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

ETAS RTPC Guide for VehicleSim Products

This memo provides step-by-step instructions for (1) Installing Real-Time VS Solvers into LabCar RT Target Computer and (2) Creating and running real-time applications using VehicleSim Products. The examples shown were made with CarSim RT 2018.0 and LabCar RT v. 5.4, but the same controls and methods are also used for BikeSim RT and TruckSim RT. Two methods are shown:

1. Running with a Simulink model in LabCar.
2. Running with an FMU/FMI module in LabCar.

This document assumes that you already have basic familiarity with both CarSim and the LabCar Operator program. Depending on your application, familiarity with MATLAB/Simulink and/or FMU/FMI.

Before you Begin

1. The Windows core version of CarSim must be installed on your computer.
2. You also need a CarSim solver license for ETAS LabCar RT.

System Requirements

We support LabCar 5 and newer; we have tested release 5.31, 5.40, 5.4.2, and 5.4.4.

CarSim	TruckSim	BikeSim
2.0 GHz Dual Core	2.4 GHz Dual Core	2.0 GHz Dual Core

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

Installing VS Solvers onto ETAS RT Target Computer

To install the ETAS-compatible VS math models, follow these steps:

1. Launch CarSim using any installed database. Make sure to put a checkmark next to "CarSim solver for ETAS LabCarRT" in the License Settings window.
2. Select the **Tools** menu command **Install RT Solvers to Target …**, (Figure 1). This opens a solver installation window (Figure 2).

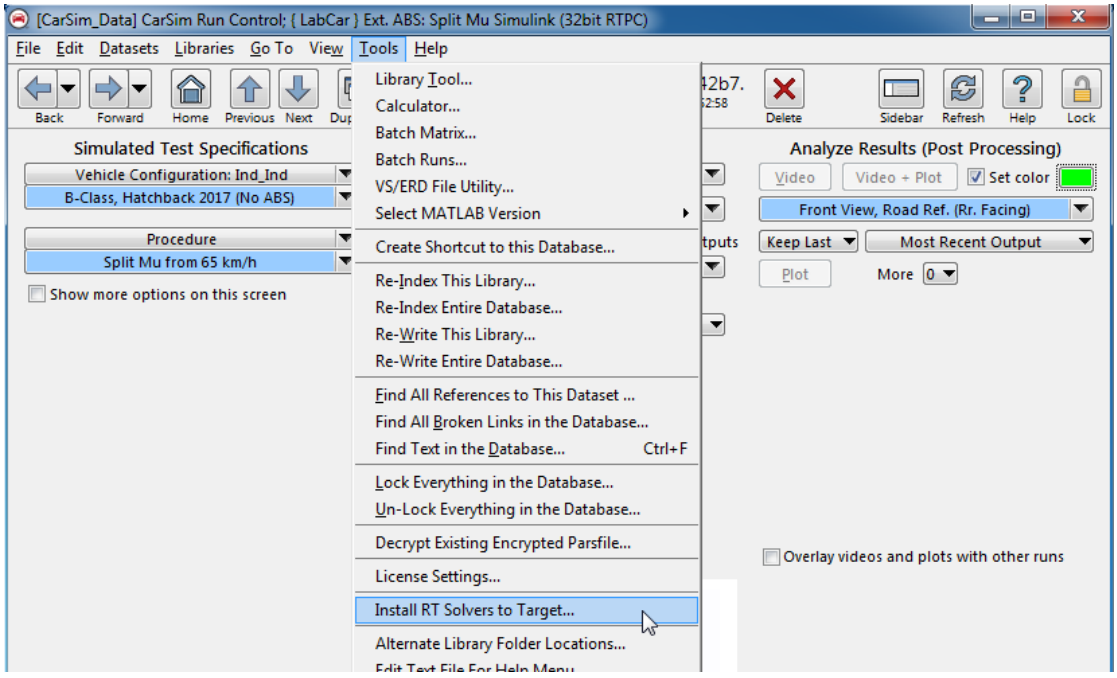


Figure 1. Tools menu to install VS RT solvers.

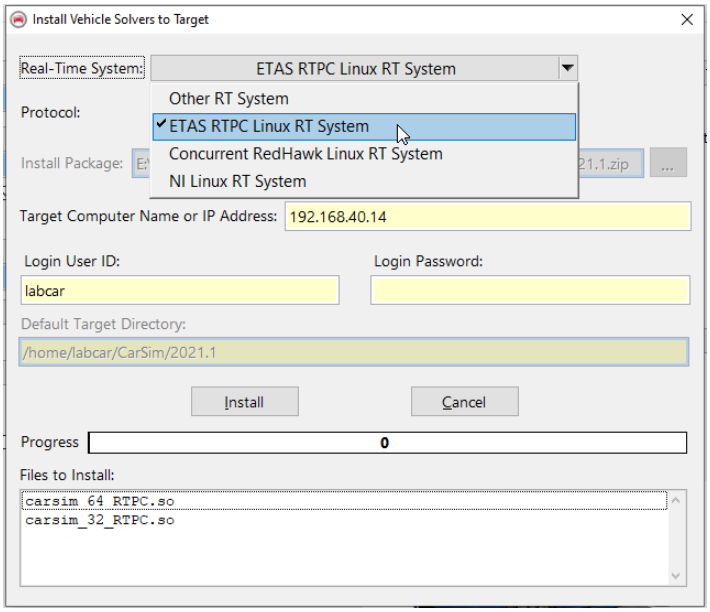


Figure 2. Select a target RT System to install Vehicle Solvers.

3. Select **ETAS RTPC Linux RT System** from the **Real-Time System** pull down menu. The solver installation package location is automatically detected (Figure 3).

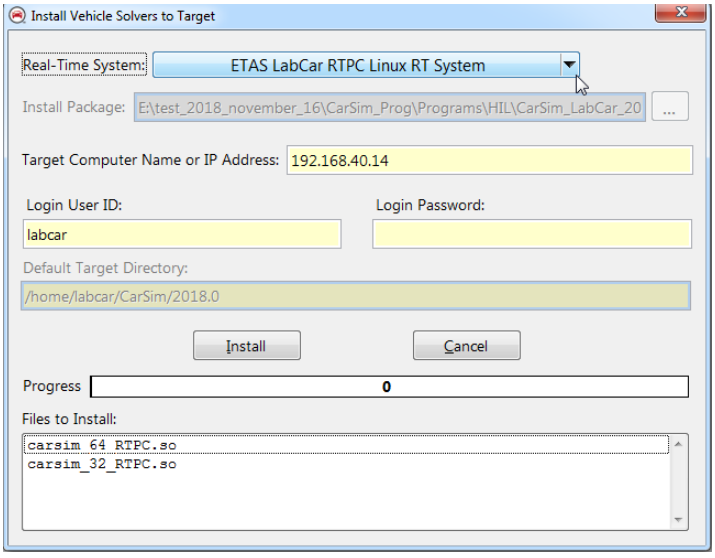


Figure 3. Install VS RT solvers to LabCar RT system target.

4. If your target computer settings are different from the display on this window, please modify the yellow fields, like target RT computer IP address, login user ID and password.

5. Click the **Install** button.

Importing Examples Datasets

> **Note** In versions earlier than 2018.0, BikeSim Windows version already contains the example datasets. Starting with version 2018.0, CPAR file is included in the `Resources` folder. Follow the instruction this section for the installation.

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 ETAS RT projects, and supporting datasets.

To build a new database, launch CarSim and select **Open Database Builder…** from the Select Recent Database window. In the Database Builder window (Figure 4), toggle on **Show all CPAR archives** (1). Select RT_ETAS.cpar (2) along with any other data groups that you would like to include in the new database. Click **Build Database from Selected Items** (3). Select the folder where you would like to save the database and click **OK**.

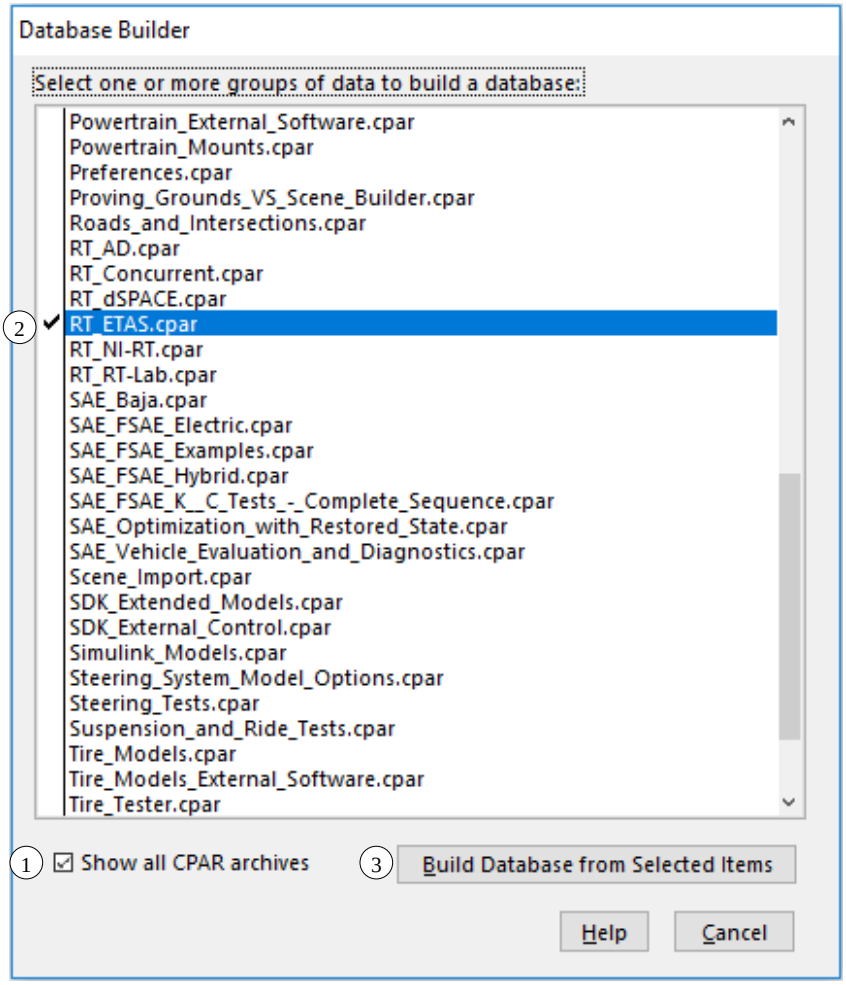


Figure 4. Database Builder

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

`CarSim_Prog\Resources\CPAR_Archives\RT_ETAS.cpar`

Use the default settings (Figure 5) to import the file. The VS browser will copy files into the current database, import some dataset files, and refresh.

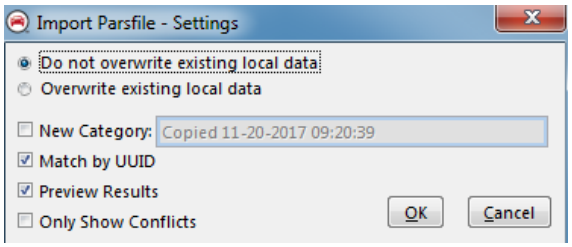


Figure 5. Import Parsfile Default Settings.

Your database now includes all the datasets that are used for the examples described in this tutorial. The examples can be found in **Datasets > RT: ETAS** (Figure 6).

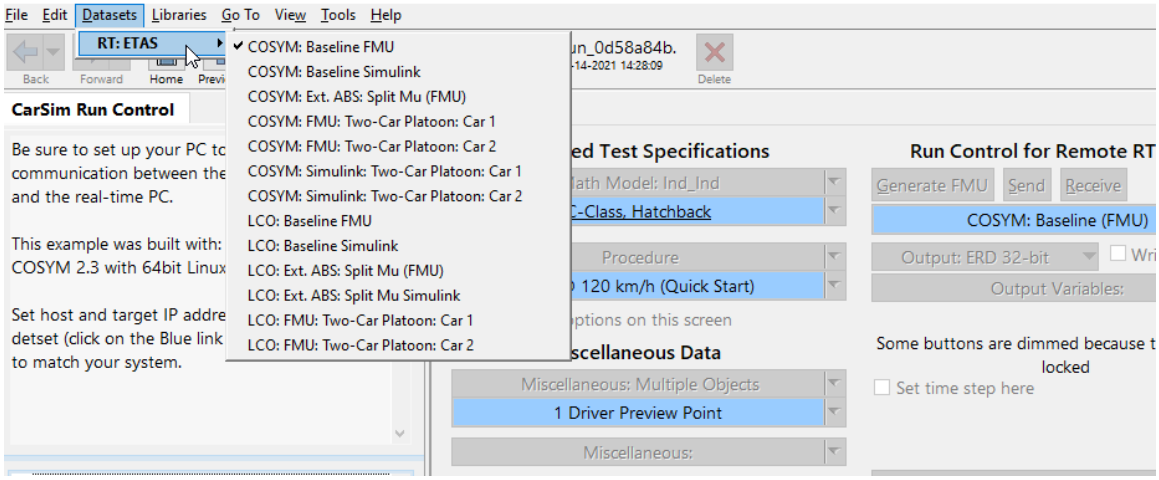


Figure 6. ETAS Examples.

LabCar Tutorial: Create & Run a Simulink RT Project

You can create a real-time simulation or test project with an existing MATLAB/Simulink model. This section describes how to create a new LabCar project from scratch using an existing CarSim ABS model example. For this tutorial, you will be monitoring system processing usage using LabCar.

Before starting this tutorial, follow the instructions of "Installing VS Solvers into LabCar RT Target Computer" and "Importing Examples Datasets." We will be using a Simulink model which is included in `RT_ETAS.cpar`

Set Up a LabCar Project

Start by creating a new LabCar project.

The Simulink model that we will be using for this example is located in: `{CarSim Data folder}\ETAS_Examples\Simulink_Models\ABS\`. The name of the Simulink model is abs_MP_CS9.mdl. If you do not see this directory, be sure to follow "Importing Examples Datasets" instruction.

1. Launch LabCar Operator. From the menu, select **File –> New Project** (Figure 7) to make the new project.

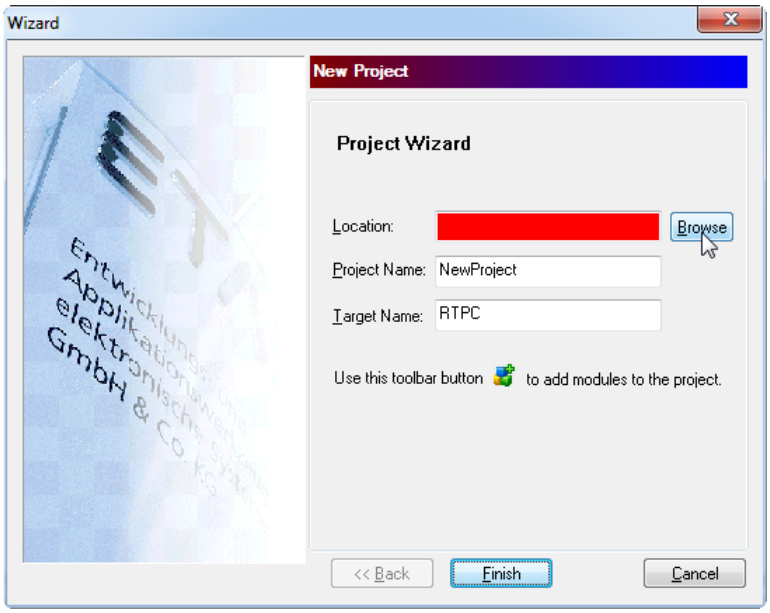


Figure 7. Create a new LabCar project.

2. Select a project location. For this example, `{CarSim Data folder}\ETAS_Examples\` is selected as a project location. Type a project name, e.g. ABS_Example (Figure 8). Click **Finish** and open the LabCar-IP main window, Figure 9.

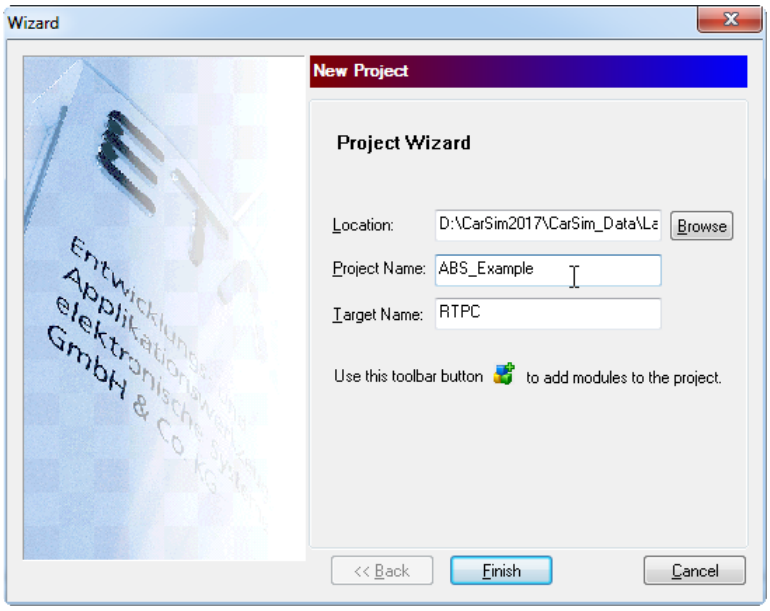


Figure 8. Select project location and name.

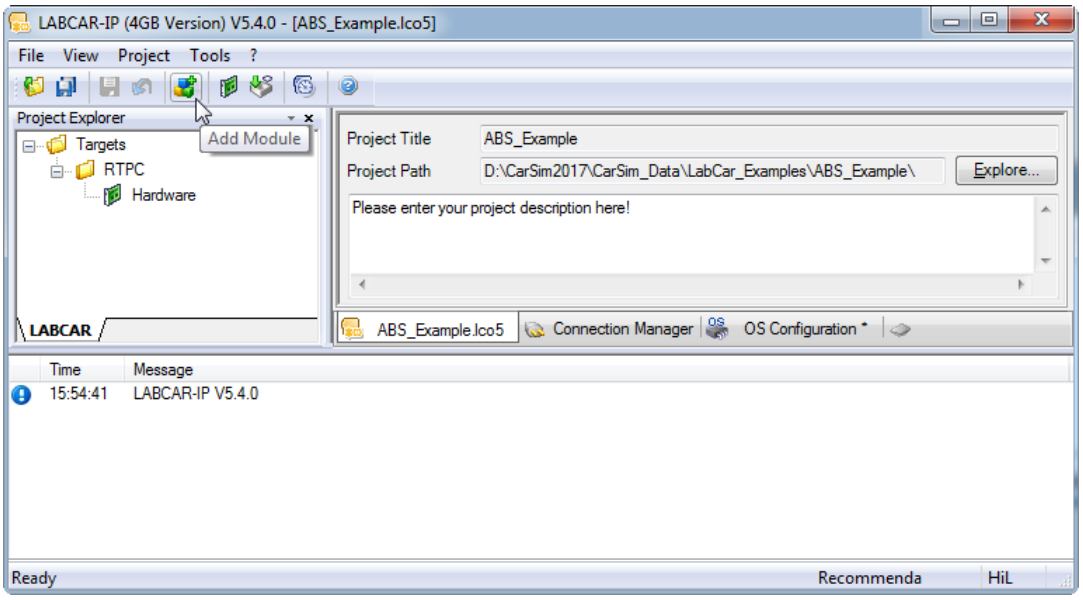


Figure 9. LabCar-IP project main window

3. Click **Add Module** icon (Figure 9). The Add Module Wizard pops up (Figure 10). Select **Add Simulink (TM) Module** and click **Next**. Select **Use existing Simulink model** (Figure 11). The ABS Simulink model is in this folder:

   `{CarSim Data folder}\ETAS_Examples\Simulink_Models\ABS`

   This should be the ABS control model that is used for several examples in CarSim.

