Model based design of
Anti-Lock Braking System (ABS)

Submitted by

Shashank Mangloor

Unique Id-2005508

Date-18th December 2020

PREFACE

Figure Table

1. Anti-Lock Braking system (ABS)

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 operates 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, technique, which were once practiced by skilful drivers before ABS, was widespread. ABS operates at a much faster rate and more effectively than most drivers could manage. Although ABS generally offers 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. Modern versions may not only prevent wheel lock under braking, but may also alter the front-to-rear brake bias.

2. Brake types

Anti-lock braking systems use different schemes depending on the type of brakes in use. They can be differentiated by the number of channels that is, how many valves that are individually controlled—and the number of speed sensors.

- Four-channel, four-sensor ABS
  There is a speed sensor on all four wheels and a separate valve for all four wheels. With this setup, the controller monitors each wheel individually to make sure it is achieving maximum braking force.
- Three-channel, four-sensor ABS
  There is a speed sensor on all four wheels and a separate valve for each of the front wheels, but only one valve for both of the rear wheels. Older vehicles with four-wheel ABS usually use this type.
- Three-channel, three-sensor ABS
  This scheme, commonly found on pickup trucks with four-wheel ABS, has a speed sensor and a valve for each of the front wheels, with one valve and one sensor for both rear wheels. The speed sensor for the rear wheels is located in the rear axle. This system provides individual control of the front wheels, so they can both achieve maximum braking force. The rear wheels, however, are monitored together; they both have to start to lock up before the ABS will activate on the rear. With this system, it is possible that one of the rear wheels will lock during a stop, reducing brake effectiveness. This system is easy to identify, as there are no individual speed sensors for the rear wheels.
- Two-channel, four-sensor ABS
  This system, commonly found on passenger cars from the late '80s through the mid-1990s, uses a speed sensor at each wheel, with one control valve each for the front and rear wheels as a pair. If the speed sensor detects lock up at any individual wheel, the control module pulses the valve for both wheels on that end of the car.
- One-channel, one-sensor ABS
  This system is commonly found on pickup trucks, SUVs, and vans with rear-wheel ABS. It has one valve, which controls both rear wheels, and a one-speed sensor, located in the rear axle. This system operates the same as the rear end of a three-channel system. The rear wheels are monitored together and they both have to start to lock up before the ABS kicks in. In this system it is also possible that one of the rear wheels will lock, reducing brake effectiveness. This system is also easy to identify, as there are no individual speed sensors for any of the wheels.

3. Basic principle

Wheel speed sensors mounted on the front and rear wheel constantly measures the rotational speed of each wheel and delivers this information to an Electronic Control Unit (ECU). The ECU detects on the one hand if the deceleration of one wheel exceeds a fixed threshold and on the other hand whether the brake slip, calculated based on information of both wheels, rises above a certain percentage and enters an unstable zone. These are indicators for a high possibility of a locking wheel. To countermeasure these irregularities the ECU signals the hydraulic unit to hold or to release pressure. After signals show the return to the stable zone, the pressure is increased again. Past models used a piston for the control of the fluid pressure. Most recent models regulate the pressure by rapidly opening and closing solenoid valves. While the basic principle and architecture has been carried over from passenger car ABS, typical motorcycle characteristics have to be considered during the development and application processes. One characteristic is the change of the dynamic wheel load during braking. Compared to cars, the wheel load changes are more drastic, which can lead to a wheel lift up and a fall over. This can be intensified by a soft suspension. Some systems are equipped with a rear-wheel lift-off mitigation functionality. When the indicators of a possible rear lift-off are detected, the system releases brake pressure on the front wheel to counter this behaviour. Another difference is that in the case of the motorcycle the front wheel is much more important for stability than the rear wheel. If the front wheel locks up between 0.2-0.7s, it loses gyrostatic forces and the motorcycle starts to oscillate because of the increased influence of side forces operating on the wheel contact line. The motorcycle becomes unstable and falls.

4. Benefits of Motorcycle ABS

A report into the benefits of Anti-Lock Braking Systems (ABS) on on-road motorcycles was commissioned by, The Commonwealth and Victorian Governments and released in 2015. The report found that:

- ABS could reduce the rate of death and severe injury from motorcycle crashes by 31%
- only around 20% of new motorcycles come with ABS as standard
- at the current rate of ABS motorcycle sales, this technology has the potential to save 22 lives between now and 2025
- This figure could rise to 35 saved lives if ABS is made standard on all new motorcycles from 2018.

