ASM Vehicle Dynamics

# Model Description

Release 2021-A – May 2021

dSPACE

## How to Contact dSPACE


| | |
|---|---|
| Mail: | dSPACE GmbH<br>Rathenaustraße 26<br>33102 Paderborn<br>Germany |
| Tel.: | +49 5251 1638-0 |
| Fax: | +49 5251 16198-0 |
| E-mail: | info@dspace.de |
| Web: | http://www.dspace.com |


## How to Contact dSPACE Support

If you encounter a problem when using dSPACE products, contact your local dSPACE representative:

- Local dSPACE companies and distributors: http://www.dspace.com/go/locations
- For countries not listed, contact dSPACE GmbH in Paderborn, Germany.
  Tel.: +49 5251 1638-941 or e-mail: support@dspace.de

You can also use the support request form: http://www.dspace.com/go/supportrequest. If you are logged on to mydSPACE, you are automatically identified and do not need to add your contact details manually.

If possible, always provide the relevant dSPACE License ID or the serial number of the CmContainer in your support request.

## Software Updates and Patches

dSPACE strongly recommends that you download and install the most recent patches for your current dSPACE installation. Visit http://www.dspace.com/go/patches for software updates and patches.

## Important Notice


This publication contains proprietary information that is protected by copyright. All rights are reserved. The publication may be printed for personal or internal use provided all the proprietary markings are retained on all printed copies. In all other cases, the publication must not be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form, in whole or in part, without the prior written consent of dSPACE GmbH.

© 2006 - 2021 by:
dSPACE GmbH
Rathenaustraße 26
33102 Paderborn
Germany

This publication and the contents hereof are subject to change without notice.

AUTERA, ConfigurationDesk, ControlDesk, MicroAutoBox, MicroLabBox, SCALEXIO, SIMPHERA, SYNECT, SystemDesk, TargetLink and VEOS are registered trademarks of dSPACE GmbH in the United States or other countries, or both. Other brand names or product names are trademarks or registered trademarks of their respective companies or organizations.

# Contents


## About This Document 7

## Introduction 9

Introduction 9

## Tutorials 11

Handling the Model 12
- How to Open an ASM Library 12
- How to Start with an ASM Demo Model via MATLAB 14
- How to Start with an ASM Demo Model Using ModelDesk 16
- How to Open a Parameterization Project 17
- How to Parameterize ASM via ModelDesk 18
- How to Plot Simulation Results in ModelDesk 22
- How to Switch the Library Type 25
- How to Handle Multi-Instances 25
- How to Display the ModelDesk Parameter Group 27

Simulating in Simulink 28
- How to Use the Simulink Accelerator Mode 29
- How to Simulate the Model and Observe the Results in Simulink 29
- How to Handle Simulink Plotters 30
- How to Update a Plot Configuration for ModelDesk for a Simulink Simulation 31
- How to Configure a Plotter in ModelDesk for a Simulink Simulation 33
- How to Animate the Simulation Results 35
- How to Change Tire and Transmission Models via ModelDesk 37
- How to Change the Suspension Kinematics Model 39
- How to Change the Drivetrain Configurations 45

Simulating on dSPACE Platforms 51
- How to Generate an OSA File for VEOS 52
- Generating the Real-Time Application for SCALEXIO 54
- How to Generate a Real-Time Application for PHS-Bus Based Platforms 54
- How to Animate the Simulation Results in MotionDesk 55
- How to Observe Results in ControlDesk 56

How to Update a Plot Configuration for ModelDesk for a dSPACE Platform ... 59
Changing the Simulation Platform ... 60
How to Use the ModelDesk Project for the Real-Time Simulation ... 60
How to Change a Parameter in ModelDesk and Observe the Results in MotionDesk and ControlDesk ... 61

Interaction Between Drivetrain Blocks and Stabilization ... 63
Interaction Between Drivetrain Blocks and Stabilization ... 63

## Troubleshooting 67

Troubleshooting in MATLAB/Simulink ... 67
Troubleshooting in ModelDesk ... 68
Troubleshooting in ControlDesk ... 69
Troubleshooting in MotionDesk ... 69
Troubleshooting with Rising Turnaround Times at Steady Operating Point ... 70

## Appendix 73

General Information ... 74
Additional Information ... 74

Roads ... 75
ABSTrack Road ... 76
FourwayJunction_2Lanes Road ... 78
FourwayJunction_4Lanes Road ... 80
HandlingTrack Road ... 82
Highway_A33_Paderborn Road ... 84
HockenheimringTrack Road ... 86
LaneChange Road ... 87
MueSplit Road ... 89
MueSplitCurve Road ... 91
NCAP_AEB Road ... 93
NCAP_LSS Road ... 94
OvalWide Road ... 96
RoadNetwork_Highway_Interchange_PB Road ... 97
RoadNetworkBasic Road ... 99
RoadNetworkRural Road ... 101
RoadNetwork_Urban Road ... 103
RoadWithParkingLane Road ... 104
RoadWorks_Basic Road ... 106
RoadWorks_Highway Road ... 108

Slalom Road ..... 110
StraightLong Road ..... 111
TestGround Road ..... 113
ThreeWayJunction_2Lanes Road ..... 115
TrafficDriverBasic Road ..... 117

Scenarios ..... 119
ABSBrake Scenario ..... 120
ACCPedalStimulus100 Scenario ..... 122
Fishhook Scenario ..... 124
FMVSS_126_SIS Scenario ..... 126
FMVSS_126_SWD Scenario ..... 128
FollowRoadEndless Scenario ..... 129
FTP75 Cycle Scenario ..... 131
LaneChange Scenario ..... 132
ManualDriving_WithRoad Scenario ..... 134
MueSplit Scenario ..... 135
MueSplitCurve Scenario ..... 137
NCAP_LSS Scenario ..... 138
SineSteer Scenario ..... 140
Slalom Scenario ..... 142
SteadyStateCornering Scenario ..... 144
StepSteer Scenario ..... 146

Index 149

# About This Document

**Content**

This tutorial introduces you to the handling of an ASM Vehicle Dynamics system. The main focus is on using the tools and the communication between them, either in Simulink simulation or in real-time simulation on dSPACE platforms.

**Symbols**

dSPACE user documentation uses the following symbols:

| Symbol | Description |
|---|---|
| DANGER | Indicates a hazardous situation that, if not avoided, will result in death or serious injury. |
| WARNING | Indicates a hazardous situation that, if not avoided, could result in death or serious injury. |
| CAUTION | Indicates a hazardous situation that, if not avoided, could result in minor or moderate injury. |
| NOTICE | Indicates a hazard that, if not avoided, could result in property damage. |
| Note | Indicates important information that you should take into account to avoid malfunctions. |
| Tip | Indicates tips that can make your work easier. |
| | Indicates a link that refers to a definition in the glossary, which you can find at the end of the document unless stated otherwise. |
| | Precedes the document title in a link that refers to another document. |

**Naming conventions**

dSPACE user documentation uses the following naming conventions:

**%name%** Names enclosed in percent signs refer to environment variables for file and path names.

< > Angle brackets contain wildcard characters or placeholders for variable file and path names, etc.

**Special folders**

Some software products use the following special folders:

**Common Program Data folder** A standard folder for application-specific configuration data that is used by all users.

```
%PROGRAMDATA%\dSPACE\<InstallationGUID>\<ProductName>
```

or

```
%PROGRAMDATA%\dSPACE\<ProductName>\<VersionNumber>
```

**Documents folder** A standard folder for user-specific documents.

```
%USERPROFILE%\Documents\dSPACE\<ProductName>\
<VersionNumber>
```

**Local Program Data folder** A standard folder for application-specific configuration data that is used by the current, non-roaming user.

```
%USERPROFILE%\AppData\Local\dSPACE\<InstallationGUID>\
<ProductName>
```

**Accessing dSPACE Help and PDF Files**

After you install and decrypt dSPACE software, the documentation for the installed products is available in dSPACE Help and as PDF files.

**dSPACE Help (local)** You can open your local installation of dSPACE Help:

- On its home page via Windows Start Menu
- On specific content using context-sensitive help via **F1**

**dSPACE Help (Web)** You can access the Web version of dSPACE Help at www.dspace.com/go/help.

To access the Web version, you must have a *mydSPACE* account.

**PDF files** You can access PDF files via the icon in dSPACE Help. The PDF opens on the first page.

# Introduction

## Introduction

**Demo model**

The ASM Vehicle Dynamics model contains the following components:

- Vehicle Dynamics
- Engine Basic
- Drivetrain
- Environment
- Soft ECU

**Vehicle Dynamics** The Vehicle Dynamics model contains:

- The vehicle structure
- The wheel suspension compliance and kinematics
- The steering system
- The brakes
- The tires

  You can select between two tire models: TMeasy and Magic Formula

The model contains different preconfigured vehicle categories, such as *SUV*, *MidSizeCar*, and *SmallCar*.

**Engine Basic** The Engine Basic model is a map-based engine model. It uses the accelerator pedal position and the engine speed as input and outputs the torque value.

**Drivetrain** The Drivetrain model contains:

- The starter
- The crankshaft
- The transmission system

  You can select between two transmission systems: *stepped automatic transmission* and *manual transmission*.
- The wheel drive

  You can select between different wheel drives: *All wheel drive*, *front wheel drive*, and *rear wheel drive*.

**Environment** The Environment model contains the Driver. It has the following functions:

- Driving the vehicle according to lateral and longitudinal specifications from a scenario.
- Gear selection in the case of manual transmission.

The model contains multiple predefined scenarios and road models, such as *MueSplitCurve*, *ABSBrake*, and *NCAP_LSS*.

**Tutorial**

The tutorial is based on the ASM Vehicle Dynamics demo project which is included in the installation. This will be the starting point for working with the ASM system. You will create a new ASM Vehicle Dynamics project based on this demo. You will simulate, parameterize, and animate the new project in Simulink simulation as well as in real-time simulation on dSPACE platforms.

The tools which might be in use are:

**ASM Vehicle Dynamics Model** Simulink model of ASM Vehicle Dynamics used for simulation and code generation

**ModelDesk** For parameterizing the ASM Vehicle Dynamics Model (see Introduction to ModelDesk (ModelDesk Basics 🕮))

**MotionDesk** For animating the ASM Vehicle Dynamics simulation data (see Introduction to MotionDesk (MotionDesk Basics 🕮))

**ControlDesk** For interactive work with the ASM Vehicle Dynamics real-time simulation, for example, start/stop maneuver, measuring of vehicle data, etc. (see Introduction to ControlDesk (ControlDesk Introduction and Overview 🕮))

# Tutorials

**Where to go from here**

Information in this section

Handling the Model.....12
Provides information on how to work with the model.

Simulating in Simulink.....28
You can simulate an ASM model in Simulink.

Simulating on dSPACE Platforms.....51
You can simulate an ASM model on dSPACE platforms.

Interaction Between Drivetrain Blocks and Stabilization.....63
In case of multi-instance use in an ASM model, it might be required to adapt the connections between flexible drivetrain blocks and the numerical stabilization.

# Handling the Model

**Where to go from here**

Information in this section

How to Open an ASM Library.....12
After successfully installing ASM on your PC, you can now easily open an ASM library.

How to Start with an ASM Demo Model via MATLAB.....14
After successfully installing ASM on your PC, you can easily start an example model of your ASM installation via MATLAB.

How to Start with an ASM Demo Model Using ModelDesk.....16
After successfully installing ASM on your PC, you can now easily start a demo model of your ASM installation using ModelDesk.

How to Open a Parameterization Project.....17
You can open a parameterization project from the model.

How to Parameterize ASM via ModelDesk.....18
You can parameterize ASM with ModelDesk.

How to Plot Simulation Results in ModelDesk.....22
You can plot simulation results in ModelDesk via the SignalInterfaces.

How to Switch the Library Type.....25
You can switch between the developer library and the operator library.

How to Handle Multi-Instances.....25
The Multi-Instance Overview button opens a dialog to get an overview of all multi-instance blocks in the model and to set the multi-instance parameters.

How to Display the ModelDesk Parameter Group.....27
You can modify the grouping in the parameter set that is displayed in the project tree of your ModelDesk experiment.

## How to Open an ASM Library

**Objective**

After successfully installing ASM on your PC, you can now easily open an ASM library.

This lesson explains how to open the Gasoline Engine library. You can start all ASM products in a similar way. The MATLAB Command Window shows you all the products that are installed and covered by your license.

**Method**

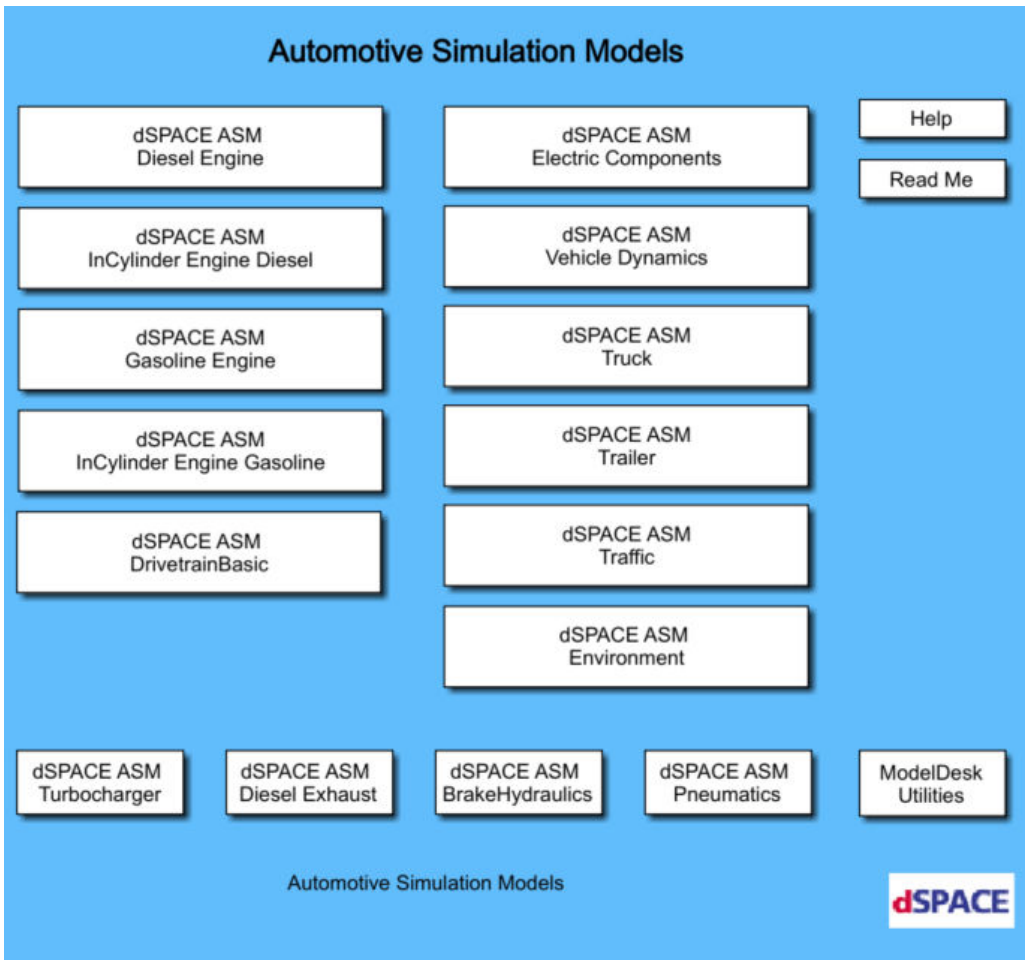
**To open an ASM library**

1 Start a MATLAB session.

When MATLAB starts, the MATLAB Command Window lists all the ASM blocksets that are installed on the PC.

