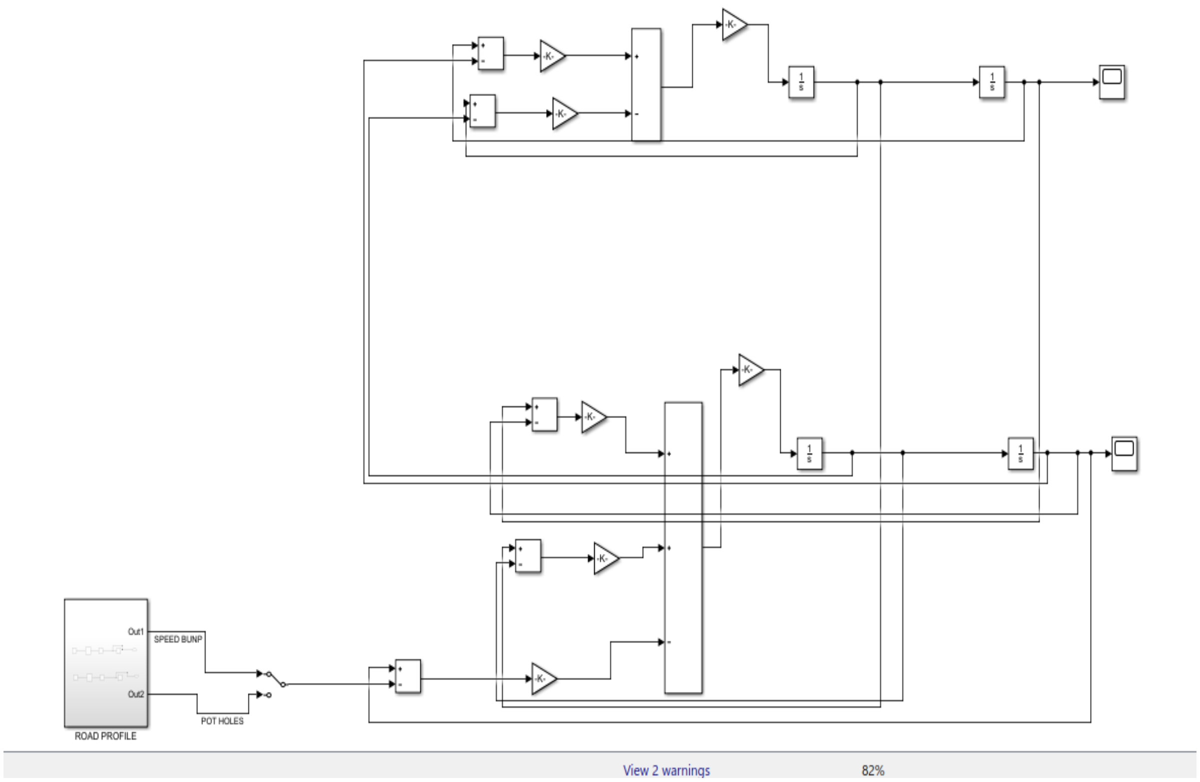


Figure 3-9: Simulink model of the passive suspension to both road profiles

3.11 Designing the Active Suspension Model

The active suspension model is designed using the control system designer app in matlab. The major difference between this model and that of the passive suspension model is the presence of an actuator whose force input is determined by the PID controller output. In order to tune or PID controller we first need to create the transfer function representation of our open loop system or in this case our passive suspension model.

3.11.1 The transfer function representation

Transfer function of a system is the ratio of the Laplace transform of the input to the Laplace transform of the output.

From our mathematical model of the open loop system;

$$\frac{d^2 z_{s1}}{dt^2} = \frac{1}{m_s}\left[-K_{s1}(z_{s1} - z_{u1}) - b\left(\frac{dz_{s1}}{dt} - \frac{dz_{u1}}{dt}\right)\right] \qquad (9)$$

$$\frac{d^2 z_{u1}}{dt^2} = \frac{1}{m_u}\left[K_{s1}(z_{s1} - z_{u1}) + b\left(\frac{dz_{s1}}{dt} - \frac{dz_{u1}}{dt}\right) - k_t(z_{u1} - q)\right] \qquad (10)$$

Our system is a single input multiple output (SIMO) system as a result our transfer function would be in a matrix form, showing how the input affect each output. However, for this report we are concerned on just the sprung and unsprung mass displacement and will thus be generating the transfer function of just the sprung mass output.

Transfer function for sprung mass displacement

Equation 9 and 10 can be rewritten in the form below;

$$m_s \ddot{z}_{s1} = -k_{s1}(z_{s1} - z_{u1}) - b(\dot{z}_{s1} - \dot{z}_{u1}) \qquad (11)$$

$$m_u \ddot{z}_{u1} = k_{s1}(z_{s1} - z_{u1}) + b(\dot{z}_{s1} - \dot{z}_{u1}) - k_t(z_{u1} - q) \quad (12)$$

Rewriting equation 11 and 12

$$m_s \ddot{z}_{s1} + k_{s1} z_{s1} + b\dot{z}_{s1} = (k_{s1} z_{u1}) - (b\,\dot{z}_{u1}) \qquad (13)$$

$$m_u \ddot{z}_{u1} + (k_{s1} + k_t) z_{u1} + b\dot{z}_{u1} = k_{s1} z_{s1} + b\dot{z}_{s1} - k_t(q) \qquad (14)$$

Taking Laplace transform (assuming initial conditions are zero)

$$Z_{s1}(s)[m_s s^2 + bs + k_{s1}] = Z_{u1}[bs + k_{s1}] \qquad (15)$$

$$Z_{u1}(s)[m_u s^2 + bs + (k_{s1} + k_t)] = Z_{s1}(s)(k_{s1} + bs) + k_t Q(s) \qquad (16)$$

Solving for $Z_{u1}(s)$ from equation 15

$$Z_{u1}(s) = \frac{Z_{s1}(s)[m_s s^2 + bs + k_{s1}]}{[bs + k_{s1}]} \qquad (17)$$

Plug in $Z_{u1}(s)$ into equation 16

$$= \frac{Z_{s1}(s)[m_s s^2 + bs + k_{s1}]}{[bs + k_{s1}]} \times [m_u s^2 + bs + (k_{s1} + k_t)] = Z_{s1}(s)(k_{s1} + bs) + k_t Q(s)$$

Solve for $\frac{Z_{s1}(s)}{Q(s)}$ and simplify

$$\frac{Z_{s1}(s)}{Q(s)} = \frac{k_t(bs + k_{s1})}{m_u m_s s^4 + (m_u + m_s)bs^3 + [k_t m_s + (m_u + m_s)k_{s1}]s^2 + k_t bs + k_t k_{s1}}$$

The highlighted equation above is the open loop transfer function for our sprung mass oscillation.

$$G_1(s) = \frac{Z_{s1}(s)}{Q(s)} = \frac{k_t(bs + k_{s1})}{m_u m_s s^4 + (m_u + m_s)bs^3 + [k_t m_s + (m_u + m_s)k_{s1}]s^2 + k_t bs + k_t k_{s1}}$$

Transfer function for the unsprung mass oscillation

$$m_s \ddot{z}_{s1} = -k_{s1}(z_{s1} - z_{u1}) - b(\dot{z}_{s1} - \dot{z}_{u1}) \qquad (18)$$

$$m_u \ddot{z}_{u1} = k_{s1}(z_{s1} - z_{u1}) + b(\dot{z}_{s1} - \dot{z}_{u1}) - k_t(z_{u1} - q) \qquad (19)$$

