755 Phoenix Drive, Ann Arbor MI, 48108, USA
Phone: 734 668-2930 • Fax: 734 668-2877 • Email: info@carsim.com
carsim.com

NI RT Target Systems

This memo describes how to use a VehicleSim (VS) product (BikeSim, CarSim, or TruckSim) with a National Instruments (NI) Real-Time (RT) target system. In this guide, we assume you are using CarSim 2019.1 or newer; the instructions are the same if you are using BikeSim or TruckSim 2019.1 or newer instead.

System Requirements

For NI ETS Real-Time system, we support and have tested LabVIEW 2015/2016/2017/2018/2019 and VeriStand 2015/2016/2017/2018/2019.

	CarSim	TruckSim	BikeSim
LabVIEW-RT	2.0 GHz Dual Core	2.4 GHz Dual Core	2.0 GHz Dual Core
VeriStand	2.0 GHz Dual Core	2.4 GHz Dual Core	2.0 GHz Dual Core

For NI Linux Real-Time system, we support LabVIEW 2015 and newer; we have tested LabVIEW 2015/2016/2017/2018/2019 and VeriStand 2015/2016/2017/2018/2019. On cRIO/cDAQ Real-Time Linux target, “Write all outputs” should not be checked. We have tested the NI Industrial Controller (IC-3173) with Linux RT system and LabVIEW 2016. We support LabVIEW 2019 with NI Linux Real-Time on PXIe controllers.

	BikeSim, CarSim and TruckSim
LabVIEW-RT	cRIO/cDAQ 1.9 GHz CPU and IC-3173 i7 CPU
	PXIe-8840 Quad-Core, PXIe-8861 and PXIe-8880
VeriStand	cRIO 1.9 GHz CPU
	PXIe-8840 Quad-Core, PXIe-8861 and PXIe-8880

NI cRIO with 1.9 GHz CPU is either cRIO-9034 (4 slots) or cRIO-9039 (8 slots). NI cDAQ with 1.9 GHz CPU is either cDAQ-9136 (4 slots) or cDAQ-9137 (8 slots).

For VeriStand on Linux RT, in VehicleSim versions 20.1 and newer, you can use FMU instead of "VeriStand Model Framework" which requires the GCC Linux compiler.

> **Note** Any firewall should be configured to allow the host and real time target to communicate.

Install Datasets and NI RT Programs

Working with NI-RT requires VS Math Models for the vehicle models to be compiled for the RT system, along with some datasets that connect to those RT VS Math Models.

> **Note** You will need the optional RT license for NI RT in order to use NI RT with BikeSim, CarSim, or TruckSim.

To install the VS Math Models and example datasets, you can use the Database Builder to build a new database, or you can import a consolidated parsfile (CPAR) to an existing database. Either method will install the necessary VS Math Models, corresponding NI RT projects, and supporting datasets. Start with a PC that has both NI RT and your VS product installed on it.

To build a new database, launch CarSim and select **Open Database Builder…** from the Select Recent Database window. In the Database Builder window (Figure 1), toggle on **Show all CPAR archives** (1). Select RT_NI-RT.cpar (2) along with any other data groups that you would like to include in the new database.

> **Note** If you are planning to do the VeriStand ABS example tutorial which is included in the Run Simulations with NI VeriStand RT section of this document, you should also select the Simulink_models.cpar from the Database Builder (4).

Click **Build Database from Selected Items** (3). Select the folder where you would like to save the database and click **OK**.

To import a CPAR file to an existing database launch CarSim and select the database that you will use with NI RT. In CarSim, navigate to **File** > **Import Parsfile (Any Export Type)** and import the following CPAR file:

`CarSim_Prog\Resources\CPAR_Archives\RT_NI-RT.cpar`

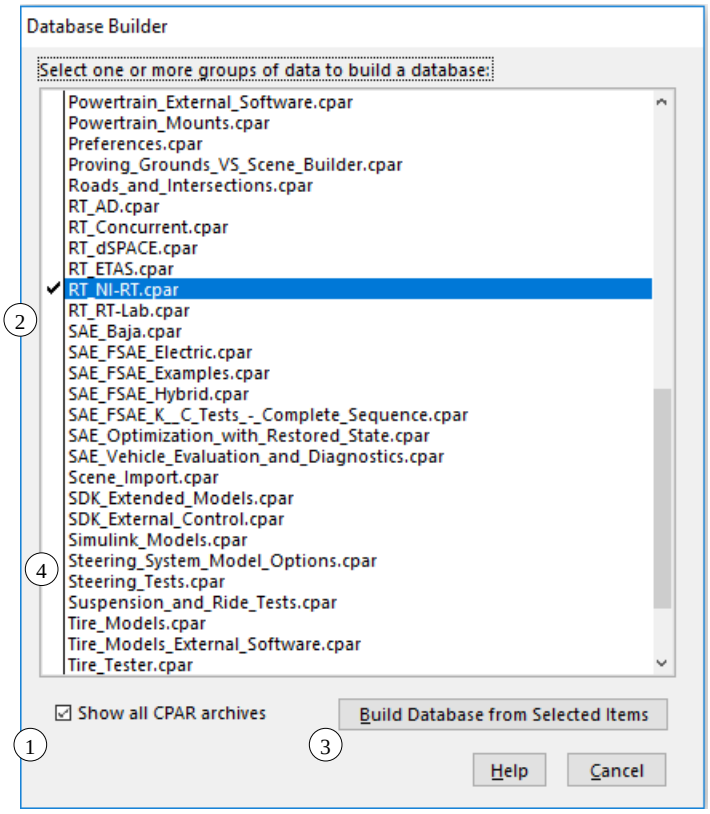

Figure 1.Database Builder

Use the default settings (Figure 2) to import the file. The VS browser will copy files into the current database, import some dataset files, and refresh. This can take a minute or two.

> **Note** You must build a new database that includes the required VS Math Models or import the CPAR file to install the required VS Math Models even if you are not planning to use the example models.

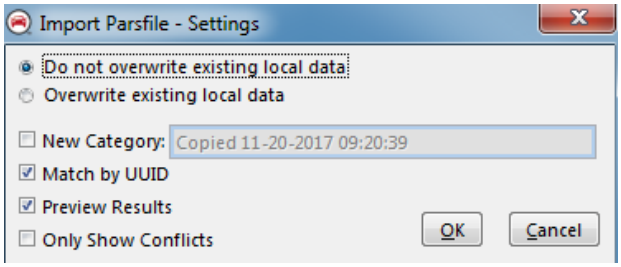

Figure 2. Import Parsfile Default Settings.

After building a new database or importing the consolidated parsfile, the NI examples can be found in **Datasets > RT: NI-RT** (Figure 3).

Your database now includes all RT VS Math Models which are needed for NI RT, in addition to the datasets that are used for the examples described in this document. The real-time VS Math Models (DLLs and LLB) are installed in the following folder:

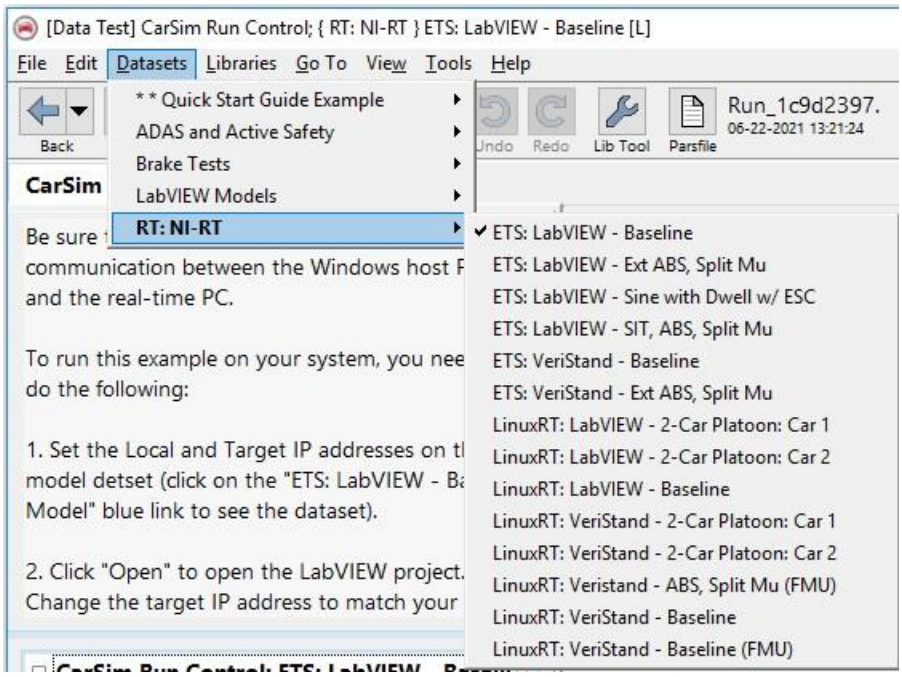

Figure 3. NI-RT Examples

`{Database root folder}\extensions\NI_RT\Solver_DLLs`

The NI RT example models are in the sub-folders of the following folder:

`{Database root folder}\extensions\NI_RT\`

Note	For NI-VeriStand users: in order to run NI-RT VeriStand in real time, you must have the LabWindows/CVI real-time engine installed on your target machine. If it is not, you will need to manually install the `cvi_lvrt.dll` file onto the target machine. We provide the `cvi_lvrt.dll` file in the folder {*Database root folder*}`\NI-RT\Solver_DLLs`. Locate it and ftp it to the target machine (username: `root`, password is empty) from the root folder to `NI-RT/System`. Make sure to set the file transfer type to binary before transferring.

Install Vehicle Solvers onto NI Linux RT Target

If you are using ETS for the target system, you can skip this section.

To install the NI-RT Linux compatible VS math models

1. Use the CarSim **Tools** menu command **Install RT Solvers to Target** … (Figure 4). You will see an installation window pop up (Figure 5).
2. Select **NI Linux RT System** from the **Real-Time System** pull down menu. The solver installation package location is automatically detected (Figure 6).

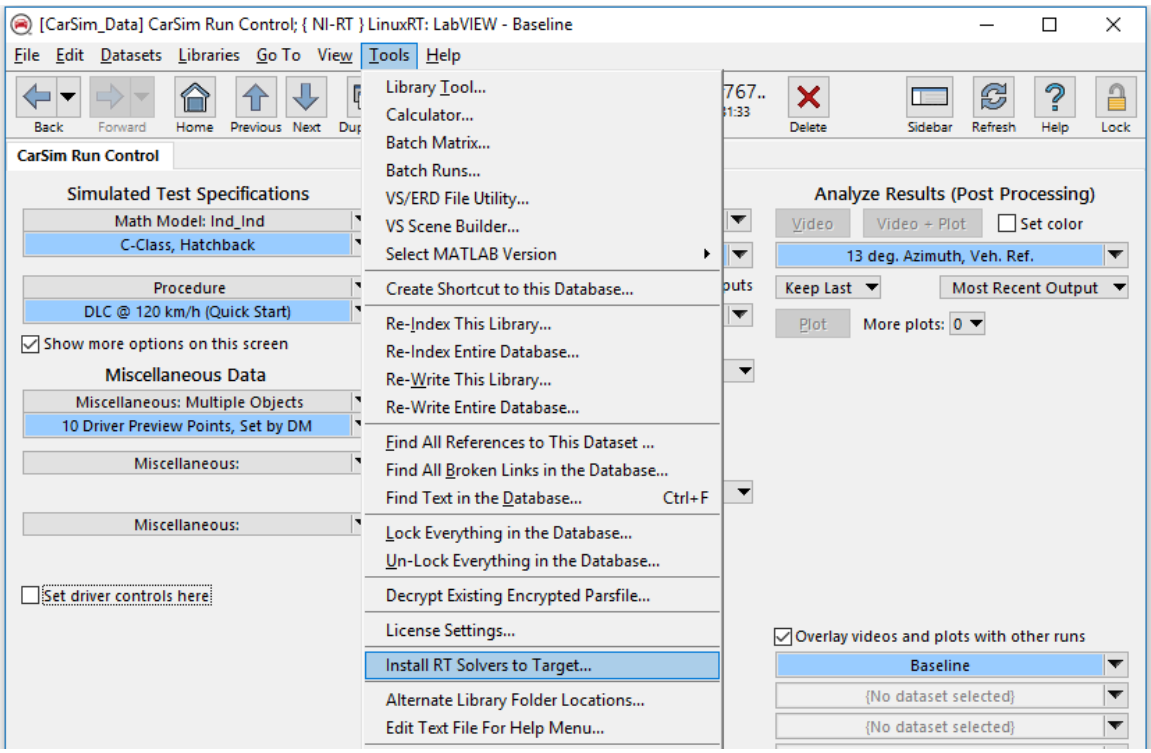

Figure 4. Tools menu: Install RT Solvers to Target.

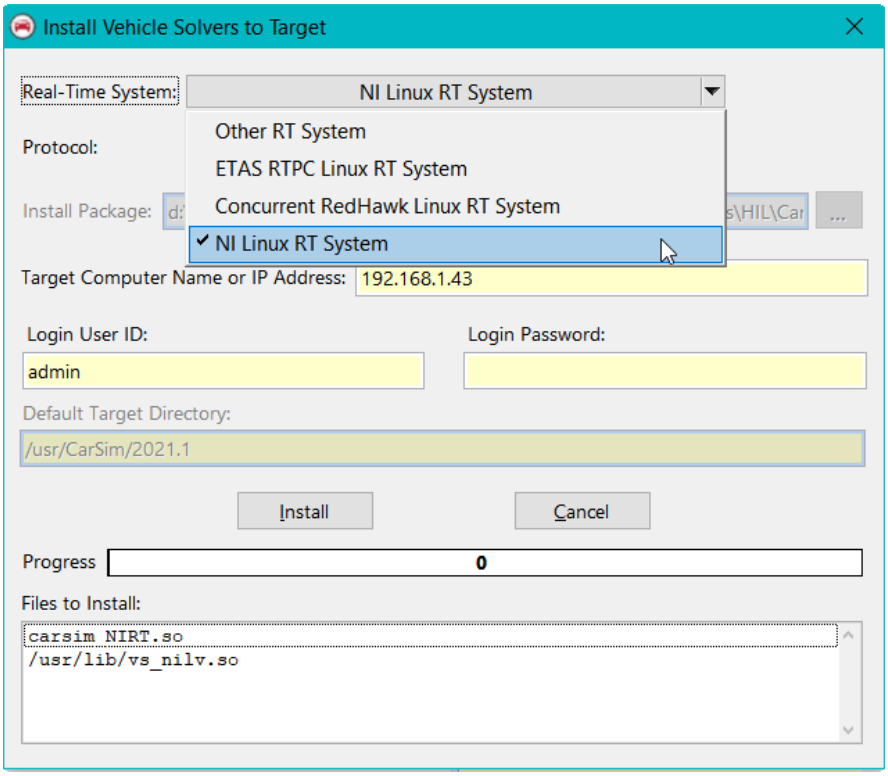

Figure 5. Select a target RT system to install Vehicle Solvers.

