

# Building a LabVIEW User Interface for a Simulink® Model with LabVIEW Simulation Interface Toolkit

36 ratings | 3.75 out of 5

## Overview

The Simulation Interface Toolkit (SIT) provides a seamless integration between The MathWorks Inc. Simulink® software and LabVIEW. The Simulation Interface Toolkit automatically generates LabVIEW code to interface with a Simulink model resulting in a flexible and easy-to-use user interface. You first must configure the Simulink® model to communicate with LabVIEW. Then you can create a LabVIEW host VI that automatically calls, runs, and interacts with the Simulink® model.

## Table of Contents

1. [Configuring the Simulink Model](#)
2. [Creating a LabVIEW User Interface](#)
3. [Creating Connections between the LabVIEW User Interface and the Simulink® Model](#)
4. [Attachments](#)

## Configuring the Simulink Model

Before LabVIEW can communicate with the Simulink® model, you must correctly configure the model in Simulink®.

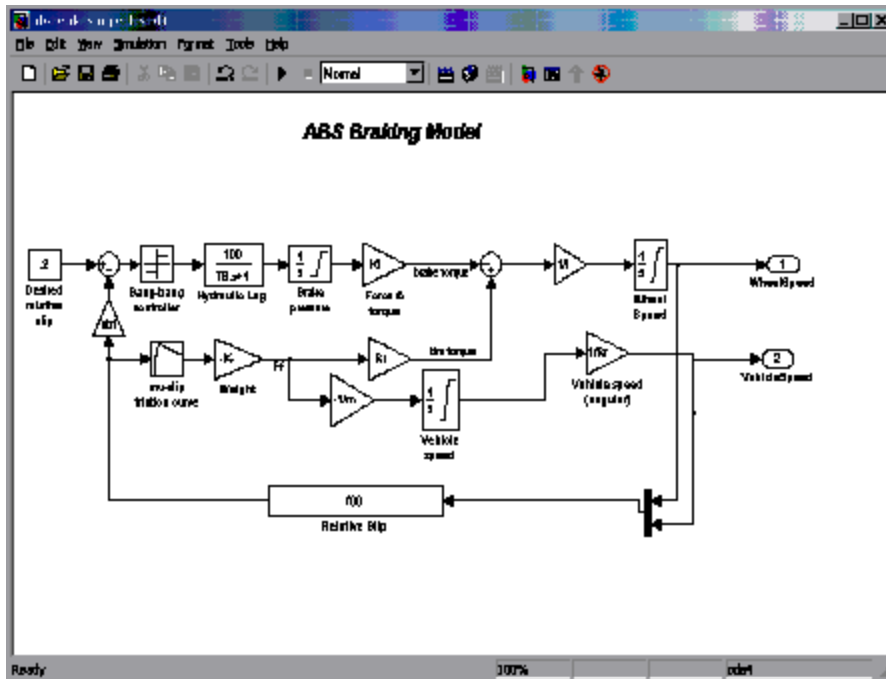
1. Save the attached files, `absbrakesingleclosed.mdl` and `abssingleclosed.vi`, to the computer.
2. Launch MATLAB® and notice how the MATLAB® command window displays the following message:

```
Starting the SIT Server on port 6011  
SIT Server started
```

The Simulation Interface Toolkit automatically installs the SIT server. If you do not see this message, you have not installed the toolkit correctly.

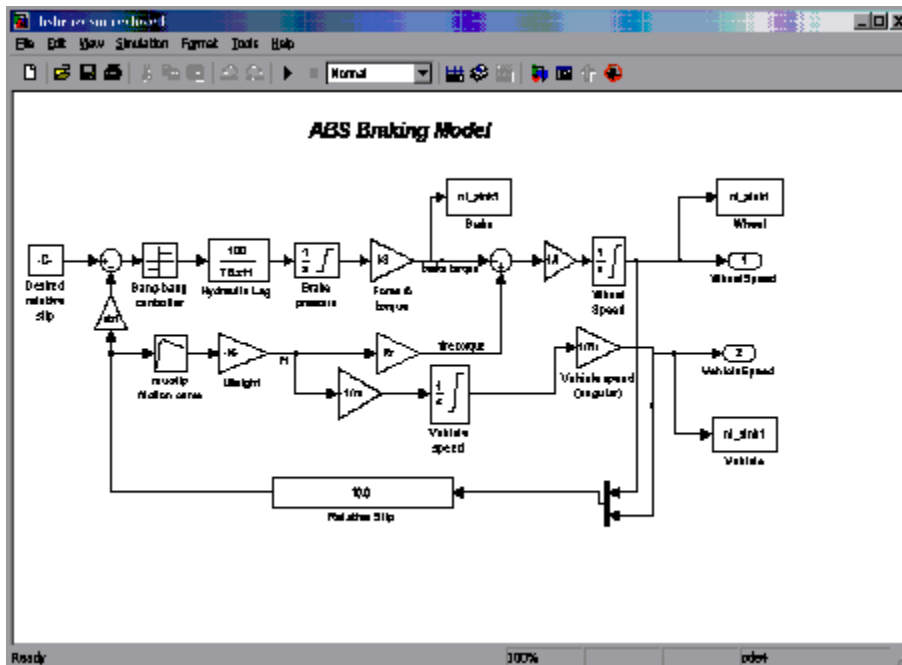
3. Enter **simulink** in the MATLAB® command window to launch the **Simulink® Library Browser** window.
4. Select **File»Open** and select the `absbrakesingleclosed.mdl`, which is file you downloaded in Step 1, to open the ABS Braking model.