5. Analysis and Physics

The wheel rotates with an initial angular speed that corresponds to the vehicle speed before the brakes are applied. We used separate integrators to compute wheel angular speed and vehicle speed. We use two speeds to calculate slip, which is determined by Equation 1. Note that we introduce vehicle speed expressed as an angular velocity (see below).

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

Equation 1

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

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

$\omega_v =$ vehicle speed divided by wheel radius

$V_v =$ vehicle linear velocity

$R_r =$ wheel radius

$\omega_w =$ wheel angular velocity

From these expressions, we see 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.

6. Modelling

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. 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.

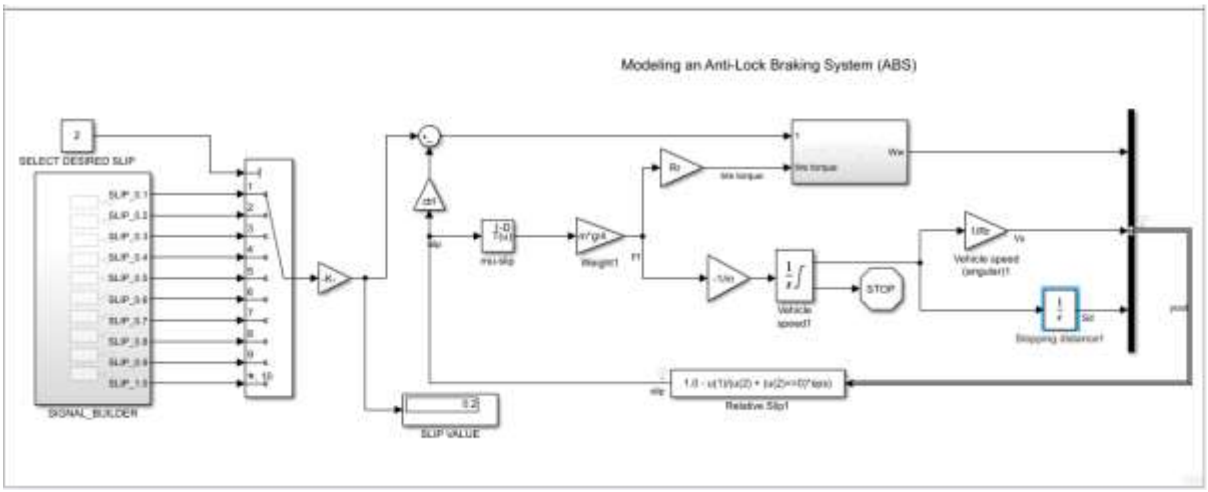

Fig 6.1 Model block diagram of ABS

Simulink is a Matlab-based graphical programming environment for modelling, simulating and analysing multi domain dynamic systems its primary interface is a graphical block diagramming tool and a customizable set of block libraries. It offers tight integration with the rest of the MATLAB environment and can either drive MATLAB or be scripted from it. Simulink is widely used in automatic control and digital signal processing for multi domain simulation and model based design.

Stateflow is a control logic tool used to model reactive systems via state machines and flow charts within a Simulink model. Stateflow uses a variant of the finite state machines notation, enabling the representation of hierarchy, parallelism and history within a state chart. Stateflow also provides state transition tables and truth tables.

Here we have modelled the Automatic Transmission Controller using the basic Simulink and State Flow diagram and analyse the model under various drive conditions.

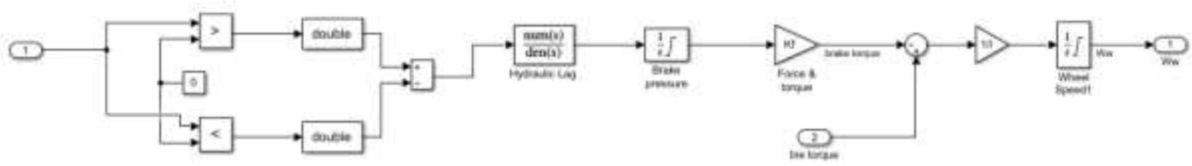

Fig 6.2 Model block diagram of ABS Calculating wheel speed for the abs simulation