2 Type `asm` in your MATLAB Command Window.

The central ASM library opens.




It provides an overview of all the accessible major ASM libraries:

- ASM Diesel Engine
- ASM InCylinder Engine Diesel
- ASM InCylinder Engine Gasoline
- ASM Gasoline Engine
- ASM Drivetrain Basic
- ASM Vehicle Dynamics
- ASM Electric Components
- ASM Trailer
- ASM Traffic

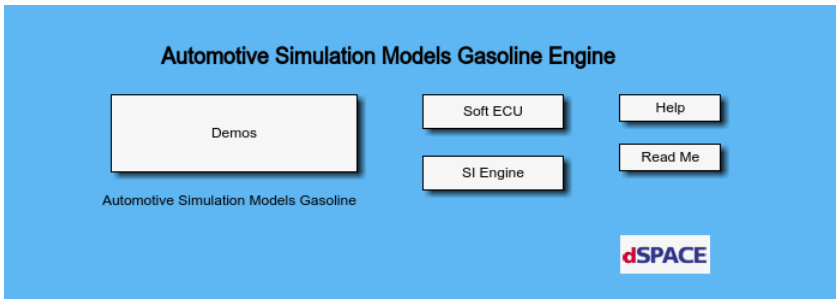
- ASM Truck
- ASM Environment

Depending on your software license, some of the libraries are grayed out. If a library is grayed, you do not have a valid license for it.

3 Double-click the **dSPACE ASM Gasoline Engine** subsystem.

The subsystem opens. Depending on the installed license, you can open the developer or operator library. For details, refer to Operator Version (ASM User Guide 🕮).

4 Double-click the **Developer** or **Operator** subsystem to open the library.




**Result**

The library opens. It is subdivided into three sections.

- On the left is a subsystem with all the demo models belonging to the library.
- The middle section contains the subsystems that are part of the library and which are contained in the example model as links.
- On the right are links to the documentation and to the last-minute information in the readme section.

**Related topics**

HowTos

How to Start with an ASM Demo Model via MATLAB..... 14

# How to Start with an ASM Demo Model via MATLAB

**Objective**

After successfully installing ASM on your PC, you can now easily start a demo model of your ASM installation via MATLAB.

This lesson explains how to start the Gasoline Engine model via MATLAB. You can start all ASM products in a similar way. The MATLAB Command Window shows you all the products that are installed and covered by your license.

**Methods to start with an ASM demo model**

You can start an ASM demo model with the following methods:

- Start an ASM demo model via MATLAB, see below.
- Generate an ASM project by using ModelDesk or by using MATLAB/Simulink. Refer to How to Create a Project Based on an ASM Demo (ModelDesk Project and Experiment Management 🕮).

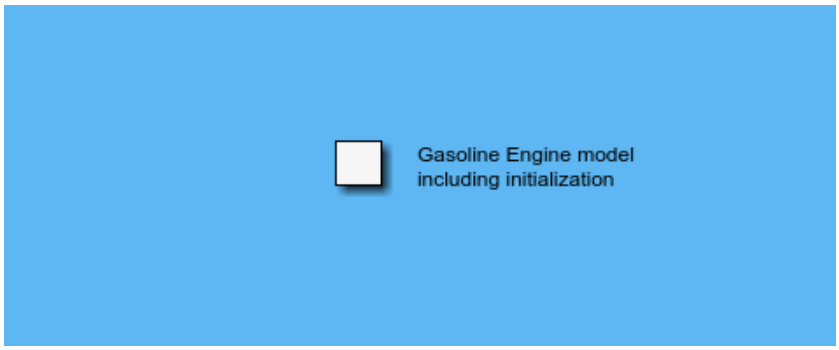
**Preconditions**

The ASM library is open. Refer to How to Open an ASM Library on page 12.

**Method**

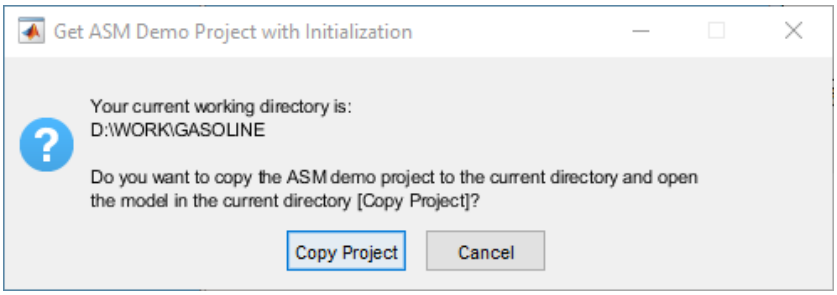
**To start with an ASM demo model via MATLAB**

1 Double-click the Demos subsystem to open a new window which presents all the demo models that are part of the library.




2 Double-click the demo model to start it.

This opens a dialog that offers you the possibility to copy the project to your current working directory.




3 Click Copy Project to copy the current demo folder to your current MATLAB work folder.

All the files related to the demo model are copied to the current MATLAB work folder. This includes the entire parameterization and the ControlDesk example layouts.

**Result**

Now the demo model opens. All the initialization files are executed and you can start simulation.

**Related topics**

HowTos

How to Create a Project Based on an ASM Demo (ModelDesk Project and Experiment Management 🕮)

# How to Start with an ASM Demo Model Using ModelDesk

**Objective**

After successfully installing ASM on your PC, you can now easily start a demo model of your ASM installation using ModelDesk.

**Methods to start with an ASM demo model**

You can start an ASM demo model with the following methods:

- Generate an ASM project by using ModelDesk or by using MATLAB/Simulink, see below.
- Start an ASM demo model via MATLAB. Refer to How to Start with an ASM Demo Model via MATLAB on page 14.

**ASM demos**

The ASM installation contains several ASM demos. The ASM demos include all the necessary files for the simulation, for example:

- Simulation model based on the ASM blocks
- Simulation applications for the simulation platforms
- ModelDesk project for parameterizing the model
- ControlDesk project for experimenting with the model
- MotionDesk project for animation (if useful)

**Preconditions**

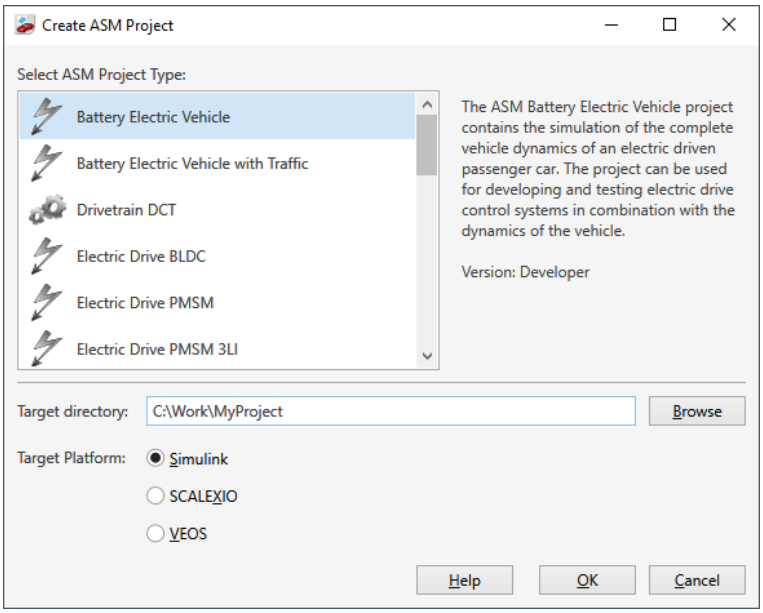
- No other project must be open.
- The license of the ASM library that is used by the ASM demos must be available.
- The ASM library must be decrypted.

**Method**

**To to start with an ASM demo model using ModelDesk**

1 On the File ribbon, click New – ASM Project.

The Define a Project dialog opens.




2 Select an ASM project type.

3 Specify the target directory. Select an empty directory or specify a new directory. You must have write permission to the directory.

4 Select the target platform to be activated in the experiment.

5 Click OK.

**Result**

ModelDesk copies all files of the selected ASM demo to the specified target folder and opens the project.

# How to Open a Parameterization Project

**Objective**

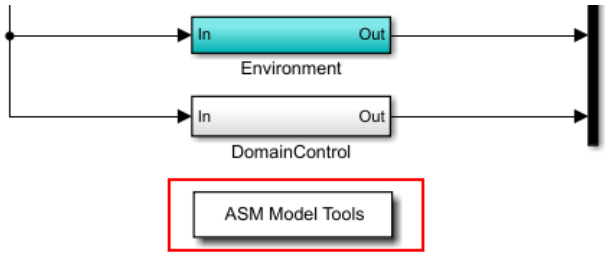
You can open a parameterization project from the model.

The parameterization project for the current model is accessible via the Open ModelDesk Project & Experiment button.

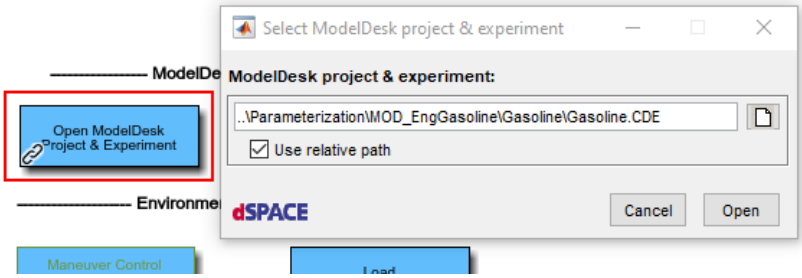
**Method**

**To open a parameterization project**

1 In the ASM model, double-click the ASM Model Tools button below the modules.




2 Double-click the Open ModelDesk Project & Experiment button.




3 In the dialog, click Open to open the selected ModelDesk project.

**Result**

ModelDesk is started and the selected project and experiment is loaded.

# How to Parameterize ASM via ModelDesk

**Objective**

The `<Projectfolder>\Parameterization` folder includes a complete ModelDesk project. You can perform the complete parameterization of the ASM model with ModelDesk.

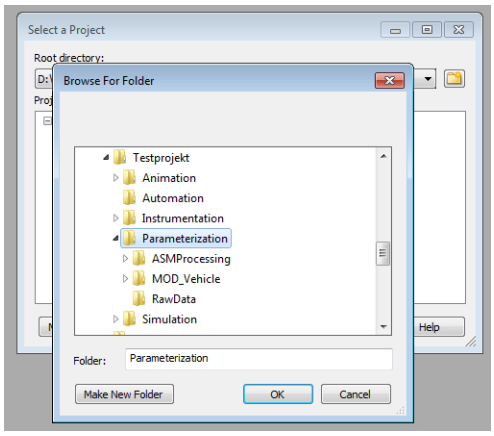
In the following method, an ASM Electric Components demo is used as an example.

**Method**

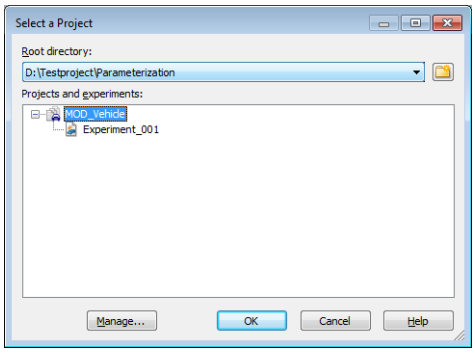
**To parameterize ASM via ModelDesk**

1 Start ModelDesk and open the project from `<Projectfolder>\Parameterization`.

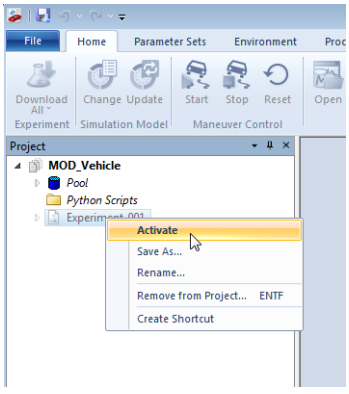
1. Click File – Open – Project + Experiment.
2. Define the root directory in `<Projectfolder>\Parameterization`.




3. Select the project.

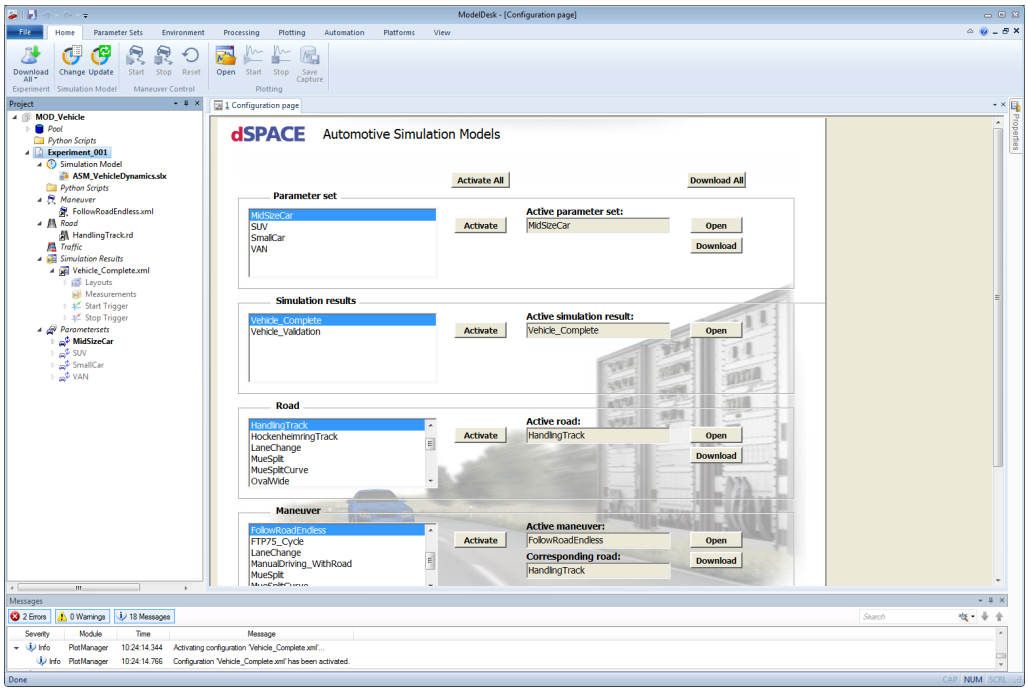



**2** Activate the demo experiment via context menu on the experiment in the Project Navigator.




The project structure is available in the Project Navigator and all parameters are available for configuration.