   Put a check mark next to **Copy Model Directory into Project**.

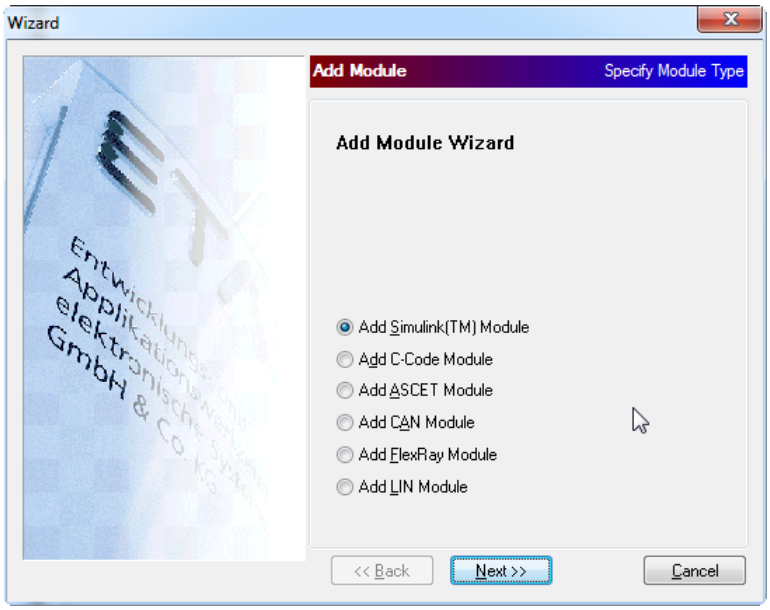


Figure 10. Select add Simulink module.

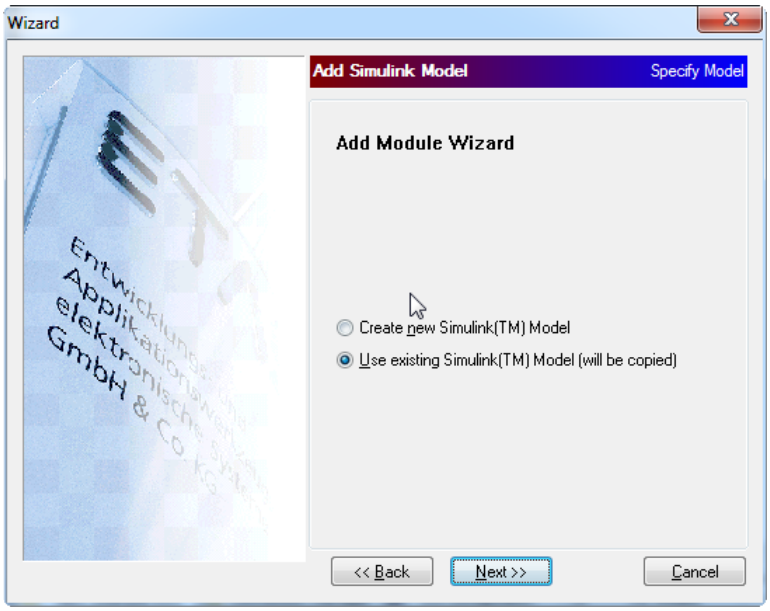


Figure 11. Select use existing Simulink Model.

4. When you click the **Finish** button, LabCar Operator adds a model into the LabCar project and automatically copies all files in the Simulink folder to the folder of the project. At this point, Simulink opens and displays the model you selected (Figure 12).

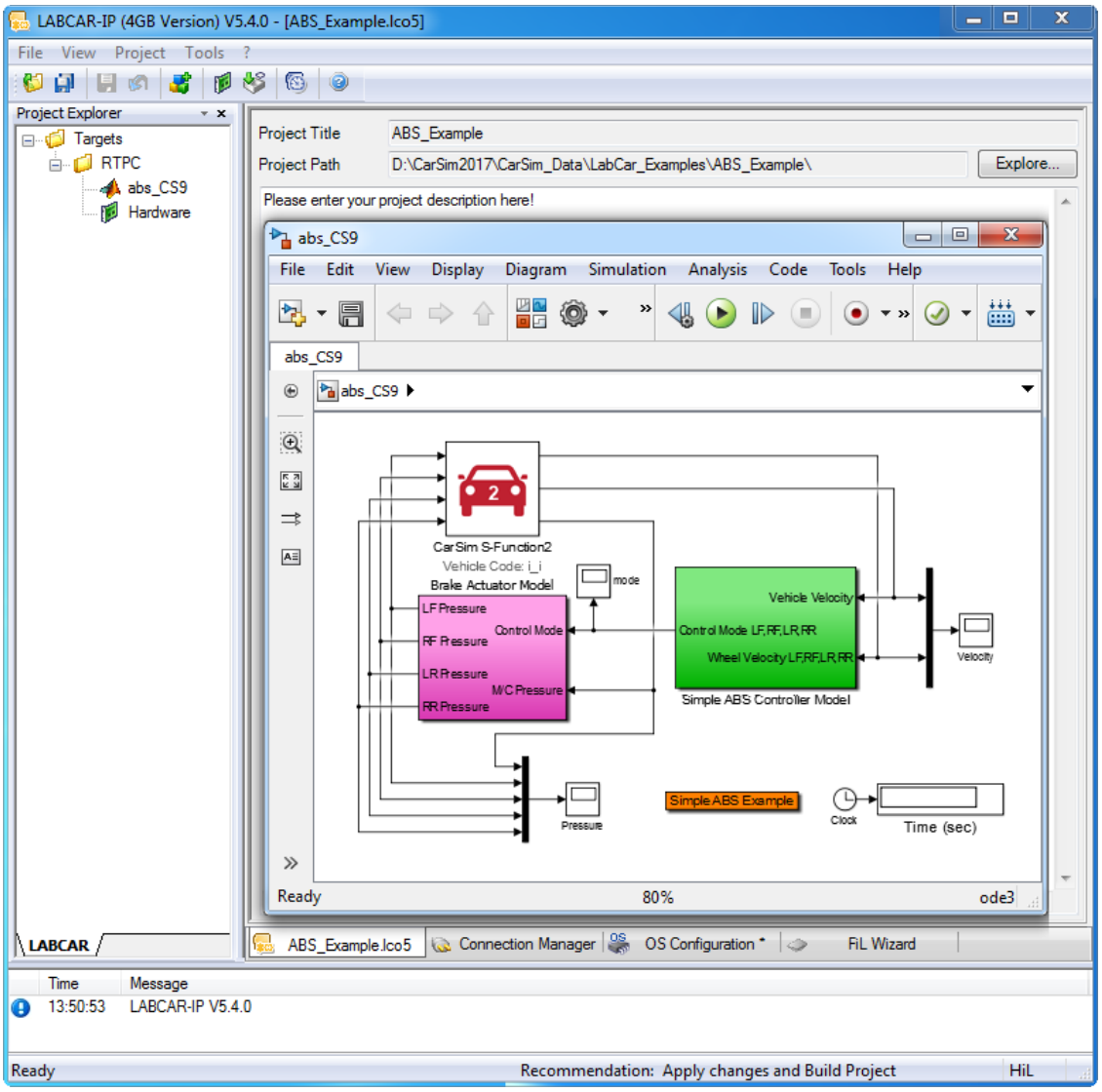


Figure 12. Project of Simulink ABS.

5. [Other Option: Copy several supporting files from the LCO_SL_ABS folder in the ETAS_Examples folder in the CarSim database to the LabCar project folder.

   The old style S-Function "vs_sf" that allows only 1 input port and 1 output port needs files car_simulink.bmp, vs_sf.c, vs_sf.mexw32 (for 32bit Matlab), and vs_sf.mexw64 (for 64bit Matlab). The S-Function2 "vs_sf2" that allows multiple ports for input and output needs the following files: vs_sf2.png, vs_sf2.c, vs_sf2.mexw32 (for 32bit Matlab) or vs_sf2.mexw64 (for 64bit Matlab). These files should be copied to the location of the Simulink model within the LabCar project. Specifically, the location should be something like

   LabCar_Folder\Target_RTPC\SimulinkModels\Project_name.]

Prepare CarSim Datasets

1. Launch CarSim and select **Datasets** > **Simulink Models** > **Ex. ABS: Split Mu** example. Press the **Duplicate** icon. When the Duplicate Dataset window pops up, enter `RT: ETAS` in **Category for new dataset**. You can leave the title as `Ex.ABS: Split Mu #1`. When running Simulink models under Windows, the connection to Simulink is handled with a dataset from the **Models: Simulink** library. That dataset specifies the Simulink MDL file, plus datasets for import and export variables. To run the same Simulink model with LabCar, the same information is needed, along with transfer information used by LabCar.
2. Use the **Models** drop-down list to select **Models: Transfer to Remote RT** (1) **(**Figure 13**)**.
3. Use the dataset select control under the **Models** control (1) (Figure 14) and choose the option **Link to New Dataset**. Give the new dataset a name and category, such as `LabCar Target` as a category and `ABS Controller` as a title.
4. Click the blue button for the new link to go to the newly created dataset (Figure 15).

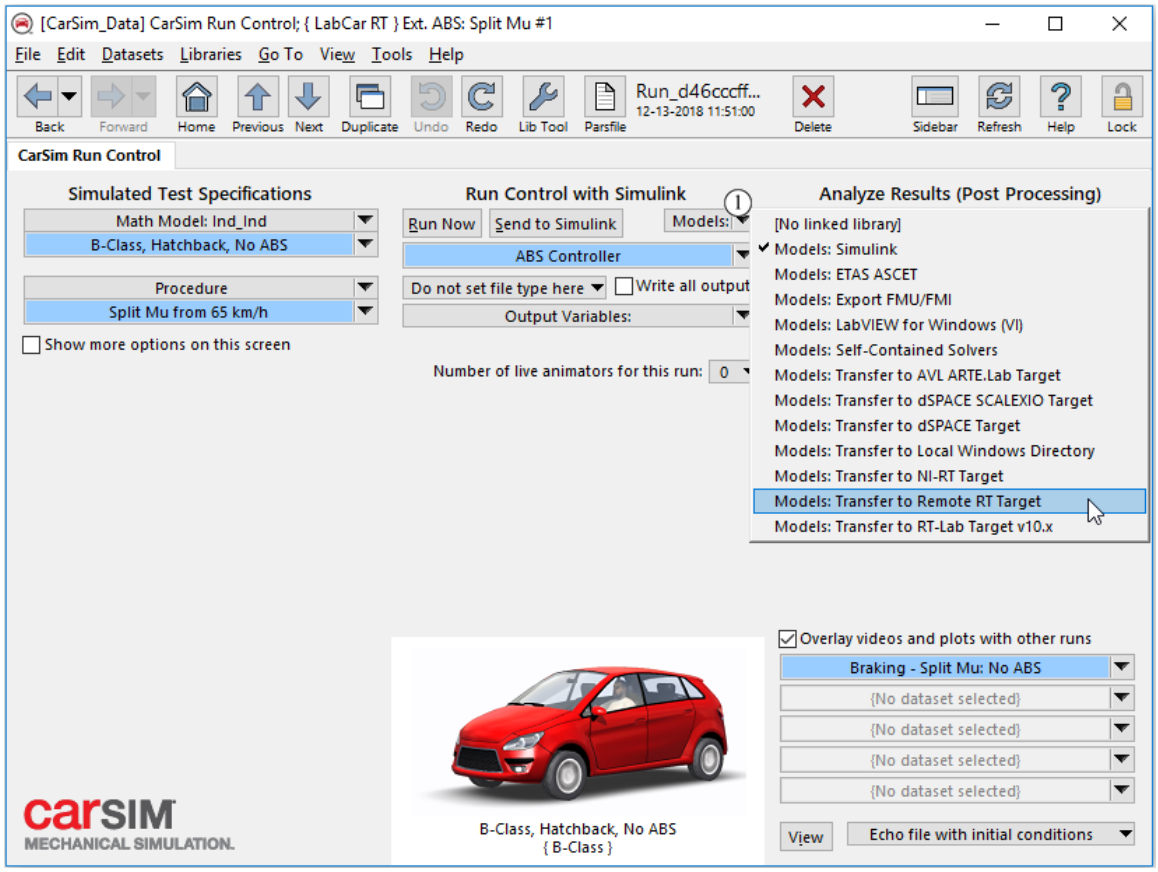


Figure 13. CarSim Run Control linked to ABS Model: change the linked Models library.

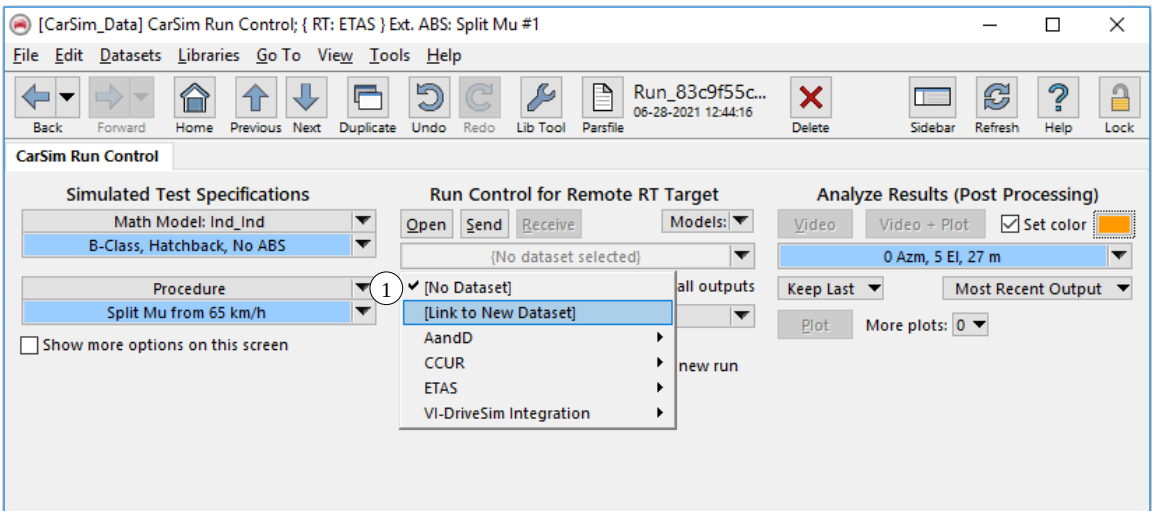


Figure 14. Make a new dataset in the Models: Transfer to Remote RT Target.

5. As shown in Figure 15, the RTPC default login ID is `labcar` and the password is empty. Put target (1) and host (2) IP addresses for file transfer between host and target on your specific network to enable live animation. The directory on the target is `/home/labcar/var` (3). Specify a Simulink model file (4), and make blue links for import and export datasets (5). The Simulink model should be in the place where you created a LabCar project (in Step 2 of Set Up a LabCar Project). In this example, this is: `{CarSimDatafolder}\ETAS_Examples\ABS_Example\Target_RTPC\SimulinkModels\Simulink_abs_MP_CS9\abs_MP_CS9.mdl`

   In the lower left corner, select **I/O Channels** > **I/O Channels: Ports**, and then select **Braking Controls** > **ABS Example – Multi Port** for a blue link.

6. Click this **ABS Example -Multi Port** blue link and you will see import and export channels setup as shown in Figure 16.

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

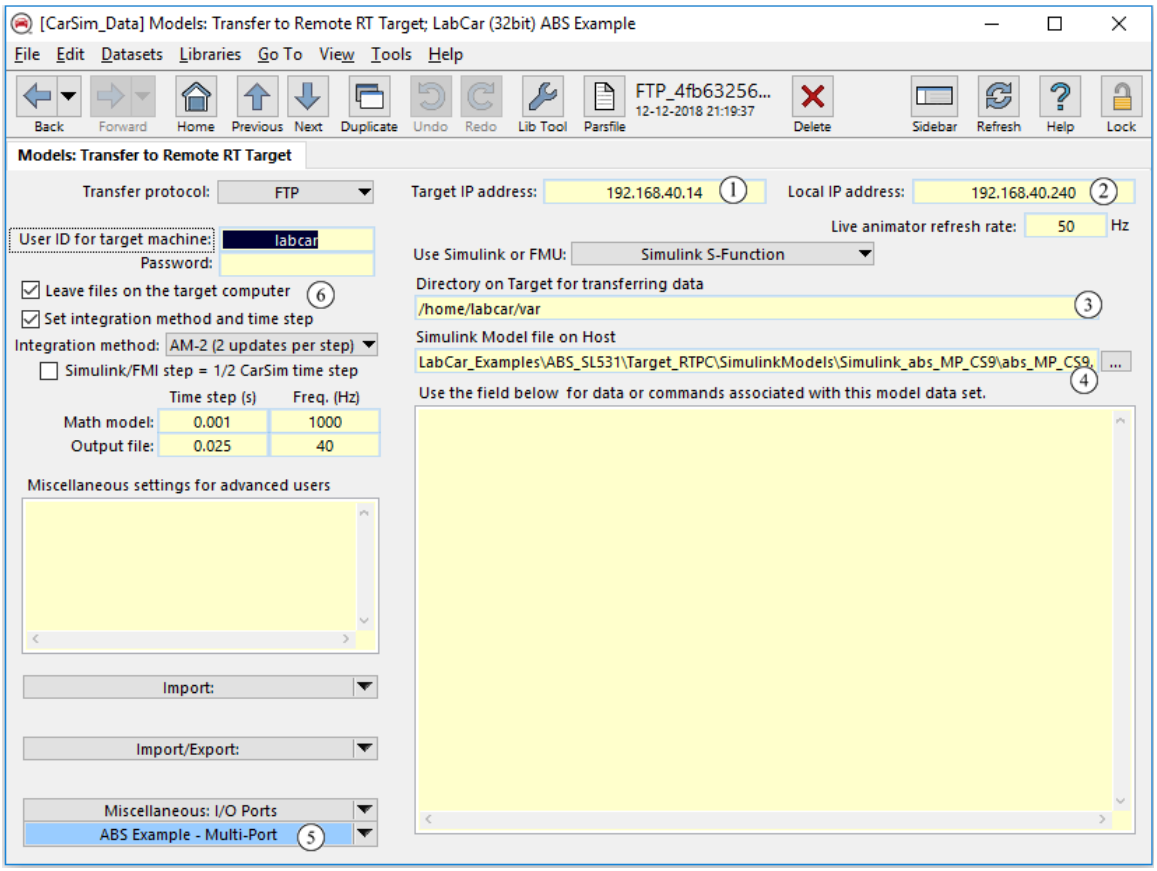


Figure 15. Specification of LabCar project and Simulink model.