Vehicle and Wheel Speed Running with ABS

Ctrl=1

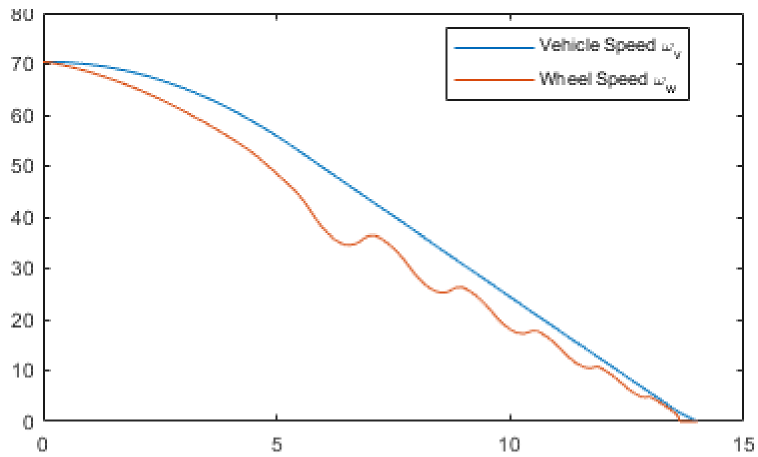

Fig 6.3 vehicle and wheel speed running with ABS

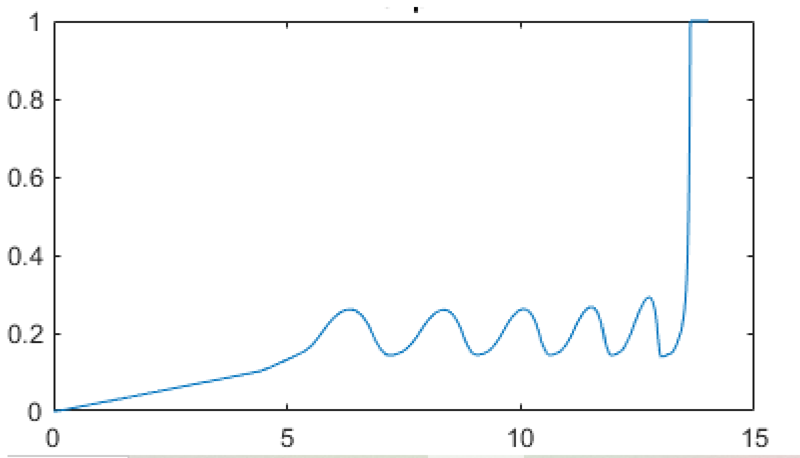

Fig 6.4 Slip of wheel running with ABS

The plots above show the ABS simulation results (for default parameters). The first plot shows the wheel angular velocity and corresponding vehicle angular velocity. This plot shows that the wheel speed stays below vehicle speed without locking up, with vehicle speed going to zero in less than 15 seconds.

Vehicle and Wheel Speed Running without ABS

Ctrl=0;