**3** Switch to the experiment's **Configuration** page.




The **Configuration** page of the experiment lets you select parameter sets for activating and downloading them to the simulation model or for activating and editing (**Open** button) them.

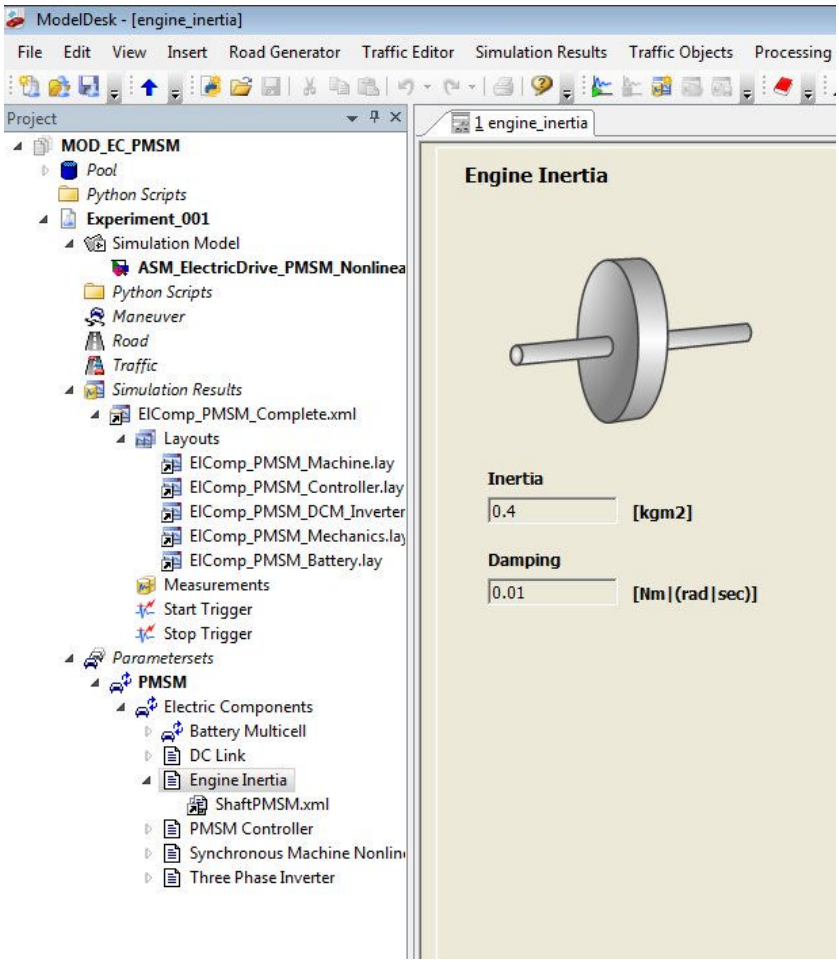
You can also activate plotting sets and open them to obtain simulation results in ModelDesk via the ModelDesk Plotting feature.

**4** Click **Activate** and then **Download** to activate and download the parameter set.

**5** Start a simulation (if possible in accelerator mode) and observe the results in the Simulink plotters, and/or ModelDesk plotting.

6 Modify, for example, the engine inertia in ModelDesk.

In the Project Navigator, select Parametersets – <ParameterSet name> – ElectricComponents – Engine Inertia.




7 Use the context menu of the current page to save and download the new setting or switch back to the main experiment page and download the current parameter setting.

8 Start a simulation (if possible in accelerator mode) and observe the results in the Simulink plotters, ModelDesk plotting, and/or MotionDesk.

9 Switch to the MATLAB workspace and look at the value of the engine inertia, `MDL.ElectricComponents.Mechanics.EngineInertia.Const_Inertia`. If you are working with the Simulink simulation, all parameters are available in the structure `MDL` in the MATLAB workspace and ModelDesk can download new values to the `MDL` structure.

```
>> MDL.ElectricComponents.Mechanics.EngineInertia.Const_Inertia
ans =
```

```
        Comment: 'Inertia'
         Author: '(null)'
         Origin: ' '
        Version: '2'
   LastModified: '16.01.2015 11:21'
          vName: 'Const_Inertia'
          vUnit: '[kgm2]'
              v: [0.4000]
```

**Result**

You parameterized the model in ModelDesk.

**Related topics**

Basics

Working with Parameter Sets (ModelDesk Parameterizing 🕮)

References

Settings (ModelDesk Road Creation 🕮)

# How to Plot Simulation Results in ModelDesk

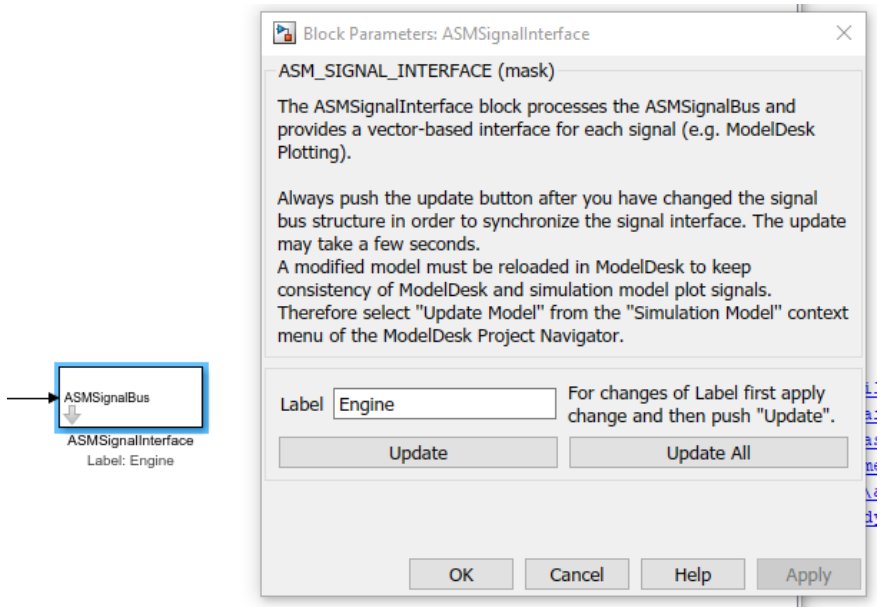
**Objective**

You can plot simulation results in ModelDesk via the SignalInterfaces.

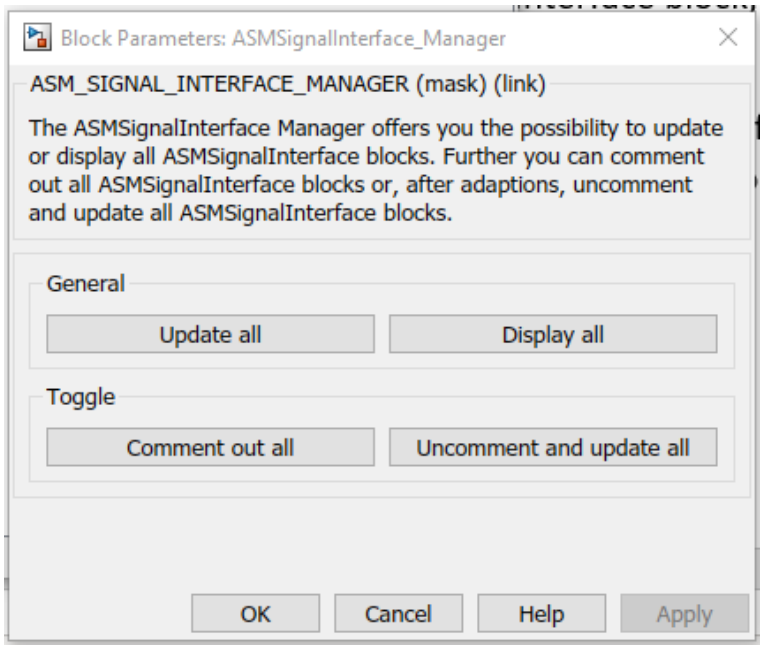
**Description**

You can use the **ASMSignalInterface** block and the ASMSignalInterface Manager to toggle the plotting.

**ASMSignalInterface block** Simulation data is transferred from the ASM model to ModelDesk via the **ASMSignalInterface** block. There is one block inside each ASM module. Refer to ASMSignalInterface (ASM User Guide 🕮).



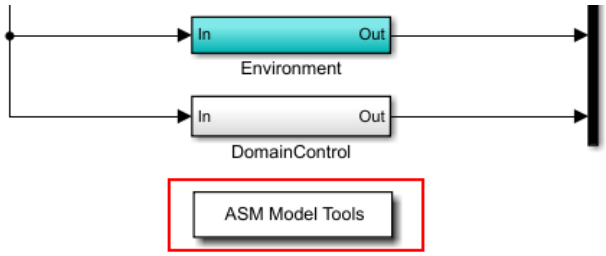

**ASMSignal Interface Manager** The ASMSignalInterface Manager collects the signals of all modules of your model. You can open it via the ASM Tools. Refer to ASMSignalInterface Manager (ASM User Guide 🕮).



**Method**

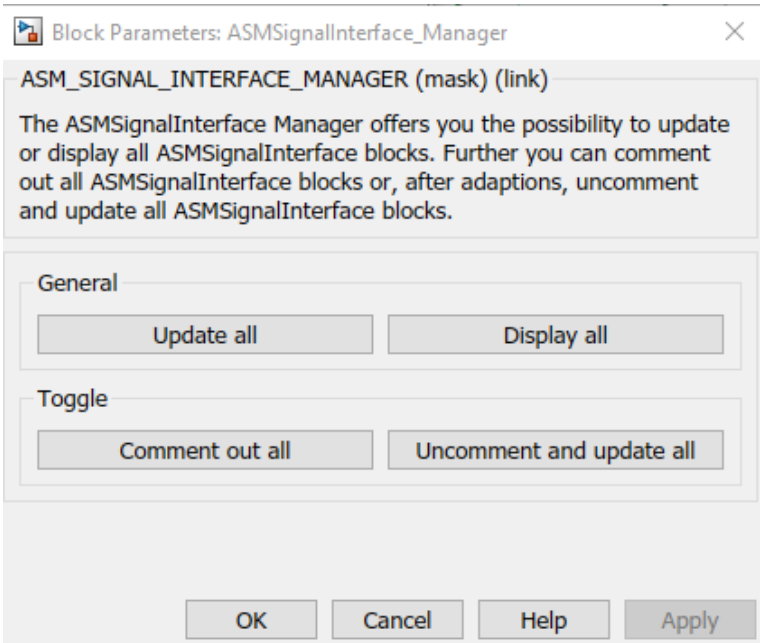
**To plot simulation results in ModelDesk**

1 In the ASM model, double-click the ASM Model Tools button below the modules.




2 Double-click the ASMSignalInterface Manager button and press one of the two buttons:

- Uncomment and update all: for plotting in ModelDesk.
- Comment out all: if you do not need plotting in ModelDesk. Commenting out the blocks will improve the simulation performance.




**Result**

You activated or deactivated plotting in ModelDesk.

**Related topics**

HowTos

How to Configure a Plotter in ModelDesk for a Simulink Simulation..... 33
How to Update a Plot Configuration for ModelDesk for a Simulink Simulation..... 31

# How to Switch the Library Type

**Objective**

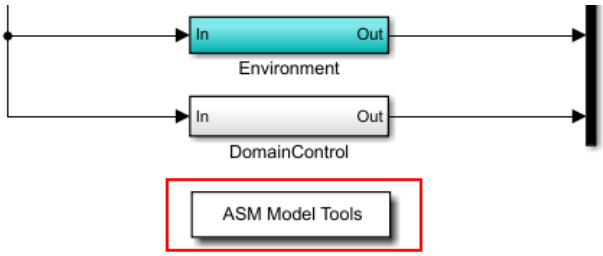
You can switch between the developer library and the operator library.

The developer library is a Simulink library containing the open-source ASM blocks. In the operator library, the ASM blocks are Simulink S-functions for offline simulation.

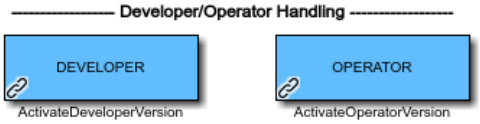
**Method**

**To switch the library type**

1 In the ASM model, double-click the ASM Model Tools button below the modules.




2 Depending on the library type, you want to switch to, double-click the DEVELOPER button or the OPERATOR button.




**Result**

You switched between the developer library and the operator library.

**Related topics**

References

Activate Developer Version (ASM User Guide 🕮)
Activate Operator Version (ASM User Guide 🕮)

# How to Handle Multi-Instances

**Objective**

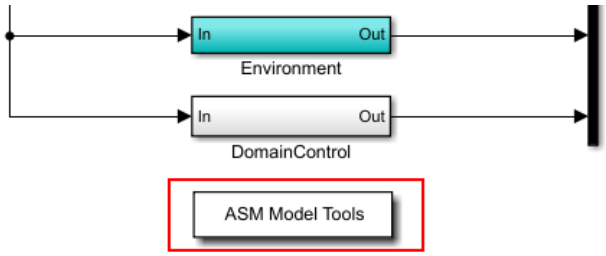
The Multi-Instance Overview button opens a dialog to get an overview of all multi-instance blocks in the model and to set the multi-instance parameters.