3. Select protocol **SSH (SFTP)** (Figure 6). You cannot use **WebDAV** for installing VS math models.

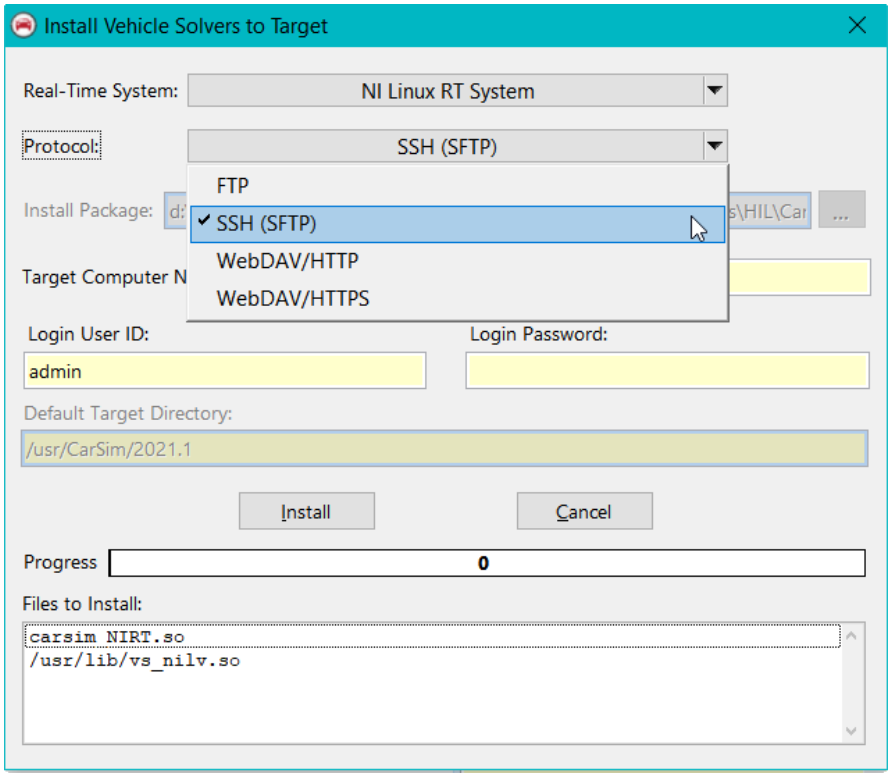

Figure 6. Install Vehicle Solvers to NI Linux RT system target.

4. Enter your target machine address into **Target Computer Name or IP Address** yellow field (Figure 6).
5. Click the button “Install” to initiate the installation process (Figure 6).

Installing the Example Datasets

To install the datasets, follow the instructions in the section “Install Datasets and NI RT Programs”.

Run a Simulation on LabVIEW RT (ETS RT Target)

The CarSim RT installation includes some examples that can be used to get started with LabVIEW RT. However, first you must prepare the model by specifying some details of your configuration.

Prepare the LabVIEW RT model

1. From the **Run Control** screen, use the **Dataset** menu to go to the run **RT: NI-RT–>ETS: LabVIEW - Ext. ABS, Split Mu** example (Figure 7).

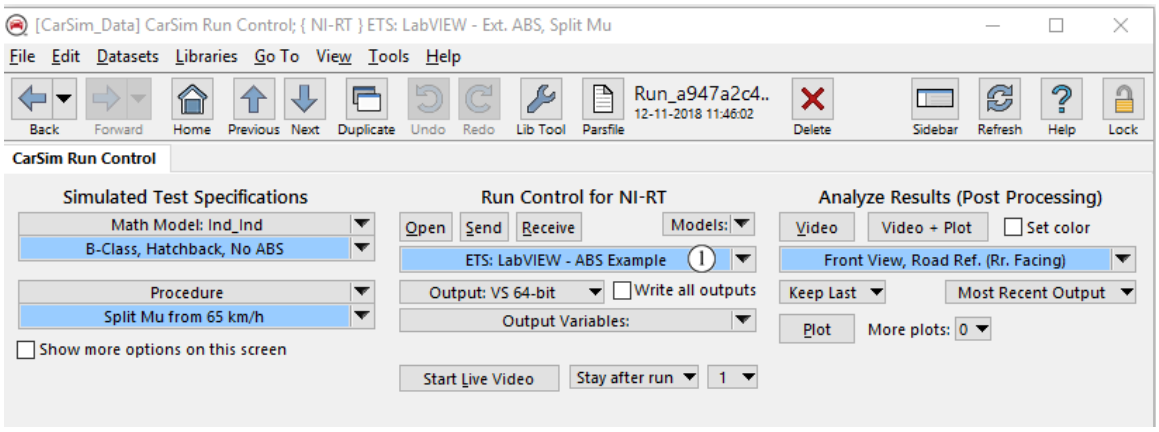

Figure 7. Example run with ABS model and LabVIEW RT.

2. Click the blue link for the NI-RT model (1) (Figure 7) to view the linked dataset (Figure 8). This screen specifies the IP addresses of the target (1) and host (2) machines and the pathname for the LabVIEW project file (3) and links to datasets that specify variables to be imported from LabVIEW (4) and exported to LabVIEW (5). Check the **Leave files on the target computer** and **Set integration method and time step** checkboxes. Select the integration method AM-2 and math model time step to 0.001 (6).

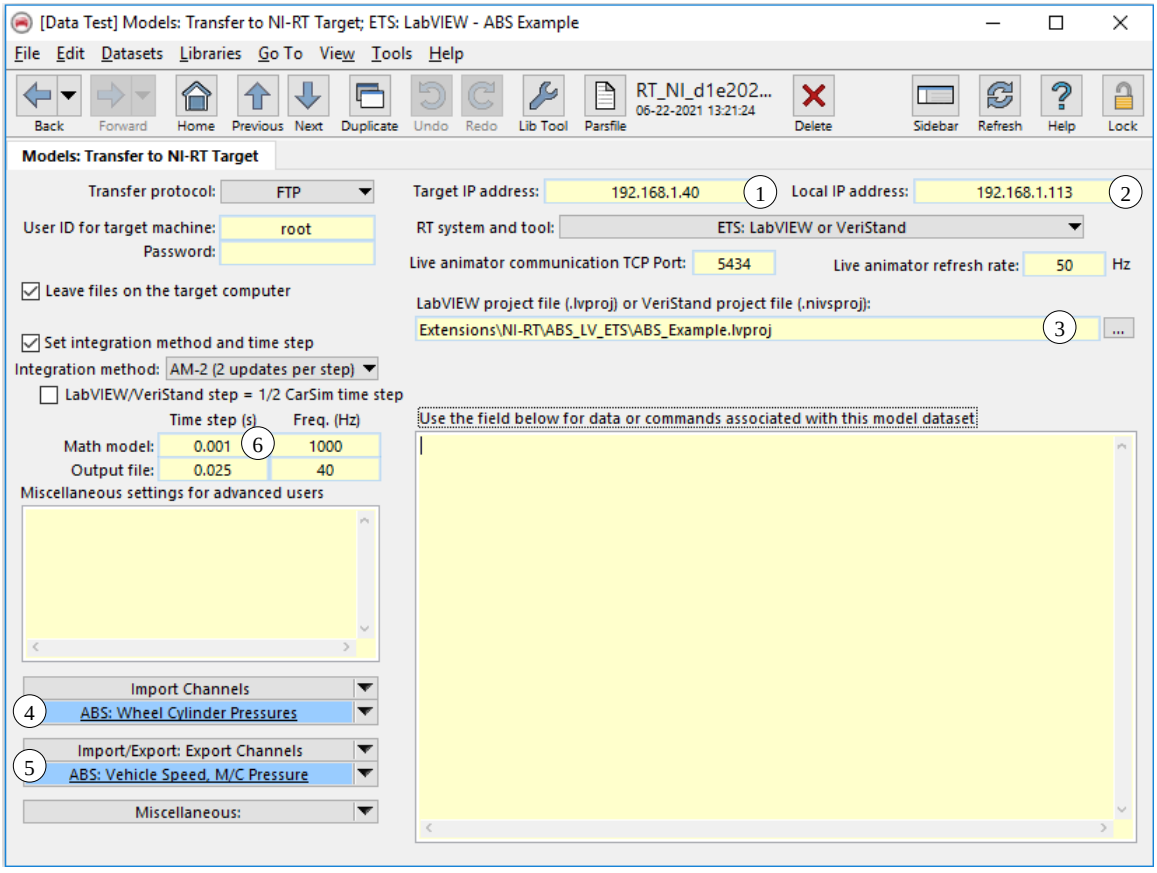

Figure 8. Dataset for LabVIEW RT ABS example.

3. Return to the **Run Control** screen.

Run the LabVIEW RT model

1. Open your LabVIEW project by clicking the **Open** button from the **Run Control** screen.

   In this example (Figure 9), there are two nodes (computers) containing the model file (VI file): `My Computer` (Windows host PC) and `PXI-8106` (LabVIEW RT target PC). You can run this example on either computer because there is no hardware included, but this document describes how to run on an RT target PC.

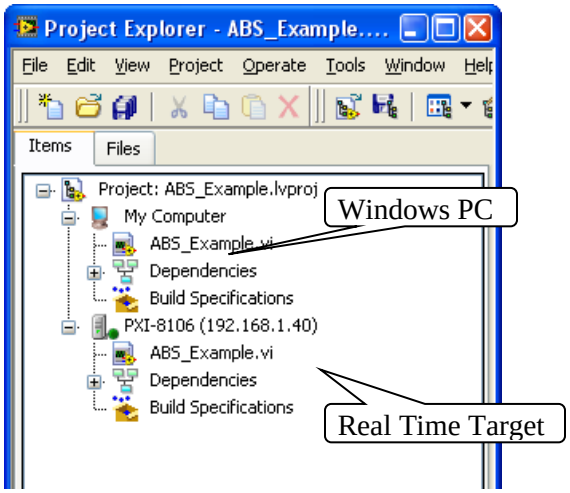

Figure 9. LabVIEW project with ABS example.

1. In CarSim, click the **Send** button on the **Run Control** screen.

   This launches the animator (to support live animation) and sends data for the simulation from the database to the target machine. This also sends a copy of the data file to the directory containing the LabVIEW project file.

2. Double click `ABS_Example.vi` on the real time node (Figure 9) to open the example on the real time target (Figure 10).

3. Click on the run button (⇨) on the LabVIEW model (or use the keyboard shortcut **Ctrl+R**) to deploy the model to the real time target and make the run.

4. After the simulation finishes, the simulation results are automatically sent back to the host Windows PC. If not, click the **Receive** button on the **Run Control** screen (Figure 7) to transfer the output files (ERD, BIN, ECHO, and LOG files) back to your VS database for post processing (plotting, animation, etc.).

Note	If you want to stop the simulation, do not click the **Abort Execution** button. This button will not release memory allocated by the VS solver. Instead, click the **STOP** button.

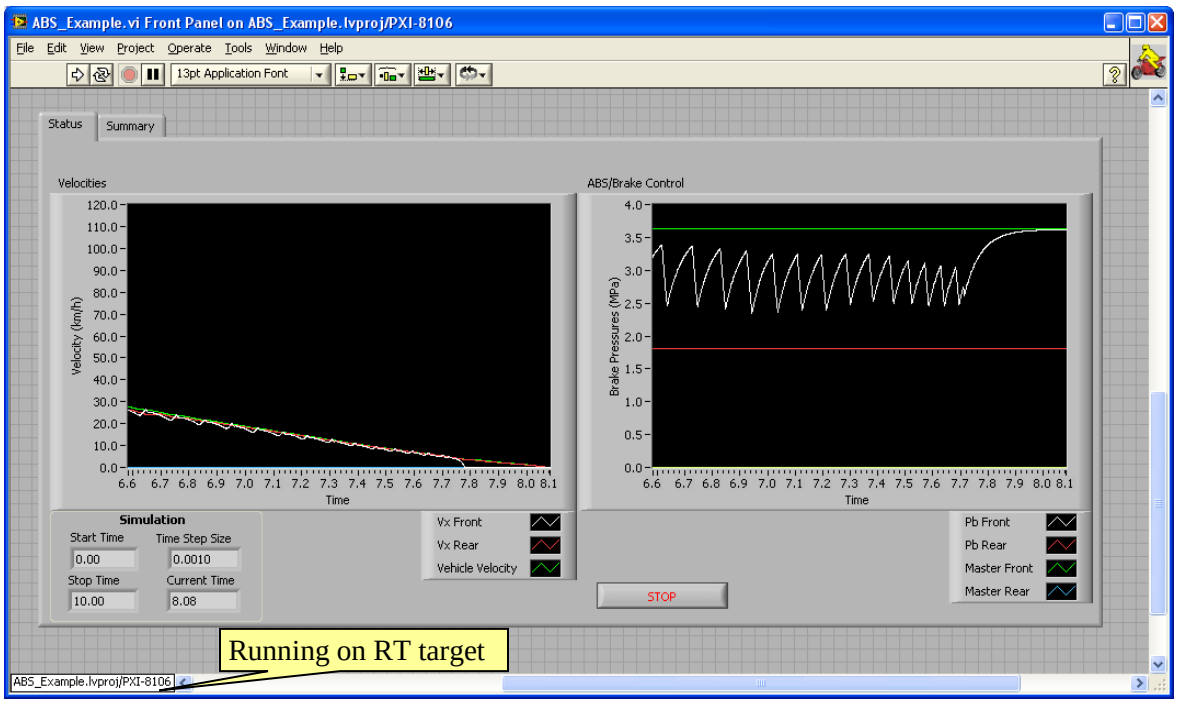

Figure 10. LabVIEW model running on the real time target.