The simulation model should look similar to the following diagram:



This Simulink® model represents an anti-lock braking system for a desired slip. The brake torque adjusts according to the current wheel speed and vehicle speed to achieve the desired slip.

5. At each point you want to monitor in the simulation model, add an **NISink** to the `absbrakesingleclosed.mdl` to allow LabVIEW to read the data at that point in the simulation model. In the **Simulink® Library Browser** window, double-click the **NI Toolkit Sinks** library. Notice the **NISink** and **NIXYGraph** blocks.
6. Place an **NISink** block in the model window of the `absbrakesingleclosed.mdl`.
7. Connect the signal to the **NISink** block. The ABS Braking Model should look similar to the following diagram:



You also can use the **NIXYGraph** block to plot and compare two signals in LabVIEW.

8. Save the Simulink® model.

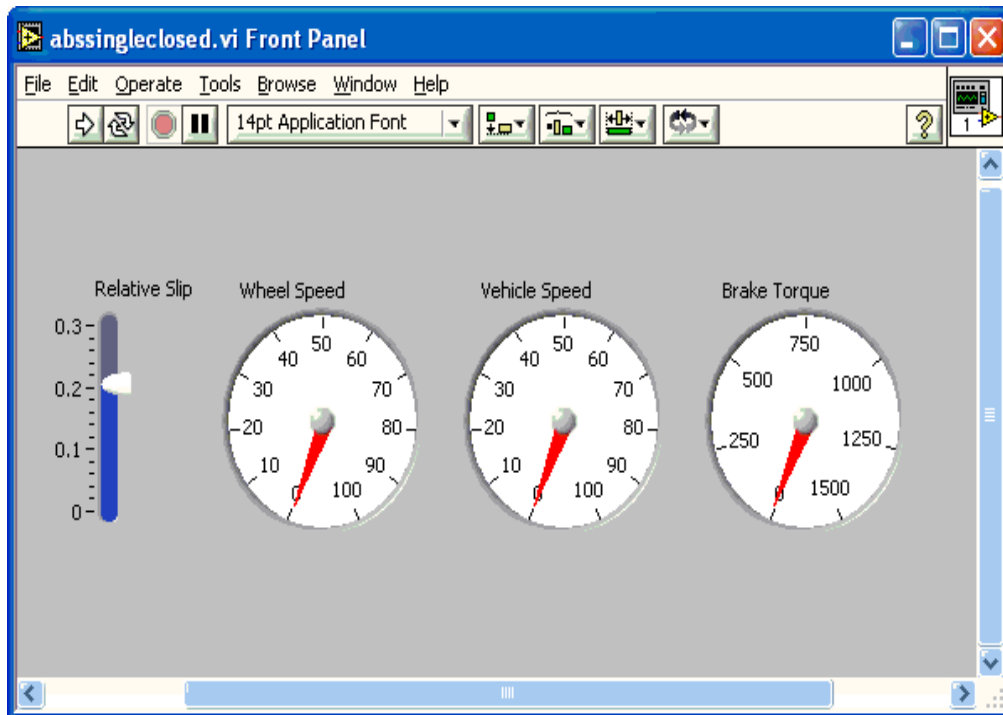
You now can create a user interface in LabVIEW for the Simulink® model. Notice that the model remains fully-operational within the Simulink® environment.

### Creating a LabVIEW User Interface

To interact with the ABS Braking Model using a LabVIEW user interface, you need to create a user interface in LabVIEW.