Rewriting equation 11 and 12

$$m_s \ddot{z}_{s1} + k_{s1} z_{s1} + b\dot{z}_{s1} = (k_{s1} z_{u1}) - (b\,\dot{z}_{u1}) \qquad (20)$$

$$m_u \ddot{z}_{u1} + (k_{s1} + k_t) z_{u1} + b\dot{z}_{u1} = k_{s1} z_{s1} + b\dot{z}_{s1} - k_t(q) \qquad (21)$$

8

Taking Laplace transform (assuming initial conditions are zero)

$$Z_{s1}(s)[m_s s^2 + bs + k_{s1}] = Z_{u1}[bs + k_{s1}] \qquad (22)$$

$$Z_{u1}(s)[m_u s^2 + bs + (k_{s1} + k_t)] = Z_{s1}(s)(k_{s1} + bs) + k_t Q(s) \qquad (23)$$

Solving for $Z_{s1}(s)$

$$Z_{s1}(s) = \frac{Z_{u1}(s)[bs + k_{s1}]}{[m_s s^2 + bs + k_{s1}]}$$

Plug in $Z_{s1}(s)$ into equation 23

$$Z_{u1}(s)[m_u s^2 + bs + (k_{s1} + k_t)] = \frac{Z_{u1}(s)[bs + k_{s1}]}{[m_s s^2 + bs + k_{s1}]} + k_t Q(s)$$

$$Z_{u1}(s)[m_u s^2 + bs + (k_{s1} + k_t)] - \frac{Z_{u1}(s)[bs + k_{s1}]}{[m_s s^2 + bs + k_{s1}]} = k_t Q(s)$$

$$Z_{u1}(s)\left[[m_u s^2 + bs + (k_{s1} + k_t)] - \frac{[bs+k_{s1}]}{[m_s s^2+bs+k_{s1}]}\right] = k_t Q(s)$$

$$G_2(s) = \frac{Z_{u1}(s)}{Q(s)} = \frac{k_t}{\left[[m_u s^2 + bs + (k_{s1} + k_t)] - \frac{[bs + k_{s1}]}{[m_s s^2 + bs + k_{s1}]}\right]}$$

Simplifying we get;

$$G_2(s) = \frac{Z_{u1}(s)}{Q(s)} =$$

$$\frac{k_t(m_s s^2 + bs + k_{s1})}{m_u m_s s^4 + (m_u + m_s)bs^3 + \left(m_u k_{s1} + m_s k_{s1} + {k_{s1}}^2 + k_t m_s + b^2\right)s^2 + (k_{s1} + bk_{s1} + k_t - b)s + n}$$

Where;

$$n = k_t k_{s1} - k_{s1}$$

Parameters

q	Road profile
$\boldsymbol{K_{s1}}$	126140N/m
$\boldsymbol{k_t}$	350000N/m

b	9495Ns/m
z_{s1}	To be computed
z_{u1}	To be computed
m_s	2045kg
m_u	630kg

Substituting in the parameters from the table above gives us our transfer function for the sprung and unsprung mass oscillation. Because of the complexity of the two equations a matlab code was need to evaluate them. Below is the matlab code used and the results gotten:

```
%PROJECT TOPIC: ANALYSIS AND SIMULATION OF AN ACTIVE SUSPENSION SYSTEM
%PROJECT SUPERVISOR: Dr N. Enoma
%Done by: Joshua Uzoamakah
         %Osagie Sarah
         %Osadolor Wanpali Duke
         %Aisosa Arasomwan
%code to calculate the transfer function for the sprung and unsprung mass
oscillation

clc
clear
close all

%value for the parameters/symbols used
ks1=126140;        %suspension spring stiffness
kt=350000;          %tire stiffness
b=9495;             %damping constant of damper
ms=2045;            %sprung mass
mu=630;             %unsprung mass
n=(kt*ks1)-ks1;

num_sprung = [kt*b kt*ks1];    %numerator for the sprung mass transfer function
```

```
den_sprung=[mu*ms (mu+ms)*b (kt*ms+(mu+ms)*ks1) kt*b kt*ks1];      %denominator
for the sprung mass transfer function

num_unsprung=[kt*ms kt*b kt*ks1];             %numerator for the unsprung mass
transfer function
den_unsprung=[mu*ms       (mu+ms)*b       ((mu*ks1)+(ms*ks1)+ks1^2+(kt*ms)+b^2)
(ks1+(b*ks1)+kt-b) n]



%generating the transfer function
G1=tf(num_sprung,den_sprung)
G2=tf(num_unsprung,den_sprung)
```

Sprung mass open loop transfer function

$$G_1(s) = \frac{Z_{s1}(s)}{Q(s)} = \frac{3.323e09\ s + 4.415e10}{1.288e06\ s^4 + 2.54e07\ s^3 + 1.053e09\ s^2 + 3.323e09\ s + 4.415e10}$$

Unsprung mass open loop transfer function

$$G_2(s) = \frac{Z_{u1}(s)}{Q(s)} = \frac{7.158e08\ s^2 + 3.323e09\ s + 4.415e10}{1.288e06\ s^4 + 2.54e07\ s^3 + 1.053e09\ s^2 + 3.323e09\ s + 4.415e10}$$

3.12 The Root Locus

The root locus is a path drawn on the real-imaginary axis showing the path the poles take as they move from their initial position when the gain is zero to infinity when the gain is at infinity. When path of the root locus starts at the poles and ends at the zero location.

The poles are the roots of the transfer function that will cause the transfer function to be equal to infinity. It can be calculated by equating the denominator to zero and finding the roots. The zeros of the transfer function are the roots of the transfer function that will make the transfer function equal to zero. The zeros of the transfer function can be calculated by equating the numerator to zero and calculating the roots of the numerator. The stability of a system is dependent primarily on the position of the poles on the real-imaginary axis. A system is stable or marginally stable as long as the poles are on the negative real axis and stability begin to diminish as the as the poles move along the real axis towards the positive real axis and becomes completely unstable when the poles cross into the real axis. For transfer functions with more poles than zeros the remaining poles goes off to infinity to connect with the zero at infinity.

To generate the root locus of a transfer function we use the 'rlocus' command as shown below;

```
rlocus(G)
```

root locus of G1(s)