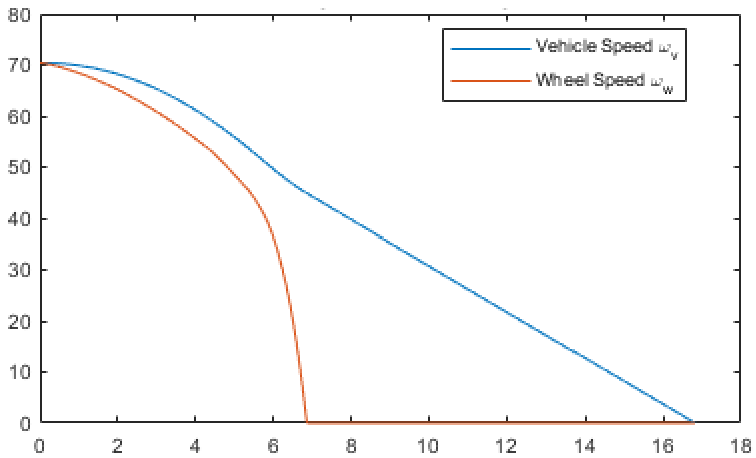

Fig 6.5 vehicle and wheel speed running without ABS

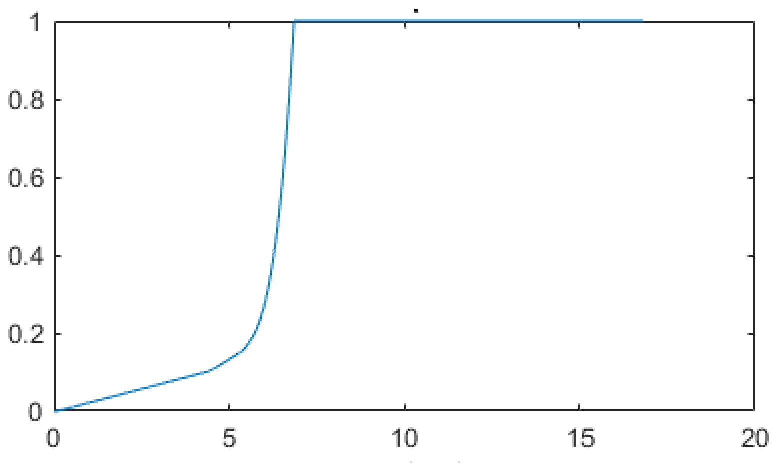

Fig 6.5 Slip of wheel running without ABS

7. Braking With ABS versus Braking Without ABS

In the plot showing vehicle speed and wheel speed, observe that the wheel locks up in about seven seconds. The braking, from that point on, is applied in a less-than-optimal part of the slip curve. That is, when slip = 1, as the slip plot shows, the tire is skidding so much on the pavement that the friction force has dropped off.

This is, perhaps, more meaningful in terms of the comparison shown below. The distance travelled by the vehicle is plotted for the two cases. Without ABS, the vehicle skids about an extra 100 feet, taking about three seconds longer to come to a stop.