Run a Simulation on LabVIEW RT (Linux RT Target)

This section describes the process of running a simulation on LabVIEW RT for Linux OS. Running a simulation on Linux OS is similar to running a simulation on ETS which was described in the previous section, but there are a few differences.

Prepare the LabVIEW RT model

The CarSim RT installation includes some examples that can be used to get started with LabVIEW RT.

1. From the **Run Control** screen, use the **Dataset** menu and select the **RT: NI RT–>LinuxRT: LabVIEW - Baseline** example (Figure 11).
2. Click the blue link **LinuxRT: LabVIEW – Baseline** to view the linked dataset for the NI-RT model (Figure 12). Select **Linux RT: LabVIEW** from **RT System and Tool** pull-down list. Make sure Target ID or IP address, Host IP address, and LabVIEW Project File(`.lvproj`) entries are correct.
3. Return to the **Run Control** screen.

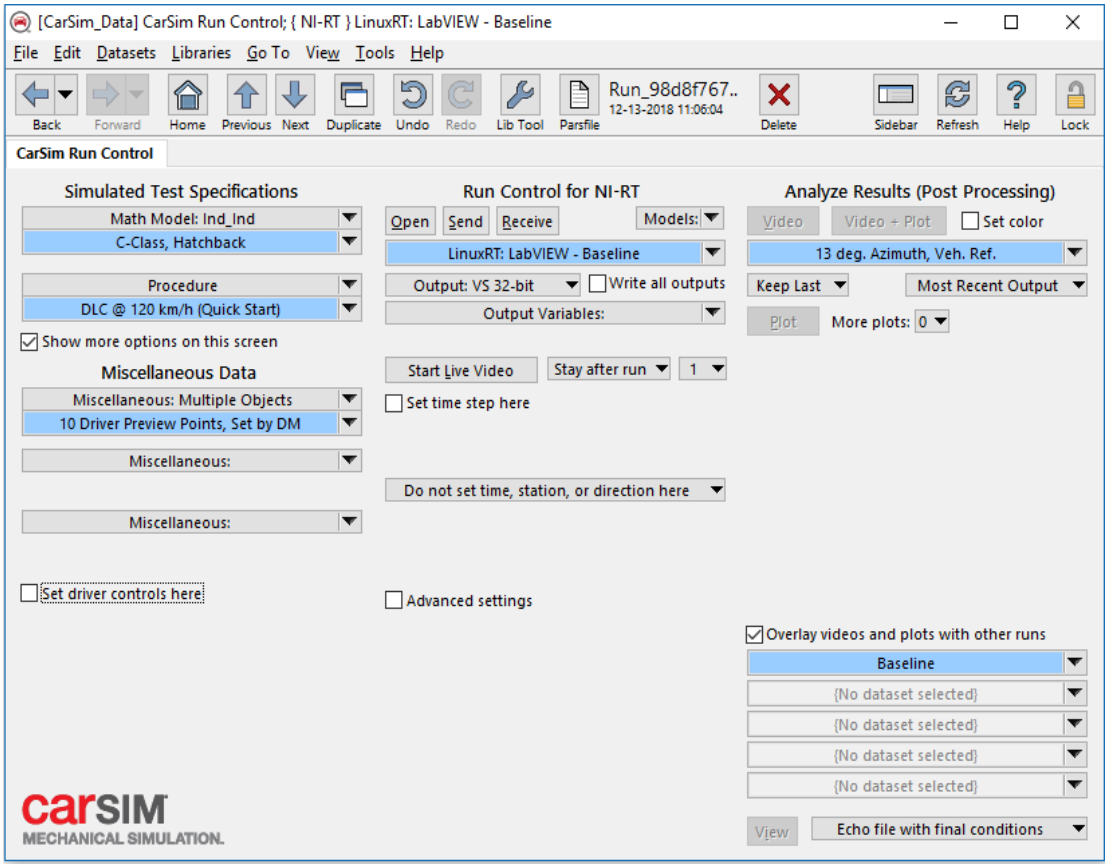

Figure 11. LinuxRT LabVIEW Baseline Example.

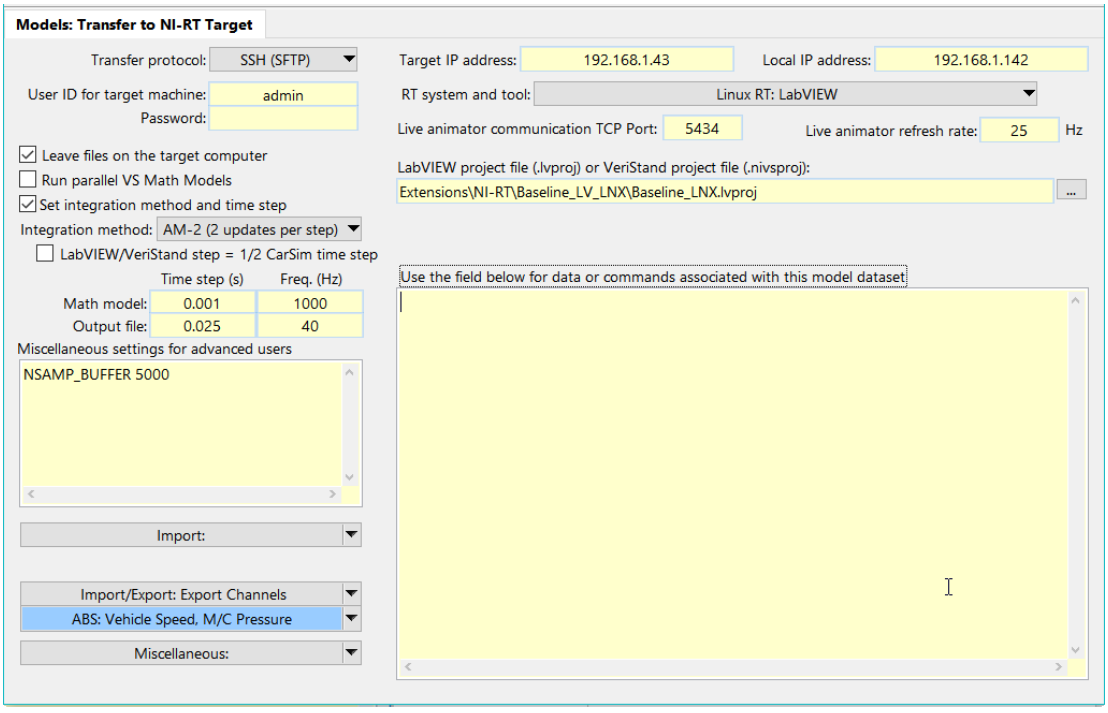

Figure 12. Dataset for Linux RT LabVIEW example.

Run the LabVIEW RT model

1. Open your LabVIEW project by clicking the **Open** button from the CarSim **Run Control** screen.

2. In this example, there are two nodes (computers), and each one has the model file (vi file): `My Computer` (Windows host PC) and `cRIO-9034` (Linux RT LabVIEW target) (Figure 13)

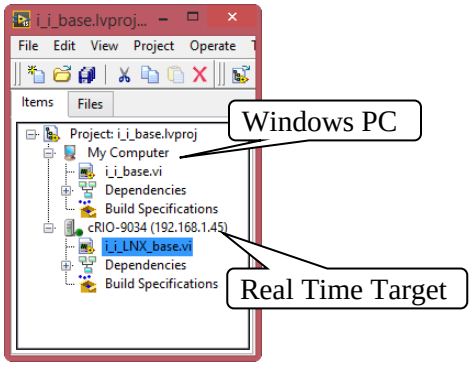

Figure 13. LabVIEW project with Baseline example.

3. Click the **Send** button on the CarSim **Run Control** screen.

   Clicking the **Send** button launches the animator (to support live animation) and sends data for the simulation from the database to the target machine. This also sends a copy of the data file to the directory containing the LabVIEW project file.

4. Double click `i_i_LNX_base.vi` on the real-time node (Figure 13) to open the example on the real-time target (Figure 14). Go to **Windows** > **Show Block Diagram** and see the connection diagram shown in Figure 15. Inside this diagram, open the Call Library Function window by clicking a car icon (there are three icons with a car image, clicking any of the icon is ok). You can see that the library name is vs_nilv.* (Figure 16).

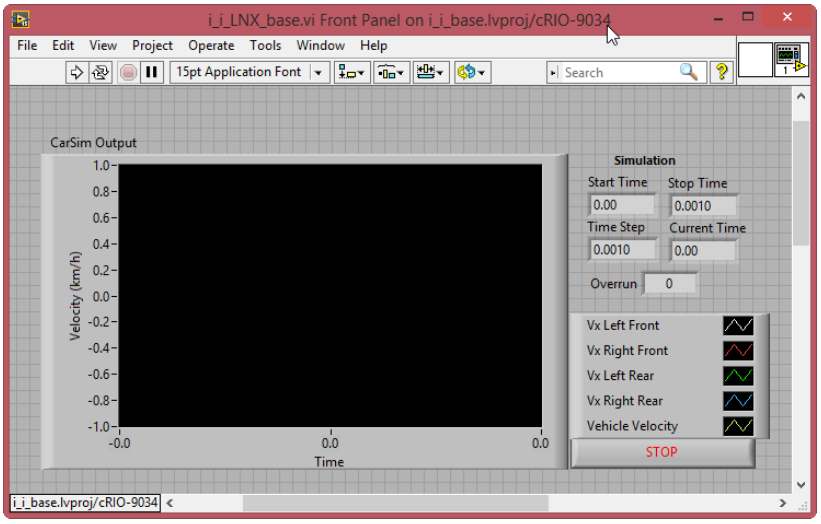

Figure 14. LabVIEW model run on the real-time target.

5. In the LabVIEW window (Figure 14), click on the run button (⇨) (or use the keyboard shortcut **Ctrl+R**) to deploy the model to the real-time target and make the run.

6. After the simulation is done, the files are automatically transferred to the host PC. If the files are not transferred, click the Receive button on the CarSim **Run Control** screen to transfer the results (ERD, BIN, ECHO, and LOG files) back to your VS database for post processing (plotting, animation, etc.).

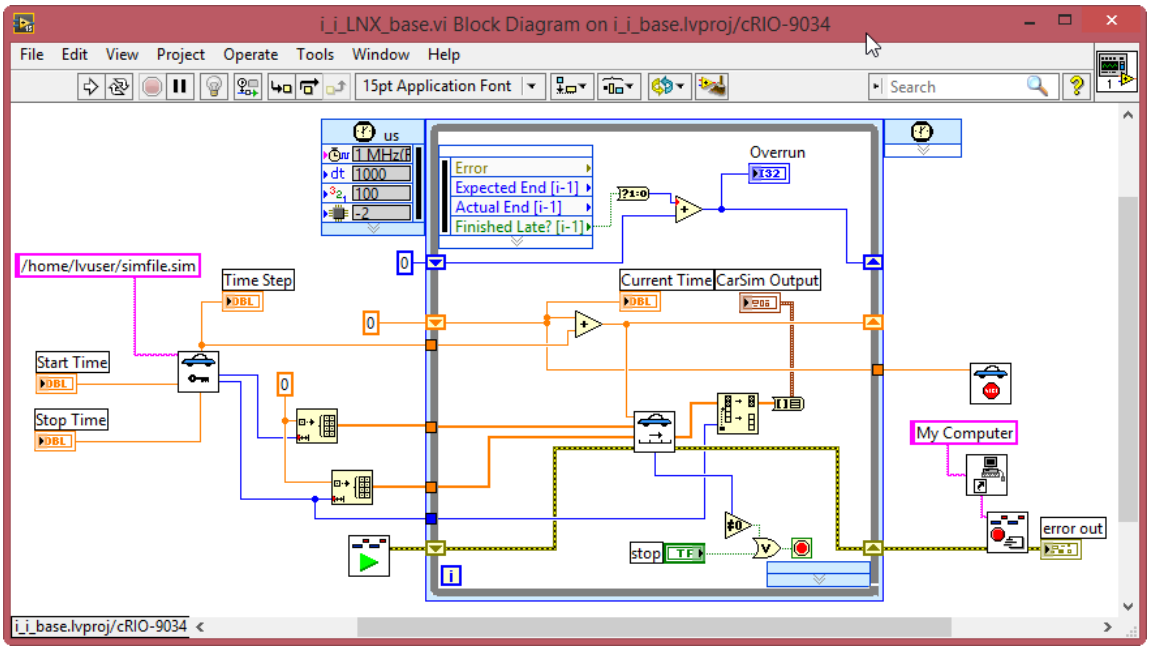

Figure 15. i_i_LNX_base.vi Block Diagram.

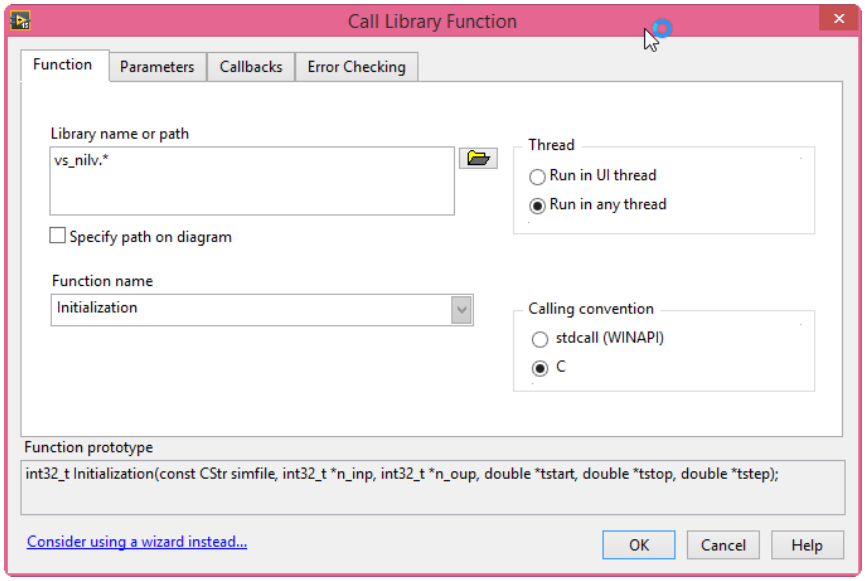

Figure 16. Call Library Function Window.

