|  |  |
| --- | --- |
|  |  |
|  |  |

Anti-Lock Braking System

## Name and Unique ID:

## Abhinav Mondal, 2005576



## Name and Unique ID:

## Abhinav Mondal, 2005576



## Name and Unique ID:

## Abhinav Mondal, 2005576



## Name and Unique ID:

## Abhinav Mondal, 2005576



Contents

[Name and Unique ID: 1](file:///C:\Users\abhin\OneDrive\Desktop\ABS_Report.docx#_Toc59093493)

[Introduction 3](#_Toc59093501)

[Working 3](#_Toc59093502)

[Modelling of Anti-Lock Braking System 4](#_Toc59093503)

[Simulink Model 5](#_Toc59093504)

[Skills Demonstrated 6](#_Toc59093505)

[References 11](#_Toc59093506)

List of Figures

Fig 1: ABS Model.…………………………………………….………………………………………………….5

Fig 2: WheelSpeedCal (Subsystem Model) ……………………………………………………………….5

Fig 3: MATLAB function ………………………………….………………………………………………....8

Fig 4: Signal1 …………………………………………………………………………………………………….9

Fig 5: Signal2 …………………………………………………………………………………………………....9

Fig6: Output visualization in Data Inspector………………………………………………………..10

## Introduction

An **anti-lock braking system** (**ABS**) is a safety anti-skid braking system used on aircraft and on land vehicles, such as cars, motorcycles, trucks, and buses.

ABS operate by preventing the wheels from locking up during braking, thereby maintaining tractive contact with the road surface and allowing the driver to maintain more control over the vehicle.

ABS is an automated system that uses the principles of threshold braking and cadence braking, techniques which were once practiced by skilful drivers before ABS was widespread. ABS operate at a much faster rate and more effectively than most drivers could manage.

Although ABS generally offer improved vehicle control and decreases stopping distances on dry and some slippery surfaces, on loose gravel or snow-covered surfaces ABS may significantly increase braking distance, while still improving steering control. Since ABS was introduced in production vehicles, such systems have become increasingly sophisticated and effective.

## Working

ABS is part of an overall stability system, commonly known as electronic stability control, which monitors wheels under heavy braking. Each wheel has a sensor attached to it. If the intelligent sensors detect that a wheel is about to lock up and stop moving, the system will release the brake. The release is only for a moment.

ABS then continuously and repeatedly apply optimum braking pressure to each wheel, meaning the system will brake just enough to not lock the wheels. When ABS is active you may feel pulsation through the brake pedal as you’re pressing it. The anti-lock system helps the driver remain in control of the vehicle rather than bringing the car to a stop. It reduces the risk of skidding even when undertaking excessive evasive manoeuvres. Therefore, it’s important to remember that the car’s braking distance may increase.

So, if you keep driving ahead straight into an obstacle, the car may not stop in time even if your instincts dictate otherwise. It’s a common misconception that ABS help reducing stopping distance.

## Modelling of Anti-Lock Braking System

The model simulates the dynamic behavior of a vehicle under hard braking conditions.

In this model the wheel speed is calculated in a subsystem named WheelSpeedCal.

The wheel rotates with an initial angular speed that corresponds to the vehicle speed before the brakes are applied. Separate integrators are used to compute wheel angular speed and vehicle speed. Two speeds are used to calculate slip, which is determined by Equation 1. Vehicle speed is introduced and expressed as angular velocity.

$$\omega_v = \frac{V}{R} \mbox{ (equals the wheel angular speed if there is no slip)}$$

**Equation 1**

$$ \omega_v = \frac{V_v}{R_r}$$

$$slip=1-\frac{\omega_w}{\omega_v}$$

$$\omega_v = \mbox{ vehicle speed divided by wheel radius}$$

$$ V_v = \mbox{ vehicle linear velocity}$$

$$ R_r = \mbox{ wheel radius}$$

$$ \omega_w = \mbox{ wheel angular velocity}$$

From these expressions, it can be observed that slip is zero when wheel speed and vehicle speed are equal, and slip equals one when the wheel is locked. A desirable slip value is 0.2, which means that the number of wheel revolutions equals 0.8 times the number of revolutions under non-braking conditions with the same vehicle velocity. This maximizes the adhesion between the tire and road and minimizes the stopping distance with the available friction.