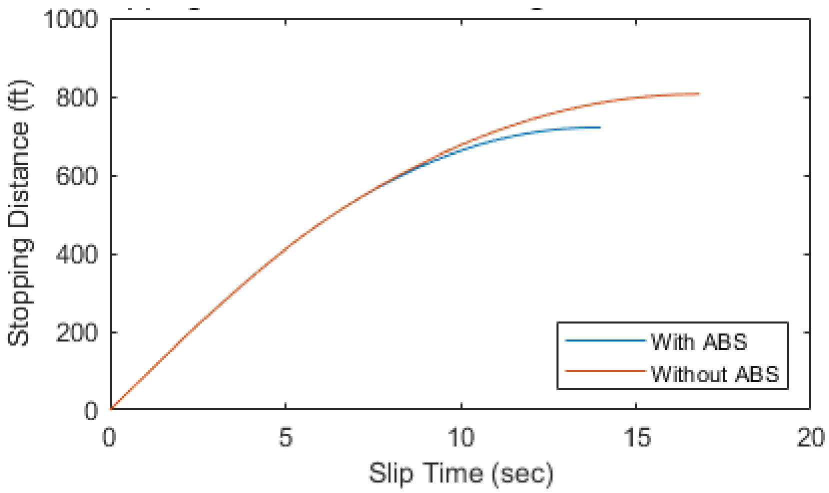

Fig 7.1 Stopping distance with and without ABS

8. Data inspector

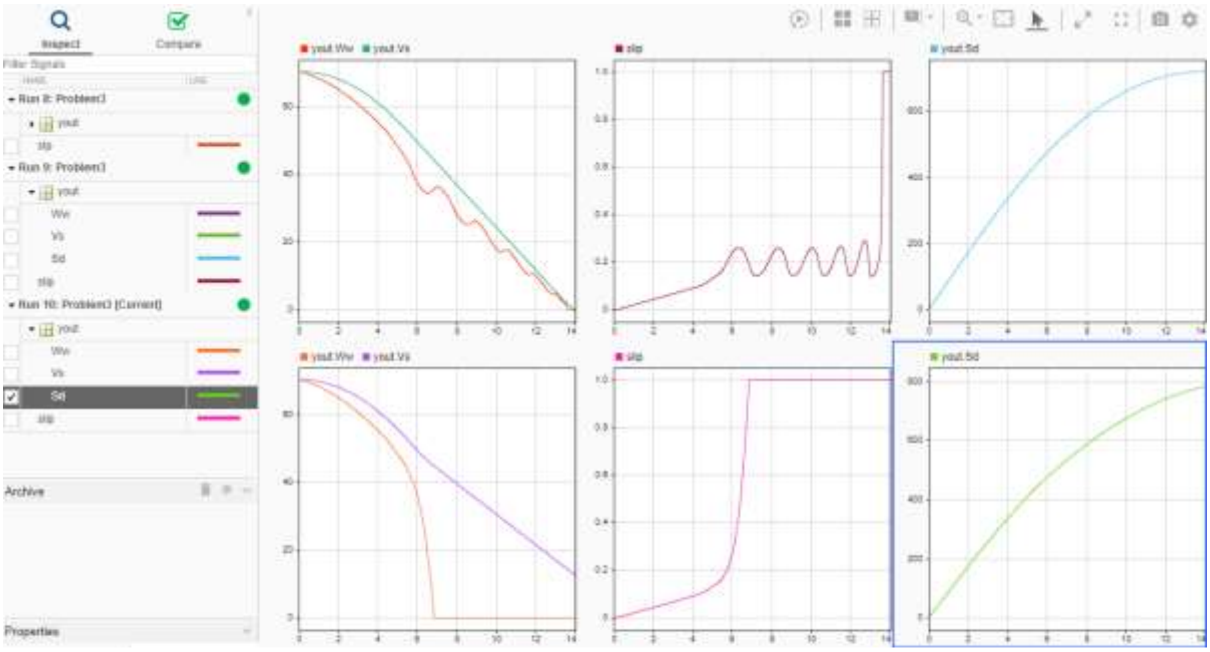

Fig 8.1 Data inspector window of ABS model

The Simulation Data Inspector visualizes and compares multiple kinds of data. Using the Simulation Data Inspector, you can inspect and compare time series data at multiple stages of your workflow. This example workflow shows how the Simulation Data Inspector supports all stages of the design cycle:

- View Data in the Simulation Data Inspector

Run a simulation in a model configured to log data to the Simulation Data Inspector, or import data from the workspace or a MAT-file. You can view and verify model input data or inspect logged simulation data while iteratively modifying your model diagram, parameter values, or model configuration.

- Inspect Simulation Data

Plot signals on multiple subplots, zoom in and out on specified plot axes, and use data cursors to understand and evaluate the data. Create Plots Using the Simulation Data Inspector to tell your story.

- Compare Simulation Data

Compare individual signals or simulation runs and analyse your comparison results with relative, absolute, and time tolerances. The compare tools in the Simulation Data Inspector facilitate iterative design and allow you to highlight signals that do not meet your tolerance requirements. For more information about the comparison operation, see How the Simulation Data Inspector Compares Data.

- Save and Share Simulation Data Inspector Data and Views.

Share your findings with others by saving Simulation Data Inspector data and views.

Here we are comparing the parameters like vehicle speed ,wheel speed , stopping distance ,slip etc. under various conditions (with abs, without abs, slip) and analyse the Variations of the above-mentioned parameters.

9. Callbacks

Callbacks are used to initialize values to model with preload, postponed options

Preload has chosen and values are initialised in it as model opens it preloads the values and keep ready for execution.

10. Matlab function Block

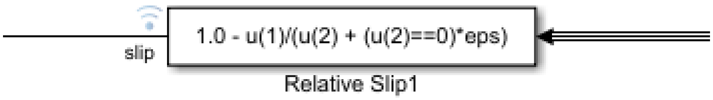

Fig 10.1 Matlab function of ABS model

MATLAB Function blocks enable you to define custom functionality in Simulink models by using the MATLAB language. They are the easiest way to bring MATLAB code into Simulink. MATLAB Function blocks support C/C++ code generation from Simulink Coder and Embedded Coder.