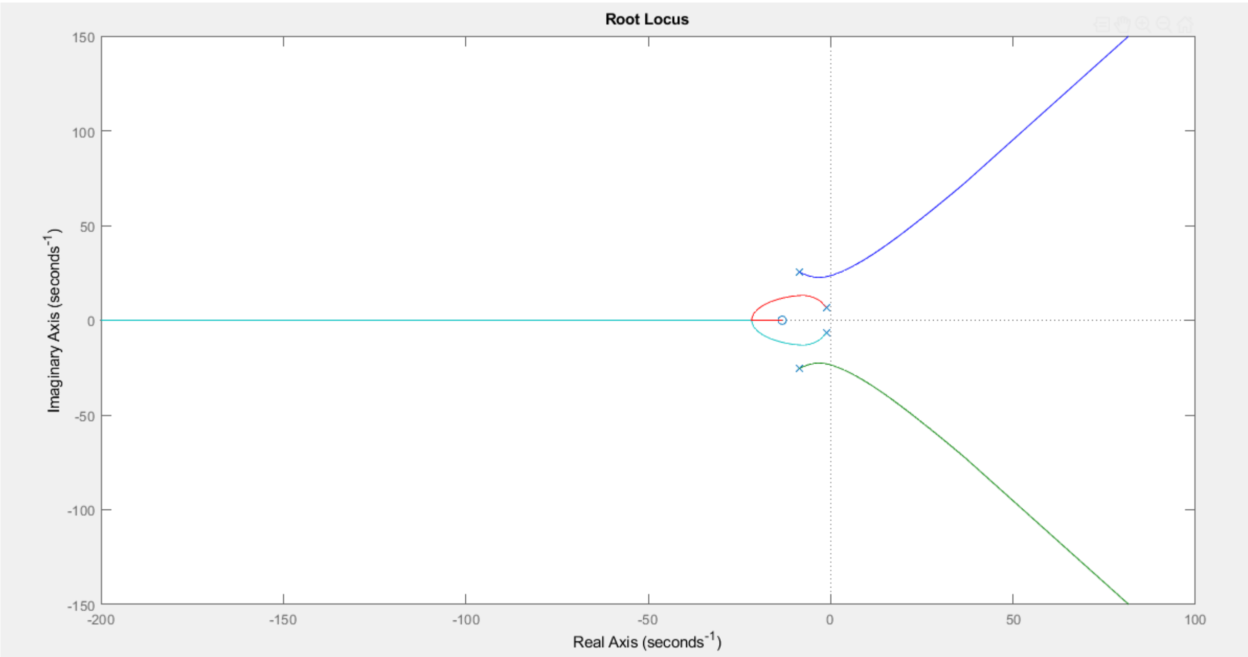


Figure 3-10: root locus of the passive suspension sprung mass

Root locus of G2(s)

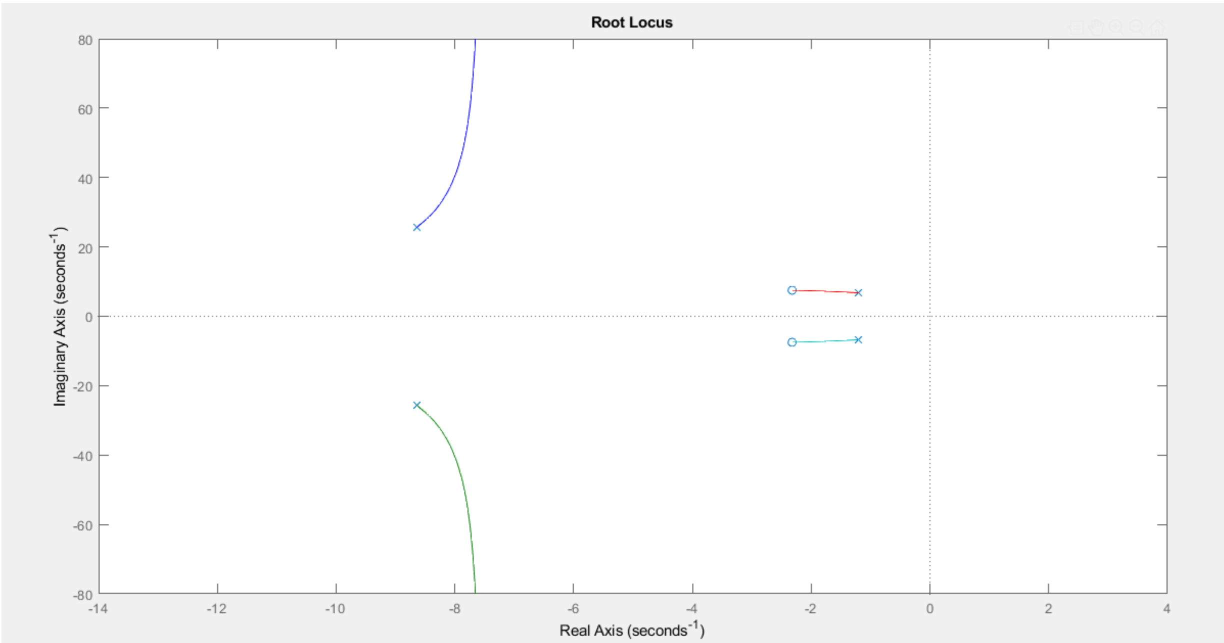


Figure 3-11: root locus of the passive suspension unsprung mass

3.13 Tuning the PID Controller Using the Root Locus Technique

In order to begin this process, we first have to launch the control system designer app and load in our plant transfer function by entering the command below on matlab;

```
controlSystemDesigner(G)
```

consider the PID-controlled closed loop system shown below:

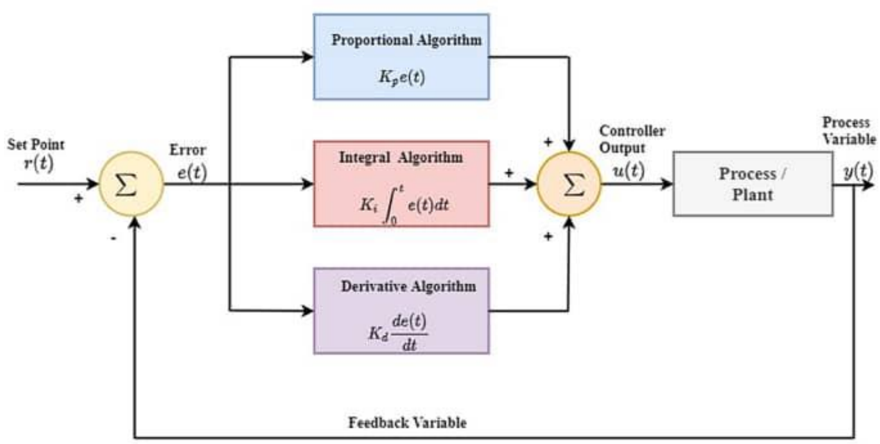


Figure 3-12: schematic of a PID controlled system

The root locus is the path the poles take as we vary the gains Kp, Ki and Kd from zero to infinity. With the control system designer app, we can manually move our poles around to monitor our plant response in real time until we get our desired response;

For all considerations we would be using G1(s) as our plant model

The transfer function for the PID controller is given by;

$G_3(s) = \frac{k_D s^2 + k_P s + k_I}{s}$

Remember the transfer function of our plant is:

$$G_1(s) = \frac{Z_{s1}(s)}{Q(s)} = \frac{k_t(bs + k_{s1})}{m_u m_s s^4 + (m_u + m_s)bs^3 + [k_t m_s + (m_u + m_s)k_{s1}]s^2 + k_t bs + k_t k_{s1}}$$

We can combine both of these transfer functions to form a closed loop transfer function with the formula

$$G = \frac{G_3 G_1}{1 + G_3 G_1} \qquad (24)$$

From equation we can see the poles of the new closed loop transfer function is going to be dependent on the value of Kp, Ki and Kd (the gains of our PID-controller)

With the root locus we are able to move around zeros, poles add or remove poles and zeros from the plant in order to properly vary the values of our gains needed to stabilize the plant.