1. Launch LabVIEW and open a new VI.
2. Select one slider control and three gauges and place them on the front panel.
3. Label the slider control *Relative Slip*.
4. Label the three gauges *Wheel Speed*, *Vehicle Speed*, and *Brake Torque*.
5. Change the **Relative Slip** scale to 0 to 0.3.
6. Change the **Wheel Speed** and **Vehicle Speed** scales to 0 to 100
7. Change the **Brake Torque** scale to be 0 to 1500.

The front panel of the VI should look like the following front panel:



8. Save this VI as `abssingleclosed.vi`. Notice that there is no block diagram code for calling the Simulink® model.

### Creating Connections between the LabVIEW User Interface and the Simulink® Model

The Simulation Interface Toolkit enables LabVIEW to run and communicate with the Simulink® model once you make the appropriate connections. MATLAB® and Simulink® must remain open to configure and run the simulation model.

1. On the front panel menu bar, select **Tools»SIT Connection Manager** to launch the **SIT Connection Manager** dialog box. Notice how the **Controls and indicators** listbox displays the controls and indicators you created on the front panel of the `abssingleclosed.vi`.
2. Click the **Load Model** button to display the **Select Host** dialog box.
3. In the **Select Host** dialog box, make sure the **Machine Name/IP** is `localhost`.

4. Click the **OK** button. By selecting localhost, LabVIEW connects to the SIT Server that MATLAB® initially started. If you choose to start MATLAB® on a separate computer with SIT 2.0 installed, you can connect to that computer by typing in the correct IP address.

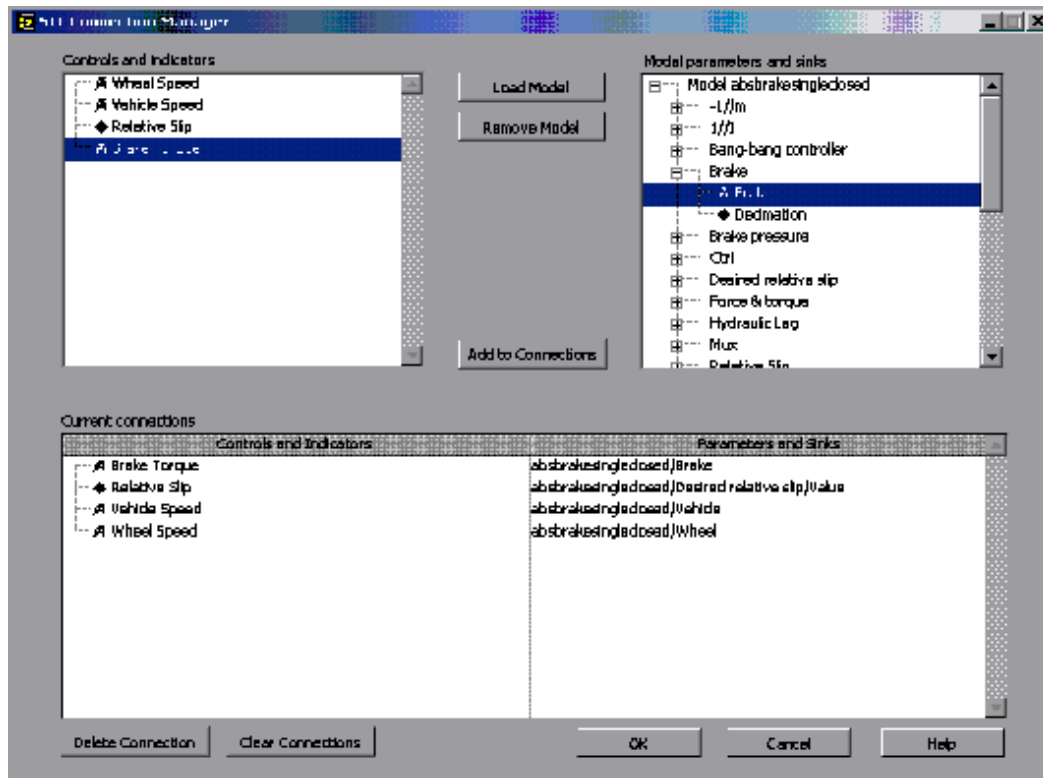
5. In the **Select Model** dialog box, navigate to the folder where you saved the `absbrakesingleclosed.mdl`. Select the `absbrakesingleclosed.mdl` file and click the **OK** button.

Notice how all the parameters and sinks associated with the ABS Braking Model appears in the **Model parameters and sinks** listbox in the **SIT Connection Manager** dialog box.