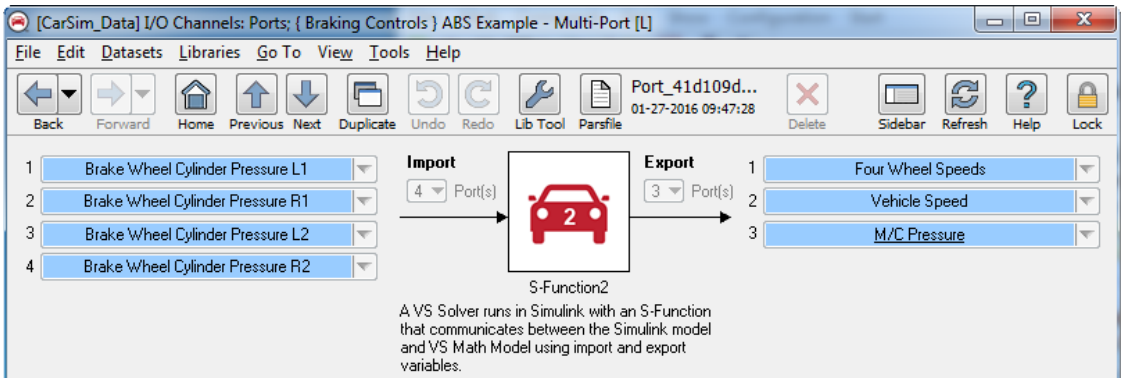


Figure 16. Import and Export channels.

Modify an Existing Simulink Model

On the CarSim Run Control screen, click the **Open** button. Pressing the **Open** button will create the simfile.sim and Run_all.par files in your project folder and launch Simulink (Figure 17). Right-click on the CarSim S-Function block and disable the library link as shown in the figure.

NOTE If Library Link is grayed out, that means this is already disabled. If you are following this tutorial and copy a Simulink model from the

ETAS_Examples folder, the link is already disabled. If you are following this tutorial by using your own Simulink model, you need to disable the link here.

Save the revised Simulink model and close MATLAB. We need to disable the link because it currently links to S-Function in CarSim_Prog/ folder. By disabling this link, it looks at the S-Function in the current project directory.

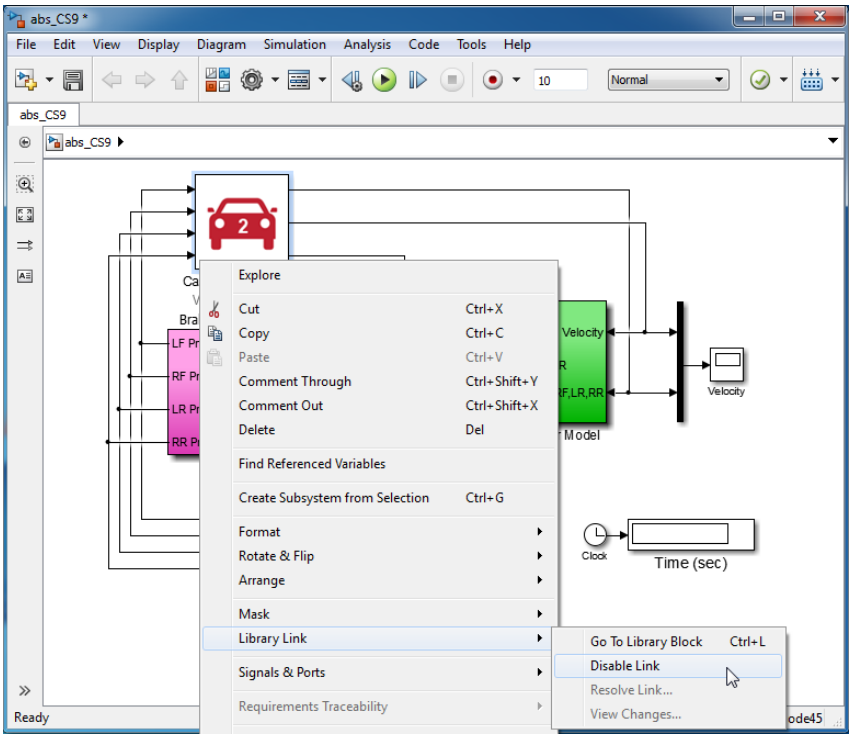


Figure 17. ABS Simulink Model, Disable Library Link.

Inside S-Function block, there are two tunable parameters: **Initial Vehicle Model State** and **System Terminate Control** (Figure 18). Even though there is an option for System Terminate Control, this option is for other Real Time systems, and it is not available for LabCar systems. However, the user can toggle between **Pause** and **Run** for **Initial Vehicle Model State** when using a Simulink model. This option is not available if FMU is used. The default of **Initial Vehicle Model State** is **Run**. When the simulation starts, the vehicle runs immediately. If this is set to **Pause**, the vehicle will not run until the Run is enabled in Experimental Control GUI.

When using an existing Simulink model and upgrading to CarSim versions 2019.1 or newer, it is necessary to replace the existing CarSim S-Function Block and disable the link. If the CarSim S-Function block is not updated, when the green play button is pressed, the "Error reported by S-Function vs-sf" error message will be displayed. To avoid the error:

1. Open an existing Simulink model and open Simulink Library Browser. Delete CarSim S-Function in the Simulink model and replace it with the one in the Library Browser (Figure 19).

2. Right click the newly replaced CarSim S-Function block and click **Library Link** > **Disable Link** to disable the link (Figure 20).

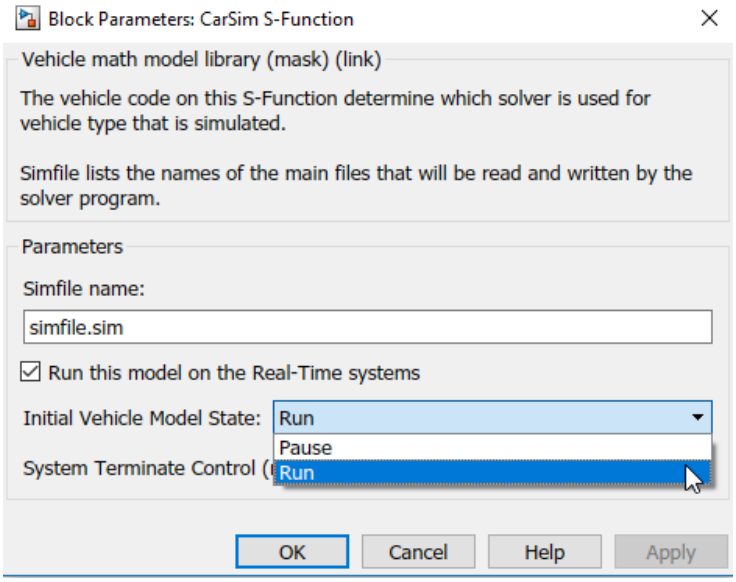


Figure 18. S-Function Block.

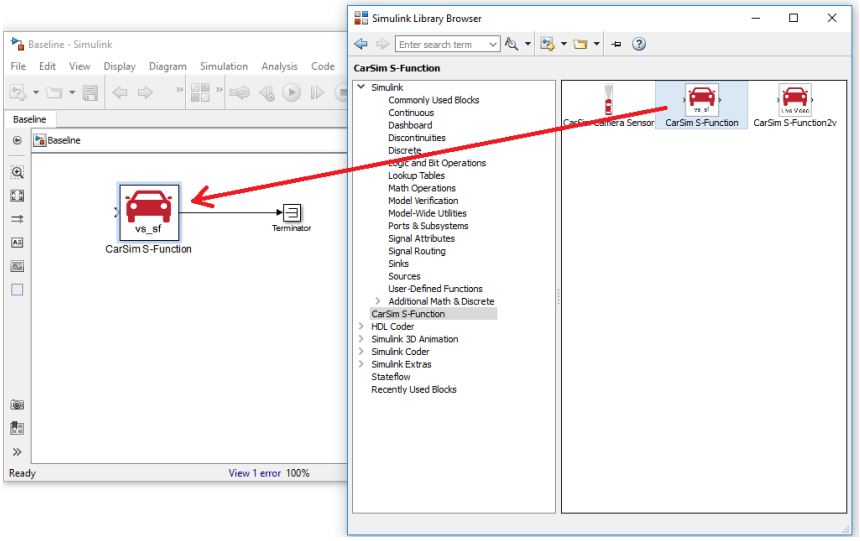


Figure 19. Replace CarSim S-Function.

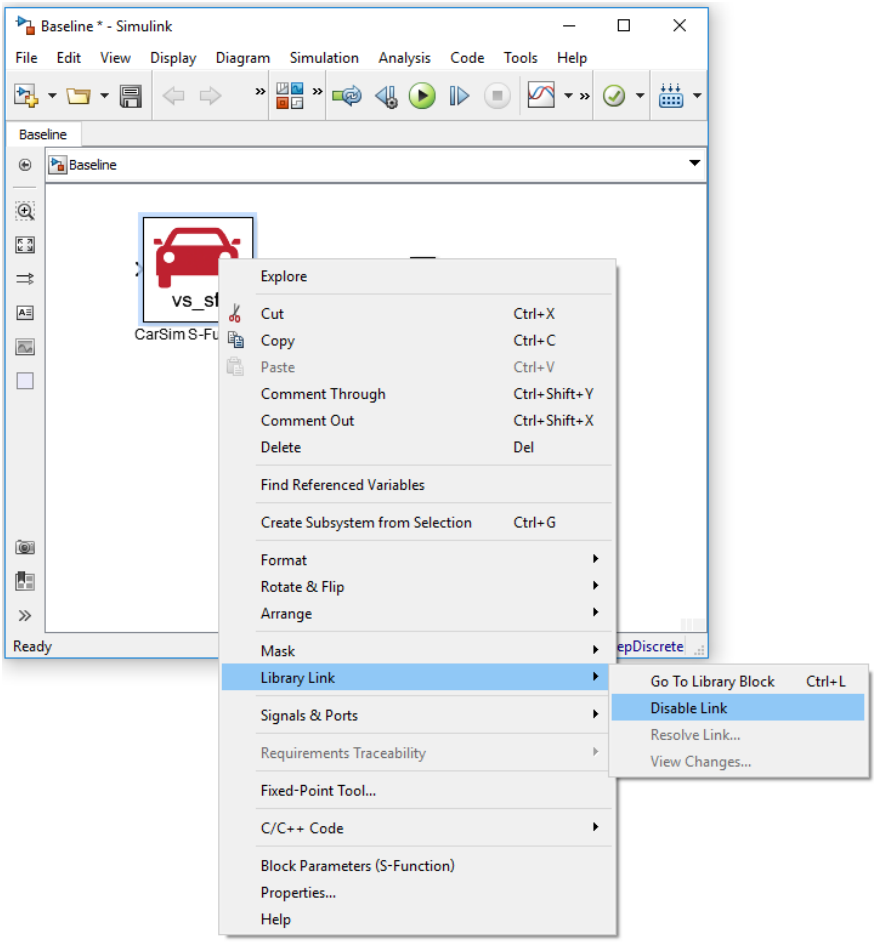


Figure 20. Disable Link.

Build the LabCar Model

Now you can build the LabCar model.

1. Go back to the LabCar Operator (Figure 21). In the OS Configuration tab, check and make sure the Enable OS Monitoring box is checked (1), click Task: Acquisition, and set period to 0.01s (2). If you do not see Enable OS Monitoring, try clicking the blue arrow  on the right side of the window. This setting allows users to observe the system processing.

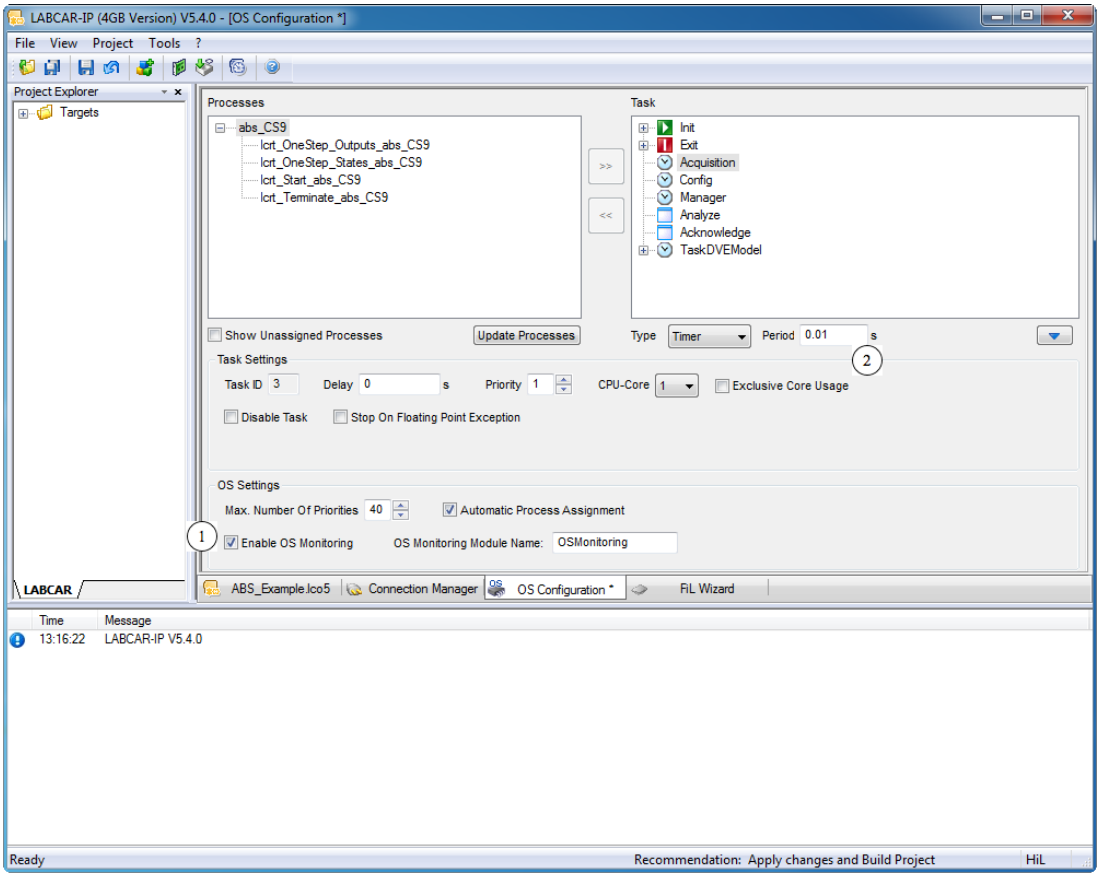


Figure 21. LabCar Operator OS Configuration.

2. Click the build icon  in LabCar Operator or from the Project menu. This operation will do the following:

    a. Launch Matlab/Simulink/Simulink coder (or called Real-Time Workshop) to convert Simulink model to C code.

    b. Download all C files and library onto RTPC target.

    c. Launch the gcc compiler to compile all C files.

    d. Link the compiled files and downloaded library to create an executable file.

    e. Start the LabCar Experiment Environment.

3. If the Experiment Environment did not automatically open, click the Open Experiment Environment icon  in LabCar Operator (or use the Tool menu Open Experiment Environment) to open the Experiment Environment.

4. From the LabCar Experiment Environment, drag the item named runtime (LABCAR > OSMonitoring > TaskDVEModel > runtime in Figure 22). Select the Oscilloscope out of the dropdown menu labeled Instrument Type (Figure 23). On the plot (Figure 22), click the y-axis and change Maximum Value to 0.005.

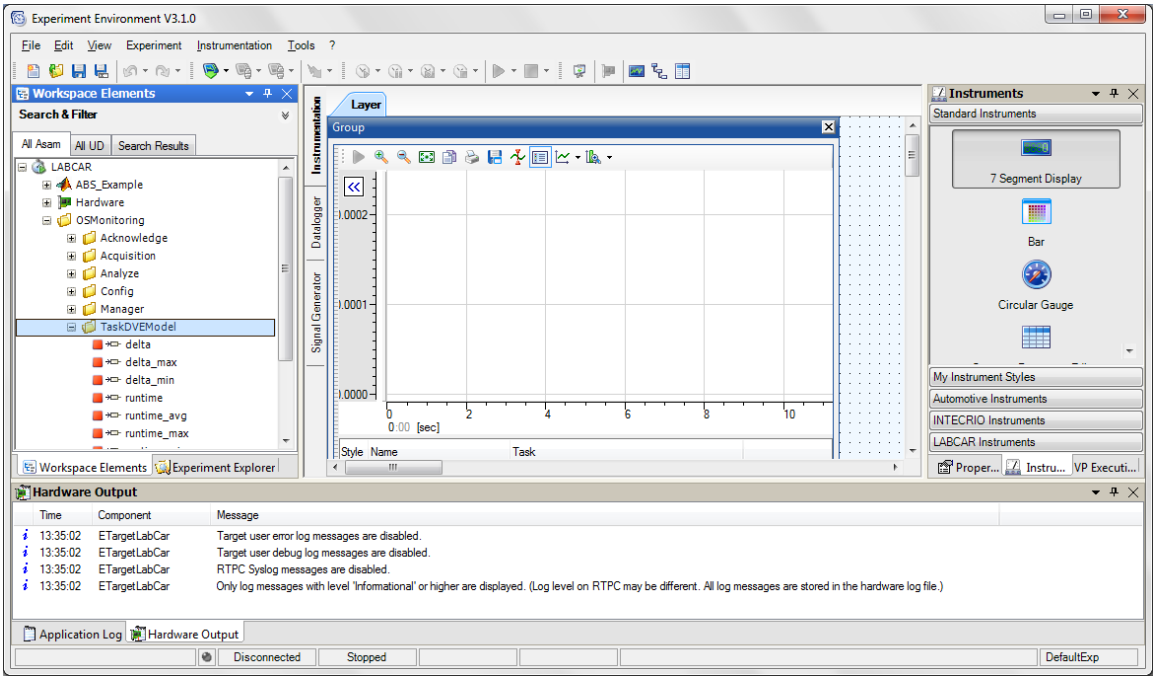


Figure 22. Real time run.

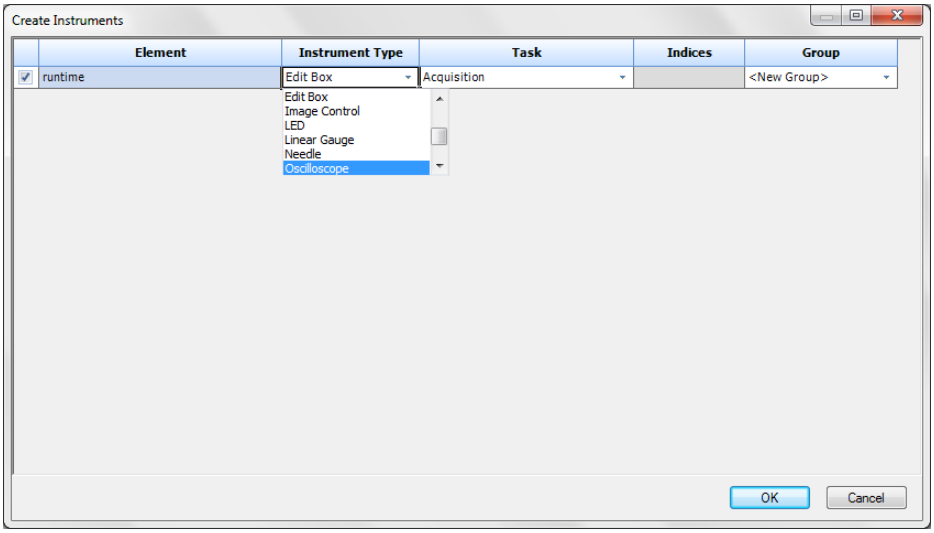


Figure 23. Instrument type settings.