After tuning our PID controller with the controlSystemDesigner we ended up with the following gains;

$$K_P = 1.5$$

$$K_I = 10$$

$$K_D = 0.1$$

Using matlab we are able to get the closed loop transfer function of our controlled plant as;

$$G = \frac{numerator}{denominator}$$

Where;

numerator = 6.116e14 s^8 + 2.936e16 s^7 + 1.17e18 s^6 + 2.494e19 s^5 + 3.029e20 s^4 + 2.379e21 s^3 + 1.196e22 s^2 + 3.978e22 s

denominator= 1.66e12 s^10 + 6.545e13 s^9 + 4.761e15 s^8 + 1.107e17 s^7 + 3.423e18 s^6 + 4.105e19 s^5 + 4.97e20 s^4 + 2.978e21 s^3 + 1.593e22 s^2 + 3.978e22 s

Now the Simulink model of our closed loop plant, using a step-input compared with the open loop plant is given by:

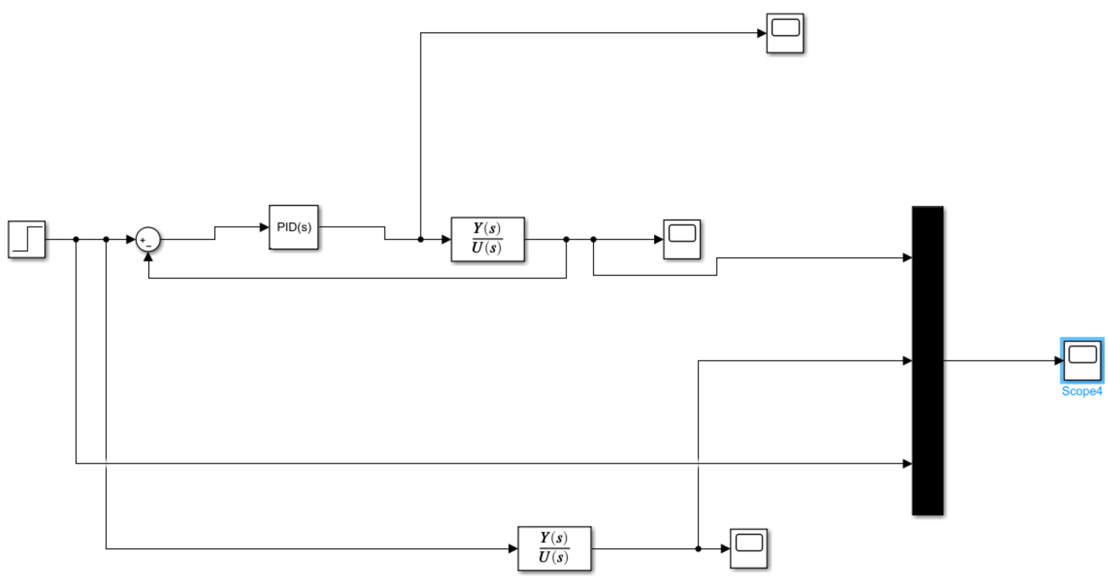


Figure 3-13: Simulink model of an active suspension system

NOTE: The PID block represent the controller where we put in the values of our gains