For more information on multi-instances, refer to Multi-Instance (ASM User Guide 🕮).

## Method

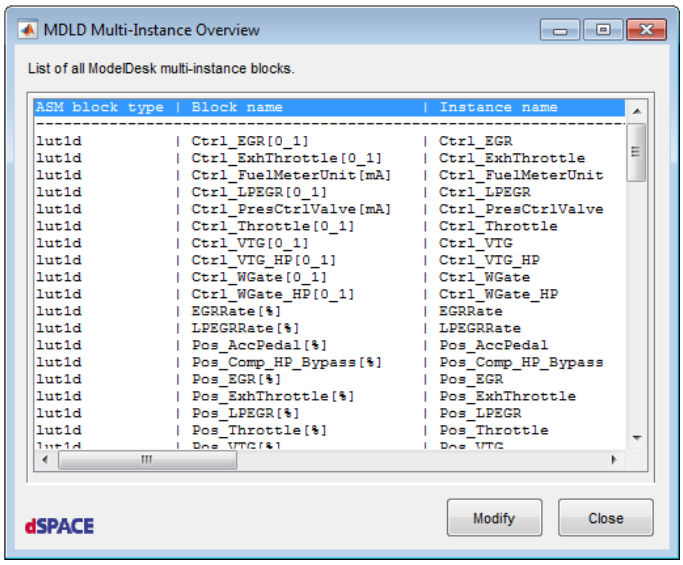
### To handle multi-instances

1 In the ASM model, double-click the ASM Model Tools button below the modules.




2 Double-click the Multi-Instance Overview button.

A dialog opens that displays the multi-instances in your model.




3 To modify a multi-instance, select it in the dialog and click Modify.

## Result

You modified multi-instances.

## Related topics

Basics

Multi-Instance (ASM User Guide 🕮)

# How to Display the ModelDesk Parameter Group

**Objective**

You can modify the grouping in the parameter set that is displayed in the project tree of your ModelDesk experiment.

For more information, refer to Basics on Grouping the View of a Parameter Set (ModelDesk Parameterizing 🕮).

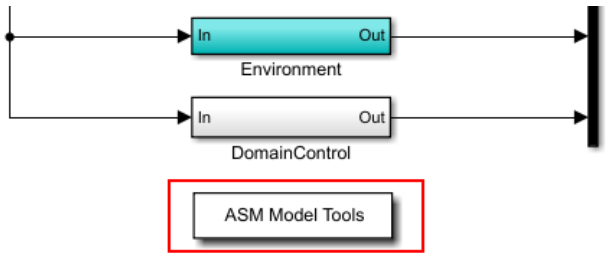
The **ModelDesk Parameter Group Overview** block in the ASM Tools opens an overview of the ModelDesk parameter group blocks that are contained in the simulation model.

For more information, refer to ModelDesk Parameter Group (ASM User Guide 🕮).

**Method**

**To display the ModelDesk parameter group**

1 In the ASM model, double-click the **ASM Model Tools** button below the modules.




2 Double-click the **ModelDesk Parameter Group Overview** button.

**Result**

The **MDLD Parameter Group Overview** dialog opens. It displays all ModelDesk parameter groups.

Select a group in the list and click **Open System** to open the group in your model.

**Related topics**

Basics

Basics on Grouping the View of a Parameter Set (ModelDesk Parameterizing 🕮)
ModelDesk Parameter Group (ASM User Guide 🕮)

# Simulating in Simulink

**Where to go from here**

Information in this section

How to Use the Simulink Accelerator Mode.....29
You can speed up the simulation by using the Simulink Accelerator mode.

How to Simulate the Model and Observe the Results in Simulink.....29
You can observe the simulation results in Simulink by using Simulink scopes.

How to Handle Simulink Plotters.....30
You can handle the Simulink plotters in your simulation model.

How to Update a Plot Configuration for ModelDesk for a Simulink Simulation.....31
If the ASMSignalBus is changed, you must update the plotting interface block and the model in ModelDesk.

How to Configure a Plotter in ModelDesk for a Simulink Simulation.....33
To plot a signal from the ASMSignalBus, you have to configure a project and an experiment in ModelDesk.

How to Animate the Simulation Results.....35
You can animate the behavior of the car during the simulation with MotionDesk. The demo model includes the block needed for preparing and sending the simulation data to MotionDesk.

How to Change Tire and Transmission Models via ModelDesk.....37
You can change tire model and transmission parameters via ModelDesk.

How to Change the Suspension Kinematics Model.....39
You can change a table-based front suspension kinematics model with the geometrically described MacPherson Strut.

How to Change the Drivetrain Configurations.....45
To change the drivetrain configuration, some subsystems of the **Drivetrain** model must be replaced and the correct drivetrain variant must be selected in ModelDesk.

# How to Use the Simulink Accelerator Mode

**Objective**

You can use the Simulink accelerator mode to speed up the simulation (Pentium 4 Processor with 2 GB RAM: up to 2.5 times faster than real time).

> **Note**
>
> You can perform all steps in this tutorial by using the accelerator mode.

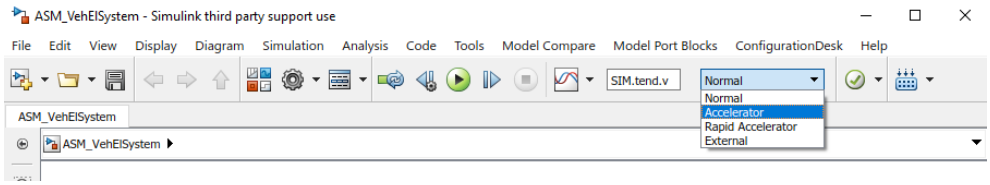
**Prerequisite**

You need a license for the Simulink Performance Tools.

**Method**

**To use the Simulink Accelerator mode**

1 In Simulink, activate the accelerator mode for the ASM demo model on the menu bar.




2 Start the simulation and observe the results on the Simulink plotters, and if available, in MotionDesk (perform a simulation of 200 s simulation time).

**Result**

The model is simulated in accelerator mode.

First, the build process for the accelerator mode is performed, which takes some time.

Whenever you perform an accelerator mode simulation and the model structure has not changed, no new code is generated, even if new parameters for the model are loaded, for example, from ModelDesk.

# How to Simulate the Model and Observe the Results in Simulink

**Starting simulation**

The demo model is prepared for a Simulink simulation. When you start simulation after you opened the demo model, a standard vehicle dynamics maneuver or engine driving cycle starts.

**Simulation results**

You can observe the simulation results in Simulink using Simulink scopes. The scopes are in the UserInterface/DISP_xyz subsystem of an ASM module. For example, you can find a scope in the Engine/Control/UserInterface/DISP_Control subsystem.

You can also watch the simulation results in oversampled model parts. For more information, refer to Watching the Simulation Results in Oversampled Model Parts (ASM Gasoline Engine InCylinder Model Description 🕮) or Watching the Simulation Results in Oversampled Model Parts (ASM Diesel Engine InCylinder Model Description 🕮).

**Method**

**To simulate the model and observe the results in Simulink**

1. In MATLAB, switch to `<Projectfolder>\Simulation` in the project and call `go` to open the model.
2. Start the simulation.
3. Open one of the scopes from the UserInterface subsystem, for example, in Engine/Control/UserInterface/DISP_Control. After a few seconds, look at the scope signals.

**Result**

You simulated the model and observed the results in Simulink.

## How to Handle Simulink Plotters

**Objective**

You can handle the Simulink plotters in your simulation model.

**Handling Simulink Plotters**

The SCOPE HANDLING GUI button opens a dialog that lets you perform the following actions:

- Disable: Comments out all Simulink plotters, displays, and xy-graphs in the model.
- Restore: Uncomments all Simulink plotters, displays, and xy-graphs in the model.
- Show: Shows a list with all the Simulink plotters, displays, and xy-graphs in the model.

The block makes directly use of the *commented* block property in Simulink. Disabled blocks are excluded from simulation and signals are terminated and grounded.

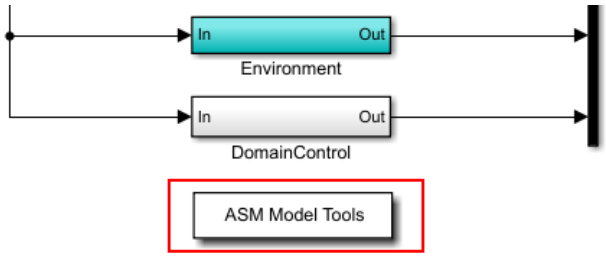
For ASM Engine Models: If you want to simulate your model with `asm_eng_testbench.m`, it is sufficient to store the simulation results in the variables under TEST_CONTROL or TEST_PLANT. No plotters are necessary.

The testbench script contains a postprocessing.

**Method**

**To handle Simulink plotters**

1 In the ASM model, double-click the ASM Model Tools button below the modules.




2 Double-click the SCOPE HANDLING GUI button.

The ASM Scope Handling GUI dialog opens.

3 In the ASM Scope Handling GUI dialog, click one of the buttons to handle Simulink plotters.

For information on the options, refer to Handling Simulink Plotters on page 30 above.

**Result**

You handled the Simulink plotters in your simulation model.

**Related topics**

References

Scope Handling User Interface (ASM User Guide 🕮)

# How to Update a Plot Configuration for ModelDesk for a Simulink Simulation

**Objective**

The ASM package includes a signal interface block for ModelDesk so that signals of the ASMSignalBus can be plotted. This block must be updated manually if the ASMSignalBus is changed.

If the respective simulation model is an operator version, it is mandatory to perform a manual update of the plotting block and plotting configuration, before using the plotting feature.

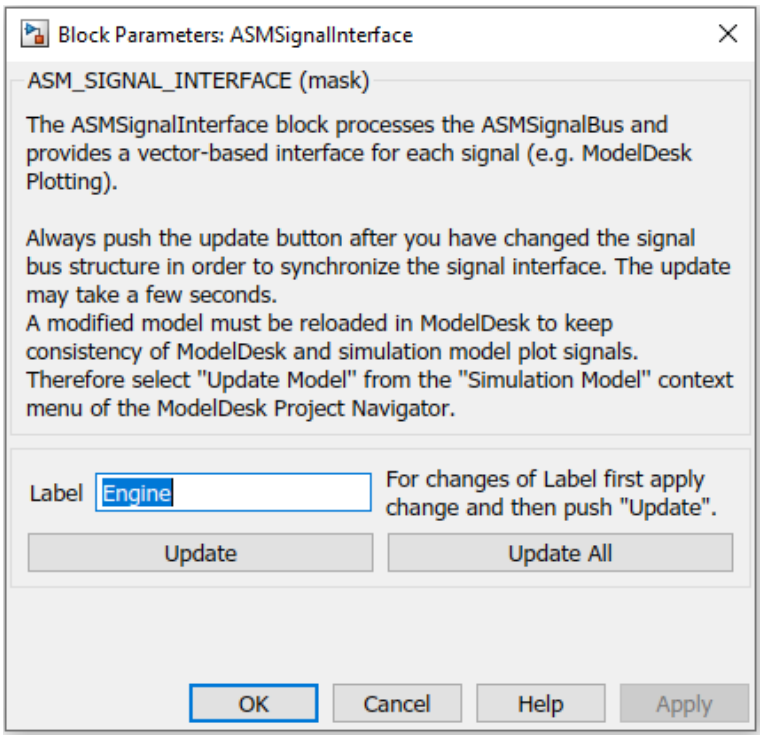
**Method**

**To update a plot configuration for ModelDesk for a Simulink simulation**

1 Inside an ASM module, open the SignalInterface subsystem.

2 Double-click the ASMSignalInterface block.

The following dialog appears with varying labels, depending on your model.

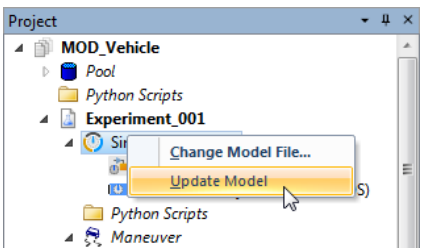



3 Click:

- Update: To update only the current block.
- Update all: To update all blocks in the model that are currently not commented out.

All the ASMSignalInterface blocks are updated. This also changes the model, which you must update in ModelDesk.

4 In ModelDesk's Project Manager, right-click the Simulation model element and select Update Model.



**Result**

The plot configuration is updated.

**Related topics**

Basics

Collecting Signals for Plotting in the ASM Model (ModelDesk Plotting 🕮)

References

Signal Selector (ModelDesk Plotting 🕮)

# How to Configure a Plotter in ModelDesk for a Simulink Simulation

**Objective**

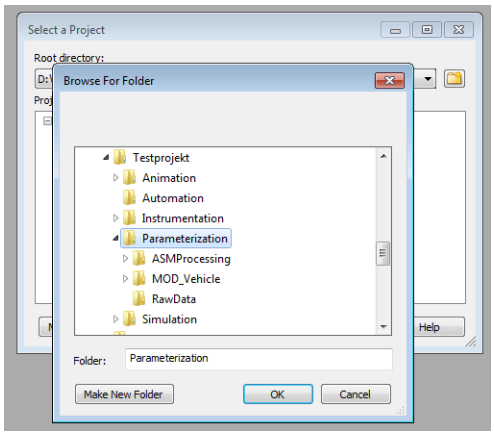
To plot a signal from the ASMSignalBus, you have to configure a project and an experiment in ModelDesk.

The `<Projectfolder>\Parameterization\MOD_<Blockset>` folder includes a complete ModelDesk project. You can completely parameterize the engine in ModelDesk. The demo project includes examples of typical plotter signals.

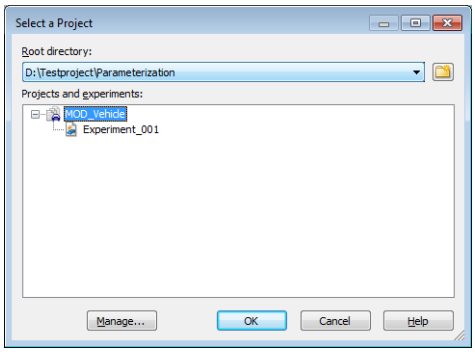
**Method**

**To configure a plotter in ModelDesk for a Simulink simulation**

1 Start ModelDesk and open the provided ModelDesk project from `<Projectfolder>\Parameterization`.
   1. Click File – Open - Project + Experiment.
   2. Select the `<Projectfolder>/Parameterization` folder.