Note If you want to stop the simulation, do not click the **Abort Execution** button. This button will not release memory allocated by the VS solver. Instead, click the **STOP** button.

Run Simulations with NI VeriStand RT (ETS)

The installation includes some examples that can be used to get started with National Instruments RT. However, first you must prepare the model by specifying details of your configuration.

Create real time build to use in VeriStand

In this section, we talk about creating a real-time model for VeriStand from a Simulink model. If you want to create your own Simulink model, you need to have NI MIT (Model Interface Toolkit). For more information, please refer to NI documentation.

1. Start CarSim.
2. From the **Run Control** screen, use the **Dataset** menu to go to **Simulink Models–>Ext. ABS: Split Mu**.
3. Click the **Duplicate** button to copy current data set to a new source. Set the **Category** as `RT: NI-RT` and set any title you want. In this manual, we will use `VeriStand: Ext. ABS, Split Mu` for the title (Figure 17).

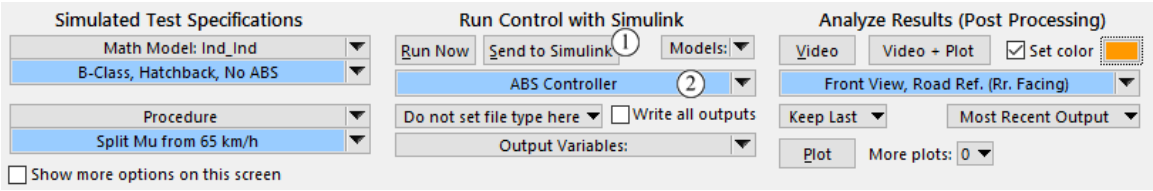

Figure 17. NI-RT VeriStand: ABS Example.

4. Click the **Send to Simulink** button (1) to open Matlab and the Simulink model.
5. Select **Save As** to save this Simulink model in a new location. In this example, we will save it in the folder `CarSim_Data\Extensions\NI-RT\ABS_VeriStand` and rename it `abs_VeriStand.slx`.
6. In Matlab, change the current folder to the new location where you saved the Simulink model in the previous step.
7. Delete the CarSim S-Function.
8. Open Simulink Library Browser and replace the S-Function you just deleted with the blocks **NI VeriStand Blocks–>NIVeriStand In1** and **NI VeriStand Blocks–>NIVeriStand Out1** (Figure 18). In this example, **NIVeriStand In1** is renamed as `ABS_Export` and **NIVeriStand Out1** as `Brake Pressure`.

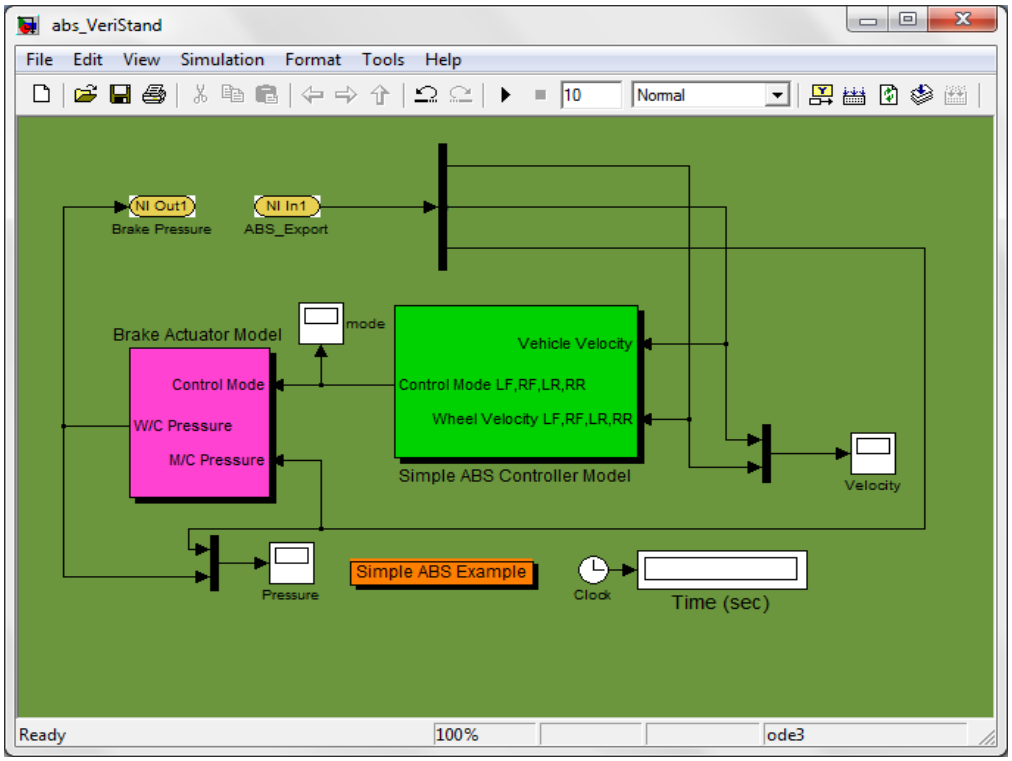

Figure 18. ABS Example Simulink Model for NI RT.

9. Now you need go back in CarSim to get Import/Export variables information. Click the blue link, ABS Controller (2 in Figure 17) and go to Import Channels/Export Channels to find the import and export variables and record them. The import and export channels are shown in Figure 19 for this example.

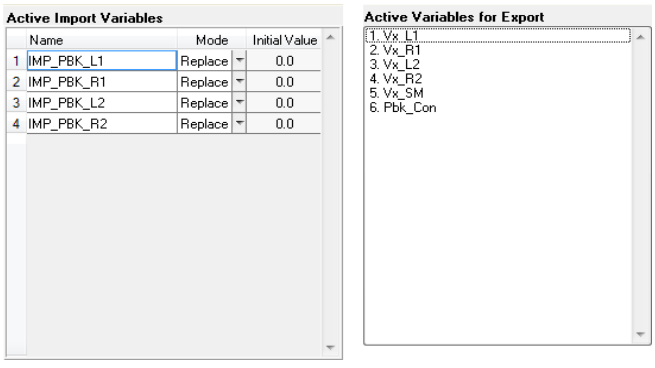

Figure 19. Active Import and Export Variables from CarSim.

10. Now, go back to the Simulink Model.
11. Double click the **NI In1** block to open its **Source Block Parameters**.
12. Change **Port dimensions** to match number of export variables (Figure 19). For this example, the number should be 6.

13. Exit back to the Simulink model and double click the **NIVeriStand Out1** block to open its **Source Block Parameters**.
14. Change **Port dimensions** to match number of import variables (Figure 19). For this example, the number should be 4.
15. In the Simulink menu, go to **Simulations–>Model Configuration Parameters...**.
16. In **Solver**, under **Solver Options**, change the **Type** to **Fixed-step** and click **Additional Option** and set the **Fixed-step size** to 0.001 second.
17. In **Code Generation** (formerly, **Real-Time Workshop)**, at **System target file**, click browse. Select **NIVeriStand.tlc** from the list of System Target Files in the window that pops up (Figure 20).

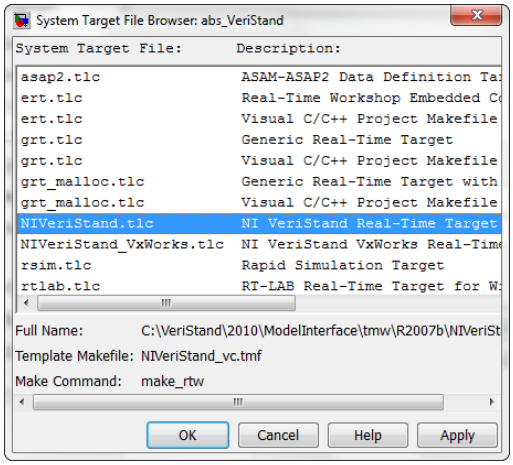

Figure 20. System Target File Browser for Simulnk model.

18. Click **Apply** and **OK** in the System Target File Browser and the Configuration Parameters window to exit to the Simulink model.
19. From the Simulink model menu, click **Code–>C/C++ Code–>Build Model** (or **Ctrl+B**) to create the various folders and files for VeriStand.

Start new NI VeriStand project

1. Start NI VeriStand.
2. Click **New NI VeriStand Project**. This will open the Create New Project window (Figure 21).
3. Enter a **Project Name**, make sure that you do not leave any spaces in the name. In this example, the project was named `ABS_VeriStand_Example`.
4. Use the browse button in the **Project Root Folder** area to find the folder where you created all the files in the last step of the previous section. For this example, the location is `CarSim_Data\Extensions\NI-RT\ABS_VeriStand`. When you find the folder in the browser, click **Current Folder**.

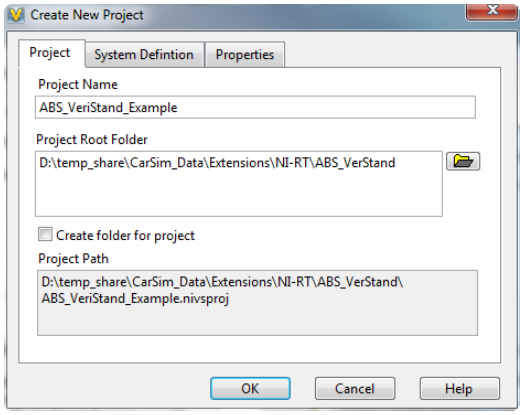

Figure 21. VeriStand New Project Creation Window.

5. Make sure the **Create folder for project** box is unchecked and click **OK**. Save and quit VeriStand.

Prepare the CarSim Dataset

1. Go back to CarSim, click the dropdown menu of **Models** and select **Models: Transfer to NI-RT Target** (Figure 22).

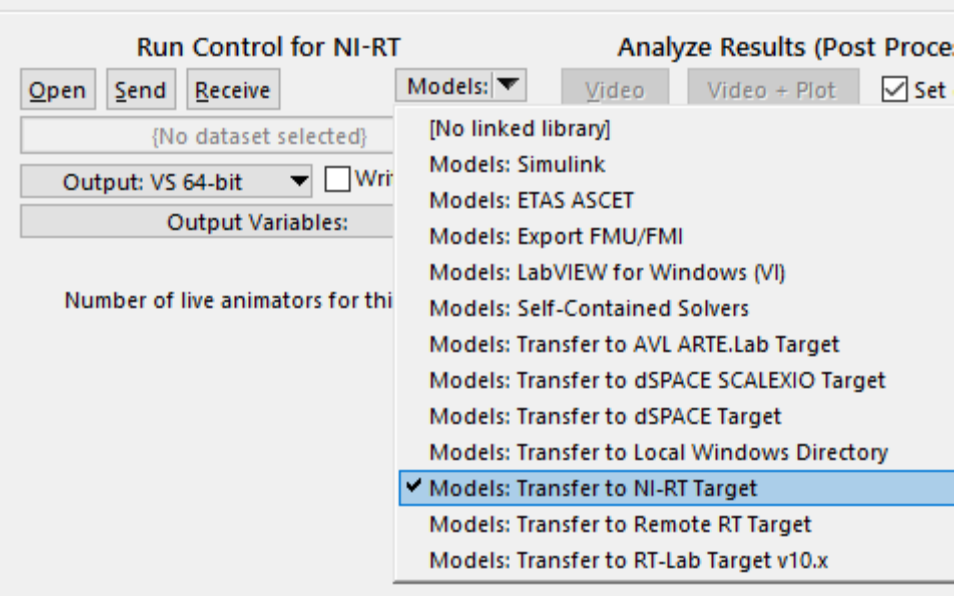

Figure 22. Set Model: Transfer to NI-RT Target.

2. Create a new model screen dataset by selecting **[Link to New Dataset]**. Enter a new dataset name and click **OK**. For this Example, the dataset was named `VeriStand: ABS Example`.

3. Click on the new blue link with the name of the dataset you just named to go to the Models dataset screen (Figure 23).