Run the Simulation

1. In the CarSim Run Control screen, change **Number of live animators for this run:** to 1.
2. Click the **Send** button. You should see the animator program prepare for live animation.
3. In the LabCar Experiment Environment, click the Download icon  (or use the Experiment menu Download->Labcar) to download the experiment to the target PC.

4. Start running by clicking the icon  . When the model is running, you should see the vehicle moving via live animation. Click the icon  to stop the run.
5. After stopping the run, the results are automatically transferred back to the host. If there are any problems, you can manually transfer results back to the host by clicking the **Receive** button on the CarSim Run Control screen. However, this is normally not necessary.

In 2019.1 version, the `vs_state` parameter was introduced. By using this parameter, a button control can be added like the one shown at the top right corner in Figure 24. The Baseline project which is included in the shipping cpar file contains this button as an example. The parameter `vs_state` cannot be used with FMU models.

The value of the parameter `vs_state` is defined as:

- **Pause: 1**
- **Run: 2**

The user can pause/run simulation by using this button. To stop the simulation, press the stop icon  .

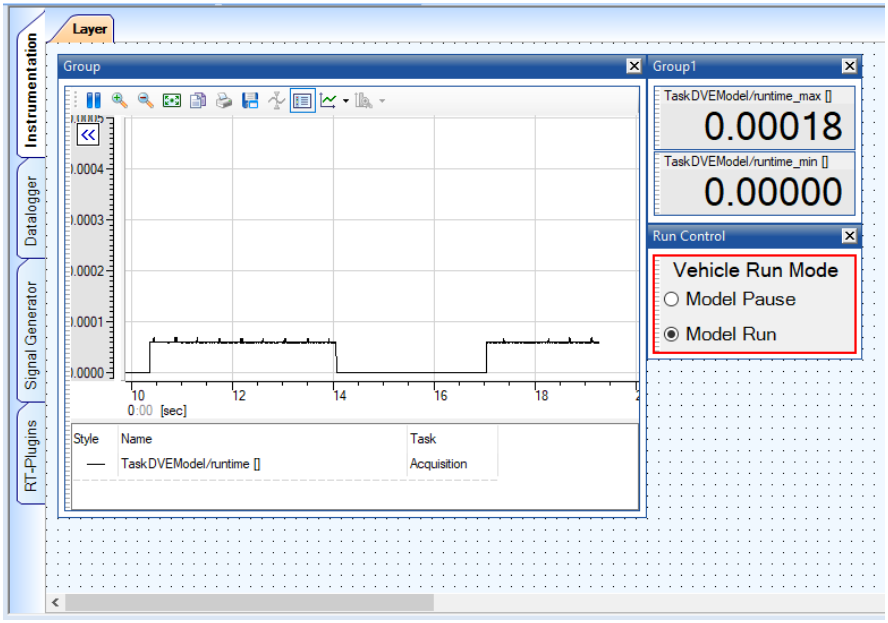


Figure 24. Vehicle Model Pause/Run Option.

Note In Experiment Environment, the icons  or  are for the system run control. When the stop icon is pressed, the whole system stops. Using Vehicle Model Pause/Run from vs_state parameter, it will run/pause just vehicle model, and other hardware/software in the system can still be running.

LabCar Tutorial: Create & Run an FMU/FMI RT Project

The Functional Mockup Unit (FMU) is a function component which implements with the interface called FMI. It consists of one zip-file with extension ".fmu" and contains all necessary components to utilize the FMU. The FMI standard consists of two main parts: a) FMI for Model Exchange and b) FMI for Co-Simulation. The first version, FMI 1.0, was published in 2010, followed by FMI 2.0 in July 2014. The LabCar from version 5.3.0 can import FMI Co-Simulation version 1.0 (FMI 1.0).

CarSim/TruckSim/BikeSim, started supporting FMI version 1.0 and 2.0 in Co-Simulation mode on the Windows platform in version 2016.0. We pack the vehicle solvers as an FMU. In version 2017.0, we extended our vehicle solvers FMU to run on the LabCar RT target. On the RT system, LabCar supports Co-Simulation FMI 1.0. The FMU module allows you to run in real-time with hardware in the loop without Matlab/Simulink/Simulink Coder (RTW). You can directly link modules based on FMU or hardware I/O to the vehicle model.

Set Up a LabCar Project

Start by creating a new LabCar project.

1. From the menu of LabCar Operator select **File –> New Project** (Figure 25) to make a new project.

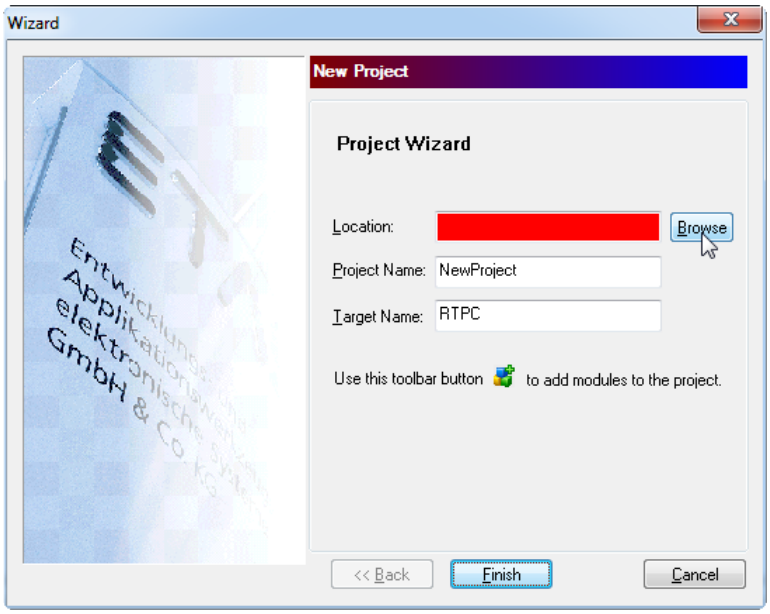


Figure 25. Create a new LabCar Project.

2. Create or select a project location and type in a project name, e.g. `ABS_FMU` (Figure 26). In this example, we are creating a project in `CarSim_Data\ETAS_Examples\`. Click on **Finish** and open the LabCar-IP main window, Figure 27. Leave LabCar-IP open but minimize the screen.

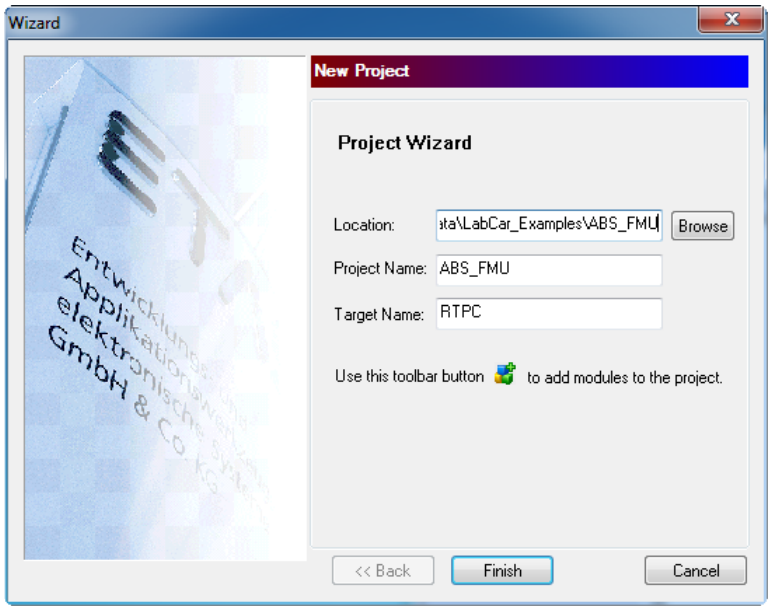


Figure 26. Select Project location and name.

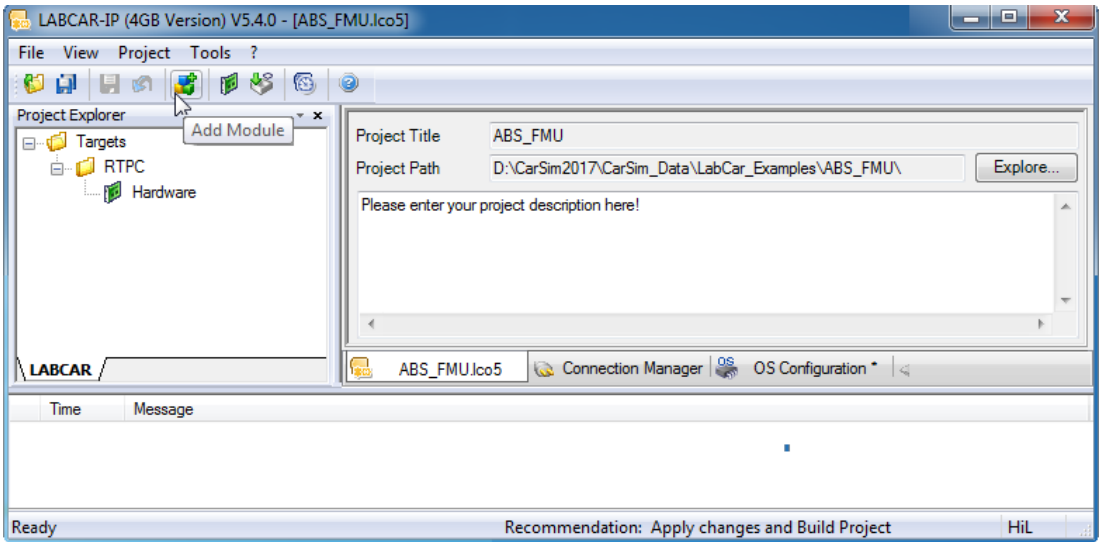


Figure 27. LabCar-IP Project main window.

3. Start CarSim.
4. In CarSim, go to the **Run Control** library and select `<RT: ETAS> Ext. ABS: Split Mu Simulink (32bit RTPC)` example. Copy the dataset for your new run and rename it to `<RT ETAS> Ext. ABS: Split Mu Simulink (FMU)`. In Models: make sure that **Transfer to Remote RT Target** is selected.
5. The blue link, `LabCar (32bit) ABS Example`, is for ABS Simulink example. Copy and Link this dataset and rename it to `LabCar ABS Example (FMU)`

6. Click the model blue link `LabCar ABS Example (FMU)` to go to Models: Transfer to Remote RT Target dataset screen (Figure 28). In **Use Simulink or FMU** pull down menu, select FMU for LabCar (FMI 1.0). In **Vehicle Solver FMU path and file name**, enter

   `{CarSim Data folder}\ETAS_Examples\ABS_FMU\vs_fmu1.fmu`

   (this is the directory where you created a new LabCar project)

   If an FMU name other than **`vs_fmi1.fmu`** is used, you must recreate the shared object. Please refer to Function Mock-up Unit (FMU) of LABCAR-OPERATOR User's Guide. For this example, it is recommended that you leave the file name as vs_fmi1.fmu

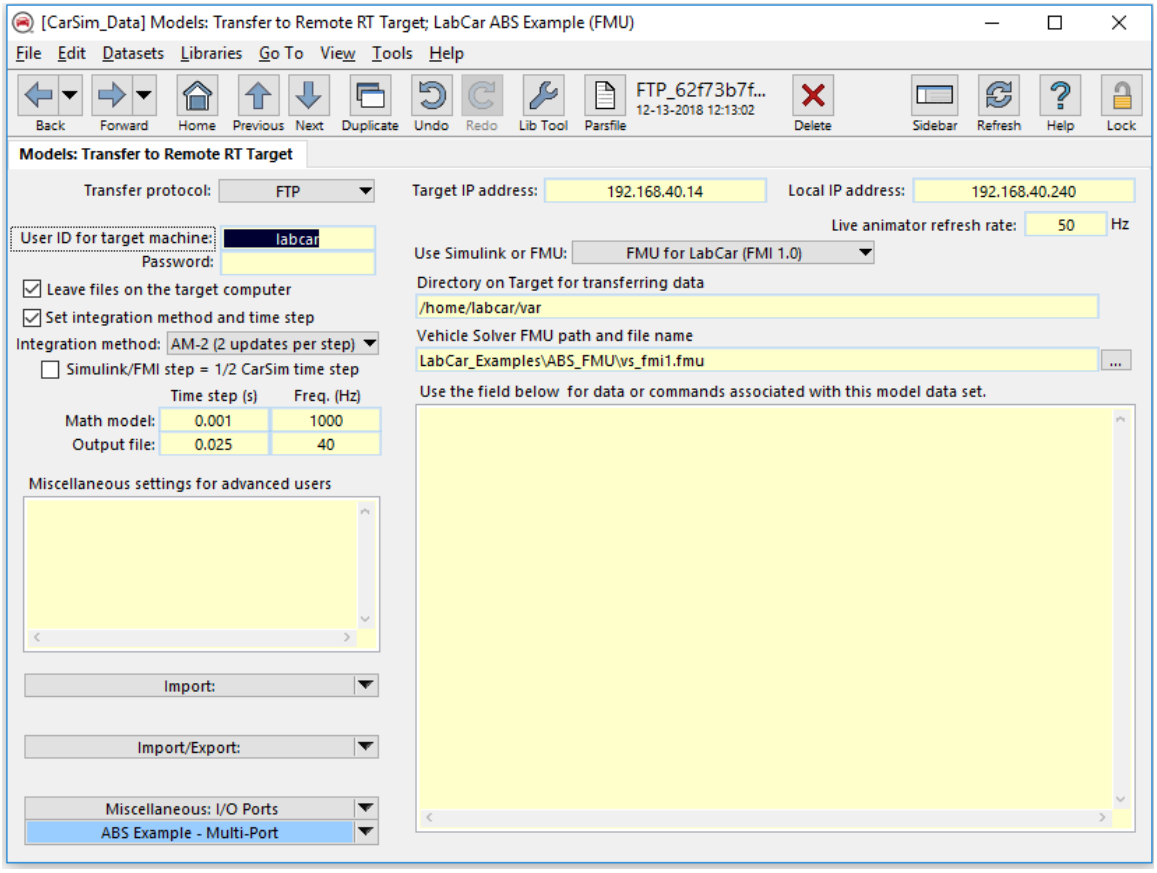


Figure 28. FMU model screen settings.

7. Go back to the Run Control Screen. The control button will change from **OPEN** to **Generate FMU** from "**Open**", see Figure 29.

8. Click "**Generate FMU**" to generate the FMU for this setting. And you get confirmation message as shown in Figure 30. Do not click the **OK** button yet. Leave this confirmation message on.

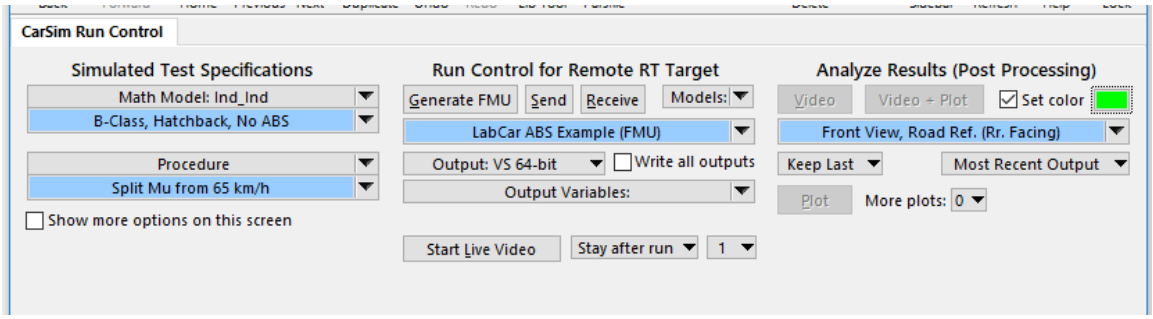


Figure 29. FMU Run Control screen.

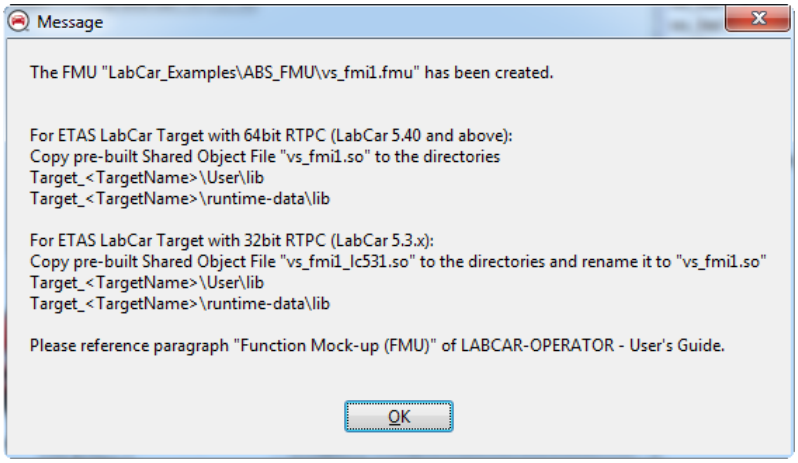


Figure 30. Confirmation message after generating FMU.

9. Open LabCar main project window (see Figure 27) and click **Add Module** icon. An add module wizard pops up, Figure 31. Select **Add C-Code Module** and click the **Next**.

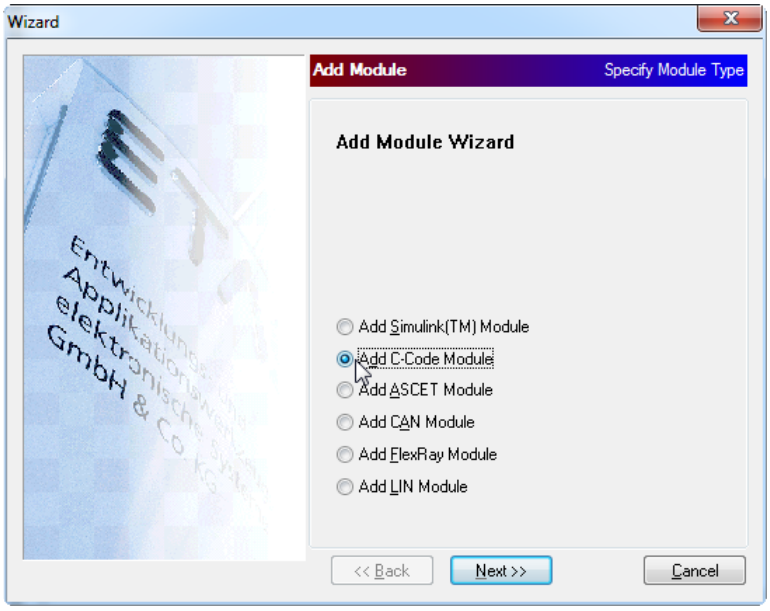


Figure 31. Select Add C-Code Module.