Root locus of the closed loop transfer function G

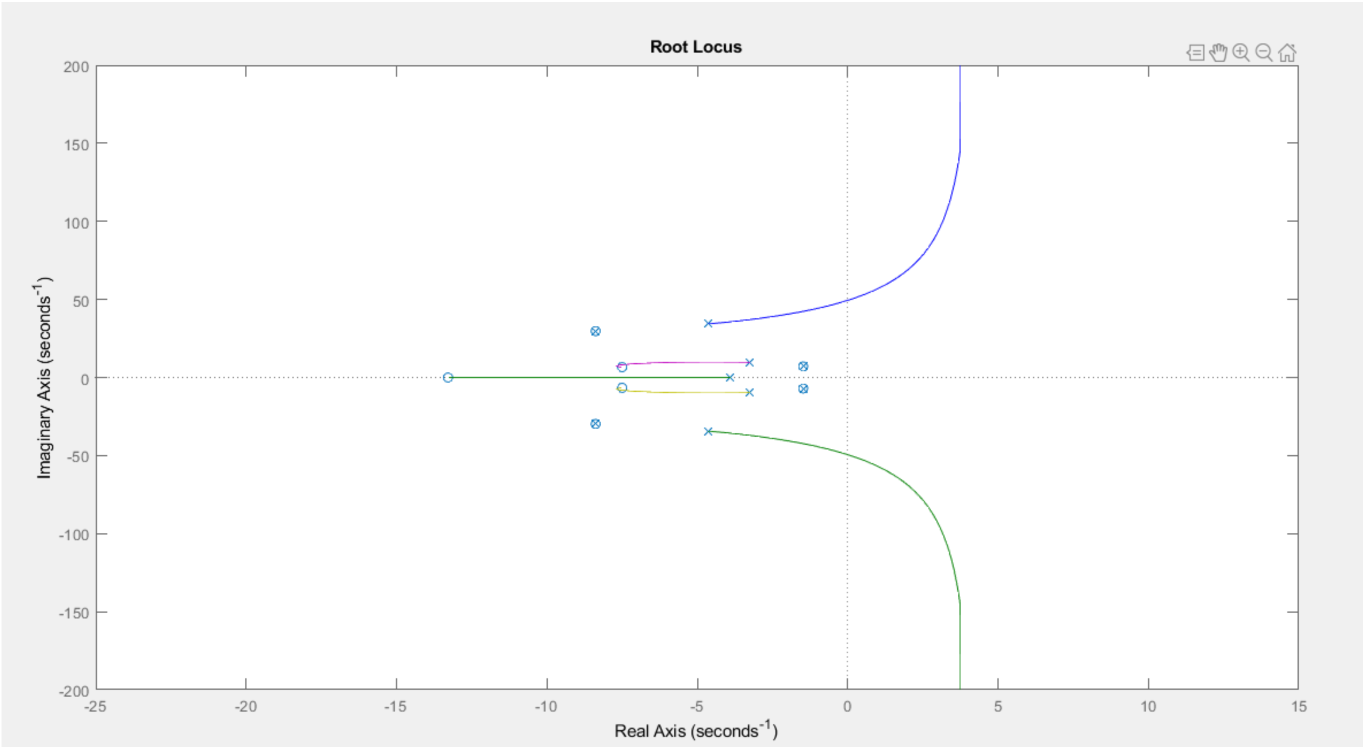


Figure 3-14: root locus of the active suspension system

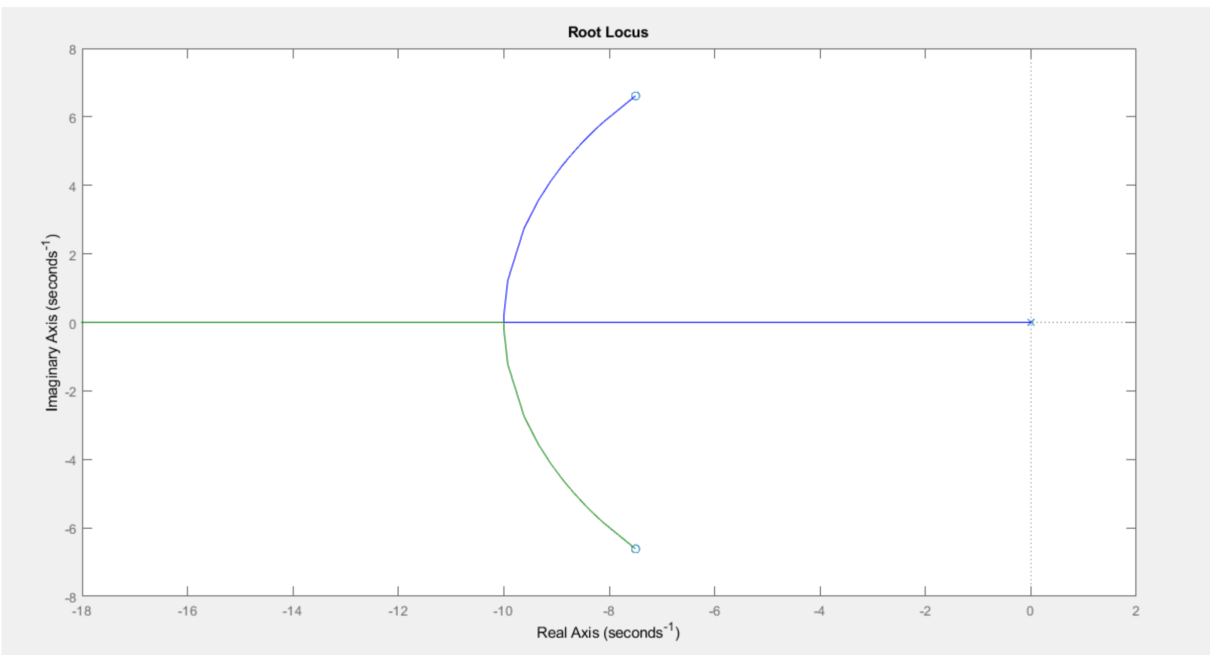


Figure 3-15: Root locus of the PID controller

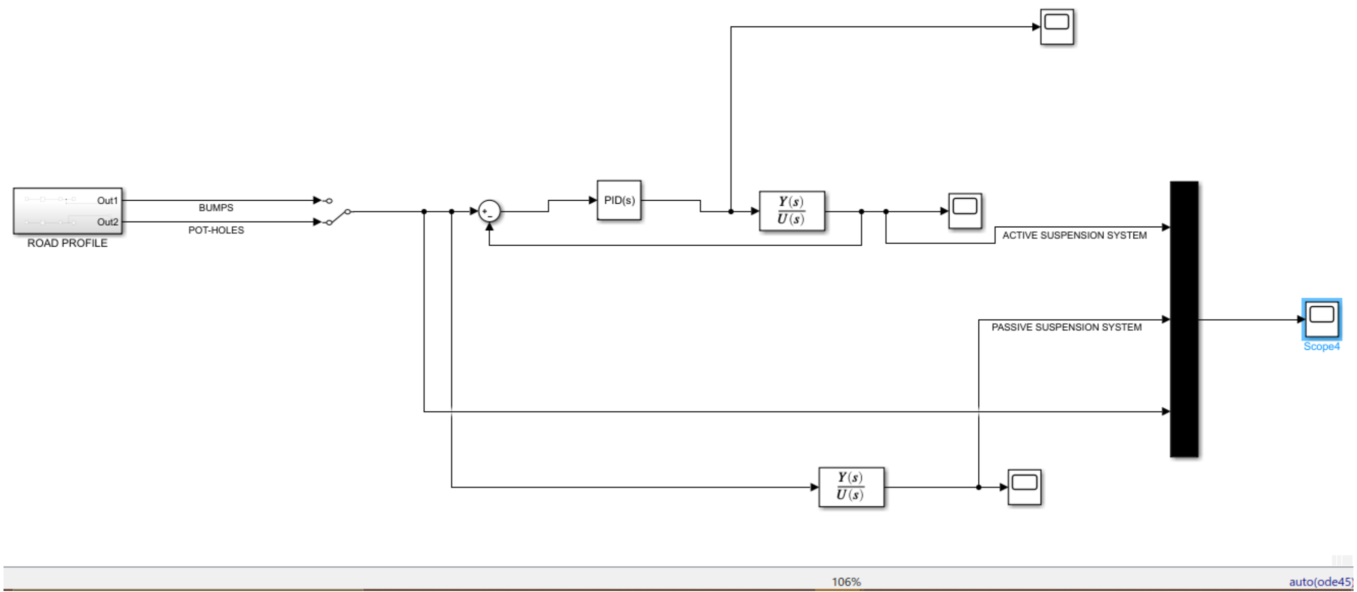


Figure 3-16: Simulink of the Active suspension compared to passive suspension compared to both road profile

Chapter 4

4 Result and Discussion

4.1 Simulated Results

The figures below represented all of the simulated results generated by MATLAB Simulink:

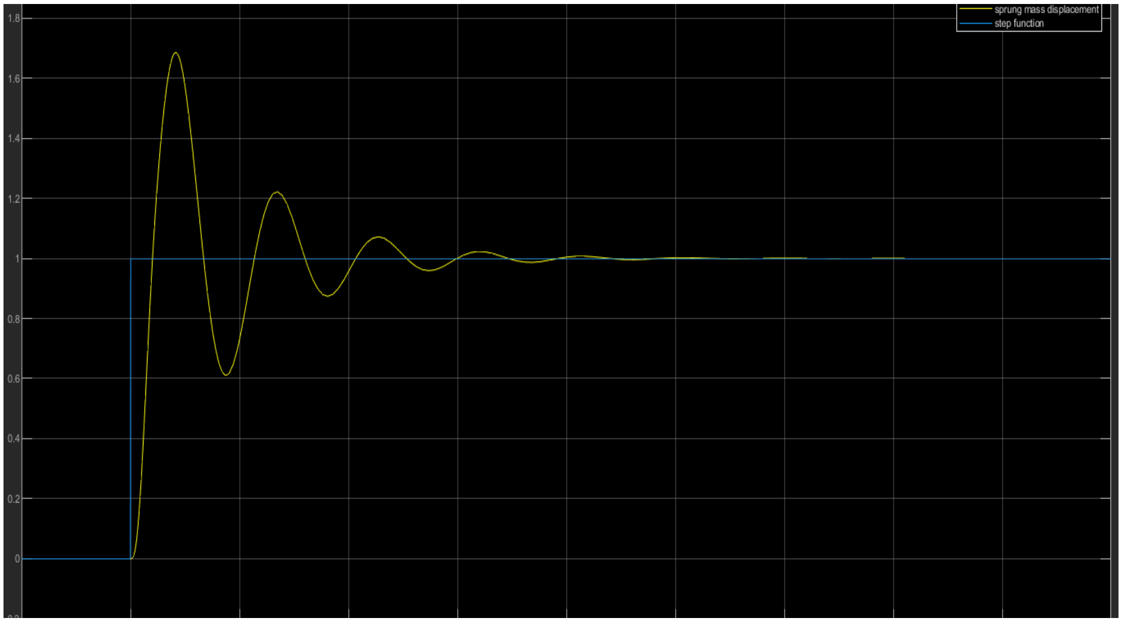


Figure 4-1: graphical representation of the step response of the passive suspension system

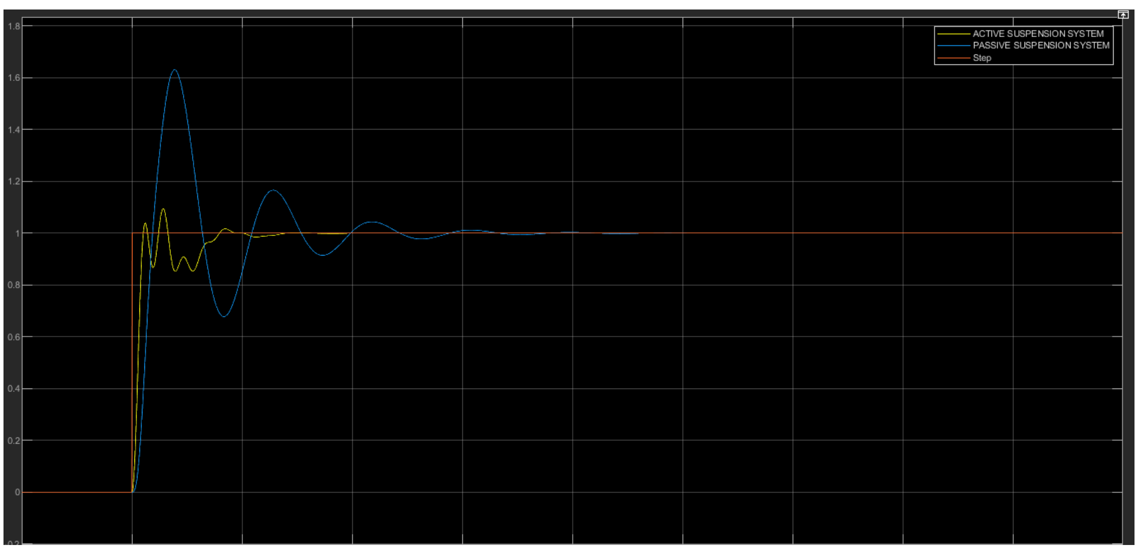


Figure 4-2: Step-response of the active suspension compared to the passive suspension

From the figure above we can observe that when both models were subjected to an excitation like a step response both models exhibit stability because they eventually damping out, however the active suspension model showed a more desired performance we look for in a vehicle because it damped out the vibrations faster and had a significantly lower amplitude than that of the passive suspension which means that a lot less vibration would be transferred to the passengers in the vehicle.