## Simulink Model

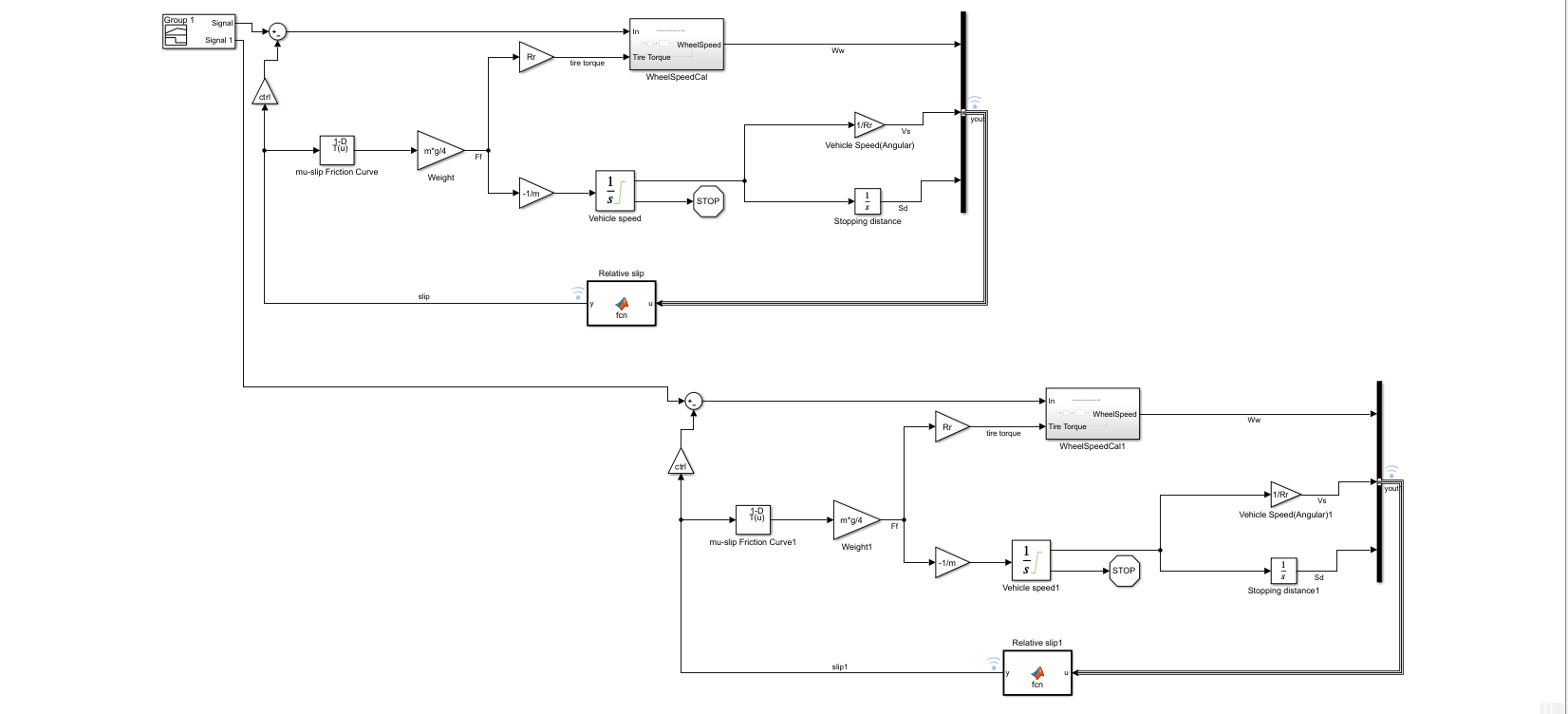


Fig1: ABS Model

The model represents a single wheel, which is replicated once to create a model for a multi-wheel vehicle.