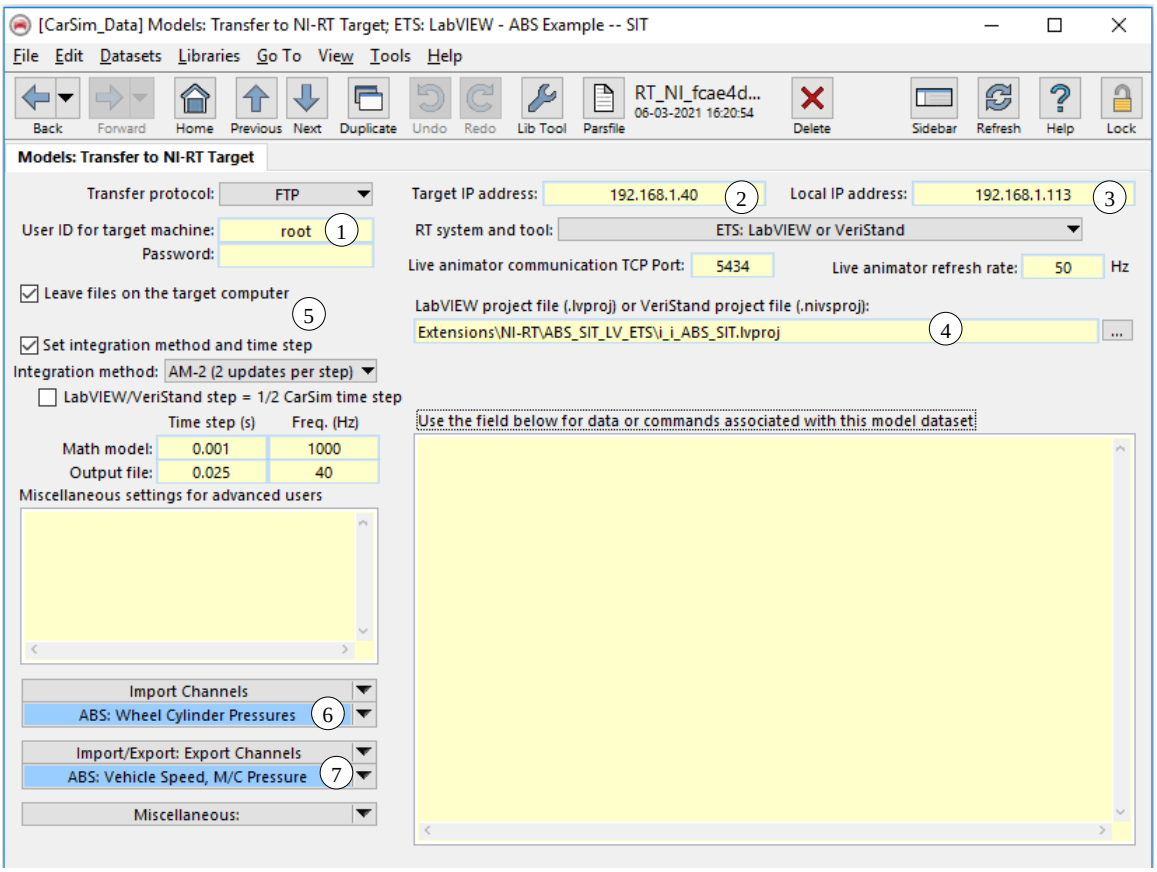

Figure 23. Dataset for NI RT ABS example.

4. Set the User ID (1) for target machine as root.
5. Fill in the IP addresses of the target (2) and host (3) machines.
6. Click the browse button (4) and find the VeriStand Project File you created in the previous section with file extension *.nivsproj.
7. Check the **Leave files on the target computer** and **Set integration method and time step** checkboxes. Select the integration method AM-2 and math model time step to 0.001 (5).
8. Set **Import Channels** as **I/O Channels–>IO Channels: Import** and link import data (6) **Brake System: Control and External Models–>ABS**.
9. Set **Export Channels** as **I/O Channels–>IO Channels: Export** and link export data (7) **Brake System: Control and External Models –>ABS: Vehicle Speed, M/C Pressure**.
10. If a VeriStand project is open, save your project and quit Veristand (Ctrl-Q). Now click the **Open** button. This should open NI VeriStand Project Explorer window (Figure 24).

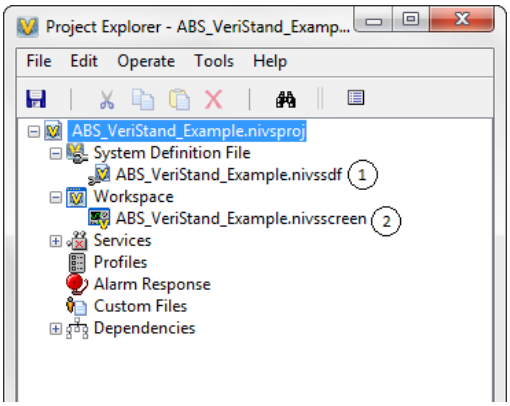

Figure 24. VeriStand project with ABS example.

Run the National Instruments RT model

1. Expand **ABS_VeriStand_Example.nivsproj–>System Definition File**. Double click the system definition file **ABS_VeriStand_Example.nivssdf** (1) (Figure 24) to open the System Explorer window (Figure 25).

2. Click on Controller to open Controller Settings window (Figure 26). Set the Operating System (1) and IP Address (2) for your target system and set the Target Rate (3) to match your CarSim setting. Save the settings.

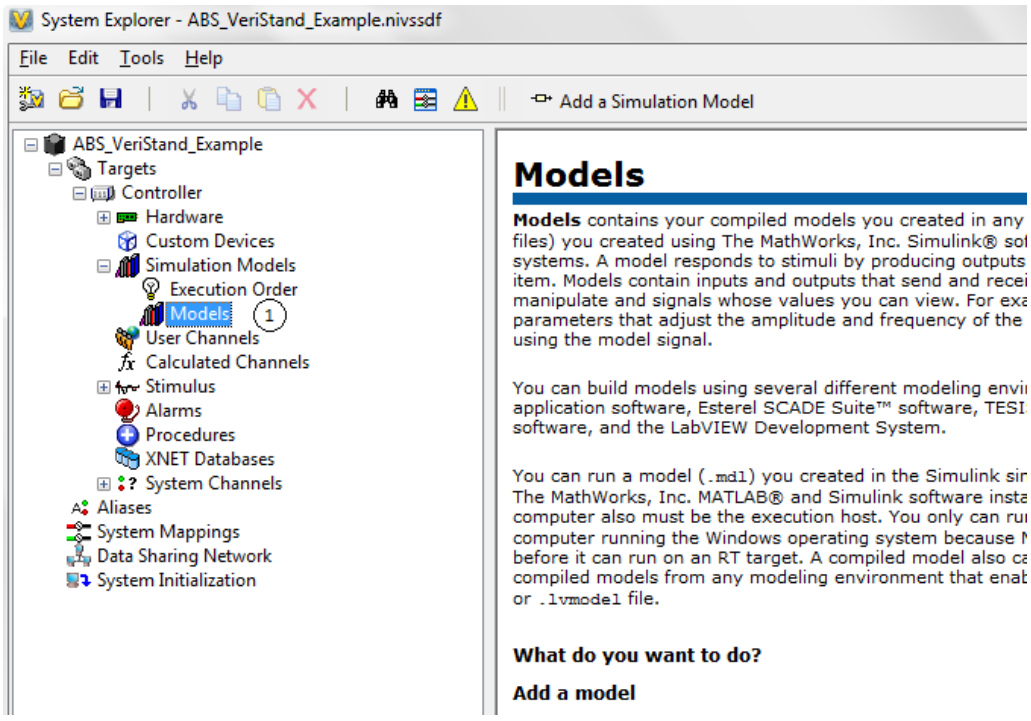

Figure 25. VeriStand System Explorer.

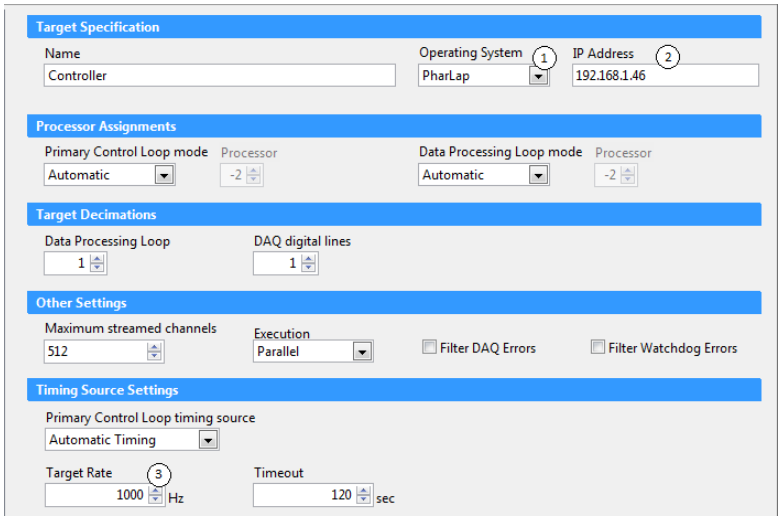

Figure 26. VeriStand Controller Settings.

3. Using File Explorer, locate the file `carsim_LVRT.dll` which should be in the folder `CarSim_Data\Extensions\NI-RT\Solver_DLLs`. Copy it into the VeriStand project root folder. In this example, it should be `CarSim_Data\Extensions\NI-RT\ABS_VeriStand`.

> **Note:** It is very important to copy the *.dll to the VeriStand project root folder. Otherwise, some necessary CarSim related files won't get written in the correct folder.

4. On the left side of the System Explorer window, expand **Controller–>Simulation Models**. Right click **Models** (1) (Figure 25) and select **Add Simulation Model**.

5. Under **Path**, click the browser button and point to `carsim_LVRT.dll, the` file that you just copied to your project folder. Select **Initial State Paused**. Repeat the same process to add another Simulation model, `abs_VeriStand.dll`. The model specification and the settings should look something like Figure 27.

6. Now, since you have two models, you need to connect them so that they can interact. From the main menu, go to **Tools Edit–>Mappings**. This will open the System Configuration Mappings window (Figure 28).

7. Expand **Simulation Models–>Models** in both the **Sources** and **Destinations** area. For this example, you must connect every Outport variable of **ABS_VeriStand** with every Inport of **carsim_LVRT**, and every Inport variable of **carsim_LVRT** with every Outport variable of **ABS_VeriStand**.

8. Match each variable to its correcting counterpart. Based on the order each variable appears in CarSim Import/Export, connect each with its corresponding number. In this example, since the import variables are ordered **PBK_L1**, **PBK_R1**, **PBK_L2**, **PBK_R2**, they link to Brake Pressure (1,1), (2,1), (3,1), (4,1), respectively (Figure 29. Also see: Figure 19).

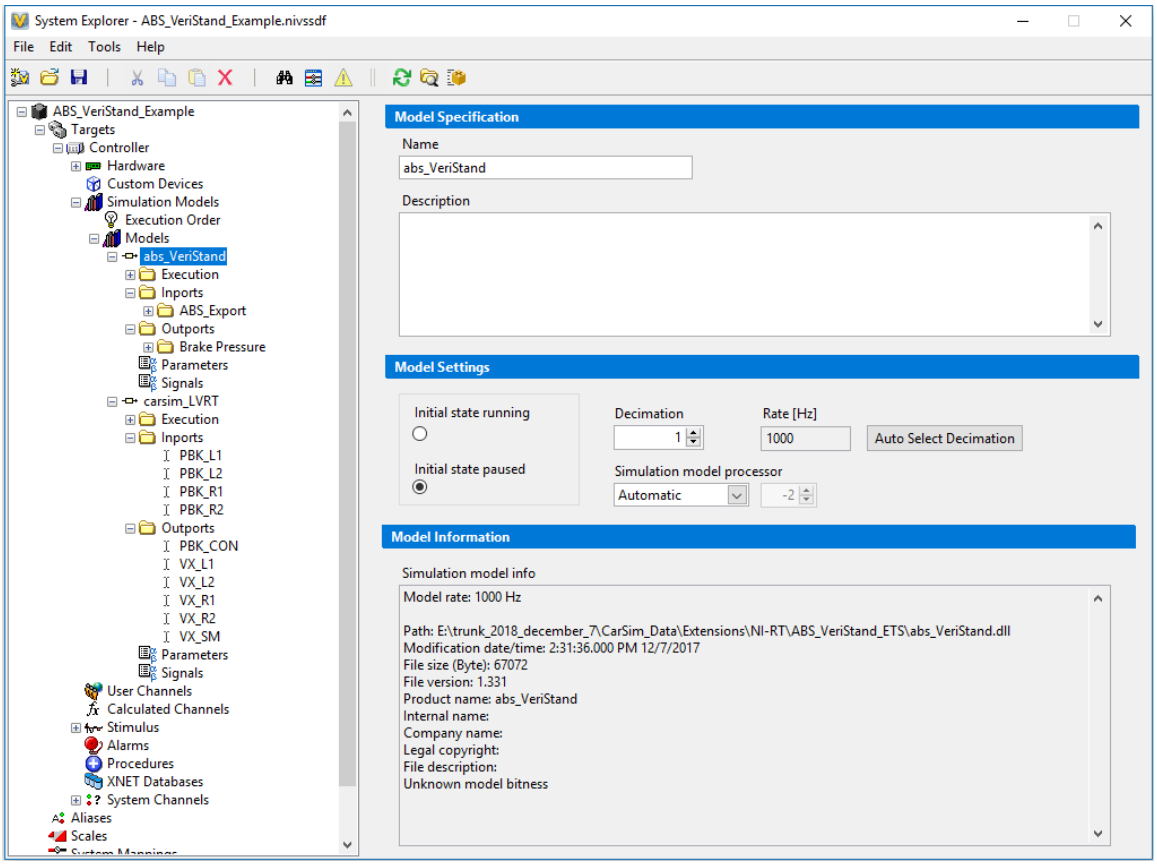

Figure 27. VeriStand after adding simulation models.

9. In the same way, since the export variables in CarSim are ordered **Vx_L1**, **Vx_R1**, **Vx_L2**, **Vx_R2**, **Vx_SM**, and **Pbk_Con**, they should be linked to ABS_Export (1,1), (2,1), (3,1), (4,1), (5,1), and (6,1), respectively. Again, refer to Figure 29.
10. To connect two variables, click one of them in the **Sources** section and the other in the **Destinations** section, and then click the **Connect** button (Figure 28).
11. Do this for every set of variables. For this example, there should be ten connections total.
12. Also, connect **carsim_LVRT>Execution>Model Command** and **abs_VeriStand>Execusion>Model Command** so you can do a model control from the Veristand User Interface (Figure 30).
13. Exit to the System Explorer (Figure 25), save and exit back to the VeriStand Project Explorer window (Figure 24).
14. Open the workspace window (click (2) in Figure 24). In this window (Figure 30), you can add some objects (e.g., speed meter) and keep track of all the variables while running.