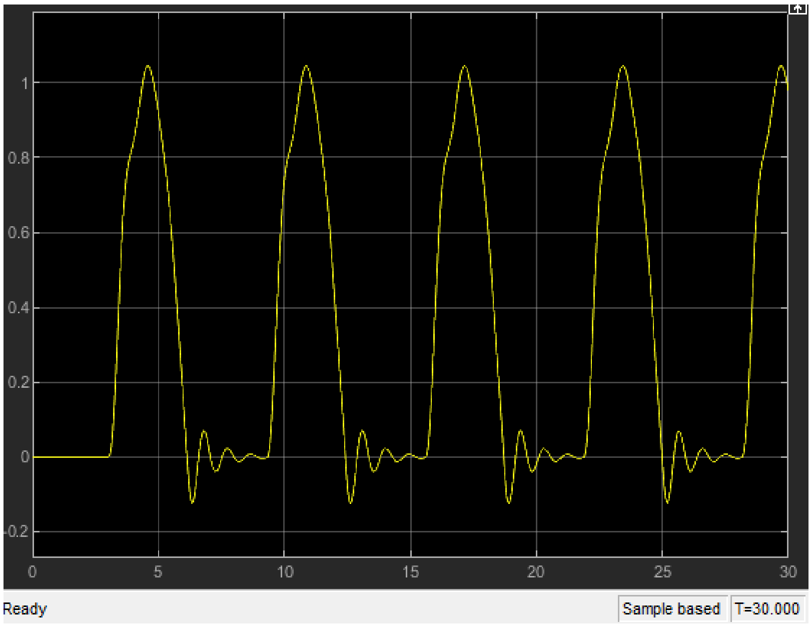


Figure 4-3: passive suspension sprung mass oscillation when vehicle is subjected to a speed bump

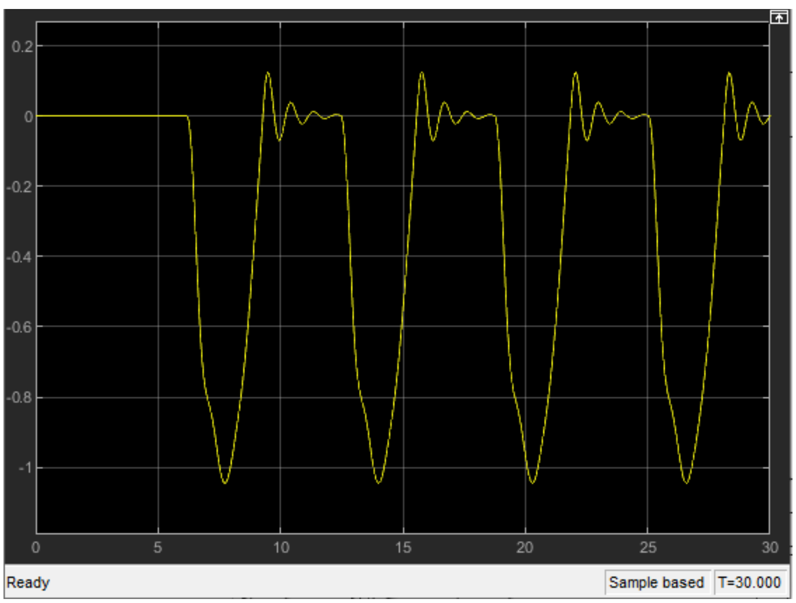


Figure 4-4: passive suspension sprung mass oscillation when subjected to a pot hole

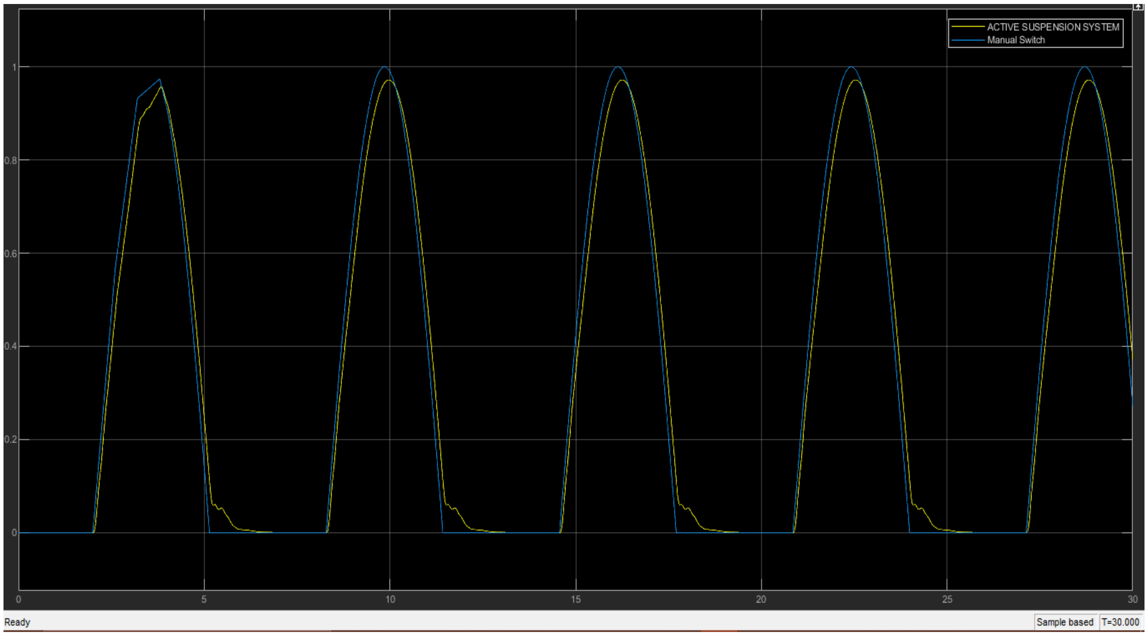


Figure 4-5: Active suspension sprung mass oscillation when vehicle is subjected to a speed bump