10. Select Use existing FMU model (V1.0 Co-Simulation), see Figure 32. Browse to the location where FMU file was generated in the previous step (Figure 33):

    `{CarSim Data folder}\LabCar_Examples\vs_fmu1.fmu`

    Press the Finish button. You should get a message, “FMU module is added successfully.”

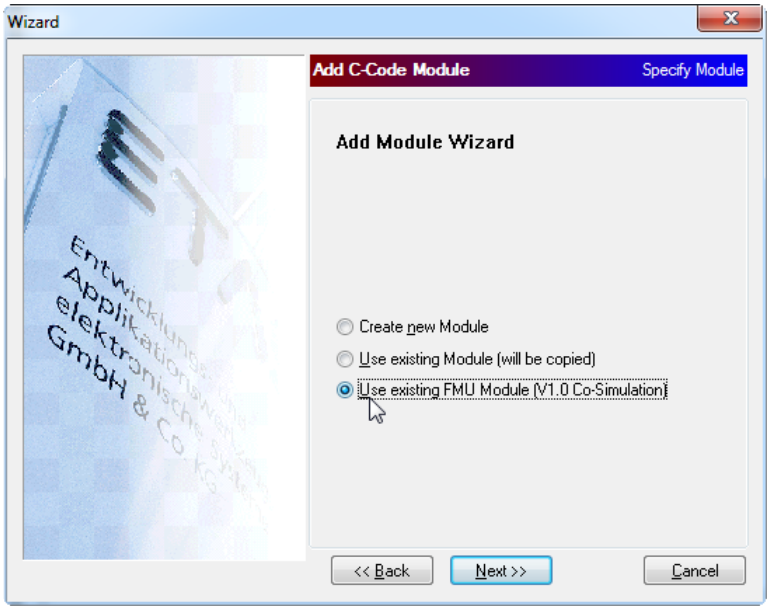


Figure 32. Select Use existing FMU Module.

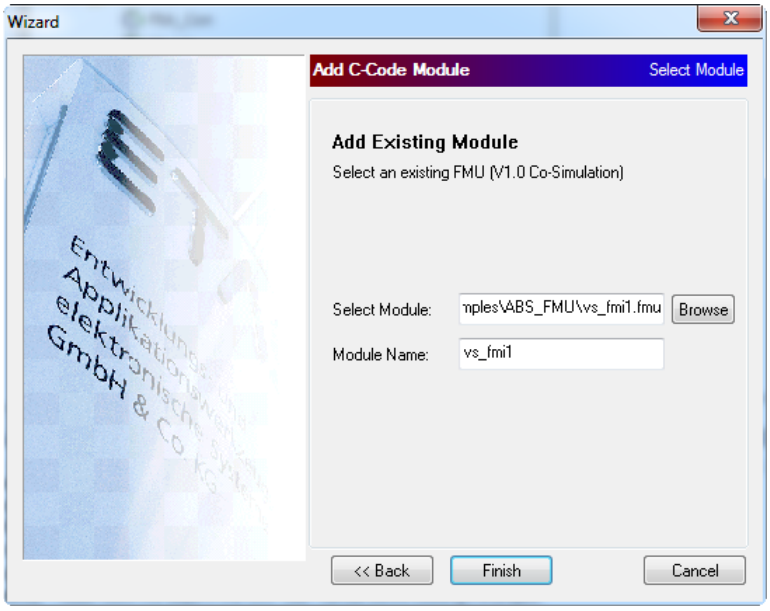


Figure 33. Add the vehicle solver FMU “vs_fmi1.fmu.”

11. Follow the instruction in this confirmation message (Figure 30). You should have a `vs_fmi1.so` in the directory where you created the LabCar project. In this example, the

file should be in:
`{CarSim Data folder}\ETAS_Examples\ LCO_FMU_SL_ABS\`

- If you are using LabCar 5.40 and above, copy `vs_fmi1.so` from the above folder to the following folders:

  `{LabCarProjectFolder}\Target_RTPC\User\lib`

  `{LabCarProjectFolder}\Target_RTPC\runtime-data\lib`

- If you are using LabCar 5.3.x, you should copy `vs_fmi1_lc531.so` to the above directory and rename it to `vs_fmi1.so`

12. In the LabCar window (Figure 27), click Add Module icon to add the Simulink model `ETAS_Examples\Simulink_Models\ABS_Controller_Only\abs_ctrl.mdl` (Figure 34). Make sure to check mark **Copy Model Directory into Project.**

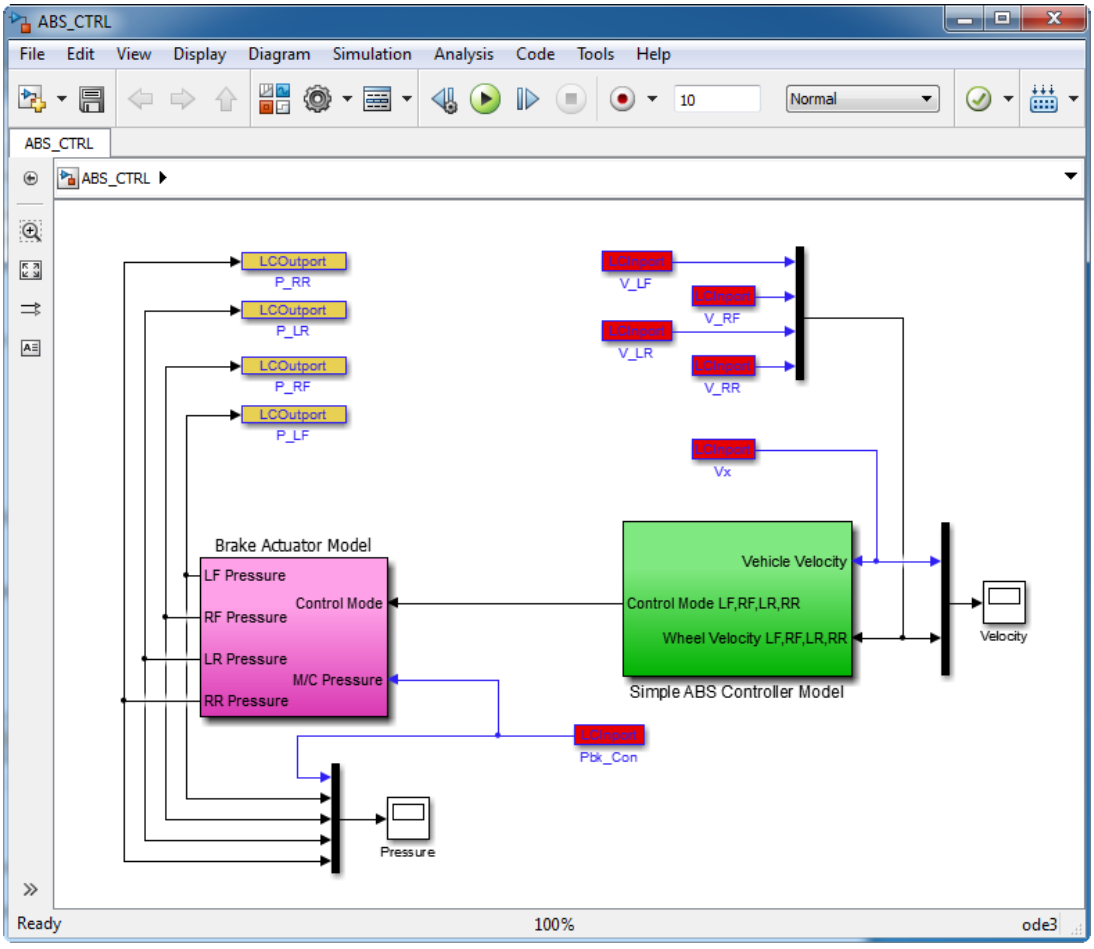


Figure 34. ABS Controller Simulink Model (without vehicle solver).

13. On the LabCar main project window, click the Connection Manager tab to make the connection between FMU and Simulink (Figure 35). Once you make all connections, select **File** -> **Save**.

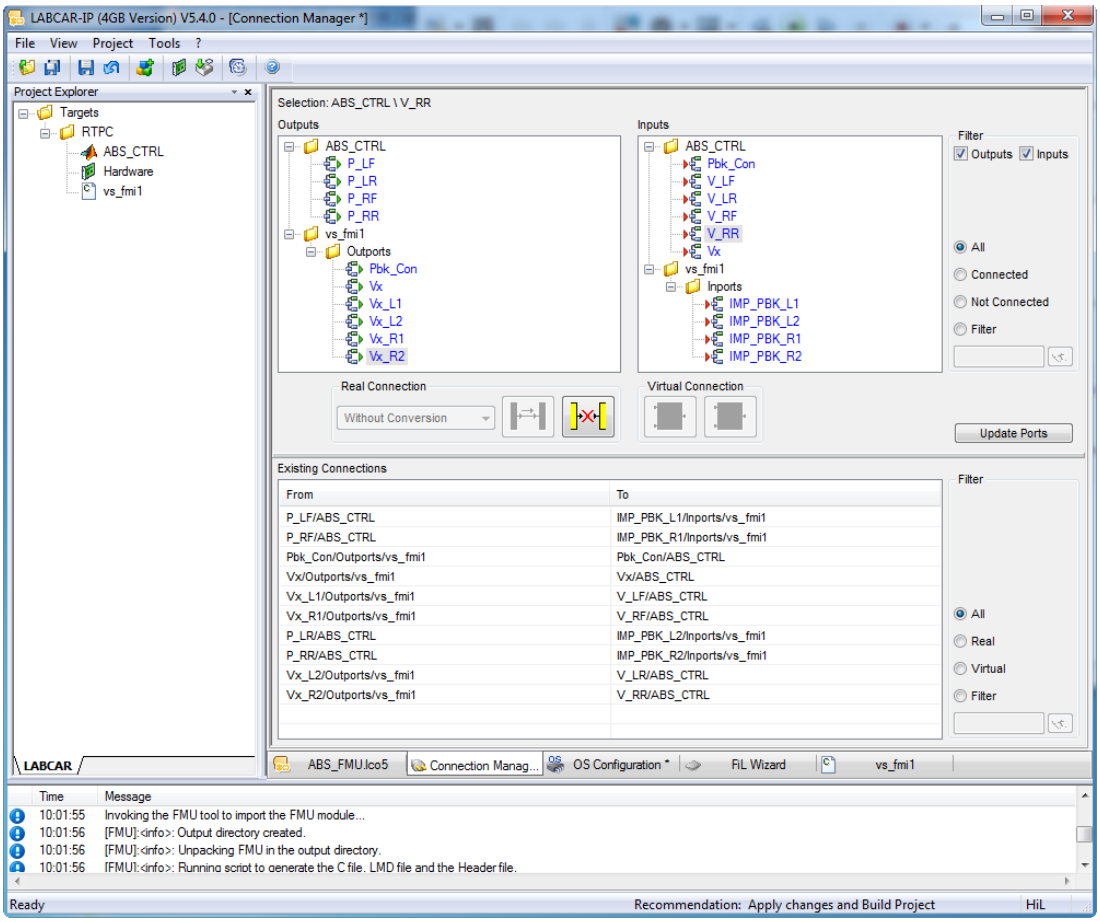


Figure 35. LabCar Project – connecting FMU and Simulink.

Build the LabCar Model

Now you can build the LabCar project from the LabCar Operator window.

1. Go back to the LabCar Operator (Figure 35). In the OS Configuration tab, check and make sure the Enable OS Monitoring box is checked and set update timer period to 0.01s. If you do not see Enable OS Monitoring, try clicking a blue arrow  on the right side of the window.

2. Click the build icon  in LabCar Operator or from the Project menu. This operation will do the following:

    a. Launch Matlab/Simulink/Simulink coder (or called Real-Time Workshop) to convert Simulink model to C code.

    b. Download all C files and library onto RTPC target.

    c. Launch the gcc compiler to compile all C files.

    d. Link the compiled files and downloaded library to create an executable file.

    e. Start the LabCar Experiment Environment.

3. If the Experiment Environment did not automatically open, click the Open Experiment Environment icon  in LabCar Operator (or use the Tool menu Open Experiment Environment) to open the Experiment Environment.
4. On the LabCar Experiment Environment, create plots or any kind of display if you need. For example, you can drag OSMonitoring > Acquisition > runtime to the Layer window, change Instrument Type to Oscilloscope. Change the Maximum Value to be 0.0005 and observe the runtime.

Run the Simulation

1. From the CarSim Run Control screen, click the **Send** button and you should see the animator program prepare for live animation.
2. In the LabCar Experiment Explorer, click the Download icon  (or use the Experiment menu Download->Labcar) to download the experiment to the target PC.
3. Start running by clicking the icon  . When the model is running, you should see the vehicle moving via live animation. Click the icon  to stop the run.
4. After stopping the run, the results are automatically transferred back to the host. If there are any problems, you can manually transfer results back to the host by clicking the **Receive** button on the CarSim Run Control screen. However, this is normally not necessary.

Running Parallel Solver Models

This section covers the settings that differ between running a Multiple Vehicle example in LabCar rather than on Windows. For more information regarding the standard setup for a Parallel Solvers example, refer to the 'Simulations with Multiple Vehicles' technical memo. Running Parallel Solvers models in LabCar is only supported with FMU at this time.

From the Parallel Solvers screen (Figure 36), use the drop-down menu ① to select LabCar as the Run platform.

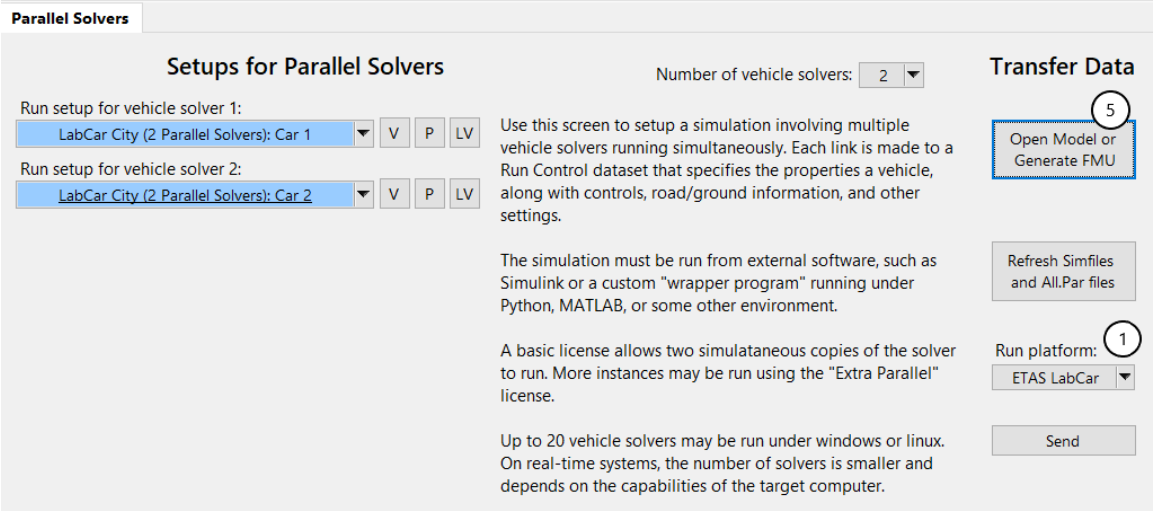


Figure 36. LabCar Parallel Solvers Screen Setup.

Each Run control included in the Parallel Solver screen will need a uniquely defined 'Models: Transfer to Remote RT Target' dataset to specify separate live animator communication TCP Ports ② and FMU files ③ for each parallel solver (Figure 37).

> **Note** The FMU files must be located in the LabCar project folder. The LabCar "Target Name" must use default "RTPC". When the FMU files are generated, the browser will copy the binary files to the proper location on the target.

On the Models screen for each vehicle, toggle on the **Run multiple solvers on the target machine** ④ setting.

From the Parallel Solvers screen (Figure 36), click Generate FMU ⑤ .

In Labcar, add the FMU Modules for each parallel solver and ensure that CarSim I/O between solvers is appropriately set up in the Connection Manager.

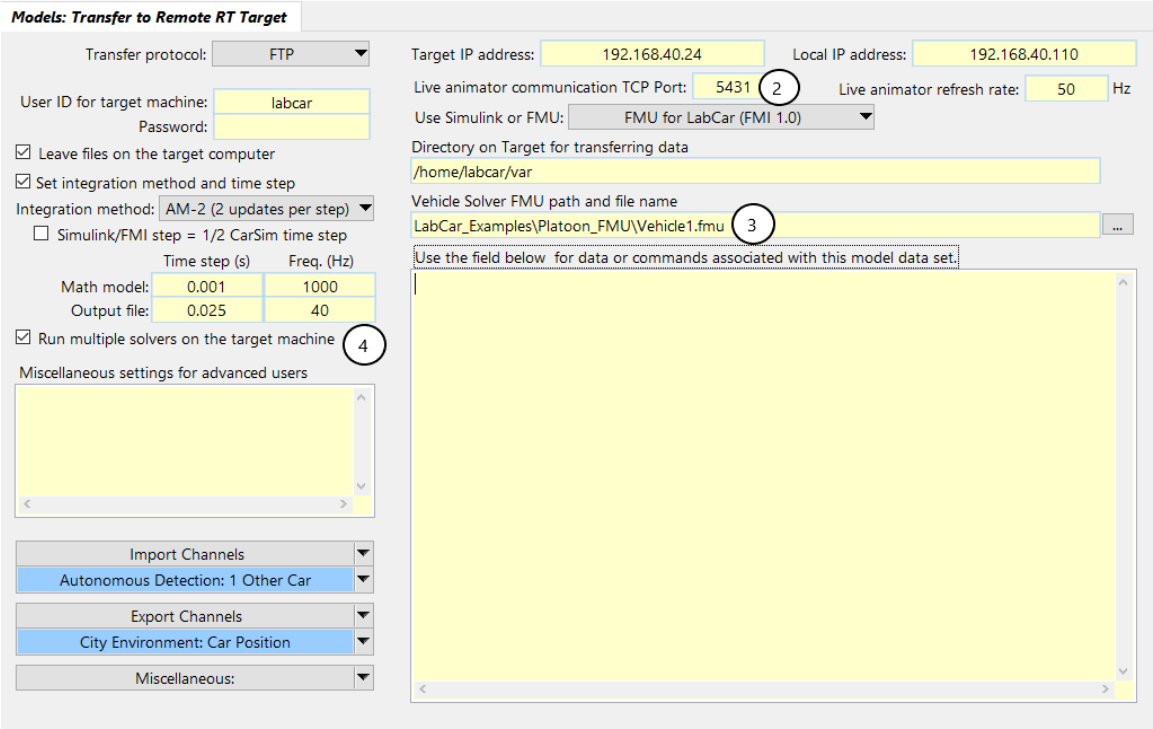


Figure 37. LabCar Parallel Solvers Models Screen Setup.

In the OS Configuration panel, for each Parallel Solver instance, define a separate Task for the *cmod_config_* and *cmod_process_* processes as shown in Figure 38. For the Tasks that have the *cmod_process_* process attached, ensure the timer value matches the Run's step time.

After this, the model can be built and run as normal.