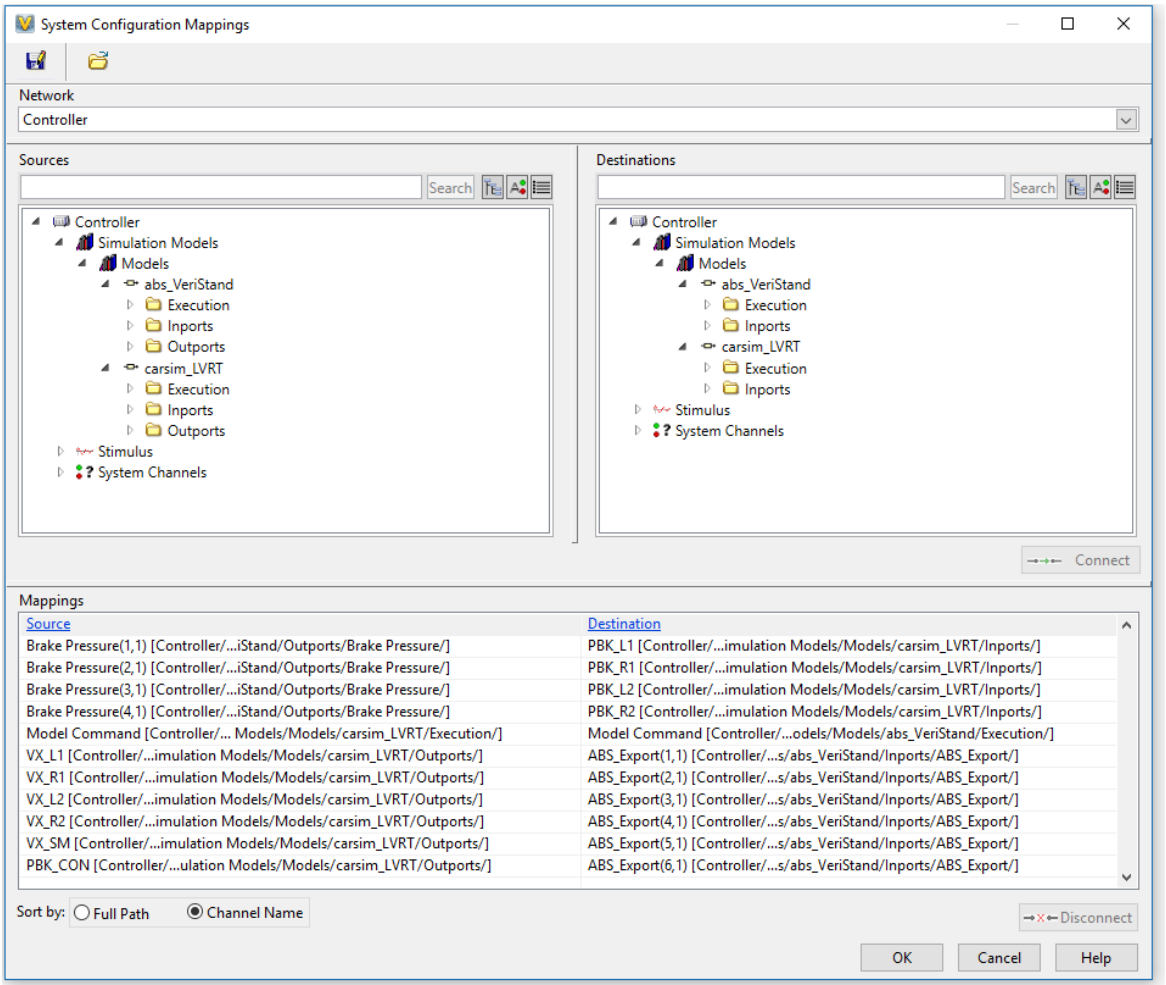

Figure 28. VeriStand System Configuration Mappings.

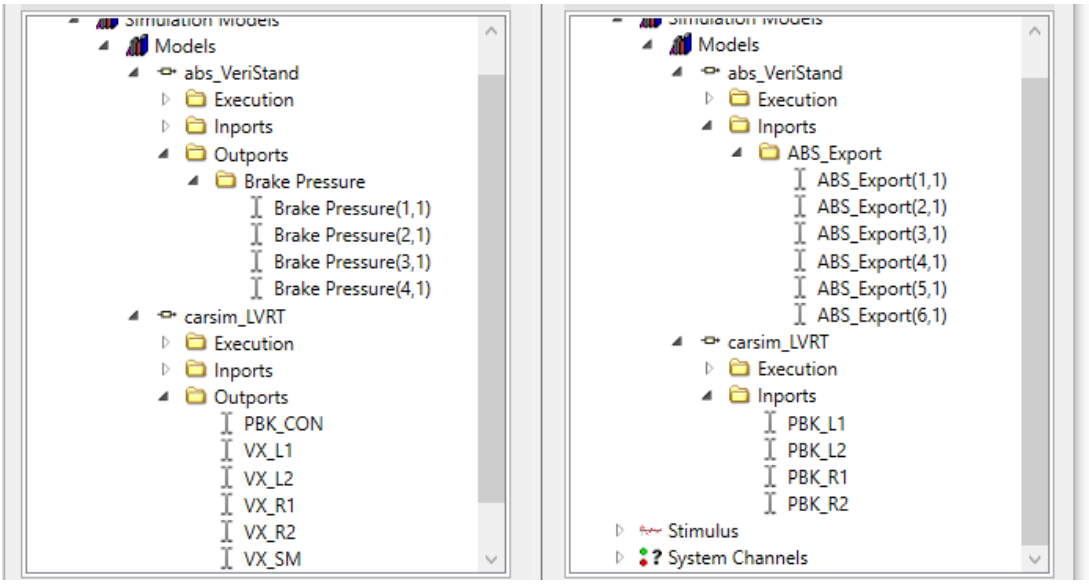

Figure 29. Connection pairs based on CarSim Import/Export.

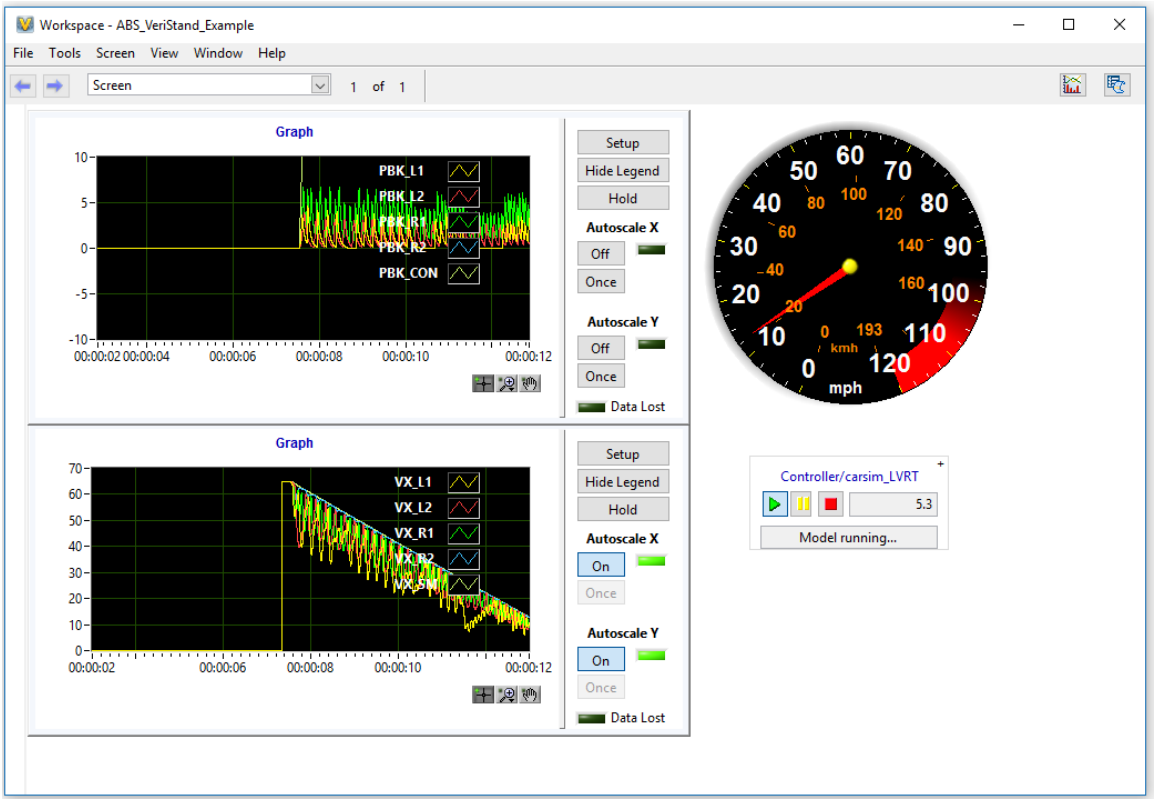

Figure 30. ABS_VeriStand_Example Running.

15. Go back to the CarSim GUI and make sure the Live Video button is 1. Press the **Send** button. This should open the Live Animation. To run in NI VeriStand, go back to the VeriStand Project Explorer (Figure 24) and click on **Operate–> Deploy**. This will automatically open the workspace and allow you to monitor the objects you added to the workspace as it runs. The workspace setup for this example is shown in Figure 30.

Run with NI VeriStand with Linux OS

The installation includes some examples that can be used to get started with National Instruments RT. However, first you must prepare the model by specifying details of your configuration.

Start new NI VeriStand project

1. Start NI VeriStand.
2. Click **New NI VeriStand Project**. This will open the Create New Project window (Figure 31).
3. Enter a **Project Name**, make sure that you do not leave any spaces in the name. In this example, the project was named `ABS_VeriStand_Example`.

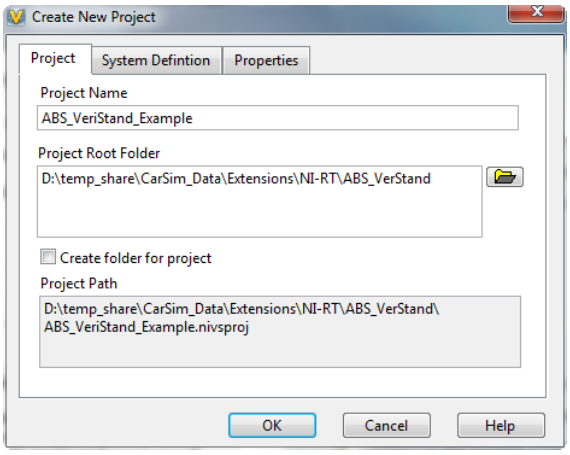

Figure 31. VeriStand New Project Creation Window.

4. Use the browse button in the **Project Root Folder** area to find the folder where you created all the files in the last step of the previous section. For this example, the location of the folder is `CarSim_Data\Extensions\NI-RT\ABS_VeriStand`. When you find the folder in the browser, click **Current Folder**.

5. Leave the **Create folder for project** box unchecked and click **OK**.

Configure CarSim to Use the C/C++ Compiler

Starting with NI RT Linux version 2019, you can transfer the CarSim model to VeriStand by using the C/C++ compiler interface or by exporting an FMU. To use the C/C++ compiler, follow the steps listed below. To export an FMU, follow the steps in the next section, *Configure CarSim to Export an FMU File.*

1. Start CarSim.

2. From the **Run Control** screen, use the **Dataset** menu and select **NI-RT** > **LinuxRT: VerisStand – ABS Example** (Figure 32). Click the dropdown menu of **Models** and confirm that **Models: Transfer to NI-RT Target** is selected.

3. Confirm that the Models blue link is set to `LinuxRT: VeriStand – ABS Example` (1) (Figure 32). Click the blue link to view the linked dataset for the NI-RT model (Figure 33).

4. Select `Linux RT: VeriStand Model Framework (GCC Compiler Required)` from the **RT System and Tool** pull-down list (1) (Figure 33). Make sure **Target ID or IP address** (2), **Local (host) IP address** (3), **VeriStand Project File (.nivsproj)** (4), and **GNU C/C++ Compiler (Eclipse)** (5) are set correctly. If you do not have the GNU C/C++ Compiler (Eclipse), you must download it from the National Instrument website.

5. Go back to the **Run Control** screen and click the **Build** button. This will generate the *.so file called `vs_nivs.so` inside the project folder. This is your model file.

6. Click the **Send** button on the **Run Control** screen.

7. Now click the **Open** button.

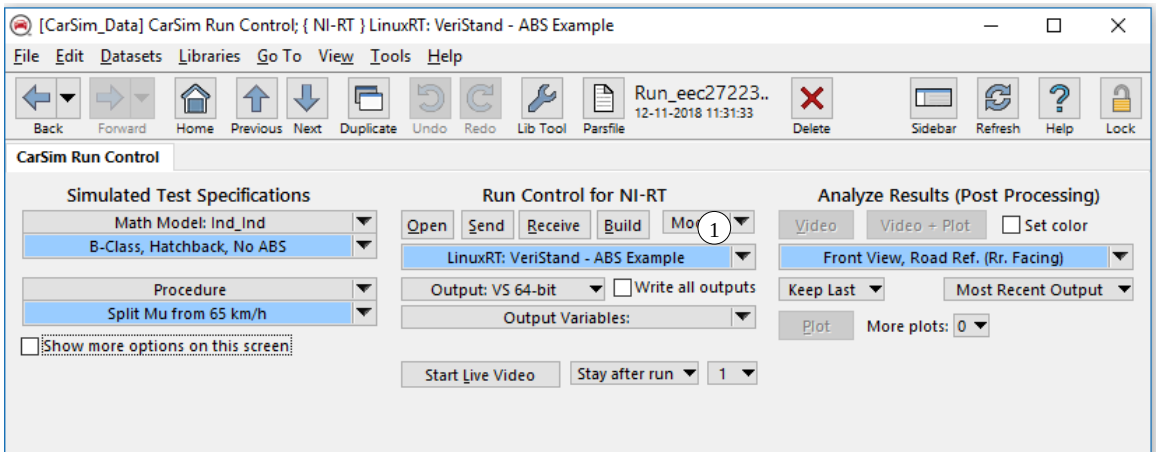

Figure 32. LinuxRT: VeriStand - ABS Example.