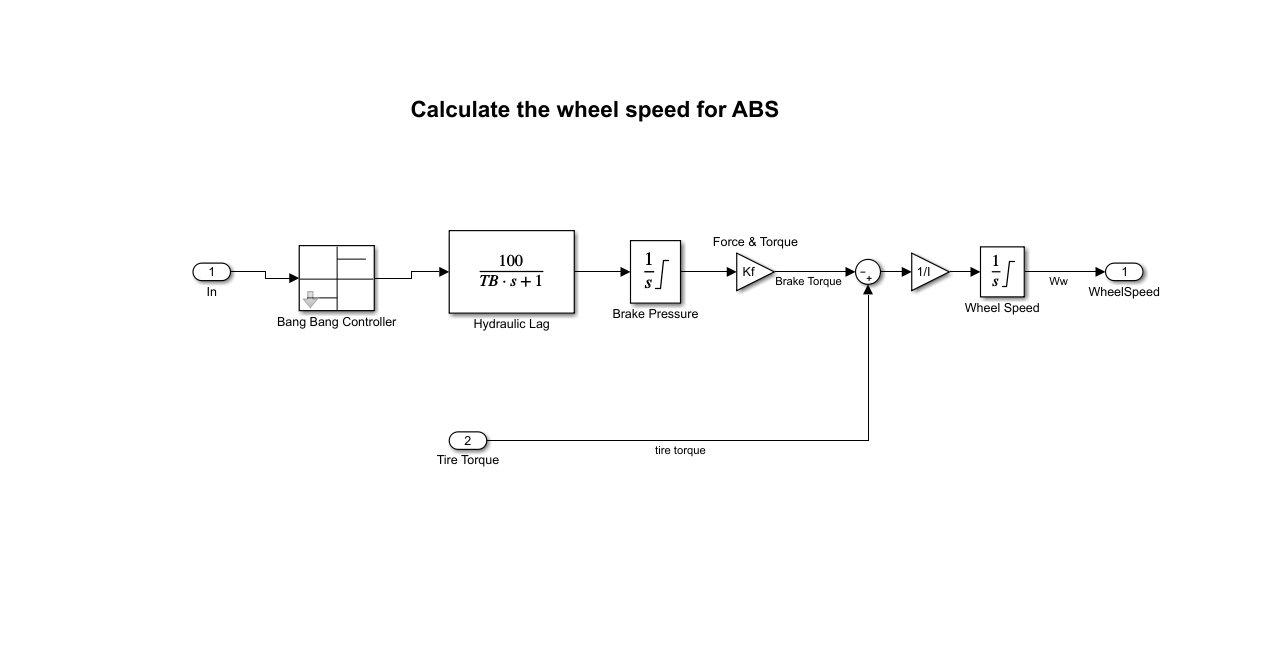


Fig2: WheelSpeedCal (Subsystem Model)

## Skills Demonstrated

* **Callbacks**

Callbacks are commands you can define that execute in response to a specific modelling action, such as opening a model or stopping a simulation. Callbacks define MATLAB expressions that execute when the block diagram or a block is acted upon in a way.

Simulinkprovides model, block, and port callback parameters that identify specific kinds of model actions. You provide the code for a callback parameter. Simulink executes the callback code when the associated modelling action occurs.

The following callbacks are demonstrated for the ABS model:

1. PreLoadFcn (Pre-load Function): This callback is executed before the model is loaded. ABS model has a pre-load function callback i.e. **initdata**

Which initializes all the parameters in the base workspace.

1. StopFcn (Stop Function): This callback is executed after the simulation stops. Output is written to workspace variables and files before the StopFcn is executed. ABS model has a stop function callback i.e.

**if exist ('ABS\_output')**

**plots;**

**end**

which means after the simulation stops and if ABS\_output exists in the

workspace then plot the graphs.

1. CloseFcn (Close Function): This callback is executed before the block diagram is closed. ABS model uses this callback to close all the plots if open and clear all the parameters from the base workspace.

**% Close the plots that are still open**

**h = findobj (0, 'Name', 'ABS Speeds');**

**if ~isempty(h),**

**close(h);**

**end;**

**h = findobj (0, 'Name', 'ABS Slip');**

**if ~isempty(h),**

**close(h);**

**end;**

**% Clear initial conditions data to avoid cluttering the workspace**

**clear g v0 Rr Kf m PBmax TB I slip mu ctrl h;**

* **Look-up Table**

The friction coefficient between the tire and the road surface, **mu, is an empirical function of slip, known as the mu-slip curve**. We created mu-slip curves by passing MATLAB variables into the block diagram using a Simulink lookup table(1D).