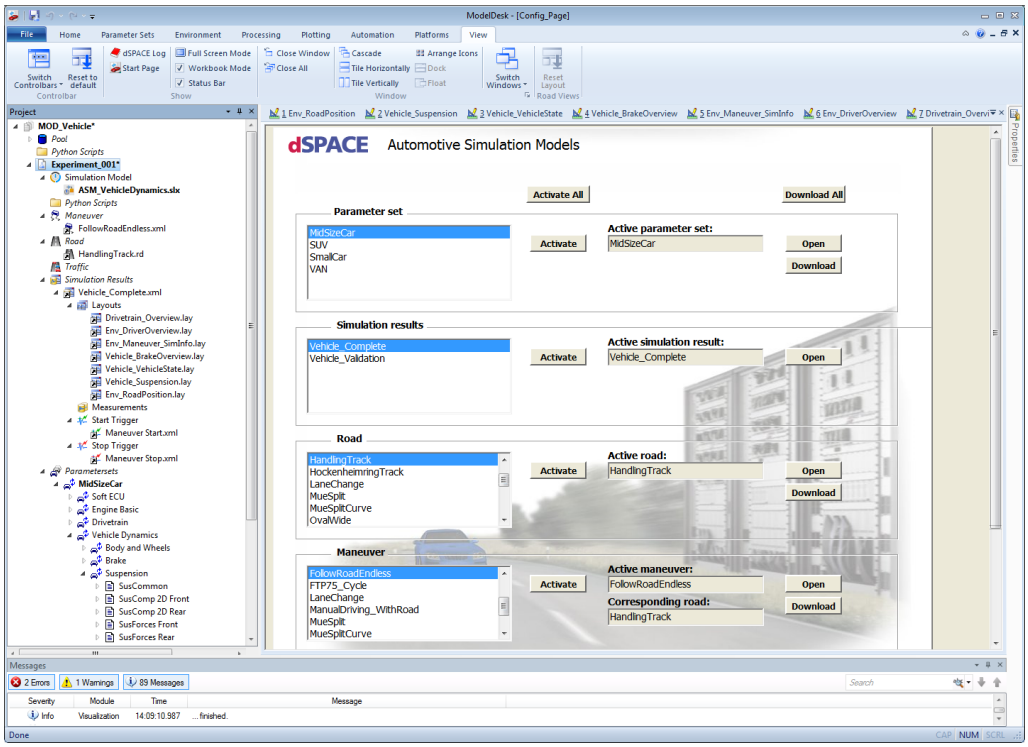


3. Select an experiment from your ModelDesk folder.

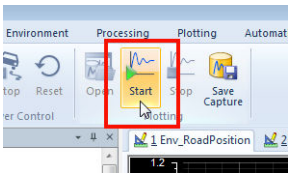



**2** Switch to the experiment's Configuration page.

**3** Select a plot configuration from the Simulation results list, click Activate and Open.



4 To start a simulation, go to the **Home** ribbon and click **Plotting – Start** (if possible in Simulink Accelerator mode).




**Result**

ModelDesk starts the simulation and plots the selected signals. You can observe the results in the ModelDesk plotters.

**Related topics**

Basics

Plotting Signals (ModelDesk Plotting 🕮)

# How to Animate the Simulation Results

**Objective**

You can animate the behavior of the car during the simulation with MotionDesk. The demo model includes the block needed for preparing and sending the simulation data to MotionDesk. In a Simulink simulation, data needed by MotionDesk is transmitted via network with the UDP protocol.

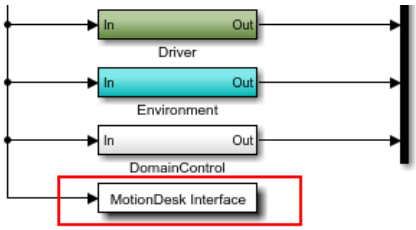
**Prerequisites**

The following steps are possible only if you have installed licensed MotionDesk software in the current dSPACE installation or on a different PC.

**Method**

**To animate the simulation results**

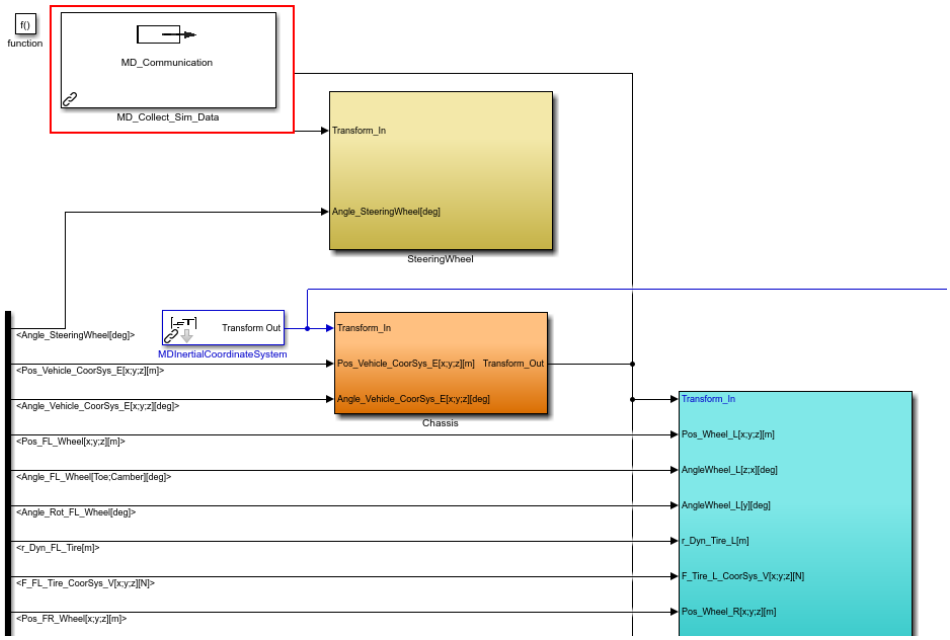
1 In the model, double-click the **MotionDesk Interface**.

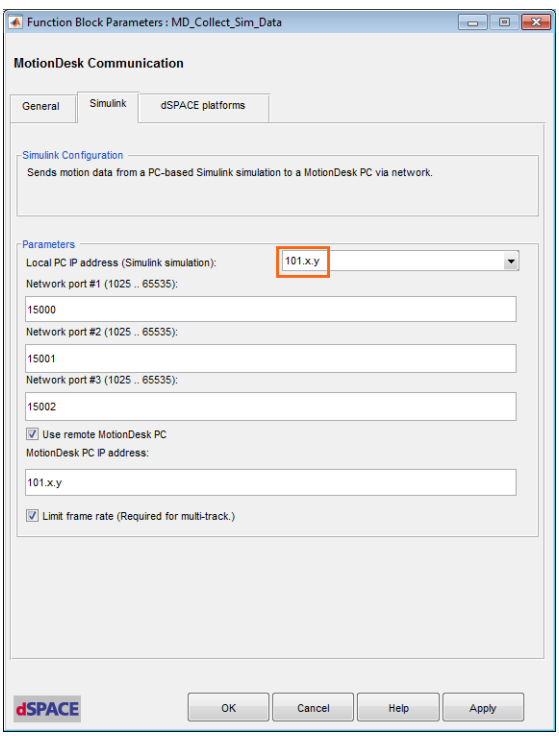



2 Open the block dialog of the **MD_Communication** block in **MotionDeskInterface/MOTION_DESK_INTERFACE** subsystem and check whether the MotionDesk PC IP address is correct.

**Tip**

If MotionDesk runs on the same PC as the simulation, enter the local PC IP address in the MotionDesk PC IP address field.



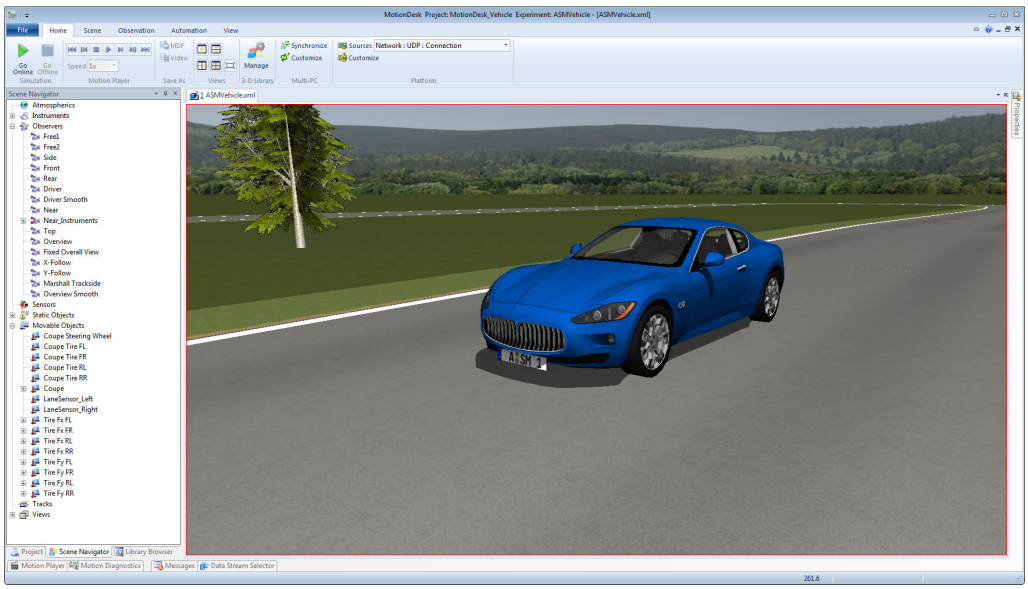
3 Start MotionDesk and select File – Open – Project + Experiment to load the experiment from `<Projectfolder>\Animation\MotionDesk_Vehicle\ASMVehicle\ASMVehicle.cdp`.

4 In MotionDesk, go to the Home ribbon and select Network: UDP: Connection from Platform – Sources.

5 Start the simulation of the model and look at the MotionDesk animation.

It will take some time for the car to start moving, because in the first two simulation seconds, there is no effect on the car.

If the animation does not work, refer to Troubleshooting in MotionDesk on page 69.



**Result**

The animation is running.

**Related topics**

Basics

MotionDesk Interface (ASM User Guide 🕮)

# How to Change Tire and Transmission Models via ModelDesk

**Objective**

The ASM Vehicle Dynamics demo model includes different tire models:

- The MagicFormula tire model
- The TMEasy tire model

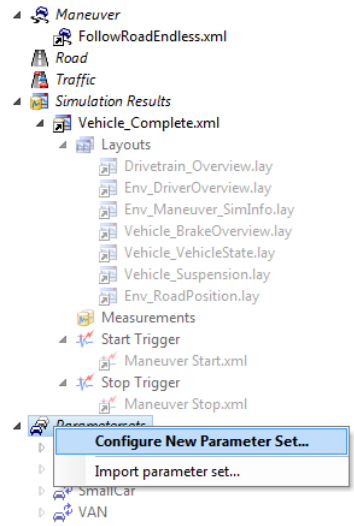
For the transmission, you can select automatic or manual transmission. The hybrid vehicle model includes the automatic transmission only. Therefore it is not reasonable to use the manual transmission. You can set the combination you want in the ModelDesk project parameter set.

## Method

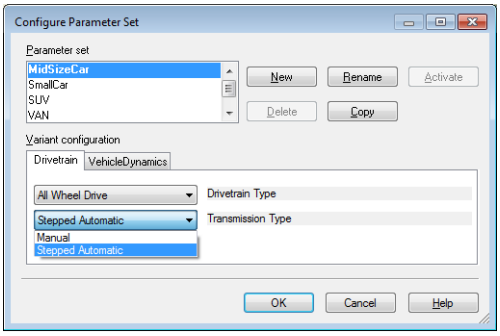
### To change tire and transmission models via ModelDesk

1 In the ModelDesk Project Navigator, open the context menu of Parametersets and select Configure New Parameter Set.

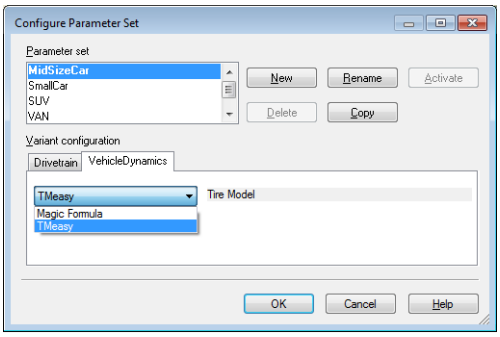
The Configure Parameter Set dialog opens.




2 Click Drivetrain and select a transmission type from the Transmission Type list.The parameter pages for configuring the necessary parameters are available in the project according to your selection.



3 Click **VehicleDynamics**. Select a tire model from the **Tire Model** list.




4 Click **OK** to close the dialog.

The parameter pages for configuring the necessary parameters are now available in the project according to your selection.

**Related topics**

Basics

Basics on Tire Models (ASM Vehicle Dynamics Addendum 🕮)

# How to Change the Suspension Kinematics Model

**Objective**

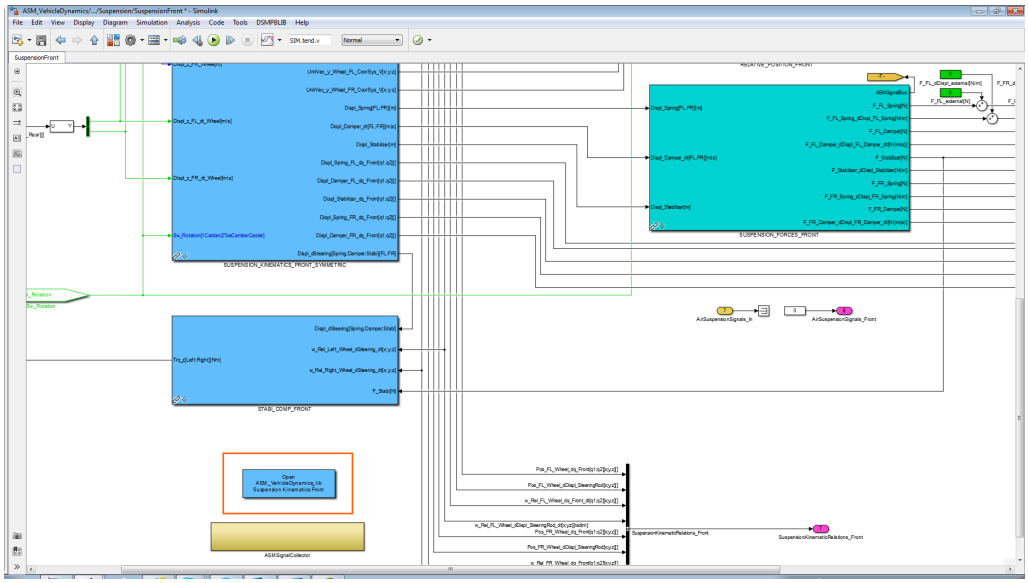
The ASM Vehicle Dynamics Package includes different simulation models for the suspension kinematics. This chapter describes how to change a table-based front suspension kinematics model with the geometrically described MacPherson Strut.