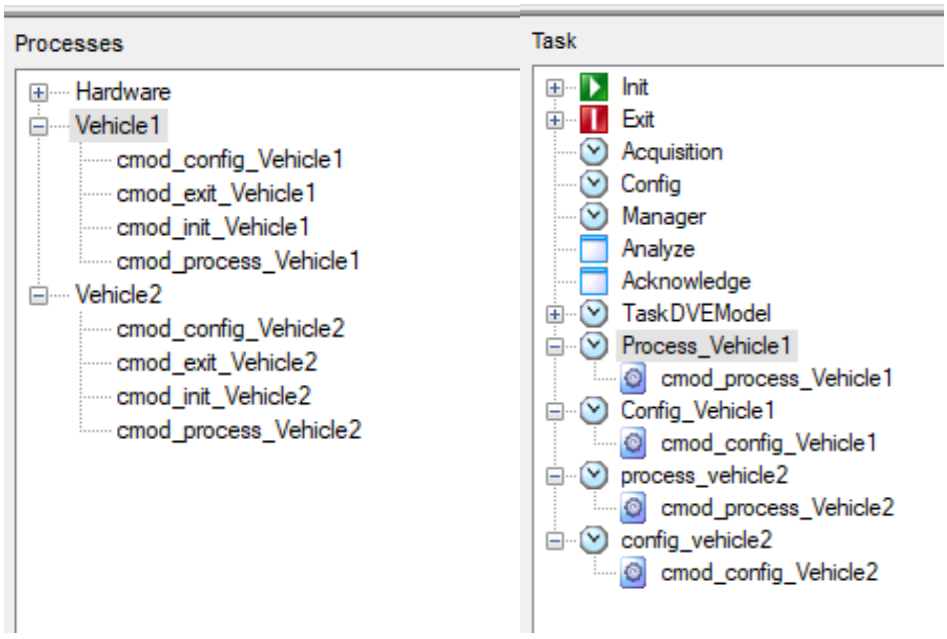


Figure 38. LabCar Parallel Solvers OS Configuration.

COSYM Tutorial: Create & Run a Simulink RT Project

You can create a real-time simulation or test project with an existing MATLAB/Simulink model. This section describes how to create a new COSYM project from scratch using an existing CarSim model example. For this tutorial, you will be monitoring system processing usage using COSYM.

Before starting this tutorial, follow the instructions of "Installing VS Solvers into ETAS RT Target Computer" and "Importing Examples Datasets." We will be using a Simulink model which is included in `RT_ETAS.cpar`

Set Up a COSYM Project

Start by creating a new LabCar project.

1. The Simulink we will be using for this example is located in:

   `{CarSim Data folder}\ETAS_Examples\ Simulink_Models\ABS\`

   The name of the Simulink model is abs_MP_CS9.mdl. If you do not see this directory, be sure to follow "Importing Examples Datasets" instruction.

2. Launch COSYM Operator. From the menu, select Create New Project to make the new project. Name the Project and provide the folder Location. For this example we will name the Project 'ABS' and designate the folder location:

   `{CarSim Data folder}\ETAS_Examples\ABS_Example`.

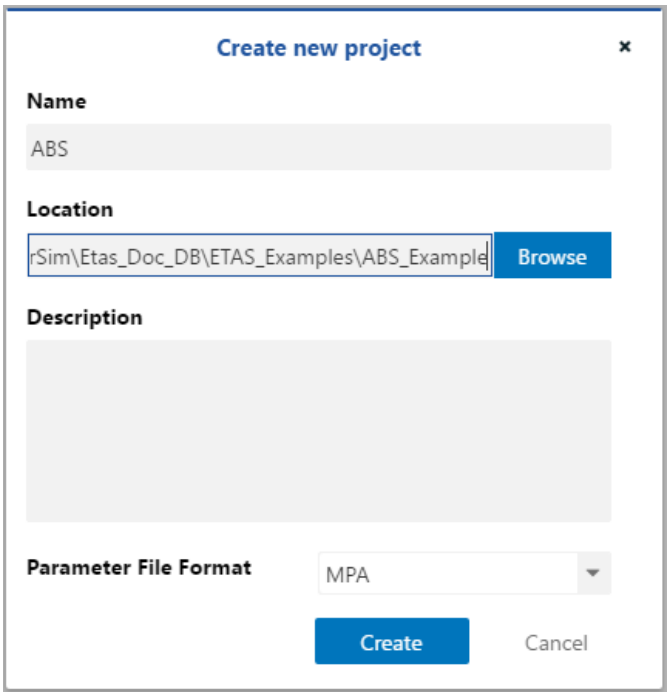


Figure 39. Create COSYM Project

3. Copy the ABS folder from the directory: `{CarSim Data folder}\ETAS_Examples\Simulink_Models\` into the directory: `{CarSim Data folder}\ETAS_Examples\ABS_Example\ABS\Library\Control`.

Prepare CarSim Datasets

1. Launch CarSim and select **Datasets** > **{Simulink Models}** > **Ex. ABS: Split Mu**. Press the **Duplicate** icon. When the Duplicate Dataset window pops up, enter `RT: ETAS` in **Category for new dataset**. You can leave the title as `Ex.ABS: Split Mu #1`. When running Simulink models under Windows, the connection to Simulink is handled with a dataset from the **Models: Simulink** library. That dataset specifies the Simulink MDL file, plus datasets for import and export variables. To run the same Simulink model with COSYM, the same information is needed, along with transfer information used by COSYM.
2. Use the **Models** drop-down list to specify the library **Models: Transfer to Remote RT (**Figure 40**)**.
3. Use the dataset select control under the **Models** control and choose the option **Link to New Dataset** (Figure 41). Give the new dataset a name and category, such as `COSYM Target` as a category and `Base Model` as a title.
4. Click the blue button for the new link to go to the newly created dataset (Figure 42).

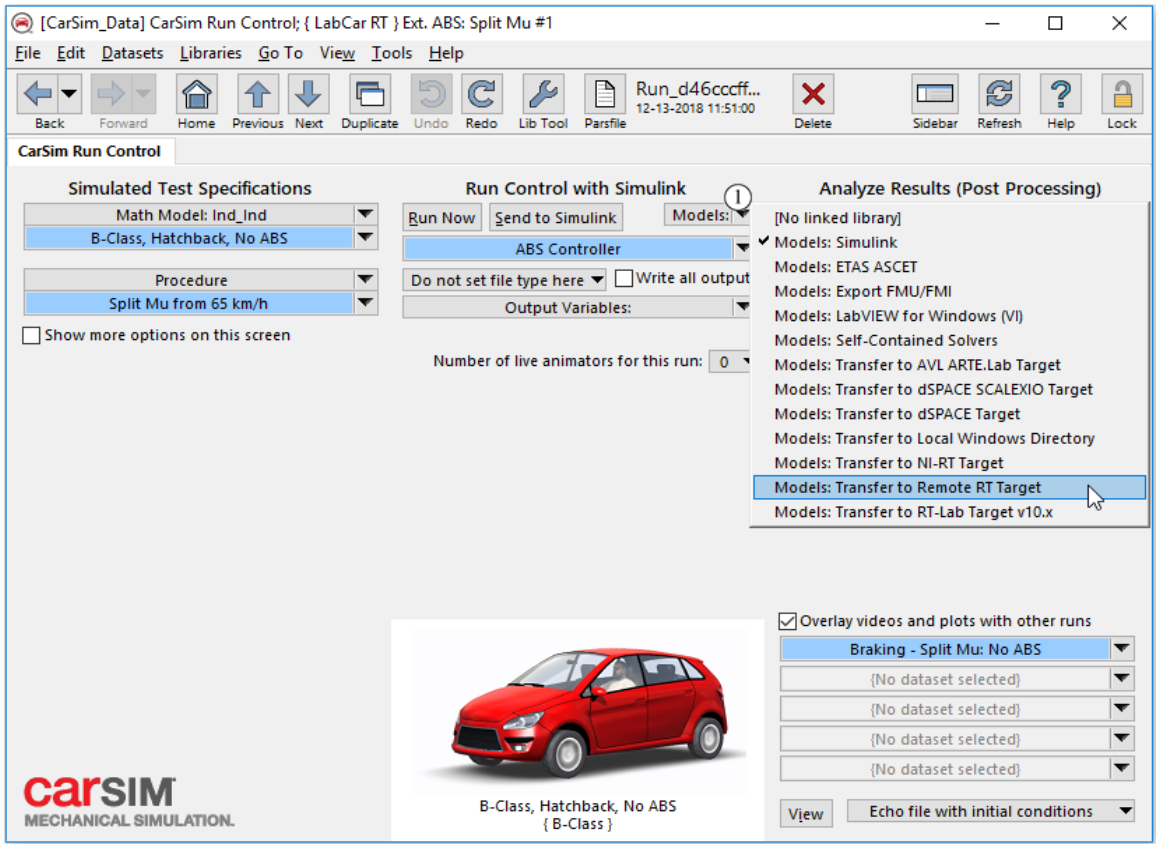


Figure 40. CarSim Run Control: change the linked Models library.

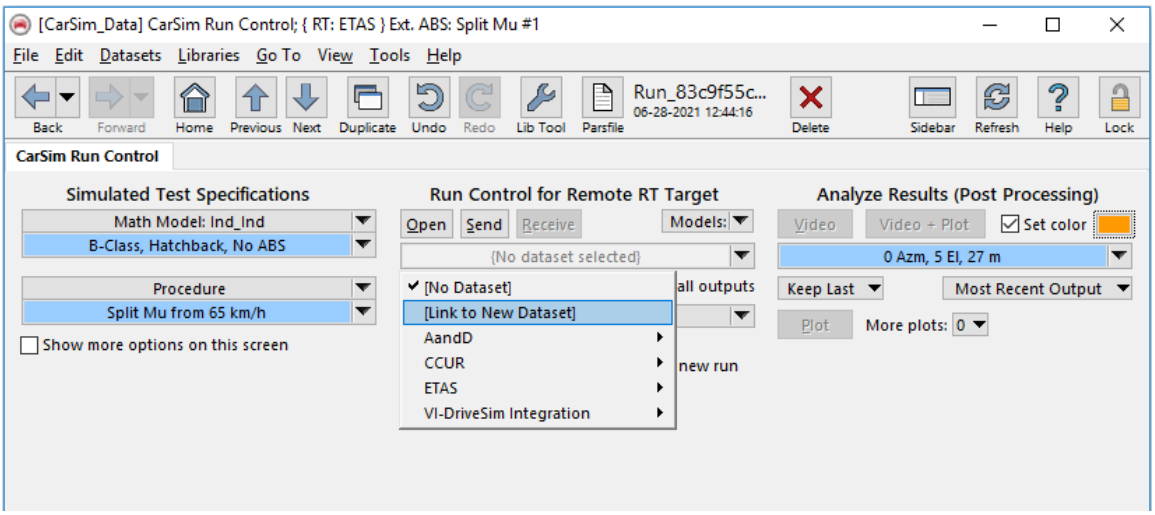


Figure 41. Make a new dataset in the Models: Transfer to Remote RT Target.

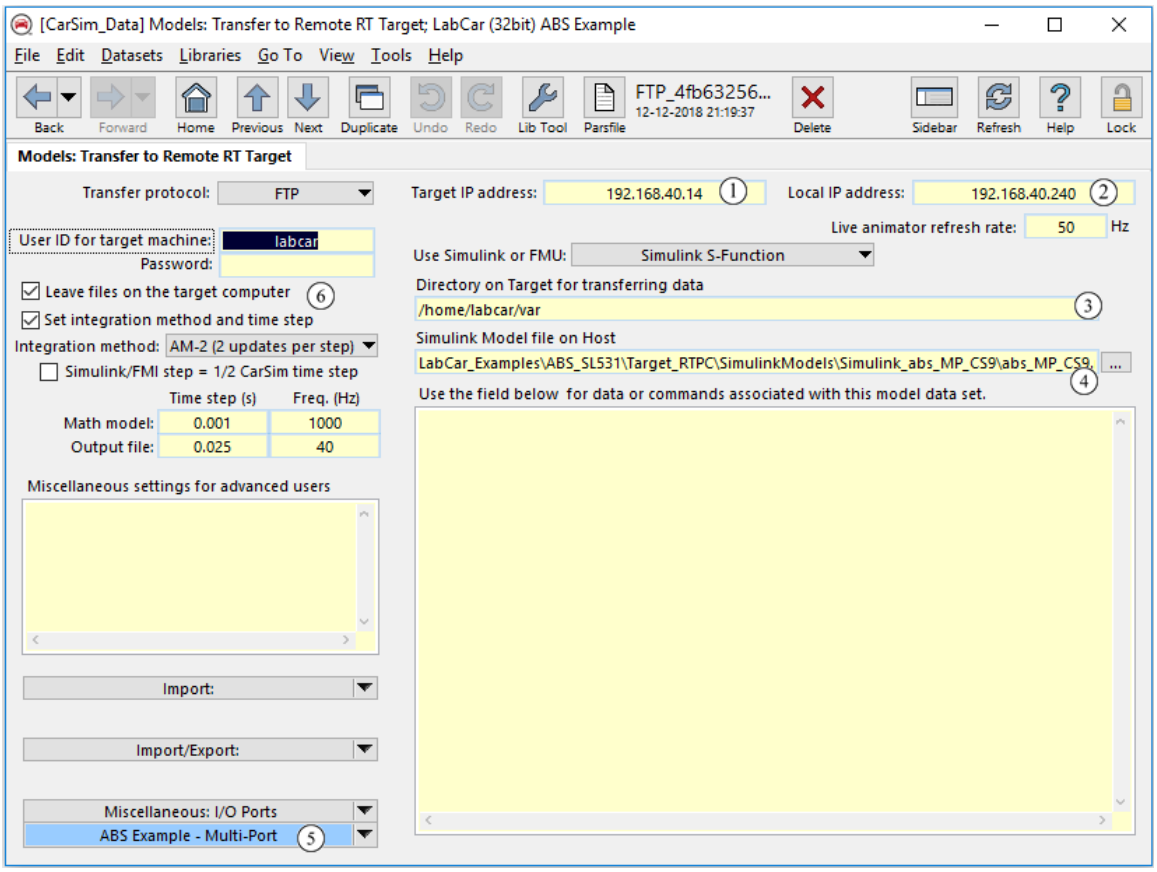


Figure 42. Specification of COSYM project and Simulink model.

5. The RTPC default login ID is `labcar` and the password is empty. Put target and host IP addresses for file transfer between host and target on your specific network to enable live

animation. The directory on the target is `/home/labcar/var`. Specify a Simulink model file and make blue links for import and export datasets as shown in Figure 43. The Simulink model should be in the place where you created a COSYM project (in Step 2 of Set Up a COSYM Project). In this example, this is:

```
{CarSimDatafolder}\ETAS_Examples\ABS_Example\ABS\Library\Control\ABS\abs_MP_CS9.mdl
```

In the lower left corner, select **I/O Channels** > **I/O Channels: Ports**, and then select **Braking Controls** > **ABS Example – Multi Port** for a blue link.

6. Click this **ABS Example -Multi Port** blue link and you will see import and export channels setup.

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

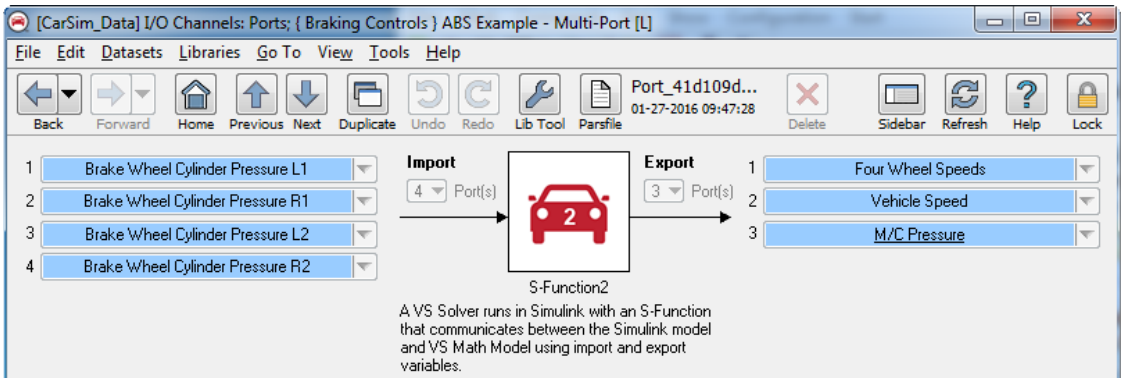


Figure 43. Import and Export channels.

Modify an Existing Simulink Model

In CarSim Run Control screen, click the **Open** button. Pressing the **Open** button will create simfile.sim and Run_all.par files in your project folder and launch Simulink (Figure 44). Right-click on the CarSim S-Function block and disable the library link as shown in the figure (note: if Library Link is grayed out, that means this is already disabled. If you are following this tutorial and copy a Simulink from ETAS_Examples\ directory, the link is already disabled. If you are following this tutorial by using your own Simulink, you need to disable the link here). Save the revised Simulink model and close MATLAB. We need to disable the link because it currently links to S-Function in CarSim_Prog/ folder. By disabling this link, it looks at the S-Function in the current project directory.

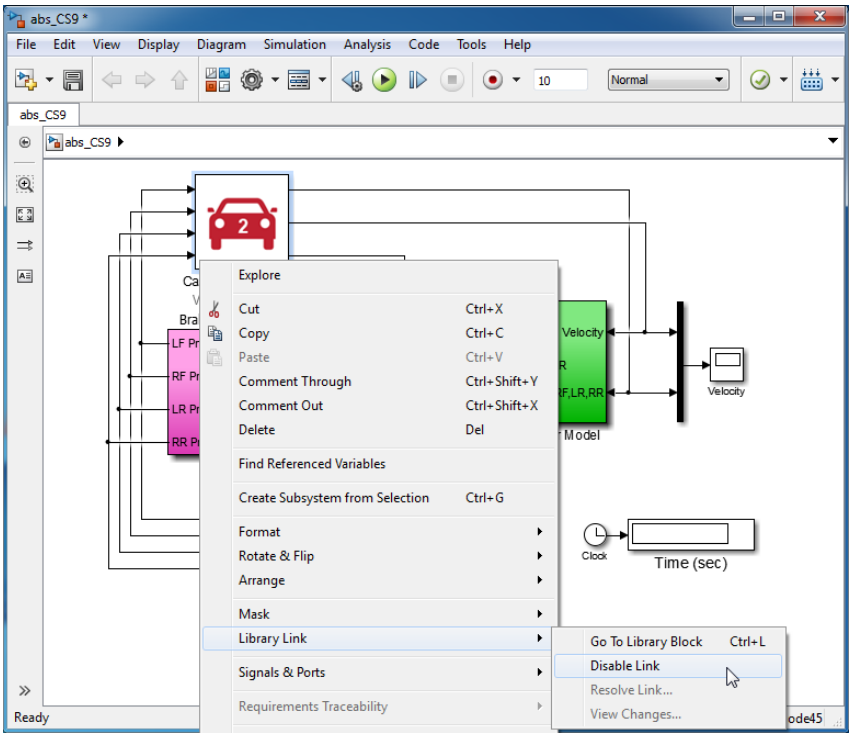


Figure 44. ABS Simulink Model, Disable Library Link.