6. To create a connection between the **Brake Torque** gauge in LabVIEW and the **Brake Torque** input in the Simulink® model, select **Brake Torque** in the **Controls and indicators** listbox and select **Brake»Brake** in the **Model parameters and sinks** listbox.

7. Click the **Add to Connections** button to add the connection to the **Current connections** table.

8. Similarly, create the remaining connections between the LabVIEW controls and indicators and the Simulink® parameters and sinks. The **SIT Connection Manager** dialog box should look similar to the following figure:

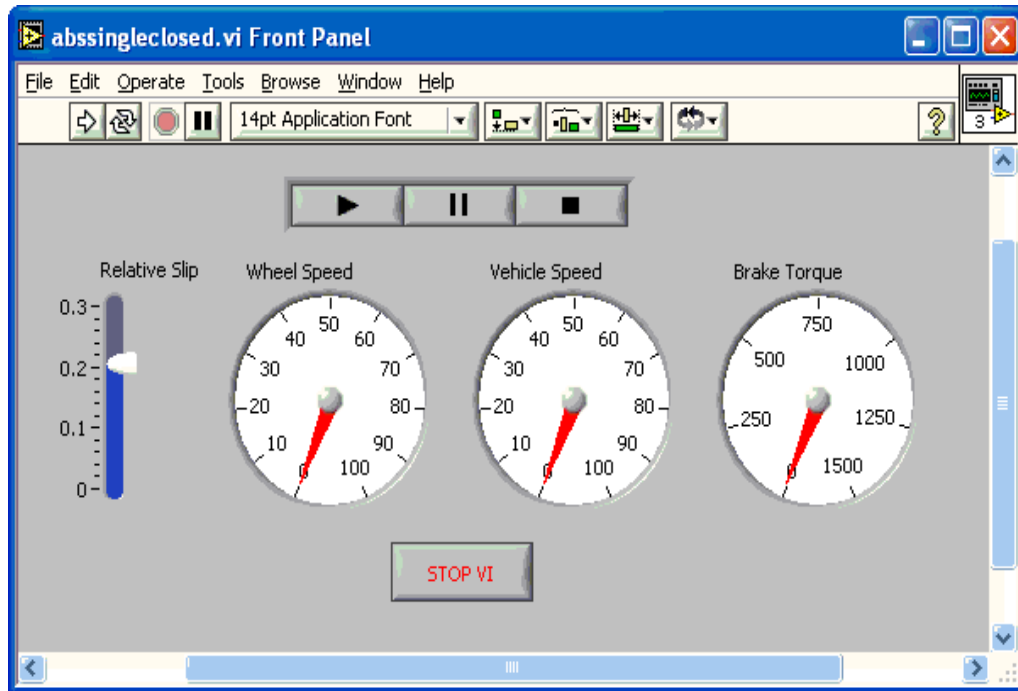



[\[+\] Enlarge Image](#)

Notice that the parameters and controls have a  icon in front of them while sinks and indicators have a  icon in front of them. Use these icons to ensure that you do not make an invalid connection.


9. Click the **OK** button.

Notice that the Simulation Interface Toolkit generates the block diagram code that governs the interaction between the LabVIEW front panel and the `absbrakesingleclosed.mdl`. The VI should look similar to the following figure:



10. Click the **Run** button,  to run the VI. The **Select Host** dialog box appears.

11. Verify that the host computer is **localhost**. Click the **OK** button. Although the VI starts do run, the Simulink® model is not running yet.

12. To run the Simulink® model, `absbrakesingleclosed.mdl`, click the **Run** button, , on the front panel.

13. Adjust the values of the **Relative Slip** slider while the Simulink® model is running. You can see the effect of relative slip on the wheel speed, vehicle speed, and brake torque.

The process described here does not require The MathWorks Real-Time Workshop®. This VI uses Simulink® to run the simulation. You also can use the Simulation Interface Toolkit together with Real-Time Workshop® to build the model into a DLL for stand-alone simulation. You need MATLAB®, Simulink®, and Real-Time Workshop® to build a model DLL and configure the user interface using the **SIT Connection Manager** dialog box. Once you create the model DLL and user interface, you need only LabVIEW to run the simulation. For hardware-in-the-loop (HIL) and prototyping applications, you can run the model DLL in LabVIEW Real-Time to provide real-time inputs and outputs and to ensure determinism.

MATLAB®, Simulink®, and Real-Time Workshop® are registered trademarks of The MathWorks, Inc. Other product and company names listed are trademarks and trade names of their respective companies.