**Method**

**To change the suspension kinematics model**

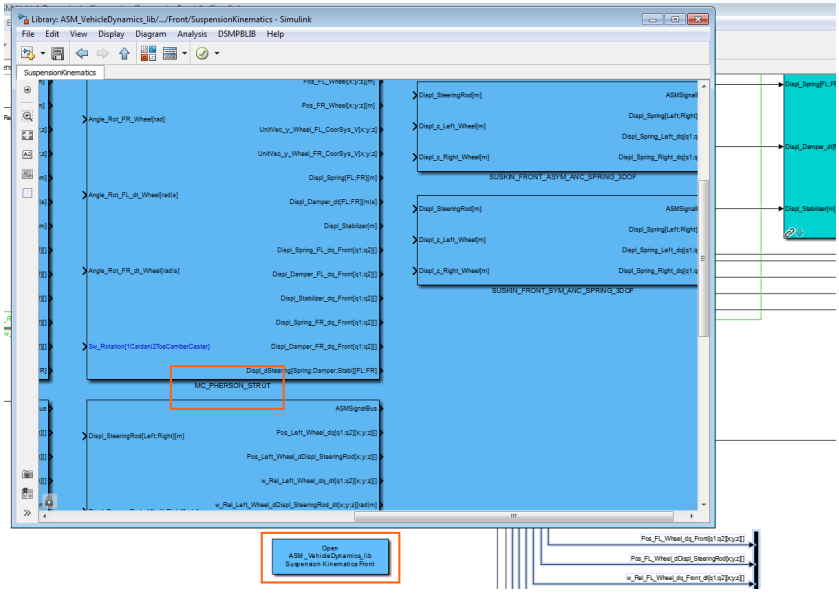
1 Open MATLAB and either create a new ASM Vehicle Dynamics demo project or open an existing project. For detailed information, see How to Open an ASM Library on page 12.

2 Open the SuspensionFront subsystem (VehicleDynamics - Plant - VehicleDynamics - Suspension - SuspensionFront).

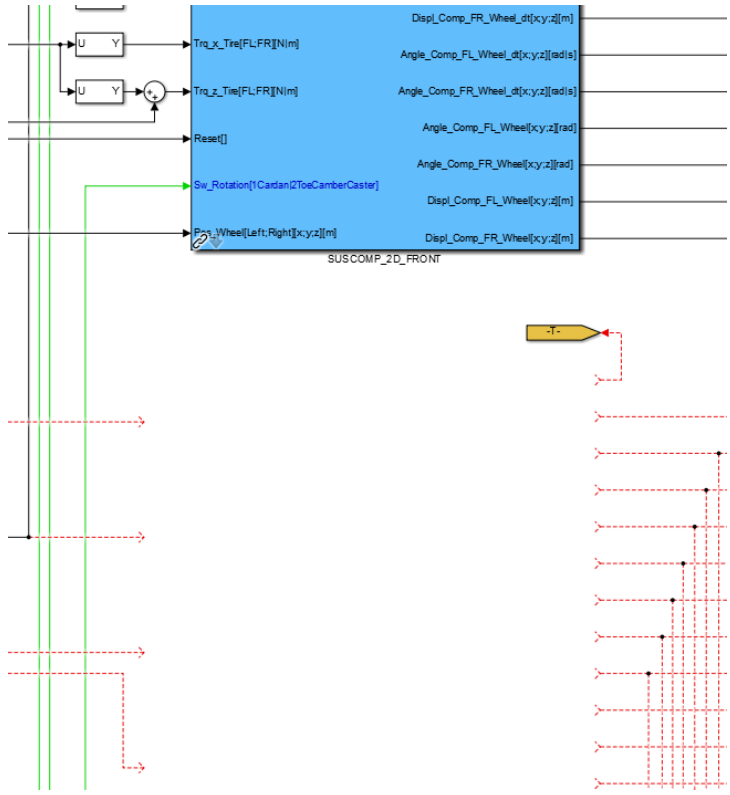


3 Double-click Open ASM_VehicleDynamics_lib Suspension Kinematics Front.

4 Copy the MC_PHERSON_STRUT block (`Ctrl` + `C` or context menu) from the opened suspension kinematics library.



**5** Change back to the Simulink model and delete the SUSPENSION_KINEMATICS_FRONT_SYMMETRIC subsystem.



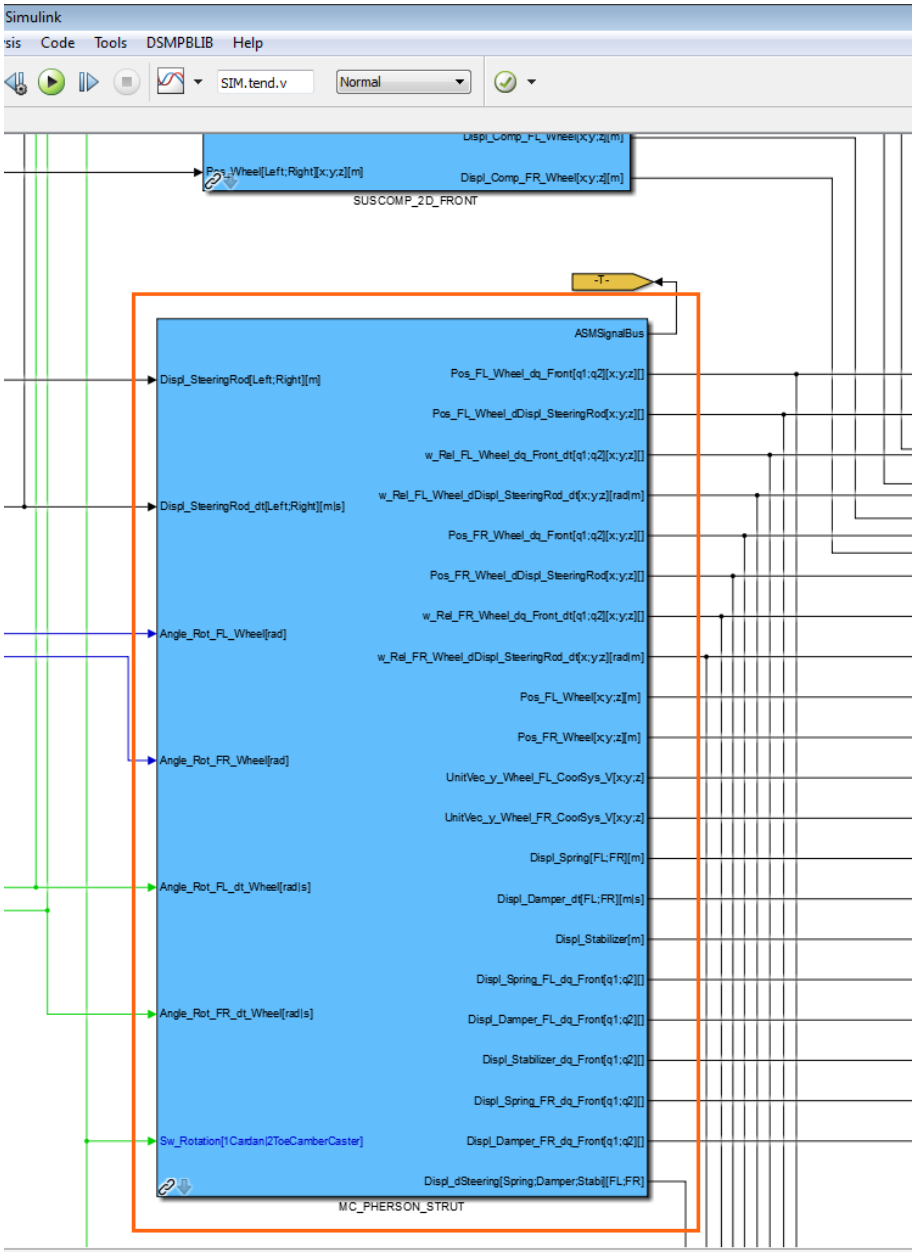

**6** Paste the copied MC_PHERSON_STRUT subsystem and connect the unconnected inport and outport signals.

**Note**

The suspension models are functions of two generalized coordinates: q_Front [q1,q2] for front suspension and q_Rear[q1,q2] for rear suspension. The table-based suspension kinematics are functions of vertical wheel displacement of left and right side, and the geometrically described MacPherson Strut is a function of rotation angle of left and right side. Thus, when replacing the table-based suspension kinematics with the geometrically described one, you have to connect inports q1 and q2 to Angle_Rot_FL and Angle_Rot_FR, respectively. For further information on suspension kinematics, refer to Suspension Subsystem (ASM Vehicle Dynamics Reference 🕮).

**Tip**

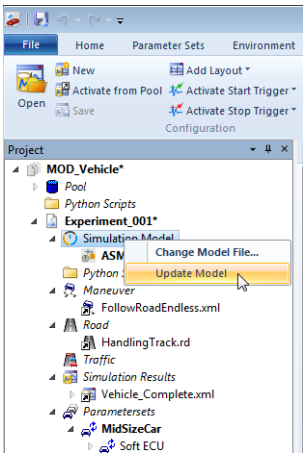
The easiest way to replace the suspension kinematics subsystem is to fit the block size of new suspension kinematics block to the one contained in the model first and then drag the new block to the same position as the former one.



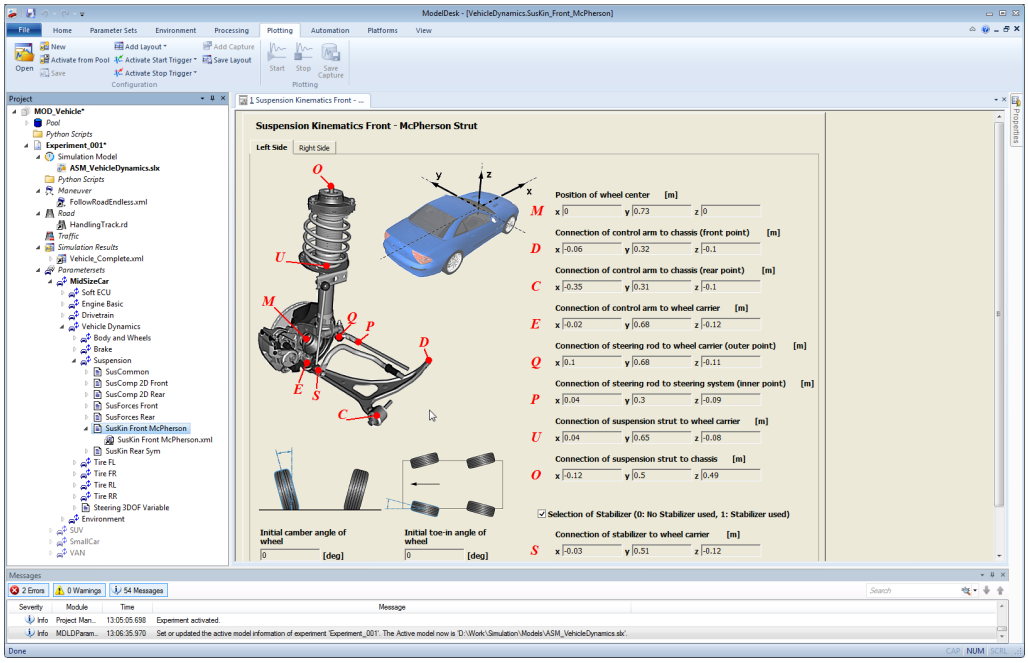

7 Save the Simulink model.

8 Open ModelDesk and start the related project (for detailed information, see How to Parameterize ASM via ModelDesk on page 18.)

**9** Update the Simulation model.

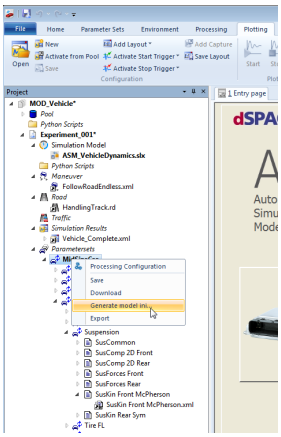


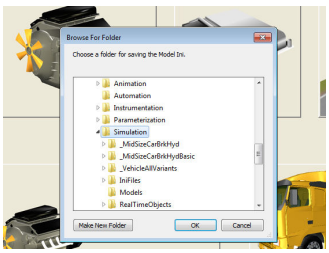

**10** Switch to the SusKin Front MacPherson parameter page and specify the parameter values according to your needs.




**11** Save the ModelDesk project.

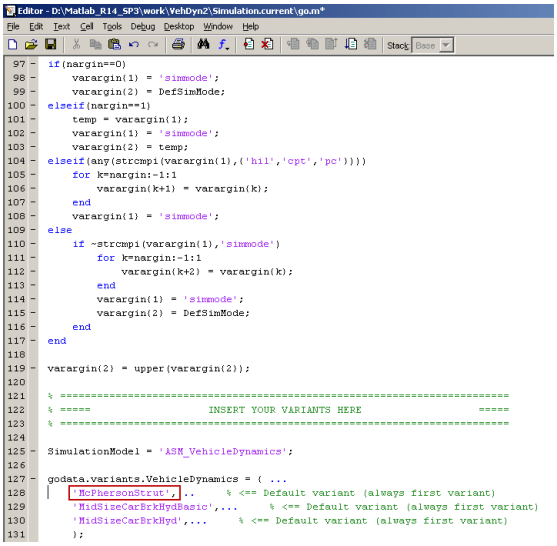
**12** Open the context menu of the **MidSizeCar** parameter set and click **Generate model ini...** to generate the new model initialization files.




**13** Switch back to MATLAB and adapt the **go** file of the demo model. The **go** file initializes the model.

In the MATLAB Command Window, type `edit go`.




**14** In the MATLAB Command Window, type **go** to start the go file.

**Result**

You replaced the table-based front suspension kinematics system of the demo model with the MacPherson strut system.

Replacing the suspension kinematics system with the asymmetric table-based system or changing suspension kinematics system of rear suspension is done in the same way.