The Look-up Table has slip as the input and mu as the output which can be seen in Fig1.

The model multiplies the friction coefficient, mu, by the weight on the wheel, W, to yield the frictional force, Ff, acting on the circumference of the tire. Ff is divided by the vehicle mass to produce the vehicle deceleration, which the model integrates to obtain vehicle velocity.

In this model, we used an ideal anti-lock braking controller, that uses 'bang-bang' control based upon the error between actual slip and desired slip. We set the desired slip to the value of slip at which the **mu-slip curve reaches a peak value, this being the optimum value for minimum braking distance.**

* **Solver selection strategy**

The solver used for this model is ode45 and the type is variable step.

The main reason of using variable step that it varies the step size during the simulation. It reduces the step size to increase accuracy when a model's states are changing rapidly and increase the step size to avoid taking unnecessary steps when the model's states are changing slowly. Computing the step size adds to the computational overhead at each step but can reduce the total number of steps, and hence the simulation time required to maintain a specified level of accuracy for models with rapidly changing or piecewise continuous states.

The reason of choosing the ode45 solver is that it has more accuracy than ode23 and ode113 when it comes to a nonstiff problem type. The simulation time taken is also quite fast. ode45 does more work per step than ode23 and ode113 but can take much larger steps.

* **MATLAB Function Block**

This block is used to create a custom functionality that can be used in Simulink models.

In ABS model this block is used to create a function that calculates the relative slip of the ABS.

The function here takes the input as yout i.e. the output of bus creator and the output is the slip.

The function written is given below:

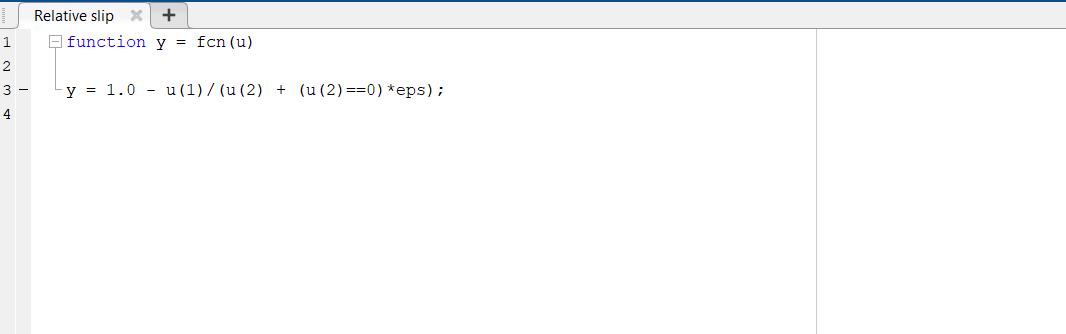


Fig3: MATLAB function

It shows up on the code editor. The variable u is the input to the function i.e. yout and y is the output i.e. the relative slip.

* **Signal Builder**

The Signal Builder block allows you to create interchangeable groups of piecewise linear signal sources and use them in a model. You can quickly switch the signal groups into and out of a model to facilitate testing.

We can create as many signals as we want using the signal builder.

For our ABS model, Signal builder is used to create test signals in order to showcase how the system is performing under various test conditions.

In the ABS model we have demonstrate a multi wheel vehicle (2 wheels)

hence two test signals are generated using the signal builder.