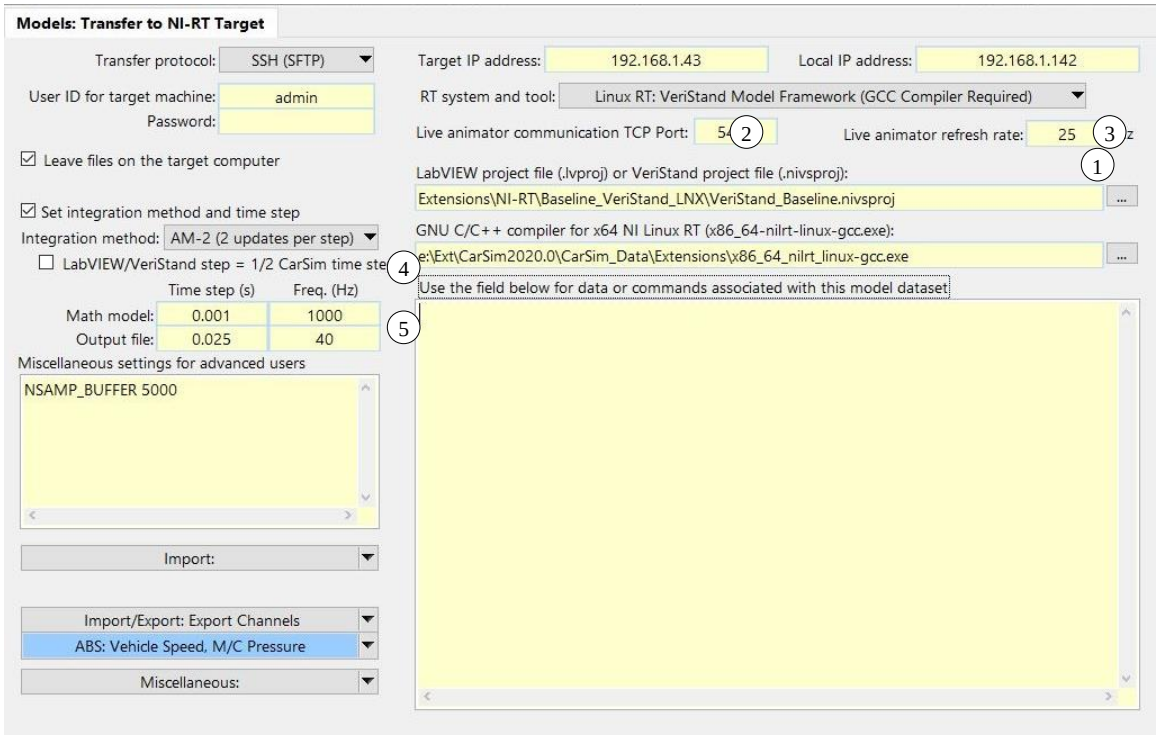

Figure 33. ABS Example Simulink Model for NI RT.

Note: To open NI VeriStand via CarSim/BikeSim, you may need to first assign NI VeriStand project file type (*.nivsproj) to be opened with NI VeriStand. For example, in Windows 7 you must find the *.nivsproj file, try to open it, and choose to **Select a program from a list of installed programs**. From there, select NI VeriStand. If it does not appear in the list of programs, you will need to browse for it. Make sure to check the box

labeled **Always use the selected program to open this kind of file**, and click **OK**. The process may vary slightly depending on the OS you are using.

Configure CarSim to Export an FMU File

If you have NI-RT VeriStand Linux version 2019, you can transfer the VehicleSim model to VeriStand by exporting an FMU file. For this example, the FMU file is already included in the project directory with the example model files. The steps are described below for simulating your own models.

1. Start CarSim.
2. From the **Run Control** screen, use the **Dataset** menu and select **NI-RT** > **LinuxRT: VerisStand – ABS Example (FMU)**. Click the dropdown menu of **Models** and confirm that **Models: Transfer to NI-RT Target** is selected.
3. Confirm that the Models blue link is set to `LinuxRT: VeriStand – ABS Example` (FMU). Click the blue link to view the linked dataset for the NI-RT FMU model (Figure 34). Select `Linux RT: VeriStand FMU/FMI` from the **RT System and Tool** pull-down list (1). Make sure **Target ID or IP address** (2), **Local (host) IP address** (3), and **VeriStand Project File (.nivsproj)** (4) are set correctly. Specify the path and filename for the FMU file (5).

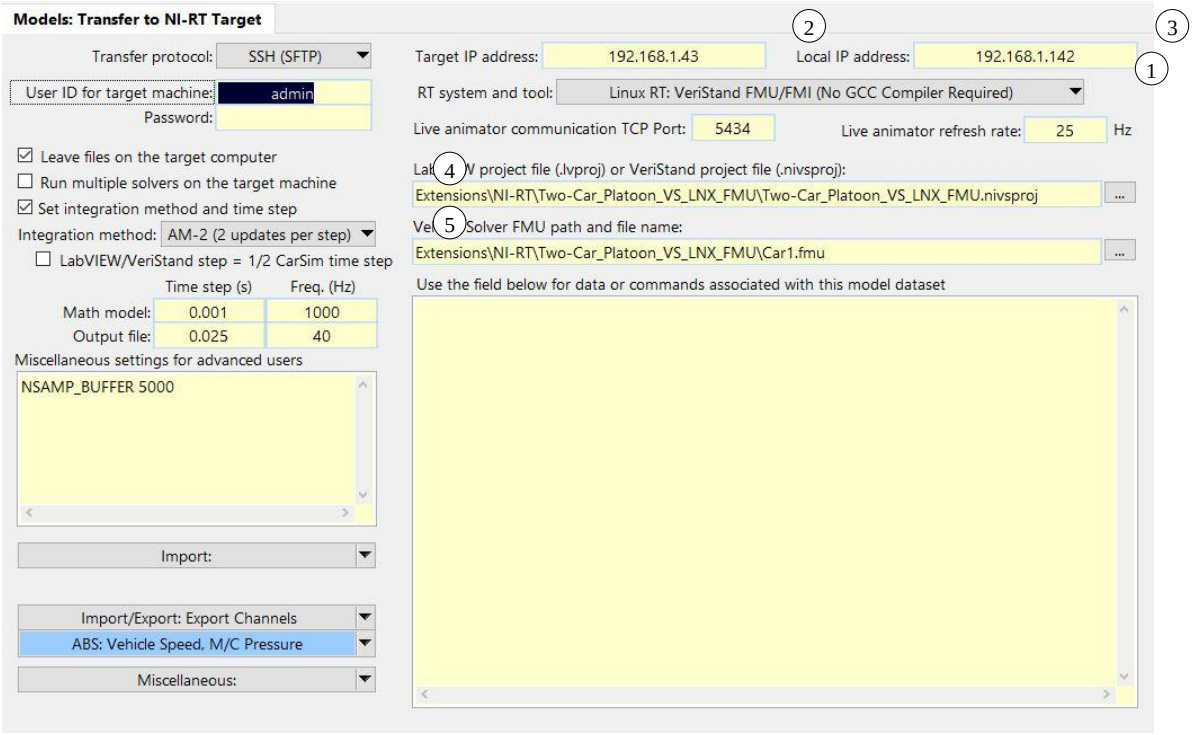

Figure 34. ABS Example FMU Model for NI RT.

4. Go back to the **Run Control** screen and click the **Generate** button. This will generate the FMU file.
5. Click the **Send** button on the **Run Control** screen.
6. Now click the **Open** button.

Run the National Instruments RT Model

1. Clicking the **Open** button in the previous step should open the VeriStand Project Explorer window (Figure 35).

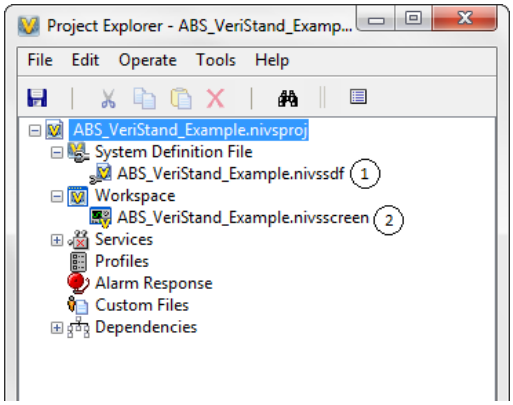

Figure 35. VeriStand project with ABS example.

2. Expand **ABS_VeriStand.nivsproj–>System Definition File**. Double click the system definition file **ABS_VeriStand_Example.nivssdf** (1) (Figure 35) to open the System Explorer (Figure 36).

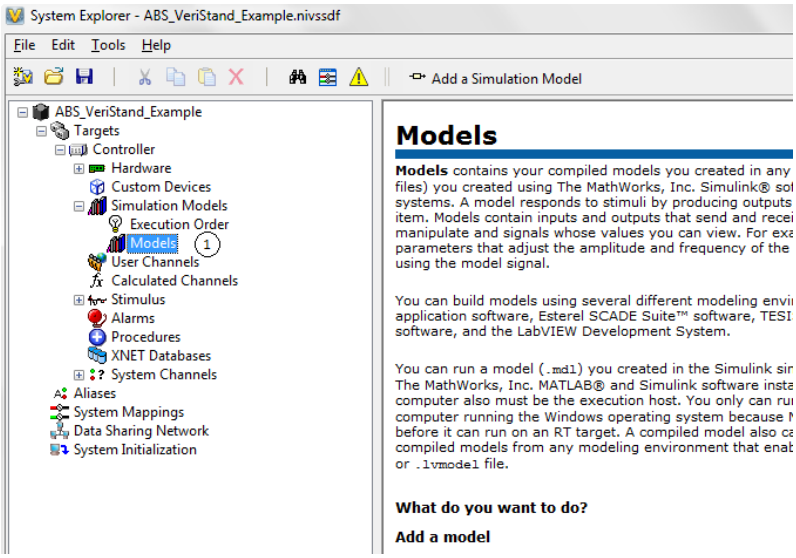

Figure 36. VeriStand System Explorer.

3. Click on the **Controller**. On the resulting screen (Figure 37), set the **Operating System** (1) and **IP Address** (2) for your target system, and set the **Target Rate** (3) to match your CarSim setting.

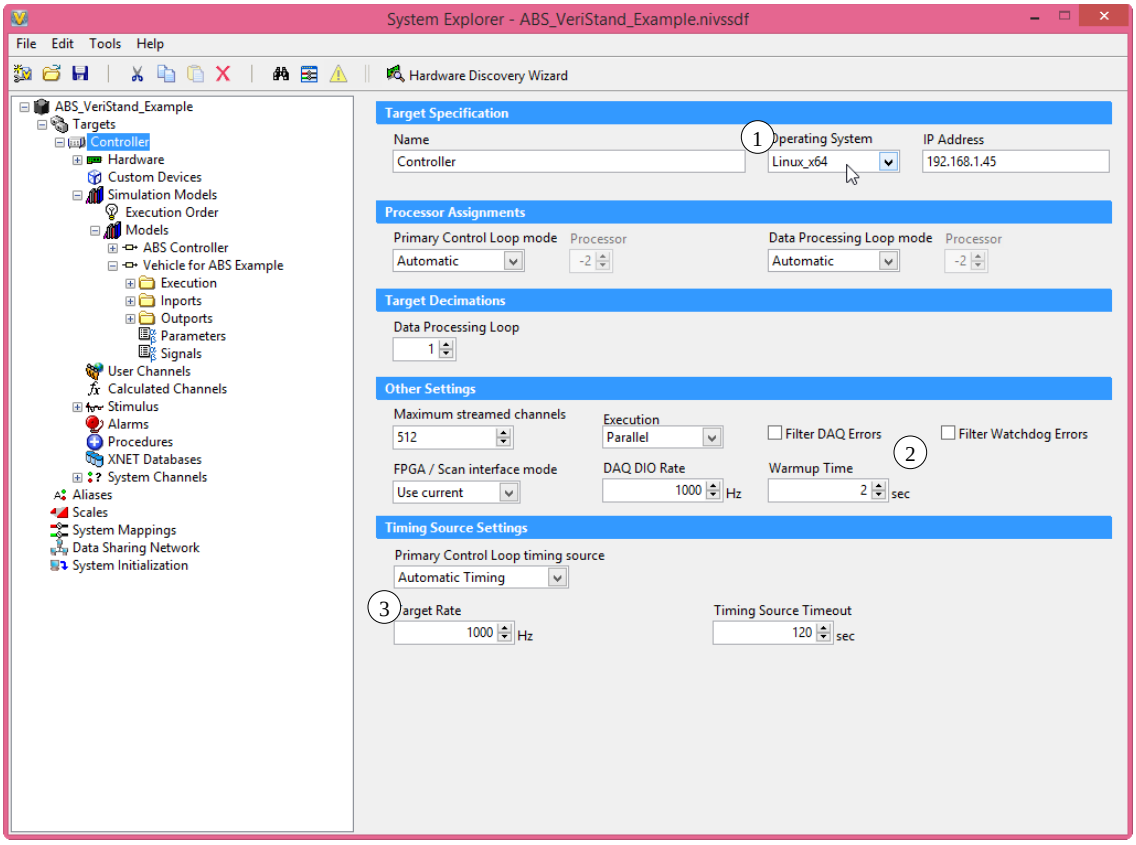

Figure 37. VeriStand Controller Settings.

4. On the left side of the System Explorer window, expand **Controller–>Simulation Models**. Right click **Models** (1) (Figure 36) and select **Add Simulation Model**.

   - If you are using the C/C++ compiler interface, then click the browser button under **Path** and locate the model file (`vs_nivs.so`) that you created in your project folder. Select it and set it to **Initial state running**. For this example, you may not need to manually point to this file as it should already be pointing to the correct file.
   - If you exported an FMU file from VehicleSim, then click the browser button under Path and locate the FMU file that you created. Select it and set it to **Initial state running**. For this example, the Veristand project may already be referencing the FMU file.

   Add the ABS controller model by clicking **Add Simulation Model** again. Click the browser button under **Path** and locate the file: `abs_VeriStand.so`. **Click OK** to see the vehicle model in VeriStand (Figure 38). This simulation model and the mapping configuration may already be included in the Veristand project for this example.