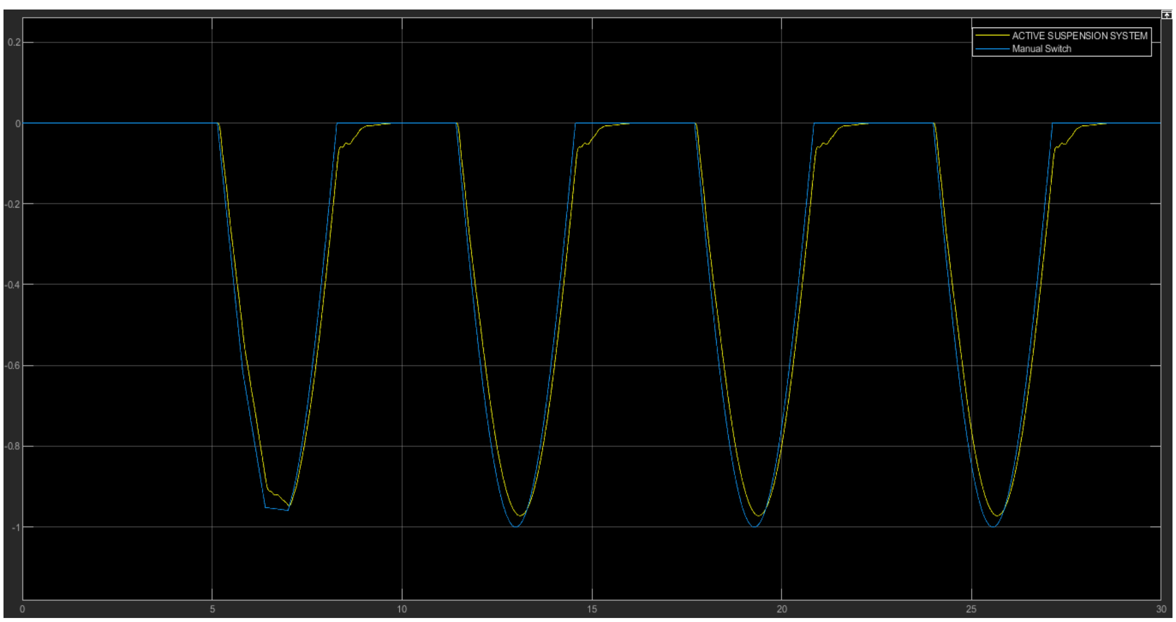


Figure 4-6: Active suspension sprung mass oscillation when subjected to a pot hole

When compared to the passive suspension system we can see that the active suspension had very little vibrations transferred to the passengers or the sprung mass.

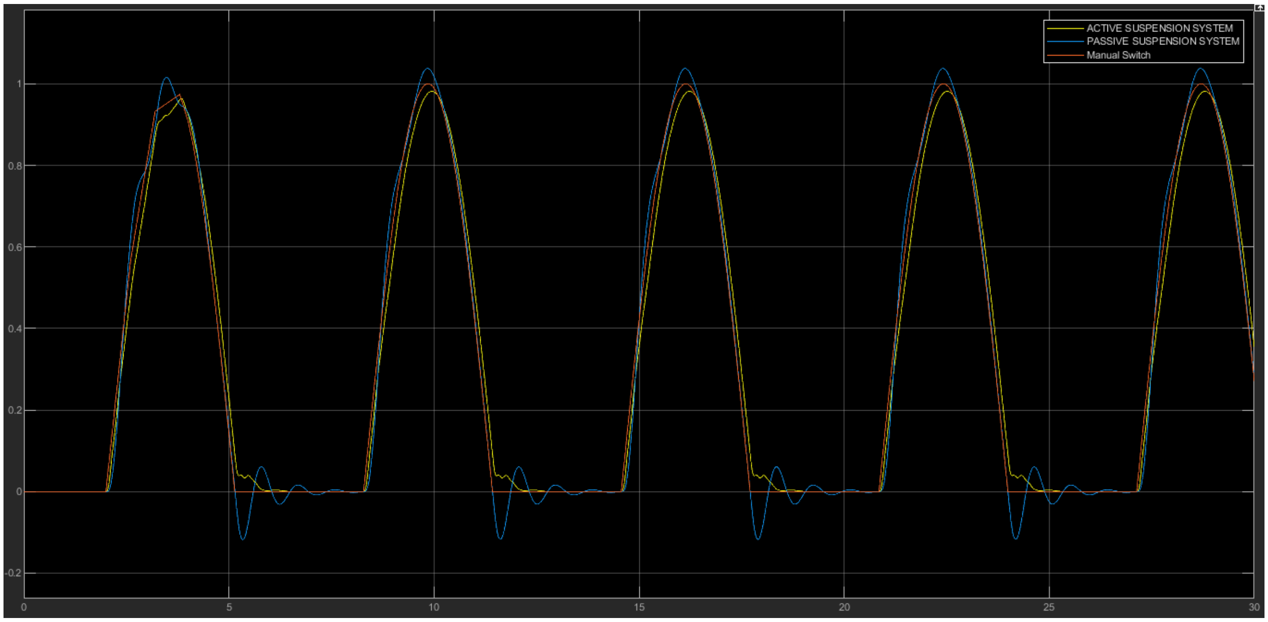


Figure 4-7: Simulated response of the active suspension compared to the passive suspension to a bumpy road profile

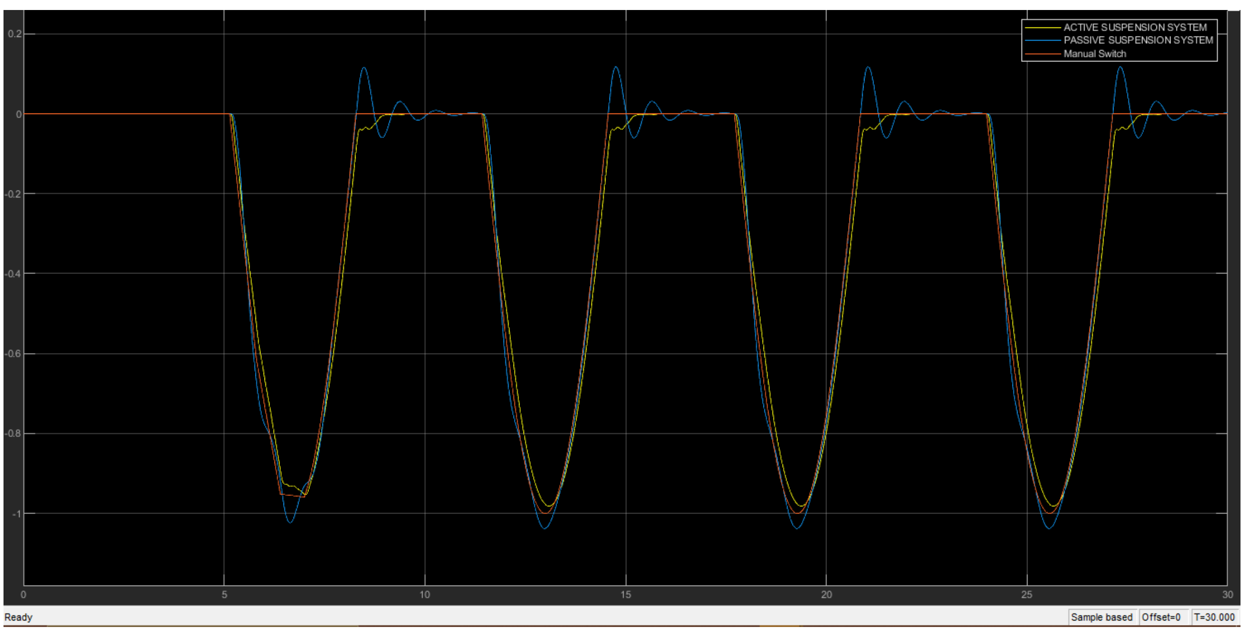
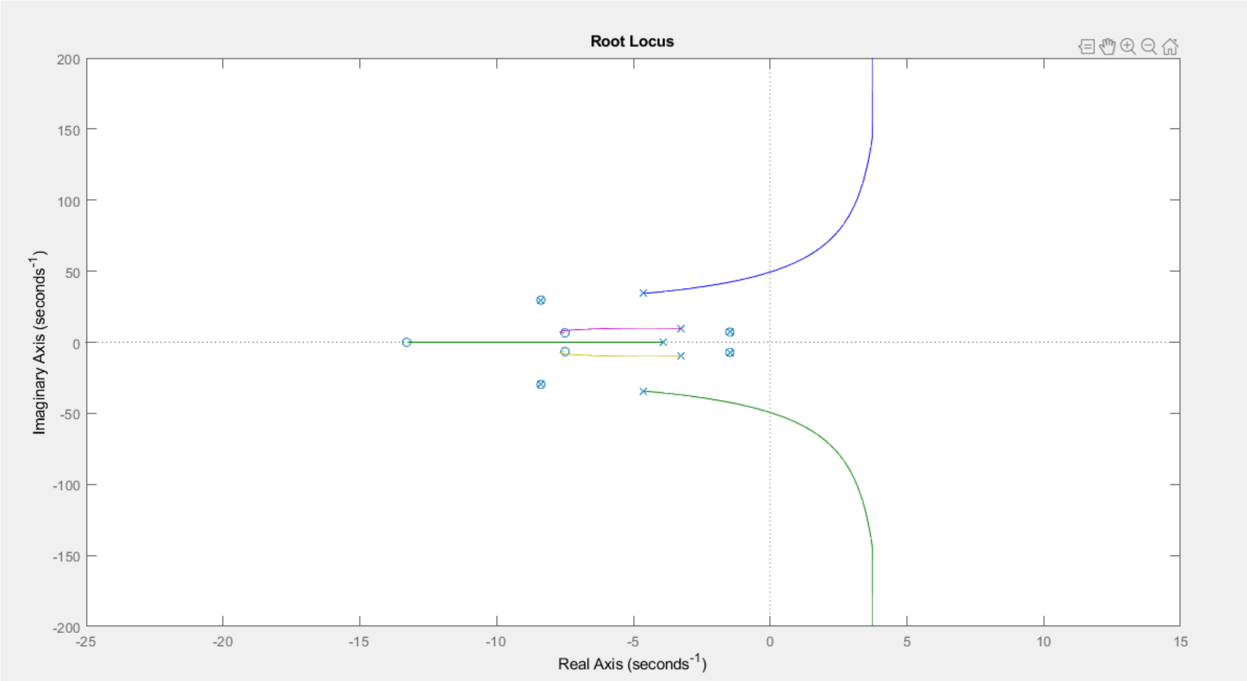