1. A Desired relative slip of 0.2

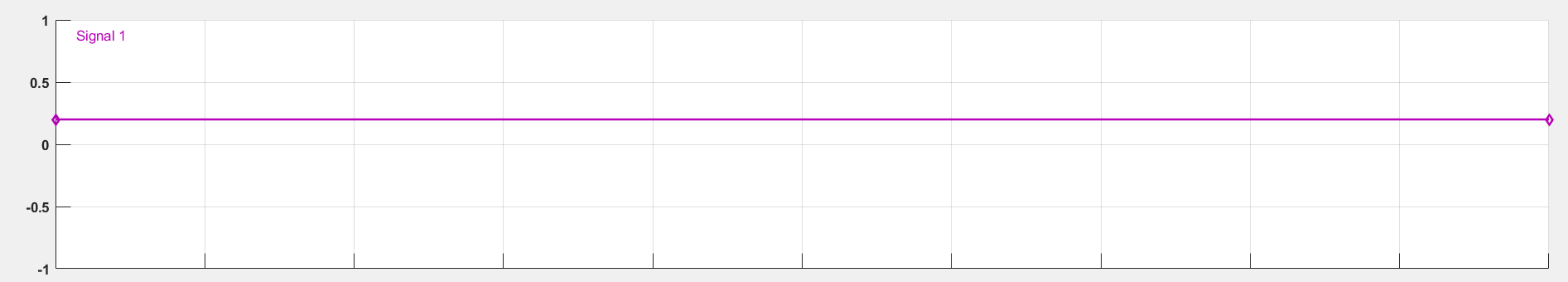


Fig4: Signal 1

This signal with a constant value of 0.2 gives the perfect output showcasing that wheel does not lock up when brake is applied.

The output plot (present in the next section- Fig6) shows that the wheel speed stays below vehicle speed without locking up, with vehicle speed going to zero in less than 15 seconds.

1. A Desired relative slip of 1

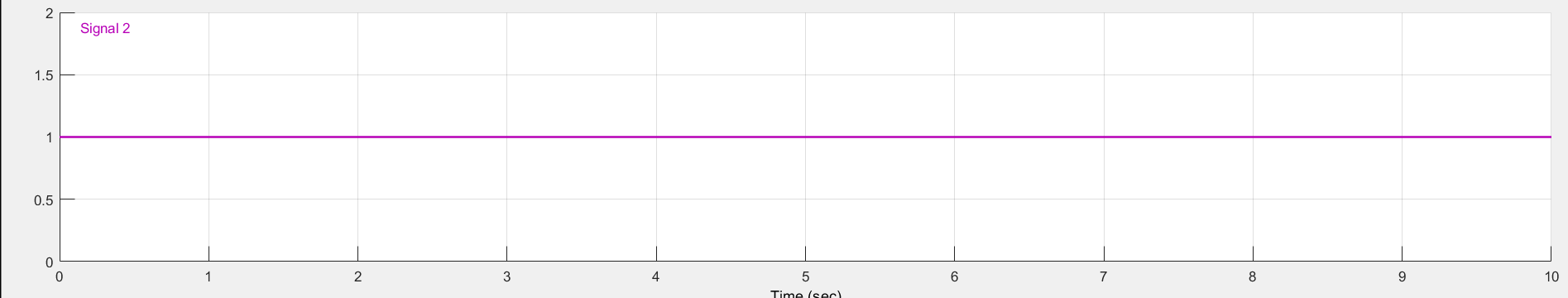


Fig5: Signal 2

This signal with a constant value of 1 is basically a test signal used to demonstrate the condition of without ABS. This disconnects the slip feedback from the controller, resulting in maximum braking.

* **Data Inspector**

The Data Inspector visualizes and compares multiple kinds of data.

Using the Data Inspector, we can inspect and compare time series data at multiple stages of your workflow.

It saves the data of the logged signals for each run. We can use this saved data and display the graphs for the output parameters.

In our ABS model we have logged two signals, first being yout and the second is slip.