Inside the S-Function block, there are two tunable parameters: **Initial Vehicle Model State** and **System Terminate Control** (Figure 45). Even though there is an option for System Terminate Control, this option is for other Real Time systems, and it is not available for LabCar systems. However, the user can toggle between **Pause** and **Run** for **Initial Vehicle Model State** when using a Simulink model. This option is not available if FMU is used. The default of **Initial Vehicle Model State** is **Run**. When the simulation starts, the vehicle runs immediately. If this is set to **Pause**, the vehicle will not run until the Run is enabled in Experimental Control GUI.

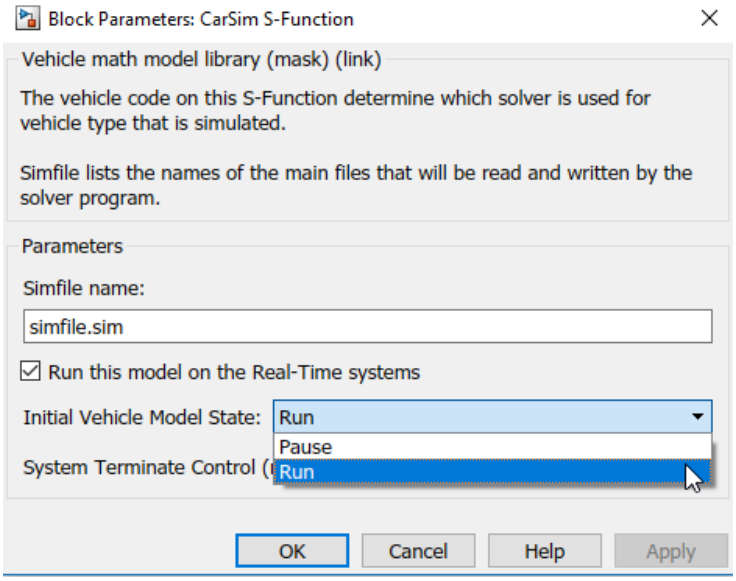


Figure 45. S-Function Block.

When using an existing Simulink model and upgrading to CarSim versions 2019.1 and newer, it is necessary to replace the existing CarSim S-Function Block and disable the link. If the CarSim S-Function block is not updated, when the green play button is pressed, the "Error reported by S-Function vs-sf" error message will be displayed. To avoid the error:

1. Open an existing Simulink model and open Simulink Library Browser. Delete CarSim S-Function in the Simulink model and replace the one in the Library Browser.
2. Right click the newly replaced CarSim S-Function block and click **Library Link** > **Disable Link** to disable the link.

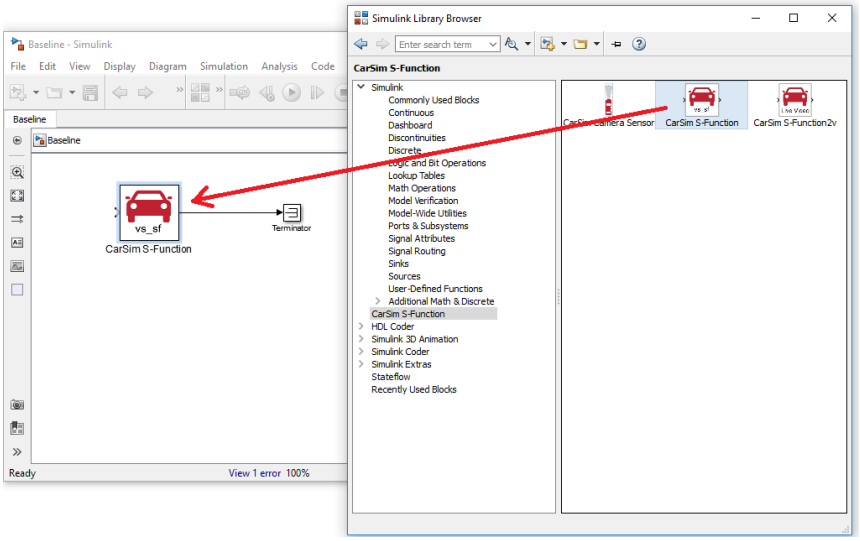


Figure 46. Replace CarSim S-Function.

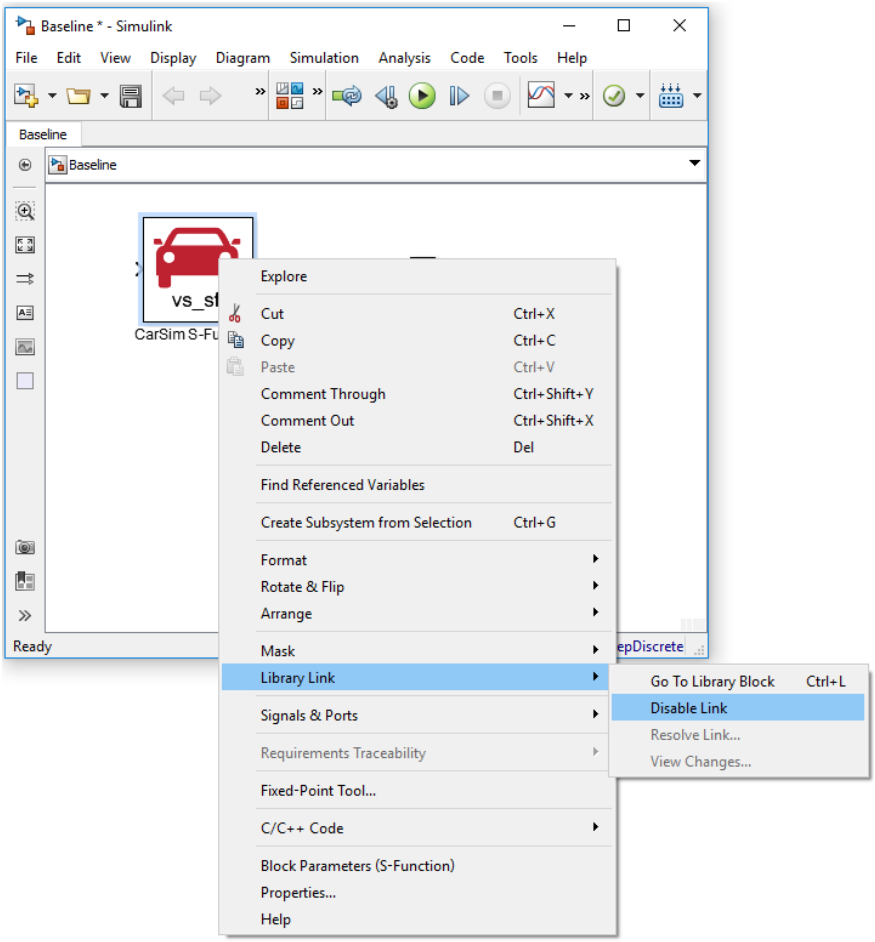


Figure 47. Disable Link.

Build the COSYM Model

1. Now you can build the COSYM model.
2. Go back to the COSYM Operator and Double-Click ‘System’ to open the system diagram view in COSYM.
3. Right-Click ‘Control’ and select ‘Create Model,’ then select RTIO model from the drop down. Name the RTIO model ‘RTIO’ and create it.
4. Drag this RTIO model into the System diagram.
5. From the CarSim Browser, click **Open** to open the baseline model.
6. In Simulink, select Tools -> System Configuration Parameters. In the Code Generation section, ensure the licrt.trc is selected. Close this options window then **Ctrl+B** to build the model and generate the necessary .lmd model for COSYM. The lmd will be populated into the folder: `{CarSim Data folder}\ETAS_Examples\ABS_Example\ABS\Library\Control\ABS\abs_MP_CS9_LABCAR_rtw`

7. Right-Click ‘Control’ and select ‘Import Model,’ then browse for the .lmd file generated in the last step.
8. Drag this model into the System diagram.

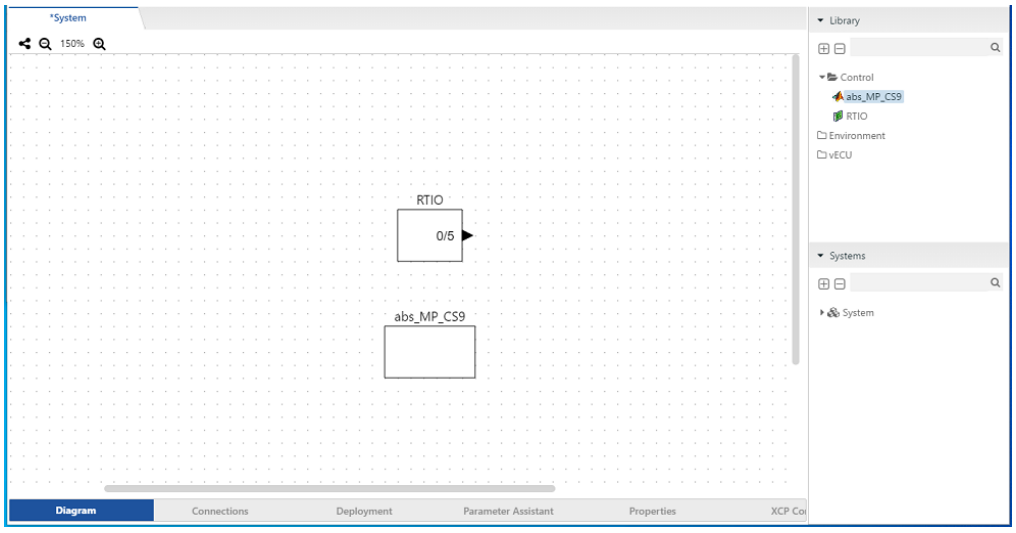


Figure 48. COSYM System Diagram.

9. In the COSYM menu bar, select Project -> Build and Run. Select the HiL target to ‘Build’ on and if prompted, allow COSYM to automatically ‘Map and Build’ the project. This process will launch Experiment Environment.

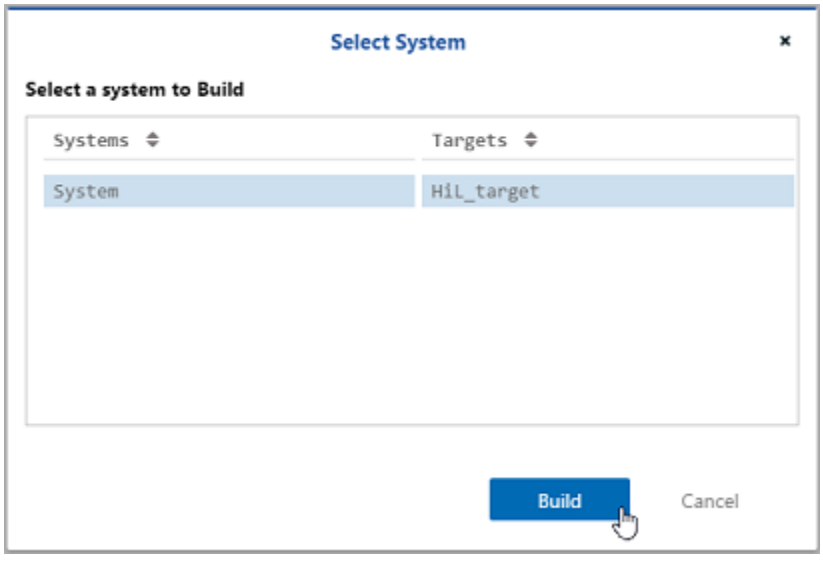


Figure 49. Build on HiL target.

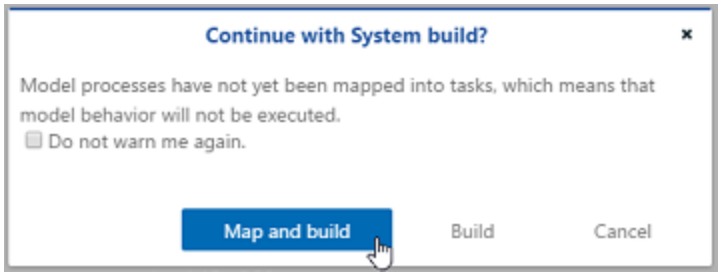


Figure 50. Map and Build.

Run the Simulation

1. In the CarSim Run Control screen, change **Number of live animators for this run:** to 1.
2. Click the **Send** button. You should see the animator program prepare for live animation.
3. In the Experiment Environment, click the Download icon  (or use the Experiment menu Download->Labcar) to download the experiment to the target PC.
4. Start running by clicking the icon  . When the model is running, you should see the vehicle moving via live animation. Click the icon  to stop the run.
5. After stopping the run, the results are automatically transferred back to the host. If there are any problems, you can manually transfer results back to the host by clicking the **Receive** button on the CarSim Run Control screen. However, this is normally not necessary.

In CarSim version 2019.1, the `vs_state` parameter was introduced, and by using this parameter, the button control can be added like the one shown at the top right corner in Figure 51. The Baseline project which is included in the shipping cpar file contains this button as an example. The parameter `vs_state` cannot be used with FMU models.

The value of the parameter `vs_state` is defined as:

- **Pause: 1**
- **Run: 2**

The user can pause/run simulation by using this button. To stop the simulation, press the stop icon  .

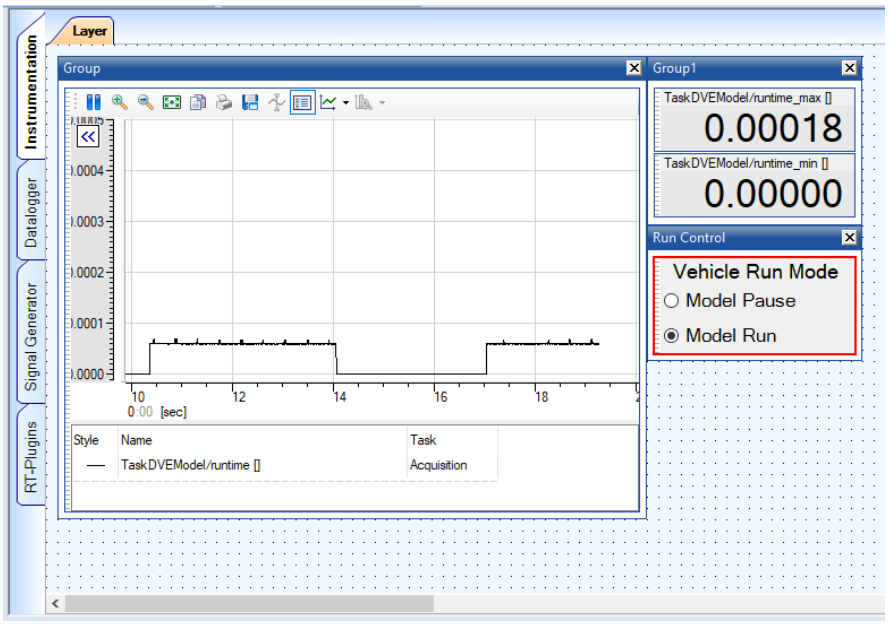


Figure 51. Vehicle Model Pause/Run Option.

Note In Experiment Environment, the icons  or  are for the system run control. When the stop icon is pressed, the whole system stops. Using Vehicle Model Pause/Run from vs_state parameter, it will run/pause just vehicle model, and other hardware/software in the system can still be running.

COSYM Tutorial: Create & Run an FMU/FMI RT Project

The Functional Mockup Unit (FMU) is a function component which implements with the interface called FMI. It consists of one zip-file with the ".fmu" extension and contains all necessary components to utilize the FMU. The FMI standard consists of two main parts: a) FMI for Model Exchange and b) FMI for Co-Simulation. The first version, FMI 1.0, was published in 2010, followed by FMI 2.0 in July 2014. Starting with version 5.3.0, LabCar can import FMI Co-Simulation version 1.0 (FMI 1.0).

CarSim/TruckSim/BikeSim, started supporting FMI version 1.0 and 2.0 in Co-Simulation mode on the Windows platform in version 2016.0. We pack the vehicle solvers as an FMU. In version 2017.0, we extended our vehicle solvers FMU to run on the LabCar RT target. On the RT system, LabCar supports Co-Simulation FMI 1.0. The FMU module allows you to run in real-time with hardware in the loop without Matlab/Simulink/Simulink Coder (RTW). You can directly link modules based on FMU or hardware I/O to the vehicle model.

Set Up a COSYM Project

Start by creating a new LabCar project.

1. The Simulink we will be using for this example is located in: `{CarSim Data folder}\ETAS_Examples\ Simulink_Models\Baseline\`

   The name of the Simulink model is Base_Model.mdl. If you do not see this directory, be sure to follow "Importing Examples Datasets" instruction.

2. Launch COSYM Operator. From the menu, select Create New Project to make the new project. Name the Project and provide the folder Location. For this example we will name the Project 'FMU' and designate the folder: `{CarSim Data folder}\ETAS_Examples\FMU_Example`.

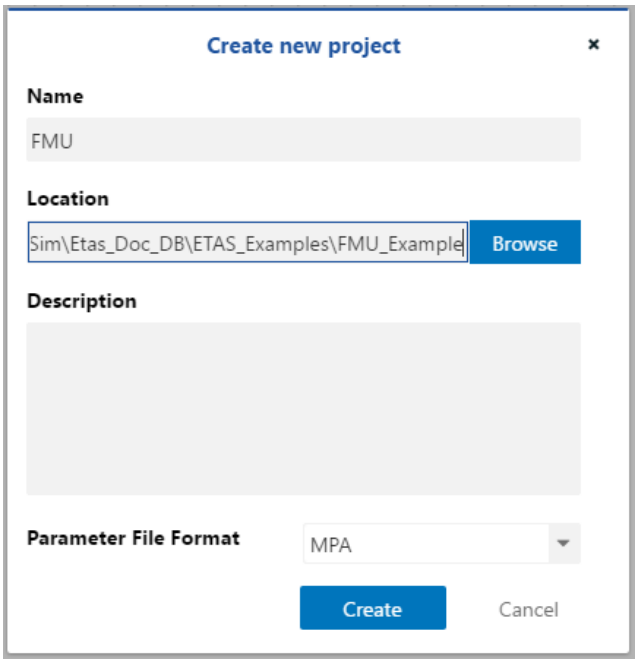


Figure 52. Create COSYM Project.

Prepare CarSim Datasets

1. Launch CarSim and select **Datasets** > *** * Quick Start Guide** > **Baseline**. Press the **Duplicate** icon. When the Duplicate Dataset window pops up, enter `RT: ETAS` in **Category for new dataset**. You can leave the title as `Baseline #1`.

2. Use the **Models** drop-down list to specify the library **Models: Transfer to Remote RT** (Figure 53).

3. Use the dataset select control under the **Models** control and choose the option **Link to New Dataset** (Figure 54). Give the new dataset a name and category, such as `COSYM Target` as a category and `Baseline FMU` as a title.