11. Lookup table

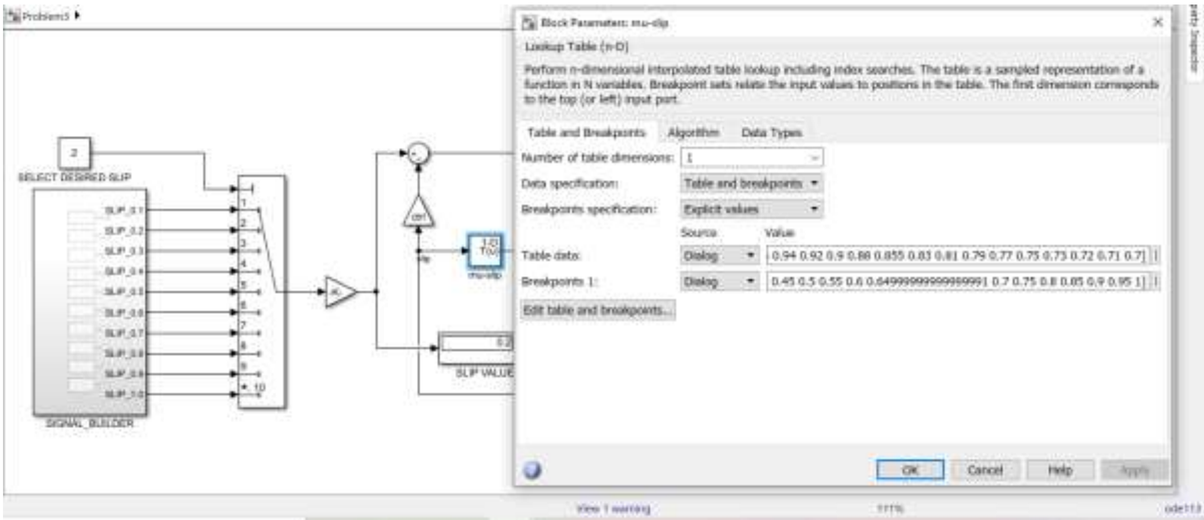

Fig 11.1 Lookup table of ABS model

A lookup table is an array of data that maps input values to output values, thereby approximating a mathematical function. Given a set of input values, a lookup operation retrieves the corresponding output values from the table. If the lookup table does not explicitly define the input values, Simulink can estimate an output value using interpolation, extrapolation, or rounding, where:

- An interpolation is a process for estimating values that lie between known data points.
- An extrapolation is a process for estimating values that lie beyond the range of known data points.
- A rounding is a process for approximating a value by altering its digits according to a known rule.

A lookup table block uses an array of data to map input values to output values, approximating a mathematical function. Given input values, Simulink performs a “lookup” operation to retrieve the corresponding output values from the table. If the lookup table does not define the input values, the block estimates the output values based on nearby table values.

12. Signal builder

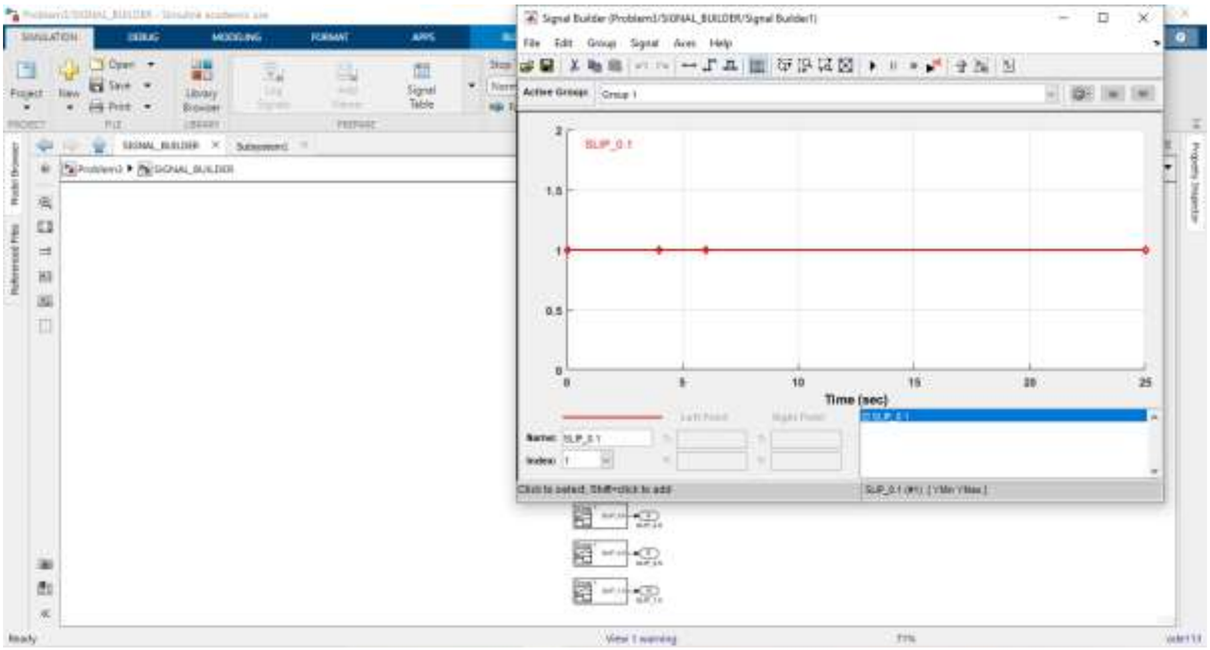

Fig 12.1 Signal builder of ABS model

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.

13. Solver

Ordinary Differential Equations

An *ordinary differential equation* (ODE) contains one or more derivatives of a dependent variable, *y*, with respect to a single independent variable, *t*, usually referred to as time. The notation used here for representing derivatives of *y* with respect to *t* is y′ for a first derivative, y″ for a second derivative, and so on. The *order* of the ODE is equal to the highest-order derivative of *y* that appears in the equation.

ODE113 is chosen

ODE113 can be more efficient than ODE45 at problems with stringent error tolerances, or when the ODE function is expensive to evaluate.

Accuracy - Low to High