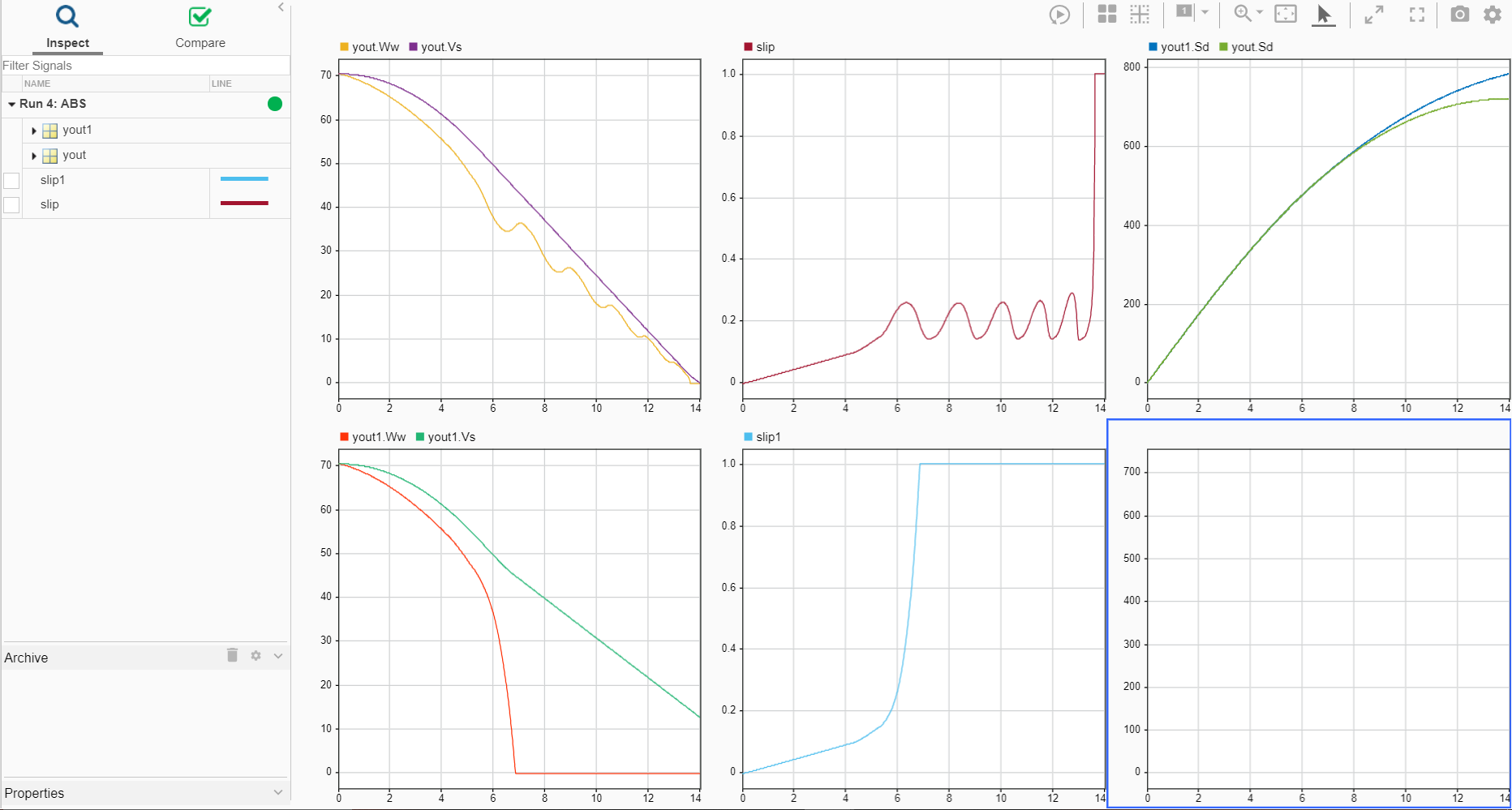


Fig6: Output visualization in Data Inspector

We can also the change the layout of the plots for proper observation and realization of the output.

There are following plots present in Fig6:

1. yout for signal1 that contains the wheel speed (yout.Ww) and vehicle speed (yout.Vs).[For desired relative slip of 0.2]
2. yout for signal2 that contains the wheel speed (yout1.Ww) and vehicle speed (yout1.Vs). [ For desired relative of 1]
3. slip for signal1.[ For desired relative slip of 0.2]
4. slip1 for signal2.[ For desired relative of 1]
5. Sd i.e. Stopping distance for both signal1 and signal2.

The distance travelled by the vehicle is plotted for the two test cases. Without ABS, the vehicle skids about an extra 100 feet, taking about three seconds longer to come to a stop.

## References

[1] <https://en.wikipedia.org/wiki/Anti-lock_braking_system>

[2] <https://www.confused.com/on-the-road/gadgets-tech/what-is-abs>

[3] <https://www.mathworks.com/help/>

[4] I. Khan, I. Hussain, M. Z. A. Shah, K. Kazi and A. A. Patoli, "Design and simulation of anti-lock braking system based on electromagnetic damping phenomena," *2017 First International Conference on Latest trends in Electrical Engineering and Computing Technologies (INTELLECT)*, Karachi, 2017, pp. 1-8, doi: 10.1109/INTELLECT.2017.8277615.