4. Click the blue button for the new link to go to the newly created dataset (Figure 55).

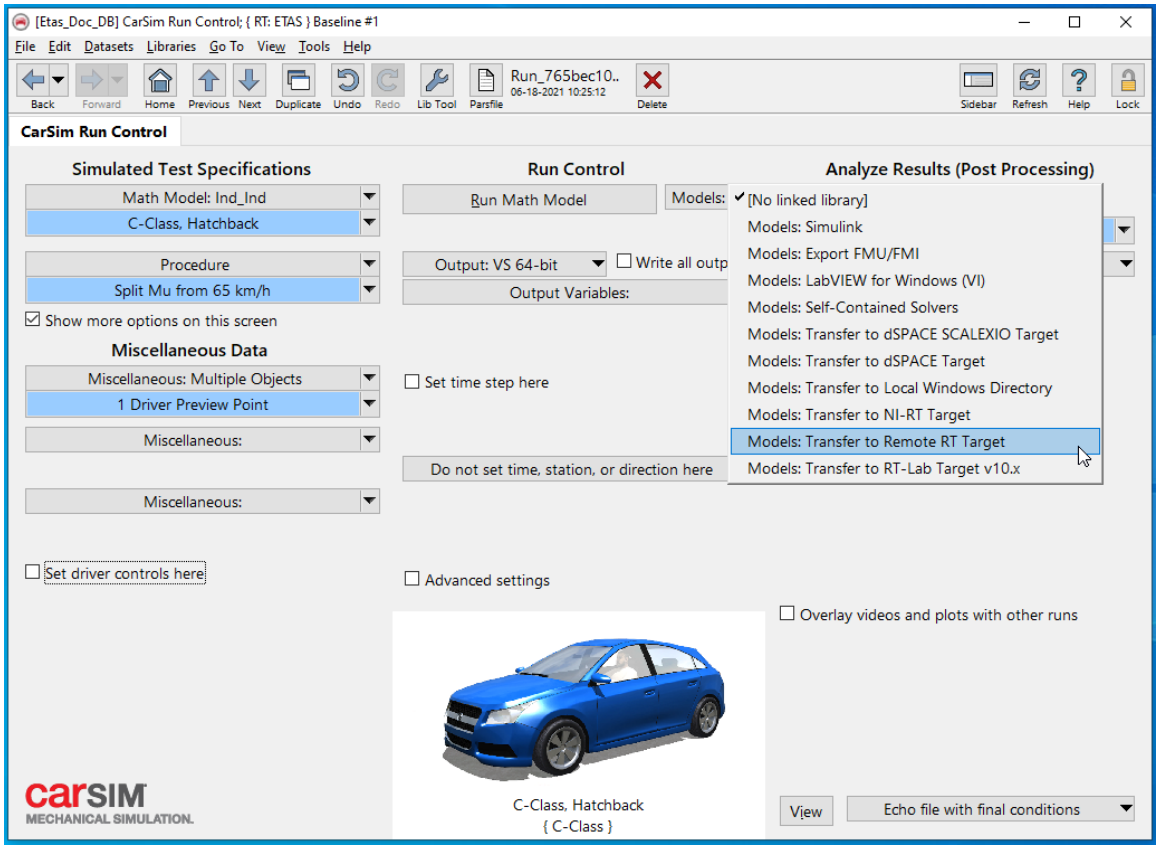


Figure 53. CarSim Run Control: change the linked Models library.

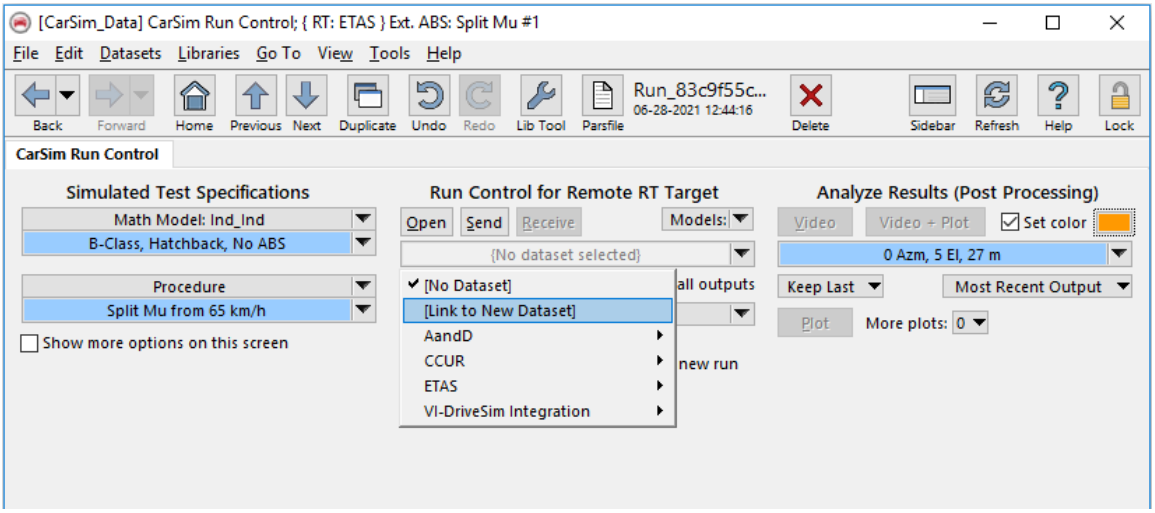


Figure 54. Make a new dataset in the Models: Transfer to Remote RT Target.

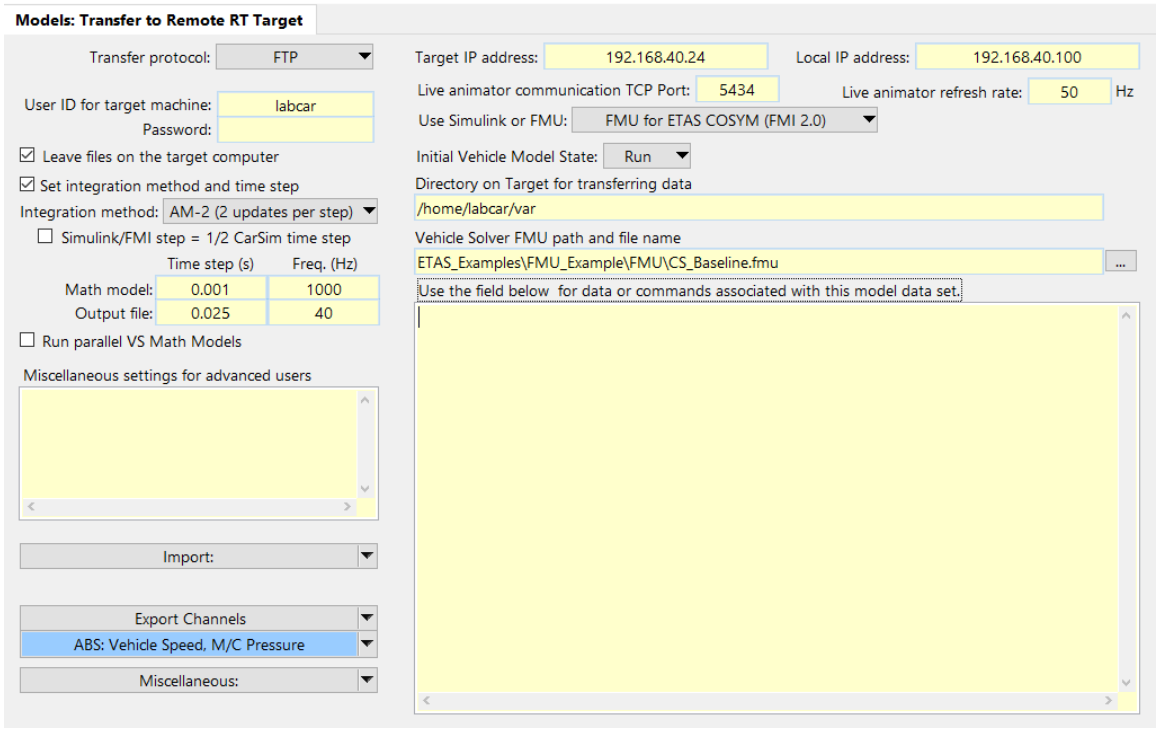


Figure 55. FMU model screen settings.

5. In the Models: Transfer to Remote RT Target dataset screen. In **Use Simulink or FMU** pull down menu, select FMU for LabCar (FMI 1.0). In **Vehicle Solver FMU path and file name**, enter:

   ```
   {CarSim_Datafolder}\ETAS_Examples\FMU_Example\FMU\
   vs_fmu1.fmu
   ```

   (This is the directory where you created a new LabCar project.)

   If an FMU name other than **`vs_fmi1.fmu`** is used, you must recreate the shared object. Please refer to Function Mock-up Unit (FMU) of LABCAR-OPERATOR User's Guide. For this example, it is recommended that you leave the file name as vs_fmi1.fmu

6. Go back to the Run Control Screen. The control button will change from **Open** to **Generate FMU**.

7. Click **Generate FMU** to generate the FMU for this setting. After the FMU files are generated, a confirmation message pops up (Figure 57).

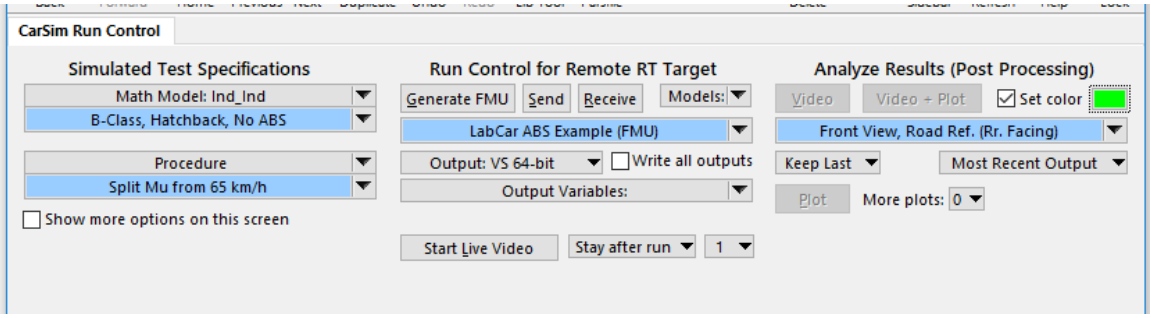
8. 

Figure 56. FMU Run Control screen.

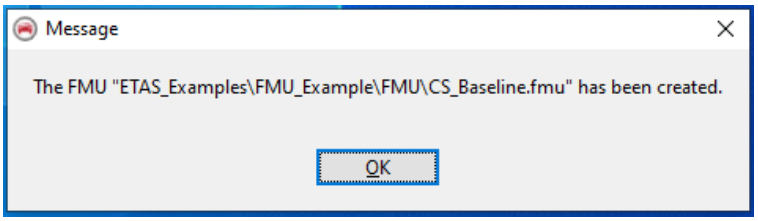


Figure 57. Confirmation message after generating FMU.

Build the COSYM Model

Now you can build the COSYM model.

1. Go back to the COSYM Operator and Double-Click ‘System’ to open the system diagram view in COSYM.
2. Right-Click ‘Control’ and select ‘Create Model,’ then select RTIO model from the drop down. Name the RTIO model ‘RTIO’ and create it.
3. Drag this RTIO model into the System diagram.
4. From the CarSim Browser, click **Open** to open the baseline model.

Right-Click ‘Control’ and select ‘Import Model,’ then browse for the fmu file generated:
`{CarSim Data folder}\ETAS_Examples\FMU_Example\FMU\vs_fmu1.fmu`

5. Drag this model into the System diagram.

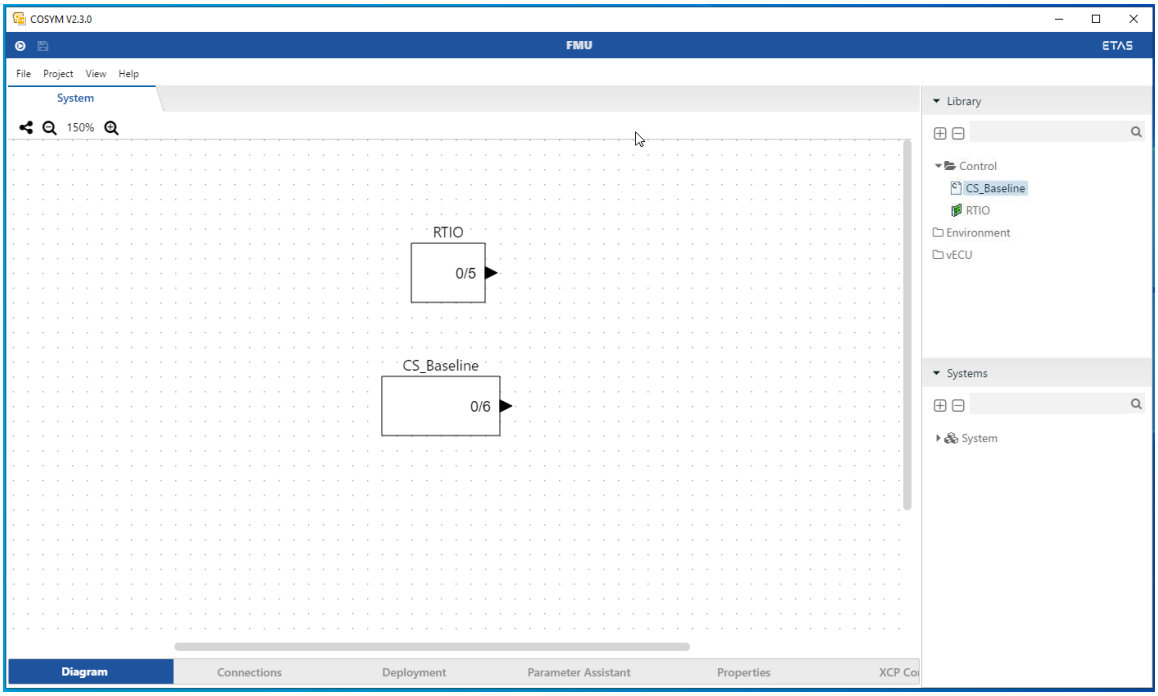


Figure 58. COSYM System Diagram.

6. In the COSYM menu bar, select Project -> Build and Run. Select the HiL target to 'Build' on (Figure 59) and if prompted, allow COSYM to automatically 'Map and Build' the project (Figure 60). This process will launch Experiment Environment.

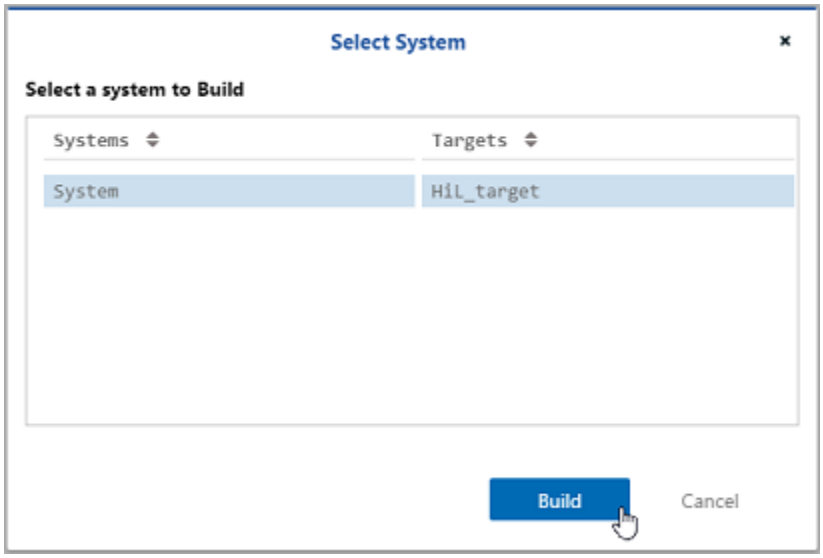


Figure 59. Build on HiL target.

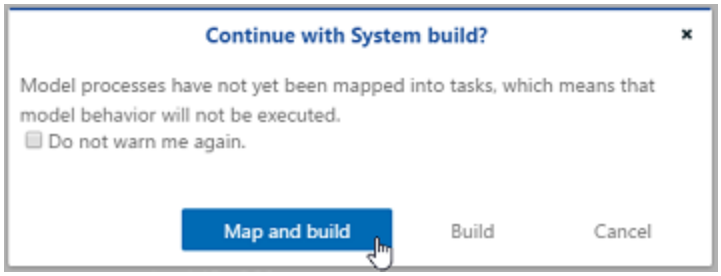


Figure 60. Map and Build.

Run the Simulation

1. In the CarSim Run Control screen, change **Number of live animators for this run:** to 1.
2. Click the **Send** button. You should see the animator program prepare for live animation.
3. In the Experiment Environment, click the Download icon (or use the Experiment menu Download->Labcar) to download the experiment to the target PC.
4. Start running by clicking the icon . When the model is running, you should see the vehicle moving via live animation. Click the icon to stop the run.
5. After stopping the run, the results are automatically transferred back to the host. If there are any problems, you can manually transfer results back to the host by clicking the Receive button on the CarSim Run Control screen. However, this is normally not necessary.

In CarSim version 2019.1, the `vs_state` parameter was introduced, and by using this parameter, the button control can be added. The Baseline project which is included in the shipping cpar file contains this button as an example. The parameter `vs_state` cannot be used with FMU models.

The value of the parameter `vs_state` is defined as:

- **Pause: 1**
- **Run: 2**

The user can pause/run simulation by using this button. To stop the simulation, press the stop icon .

> **Note** In Experiment Environment, the icons or are for the system run control. When the stop icon is pressed, the whole system stops. Using Vehicle Model Pause/Run from vs_state parameter, it will run/pause just vehicle model, and other hardware/software in the system can still be running.

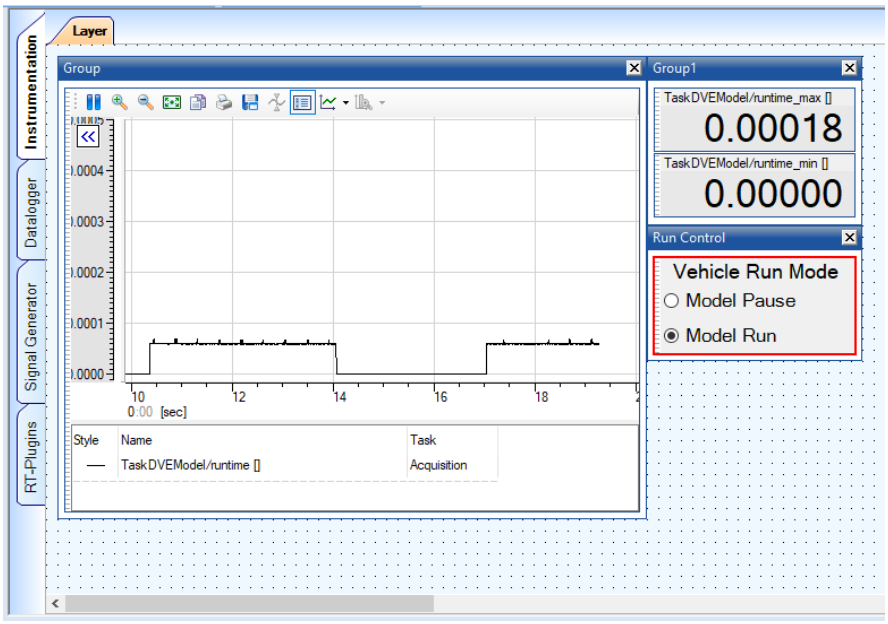


Figure 61. Vehicle Model Pause/Run Option.