Figure 4-8: Simulated response of the active suspension compared to the passive suspension system to a pot hole- road profile

Root locus of the closed loop transfer function G



The figure above is the root locus of our active suspension system, from the figure we can see that at our current controller gains all the poles are on the left half plane which means our system is stable at this configuration

The transfer function for our active suspension system is given by:

$$G = \frac{numerator}{denominator}$$

Where;

numerator = 6.116e14 s^8 + 2.936e16 s^7 + 1.17e18 s^6 + 2.494e19 s^5 + 3.029e20 s^4 + 2.379e21 s^3 + 1.196e22 s^2 + 3.978e22 s

denominator= 1.66e12 s^10 + 6.545e13 s^9 + 4.761e15 s^8 + 1.107e17 s^7 + 3.423e18 s^6 + 4.105e19 s^5 + 4.97e20 s^4 + 2.978e21 s^3 + 1.593e22 s^2 + 3.978e22 s

From previous work performed by Khalil Ibrahim (2018) (Ibrahim, et al., 2018) and other similar literatures an acceptable system performance would be one whose system requirement are in this range:

Overshoot	35% or better
Settling time	2.3 seconds or better
Rise time	0.25 seconds or better

From our controlSystemDesigner app on matlab we were able to measure our system parameter to see if it meets the system requirement and our values were:

Table 4-1: active suspension system performance

Overshoot	32.8%
Settling time	0.8 seconds
Rise time	0.04 seconds

As compared to the results we obtained from our open loop response (passive suspension)

Table 4-2: passive suspension system performance

Overshoot	60%
Settling time	4.011seconds
Rise time	0.07

From our simulated results we can see that the best performance was seen with the speed bump and pot holes this was because unlike the step response they weren't an exponential change in position but a gradual one. The step response was used to measure a general response to a worst-case scenario.

From the figure of the root locus of the closed loop system we can see that we have all our poles at the left half plane satisfying condition for stability of the system.

5 Conclusion and Recommendation

From our analysis and results we can say that the active suspension showed better system performance than the passive suspension system. With our active suspension we had a step response settling time of 0.8seconds compared to 4 seconds in the passive suspension system.

In order to successfully simulate the active suspension system a critical factor was tuning the PID controller. Tuning a PID controller means setting the proportional, integral and derivative gains of the PID controller, this was achieved using the pole placement method done with the control system designer app in matlab and we arrived at a proportional gain of 1.5, an integral gain of 10 and a derivative gain of 0.1. With the controller designed and tuned the negative feedback loop was completed with our reference signal being the road profile.

One of the problems of closed loop control system is instability which arise as a result of an improperly tuned controller, to make sure our system was stable we plotted the root locus of our system and checked if it satisfied the condition for stability (all poles must be in the left hand plane) which our active suspension system did.

As a recommendation a more robust controller like the H-infinity kind could be used for the active suspension system to increase the performance of our system, it’s also important to monitor the output from the controller in our Simulink model to make sure the controller is not demanding an output force from the actuator that is beyond the maximum possible force it can generate, that is we want to make sure we don’t run into the problem of saturation. If this active suspension system is implemented in an actual vehicle, a feedforward control should also be implemented to control the amount of noise and disturbance entering the system.

References

Auto123, 2016. [Online]
Available at: https://www.auto123.com/en/new-cars/techical-specs/audi/a8/2015/base/30-tfsi/

Barak, p., 1991. *magic number in design of suspension for passenger cars.* s.l.:SAE international.

Bhise, A. R. et al., 2016. *comparision between active and passive suspension system using matlab and simulink.* s.l.:s.n.

Boileau, 1998. *Whole body vertial biodynamic response characteristics of the seated vehicle driver: measurement and model development.* s.l.:s.n.

Distefano, J., Stubberud, A. & Williams, I., 1996. *Schaum's outline of theory and problems of feedback control system.* s.l.:McGraw-Hill.

Felipe Mrad, d. m. g. h. a. a. u. s., 2018. *optimization of the vibrational comfort of passenger vehicles through improvement of suspension and engine rubber mounting setup, shock and vibration.* s.l.:s.n.

Ibrahim, K., abdeen, a. & nasim, a.-b., 2018. *active suspension system design using fuzzy logic and linear quadratic equation.* s.l.:researchgate publication.

ISO, 1997. *mechanical vibration and shock evaluation of human exposure to whole body vibration: part 1: General requirements.* s.l.:international standards organization .

Karris, s., 2006. *introduction to simulink with engineering application.* s.l.:orchard publication.

MATHWork, n.d. *Mathwork.com.* [Online]
Available at: http://www.mathworks.com/help/index.html

Muksiar, 1976. *frequency dependent damping coefficient in lumped paramatric model of human being.* s.l.:s.n.

Nise, N., 2015. *control system engineering.* s.l.:WILEY.

Padden, 2002. *Evaluation of whole-body vibration in vehicles 2002.* s.l.:s.n.

Riduan, A. F. M., Tamaldim, N., Sudrajat, A. & Ahmad, F., 2018. *Review on active suspension system.* s.l.:s.n.

Tata, R. P., 2012. *Automotive Suspension System.* s.l.:s.n.

Yang, L., Bailo, F., Previati, G. & Gobbi, M., 2019. *robust optimization of road vehicles suspension considering the variation of the vertical stiffness.* italy: research gate publiccation.

Yatak, M. O. & Sahin, F., 2020. *Ride comfort-Road handling trade off improvement of full vehicle active suspension system by an interval type-2 fuzzy control.* s.l.:ResearchGate publication.

.