**Tip**

For visualization of geometrically described suspension, call `asm_plot_geosusp` in the MATLAB Command Window. This m-function simply plots each linkage point of geometrically described suspension and shows suspension at configuration state. For further help on this function, type `help asm_plot_geosusp` in the MATLAB Command Window or refer to asm_plot_geosusp (ASM User Guide 🕮).

**Related topics**

References

Update Model (ModelDesk Parameterizing 🕮)

# How to Change the Drivetrain Configurations

**Objective**

Several drivetrain configurations are provided in the **VehicleDynamics** library as standard demo models. There are demo models for *flexible* as well as for *rigid* drivetrains. To change the drivetrain configuration, some subsystems of the **Drivetrain** model must be replaced and the correct drivetrain variant must be selected in ModelDesk.

**Note**

To switch between flexible and rigid drivetrain, all of the following subsystems must be exchanged: **Crankshaft**, **Transmission** and **Final_Drive_Assembly**.
In your **Drivetrain** model, there must not be flexible and rigid blocks or subsystems at the same time.

**Standard drivetrain configurations**

The standard demo models for the drivetrain configurations are modeled in the **Final_Drive_Assembly** subsystems. There are two **Final Drive Assembly** subsystems, which simulate different drivetrain configurations: flexible and rigid. To switch between flexible and rigid drivetrain, different demo models are provided. If you have exchanged blocks in your model, the real-time application must be rebuilt.

- To switch between flexible and rigid drivetrain, refer to Method 1.
- To select the drivetrain variant in ModelDesk, refer to Method 2.

**Method 1**

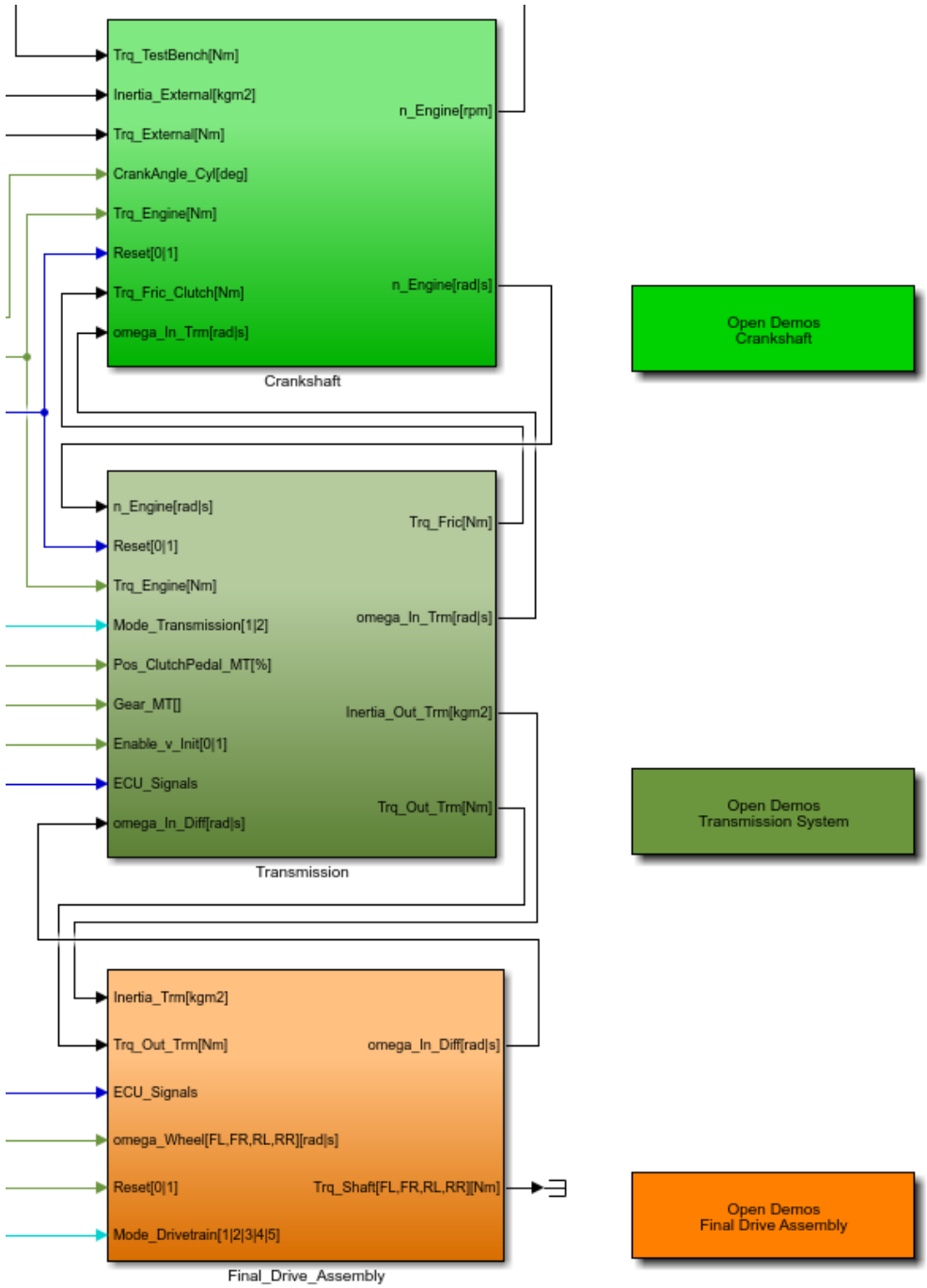
**To switch between flexible and rigid drivetrain**

**Note**

If a real-time application has to be built after changing between flexible and rigid drivetrain, the real-time paths of the parameters in the related blocks will change. Therefore, in ControlDesk, the related parameters must be reconnected.

1 Open MATLAB and create a new ASM Vehicle Dynamics demo project or open an existing project. For more information, see How to Open an ASM Library on page 12.

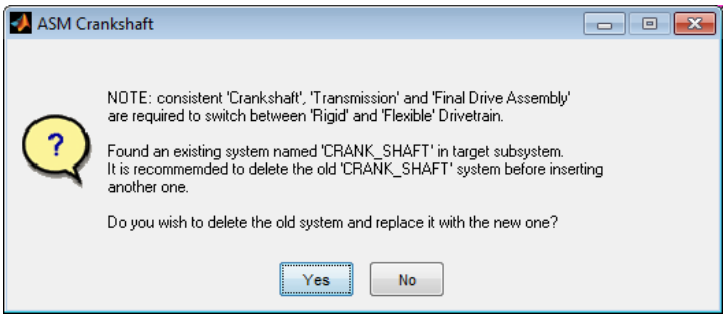
**2** Open the Drivetrain subsystem.




**3** Double-click Open Demos Crankshaft to the right of the Cranksaft subsystem to open the library.

**4** Drag a Crankshaft subsystem (flexible or rigid) from the library to the Drivetrain model.

A confirmation prompt opens asking you for confirmation to delete the old system and replace it with the new one.




Click **Yes** to replace the old block by the new one. It is connected automatically.

If you click **No**, the new block is inserted but not connected, and the old block is kept as well.

5 Repeat the steps for the **Transmission** subsystem.

Double-click the **Transmission System** or the **Transmission System Rigid** sublibrary to see the **Transmission** block included.

6 Repeat the steps for the **Final_Drive_Assembly** subsystem.

7 Save the model.

Depending on the initialized variant in MATLAB, it is possible that some parameters for the new blocks inside the **Final_Drive_Assembly** are not initialized. In this case, new initialization files have to be generated from ModelDesk after updating the project.

8 If a real-time application is needed, you must rebuild it after saving the model.

---

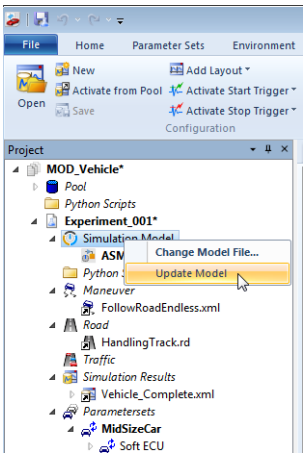
**Method 2**

**To select the drivetrain variant in ModelDesk**

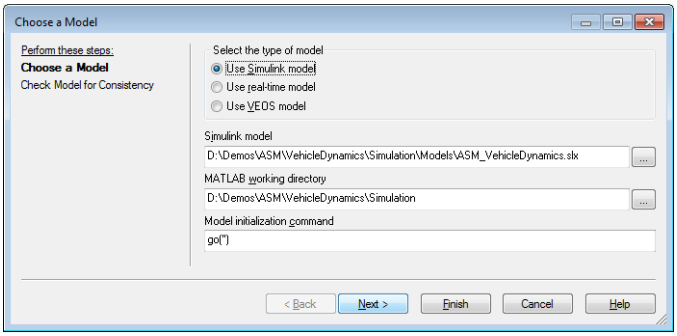
1 In ModelDesk, open the related project. Refer to How to Parameterize ASM via ModelDesk on page 18.

2 If the model has been changed, as described in Method 1, for example, you must update the model.

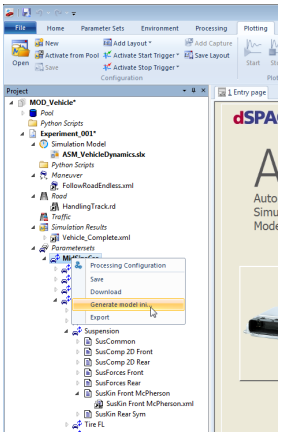
In the Project Navigator, right-click Simulation Model and select Update Model from the context menu.



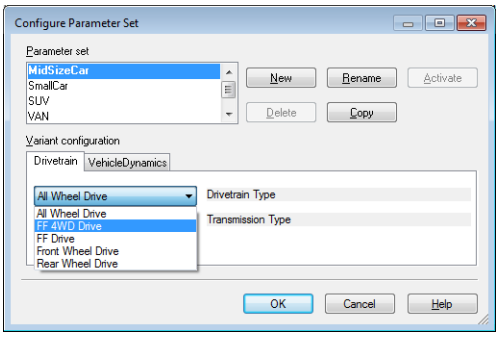

**3** Select the correct model, click Next and then Finish.




**4** In the Project Navigator, right-click Parametersets to open the context menu and select Configure New Parameter Set.



5 Select the desired drivetrain configuration and click **OK**.




After selecting the drivetrain configuration, the **Parameter** pages for configuring the necessary parameters will be available in the project.

6 Configure the parameters for the drivetrain components and save the ModelDesk project.

7 Generate new initialization files. These files contain the necessary parameters for all ASM blocks in the drivetrain model.

# Simulating on dSPACE Platforms

**Where to go from here**

Information in this section

How to Generate an OSA File for VEOS.....52
You can generate an Offline Simulation Application (OSA) file for VEOS either manually or automatically.

Generating the Real-Time Application for SCALEXIO.....54
You can generate the real-time application for SCALEXIO in ConfigurationDesk.

How to Generate a Real-Time Application for PHS-Bus Based Platforms.....54
If you have a dSPACE RTI (Real-Time Interface) license, you can also generate new code for the ASM model.

How to Animate the Simulation Results in MotionDesk.....55
You can use MotionDesk to animate the simulation results.

How to Observe Results in ControlDesk.....56
When simulating the model, you can observe the results in ControlDesk.

How to Update a Plot Configuration for ModelDesk for a dSPACE Platform.....59
If the ASMSignalBus is changed, you must update the plotting interface block and the model in ModelDesk.

Changing the Simulation Platform.....60
You can change the platform and reuse the experiment.

How to Use the ModelDesk Project for the Real-Time Simulation.....60
You can use a ModelDesk project for real-time simulation.

How to Change a Parameter in ModelDesk and Observe the Results in MotionDesk and ControlDesk.....61
You can change a parameter in ModelDesk and observe the results.

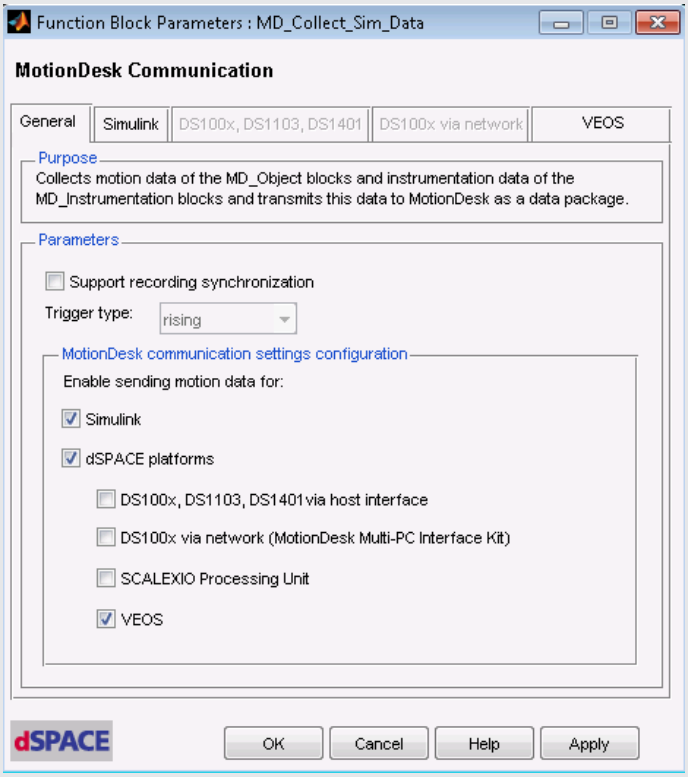
# How to Generate an OSA File for VEOS

**Objective**

You can generate an Offline Simulation Application (OSA) for VEOS either manually or automatically.

**Note**

To use MotionDesk for animation, the MotionDesk communication has to be configured for VEOS in the `…/MOTION_DESK_INTERFACE/MD_Collect_Sim_Data` Simulink subsystem.




**Possible methods**

There are two ways to generate code for VEOS:

- Manually. Refer to Method 1 on page 52.
- Automatically. Refer to Method 2 on page 53.

**Method 1**

**To generate code for VEOS manually**

1 Open the model with `go ('simmode','CPT','platform','VEOS')`. This selects the system target `dsrt.tlc` file provided by the Model Interface Package for Simulink.

2 Generate code for the ASM model.

The Model Interface Package for Simulink generates a Simulink implementation container (SIC) file containing ASM model code.

For instructions, refer to Generating Simulink Implementation Containers (Model Interface Package for Simulink - Modeling Guide 🕮).