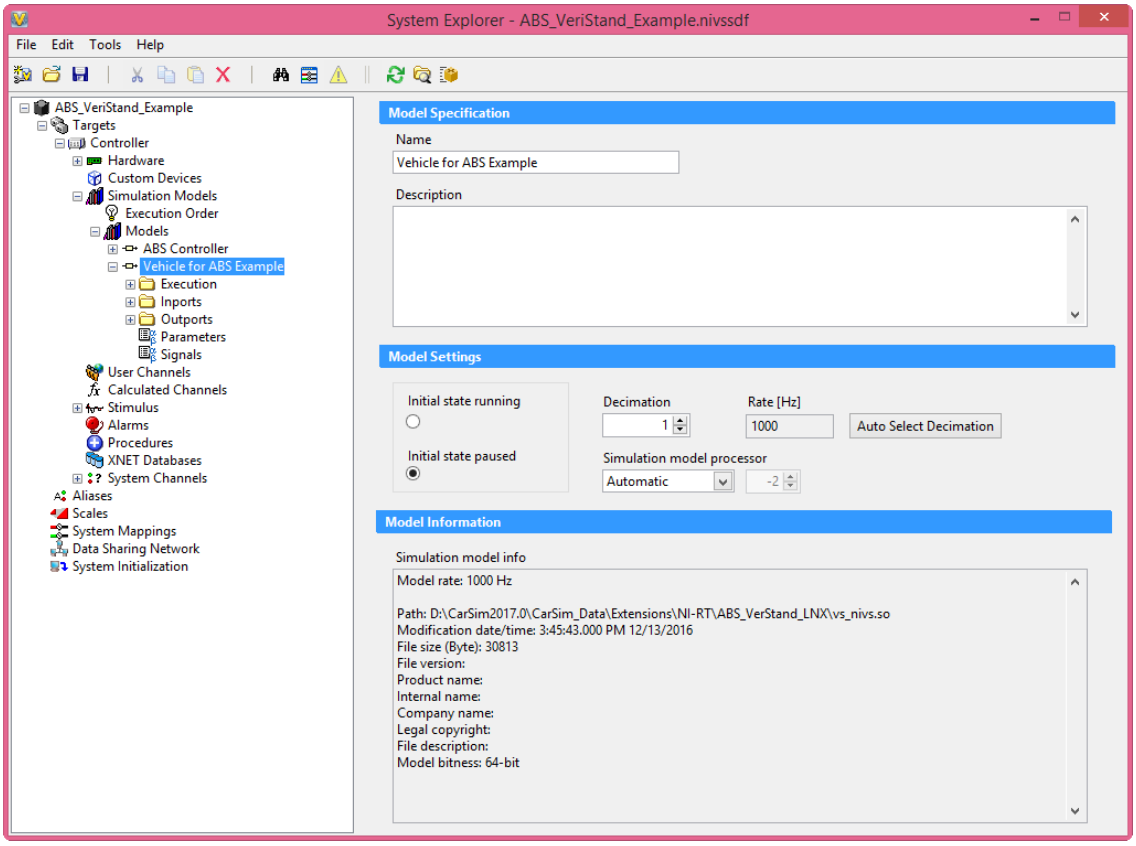

Figure 38. View of vehicle model in VeriStand.

5. Since you have two models, you need to connect them so that they can interact. From the main menu, go to **Tools Edit–>Mappings**. This will open the System Configuration Mappings window.

6. Expand **Simulation Models–>Models** in both the **Sources** and **Destinations** area. For this example, you must connect every Outport variable of **ABS_VeriStand** with every Inport of **carsim_LVRT**, and every Inport variable of **carsim_LVRT** with every Outport variable of **ABS_VeriStand**.

7. Open the workspace window by double-clicking on the workspace file in the Project Explorer (2) (Figure 35). In the workspace window, you can add objects such as a speed meter, etc. and keep track of some variables while running (Figure 39).

8. To run in NI VeriStand, click **Operate–>Run** in the Project Explorer (Figure 35). This will automatically open the workspace again and allow you to monitor the objects you added to the workspace as it runs. The workspace setup and run for this example are shown in Figure 39. Use the **Controller/ABS Controller** playback controls to start, pause, and stop the run.

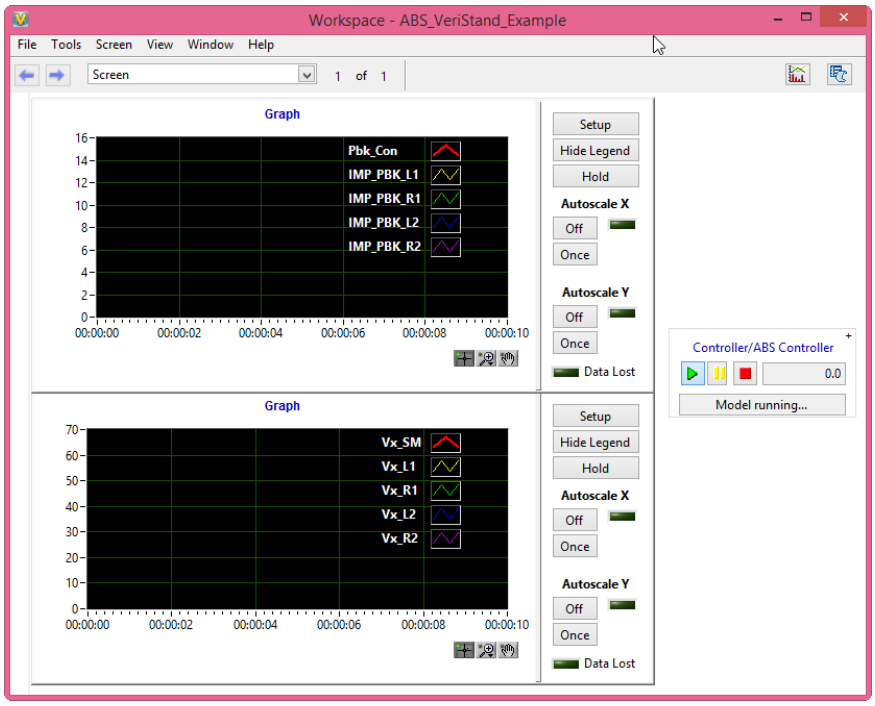

Figure 39. ABS_VeriStand_Example Running.

Running Parallel Math Models

The Parallel VehicleSim Math Models feature is available on NI Linux Real-Time Systems, versions NI 2019 and above, for VeriStand and LabVIEW-RT.

The NI-VeriStand Linux version can be run with parallel VechicleSim Math Models by exporting FMU files. This section describes how to set up a Parallel Math Model simulation in VeriStand. For more information regarding the standard setup for a Parallel Math Models example, refer to the 'Parallel VS Math Models' help document. CarSim and TruckSim include the Two-Vehicle Platoon example along with the corresponding VeriStand project.

Before you configure the CarSim model with parallel math models, you should create a VeriStand project. After you have created the VeriStand project, open CarSim and go to the Parallel VS Math Models screen by selecting **Tools > Parallel VS Math Models…**

From the Parallel Solvers screen, use the drop-down menu (1) to select **NI Linux RT** as the Run platform (Figure 40).

You can add up to 20 vehicles to the Parallel VS Math Models Screen. The number of vehicle solvers can be modified by clicking on the drop-down menu at the top of the screen (2).

Each individual vehicle model included in the Parallel Math Model screen will need a unique `Models: Transfer to NI-RT Target` dataset. To set up an individual model, click on a vehicle solver blue link (3) to open the Run Control screen for that vehicle model.

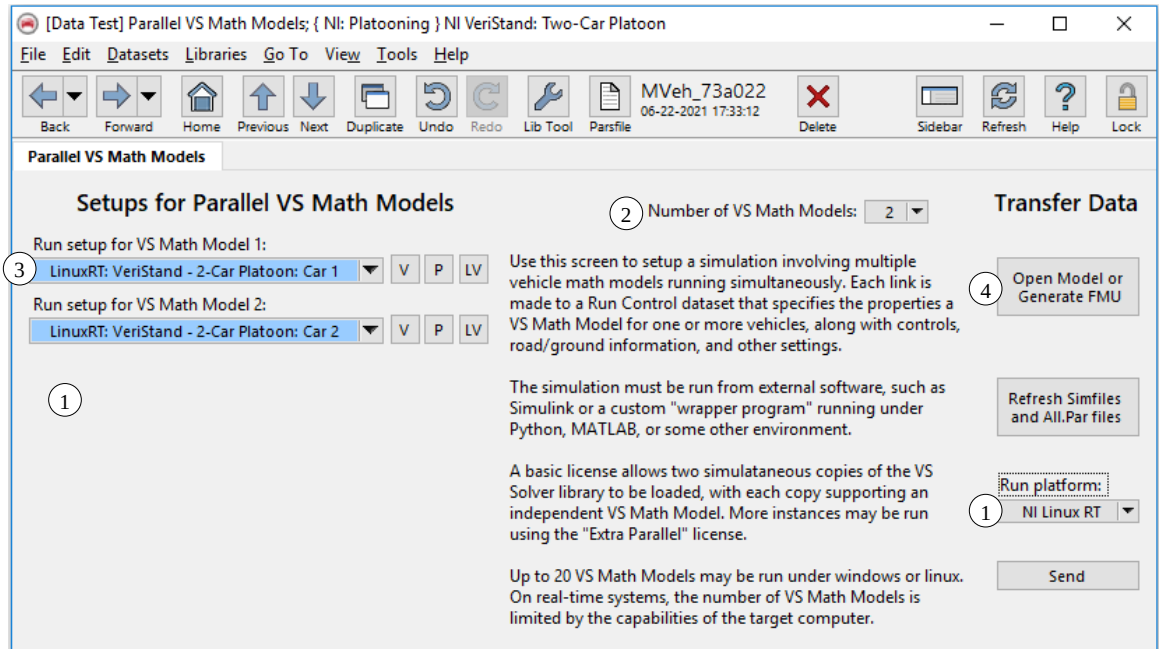

Figure 40. NI Linux RT Parallel Solvers Screen Setup.

On the Run Control screen, click on the model blue link to open the model dataset (Figure 41). For each vehicle model, specify a unique **TCP Port** (1) and **FMU file** (2). The FMU files should all be in the VeriStand project directory. In each model dataset, toggle on the **Run parallel VS Math Models** option (3).

After the vehicle math models have been defined, go back to the Parallel VS Math Models screen and click **Open Model or Generate FMU** (4) (Figure 40).

In VeriStand, create simulation models for each FMU file by importing the files. Use the Mapping tool to connect the models.

In CarSim, click **Send** on the Parallel Solvers screen. Click LV for each vehicle to open Live Video.

Follow the steps in the section, *Run the National Instruments RT Model*, to simulate the model.

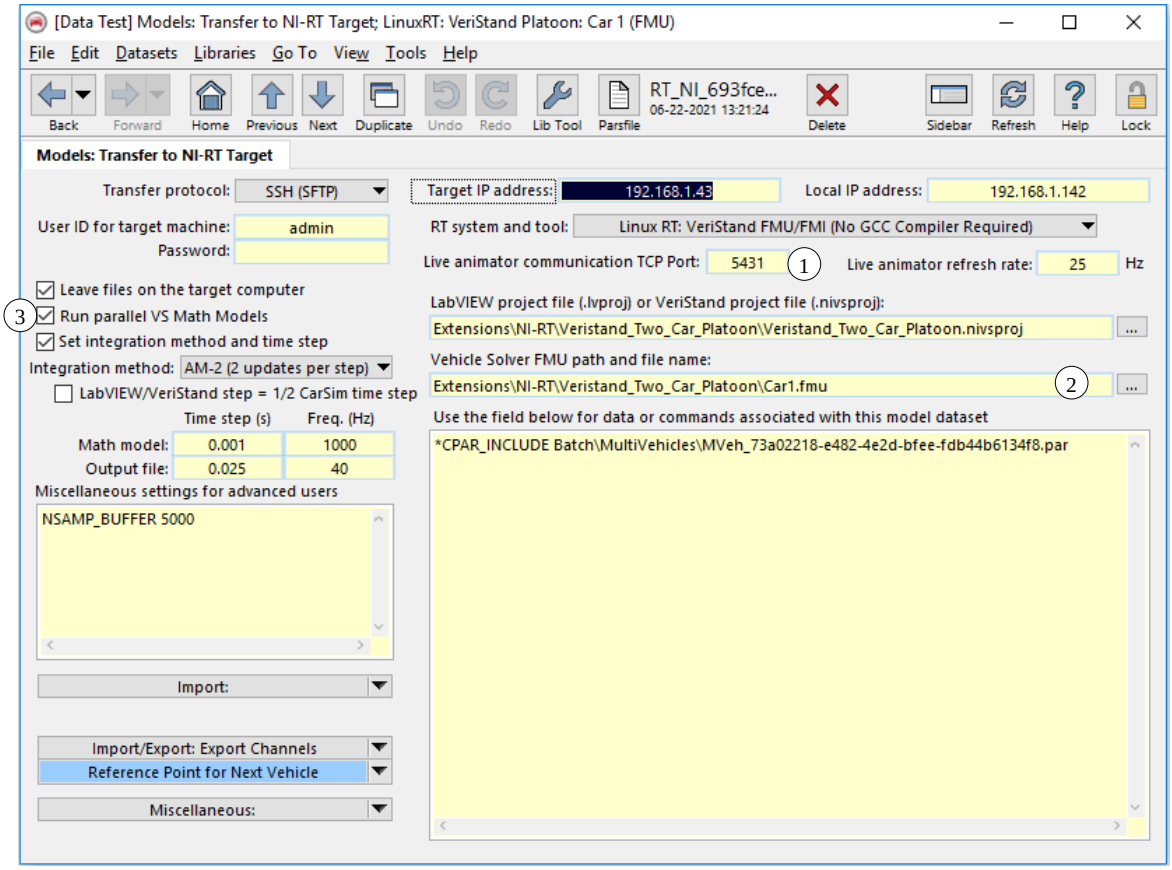

Figure 41. Model Dataset Setup for Parallel VS Math Models.