3 Create a new OSA file:

1. Start the VEOS Player.
2. On the **Home** ribbon, select **New** to create an empty OSA file.

For instructions, refer to Basics on the VEOS Player (VEOS Manual 🕮).

4 Import the SIC file to the VEOS Player to integrate the model in an Simulink simulation application for VEOS.

The VEOS Player builds the OSA for simulation on VEOS.

**Note**

ASM does not support a 32 bit version of VEOS. Make sure to select **HostPC64** as simulation target on the **Build Options** tab during import to the VEOS Player.

You can also set **HostPC64** as default simulation target via the Simulation Target Manager in VEOS.

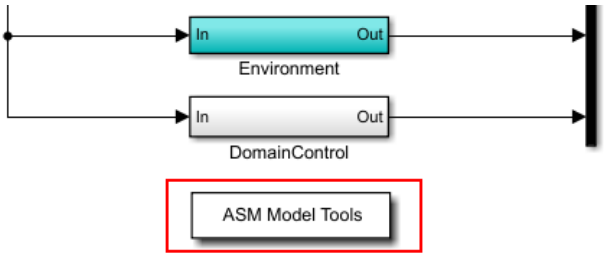
For instructions, refer to How to Import Simulink Implementations (VEOS Manual 🕮).

**Method 2**

**To automate code generation for VEOS**

1 Open the model in MATLAB.

2 In the ASM model, double-click the **ASM Model Tools** button below the modules.




3 Double-click the **Generate VEOS Application** button.

4 A **Code Generation** confirmation prompt opens. Click **Yes** to start code generation.

**Result**

You have generated code for VEOS.

**Related topics**

References

Generate VEOS (ASM User Guide 🕮)

## Generating the Real-Time Application for SCALEXIO

**Generating code for SCALEXIO**

Generation of the real-time application has to be started from ConfigurationDesk. For information on how to set up a ConfigurationDesk project and start code generation, refer to Building Real-Time Applications (ConfigurationDesk Real-Time Implementation Guide 🕮).

## How to Generate a Real-Time Application for PHS-Bus Based Platforms

**Objective**

If you have a dSPACE RTI (Real-Time Interface) license, you can generate new code of the ASM model for real-time applications. This is needed, if you want to modify the model, for example, after you have added I/O blocks for doing HIL simulation, or if you want different initial settings for the model (settings of your vehicle).

If you have a dSPACE VEOS license, you can generate new code of the ASM model for a VEOS platform (offline simulation application).

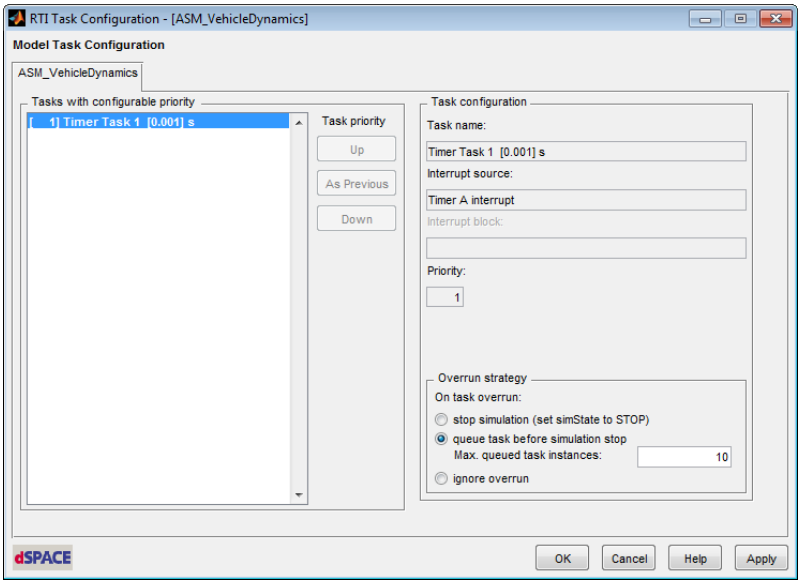
For detailed information on how to generate an offline simulation application, refer to How to Generate an OSA File for VEOS on page 52.

**Method**

**To generate a real-time application for the PHS bus based platforms**

1. Start MATLAB, switch to `<Projectfolder>\Simulation` and call `go`. If prompted, select the automatic setting of the preferences. The demo model opens with the initial default parameters specified in the `go` file.
2. From the menu bar, choose Simulation – Model Configuration Parameters or press `Ctrl+E` to open the Configuration Parameters dialog.

3 Open the RTI Simulation Options page and click Task Configuration. In the Task Configuration dialog, set the overrun strategy of timer task 1 to queue task before simulation stop and the number of maximum task instances to "10". This setting has to be made, because the initialization of the model on the real-time hardware takes some time, so overruns will occur at the beginning of the simulation. A maximum of 10 permitted overruns will prevent the simulation from stopping during initialization.




4 To start code generation, click Build Model on the toolbar or press `Ctrl+B`.

**Result**

The complete build process will take some minutes. After the build process is finished, the new real-time application is downloaded to the dSPACE platform. The initial parameters are the parameters which were loaded from the go file.

The new real-time application can be used in the same way as the original demo application.

**Related topics**

References

RTI Task Configuration Dialog (RTI and RTI-MP Implementation Reference 🕮)

# How to Animate the Simulation Results in MotionDesk

**Objective**

You can use MotionDesk for animating the simulation results. For the real-time applications in the ASM demo model, MotionDesk and ControlDesk must run on the same PC.

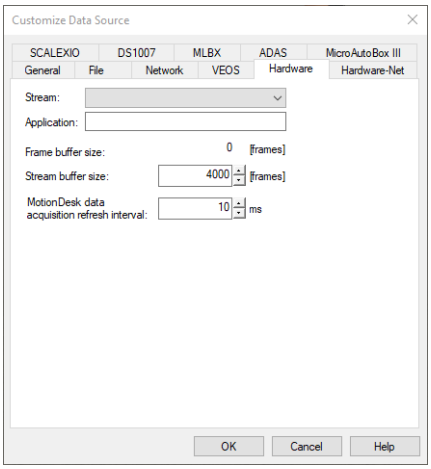
**Method**

**To animate the simulation results in MotionDesk**

1 Start MotionDesk and open the experiment from the demo project in the `<Projectfolder>/Animation` folder.

2 On the Home ribbon, select Platform – Customize.

The Customize Data Source dialog opens.




3 Open the page of your platform type and specify the simulation platform in the Stream list.

4 Start the maneuver in ControlDesk.

5 Start the animation in MotionDesk.

# How to Observe Results in ControlDesk

**Objective**

When simulating the model, you can observe the results in ControlDesk.

The ASM demo model contains preconfiguered real-time application files for the SCALEXIO and VEOS platforms. You can use these files to observe simulation results with the preconfigured ControlDesk experiment.

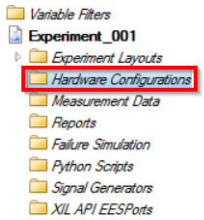
**Method**

**To observe results in ControlDesk**

1 Start ControlDesk.

On the ribbon, click File – Open – Open Project and Experiment to open the preconfigured ControlDesk experiment. Select `<Projectfolder>\Instrumentation\CD_<Blockset>\CD_<Blockset>.CDP`.

2 In ControlDesk, right-click **Hardware Configurations** in the project tree. On the context menu, select **Add Platform/Device** to add the platform.




Select your simulation platform and click **Next**. Import the prebuilt real-time application from `<Projectfolder>\Simulation\RealTimeObjects\<platform>\ASM_<Blockset>.sdf`.

3 On the **Platforms** ribbon, select **Register Platforms** to register your platform.

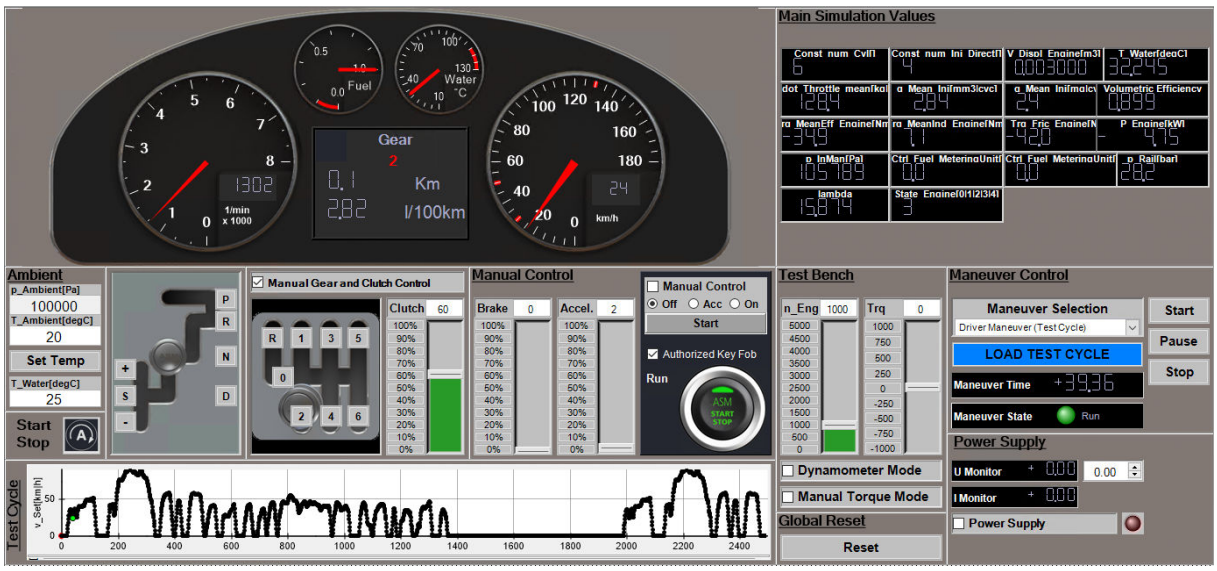
4 Open the platform's context menu.

Open the predefined layout in the **Experiment Layouts** folder in the project tree of the ControlDesk experiment.

5 Save the experiment.

On the **Home** ribbon, select **Go Online** to start online calibration.

The application runs on the selected platform. You can observe the Simulation results on the available layouts.




6 Use the instruments in the dashboard layout to control the maneuver, to start and stop the engine and to observe variables such as vehicle speed and engine speed.

You can also select whether you would like to drive with the ASM driver model, with stimulated input signals, or with a manual control, by changing the maneuver selection. In real-time simulation, you have to set the pedal positions and the gear switch yourself, for example, by using the instruments of the Manual Control area in the dashboard layout.

Depending on the ASM used, you can use further layouts to observe engine or vehicle dynamics signals.

**Result**

You observed results in ControlDesk.

**Note**

For some ControlDesk projects, a map layout is available that provides a browser instrument to show a street map with the current position of the simulated vehicle depending on its GPS coordinates. The visualization of the street map is based on OpenStreetMap data. Refer to Additional Information on page 74.

**Note**

If you want to update the real-time application, note the following points:
For RTI and VEOS, the real-time code is generated and saved in the MATLAB working directory, which is typically the `Simulation` folder. When you generate the code repeatedly in the same directory, ControlDesk detects the update and prompts you to reload the real-time application when going online.

**Note**

Generating new code with an RTI platform (e.g., DS1007) creates a host service named HostService by default. The provided ControlDesk experiments expect a host service named Periodic Task1 (default name for SCALEXIO and VEOS platforms). If you do not manually rename the default host service for the RTI platform to Periodic Task1, the ControlDesk Message Viewer displays the following messages:

```
ControlDesk NG MeasurementConfiguration:
Warning …:Could not create the signal for variable 'XYZ’ on
platform 'ABC' with raster 'Periodic Task 1'.
ControlDesk NG MeasurementConfiguration:
Warning …:Could not create the signal for the following
variable on platform ABC with raster Periodic Task 1: XYZ
```

These messages depict the difference between the existing name and the expected name of the host service. However, this behavior has no negative effect on the functionality of the ControlDesk experiment. The signals are connected with the present host service.

# How to Update a Plot Configuration for ModelDesk for a dSPACE Platform

**Objective**

The ASM package includes a signal interface block for ModelDesk so that signals of the ASMSignalBus can be plotted. This block must be updated manually if the ASMSignalBus is changed.

If the respective simulation model is an operator version, it is mandatory to perform a manual update of the plotting block and plotting configuration, before using the plotting feature.

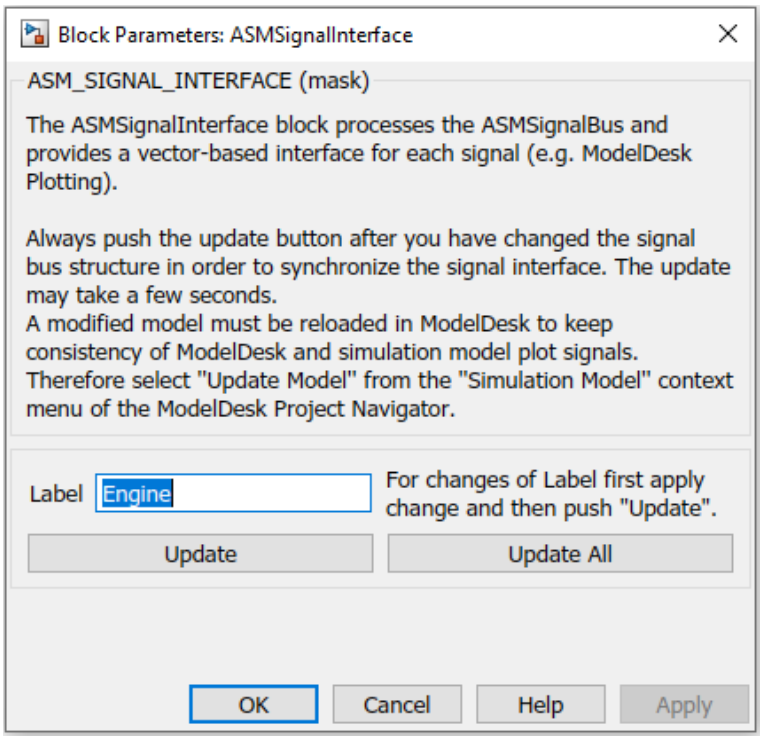
**Method**

**To update a plot configuration for ModelDesk for a real-time simulation**

1 Inside an ASM module, open the SignalInterface subsystem.

2 Double-click the ASMSignalInterface block.

The following dialog appears with varying labels, depending on your model.




3 Click:

- Update: To update only the current block.
- Update all: To update all blocks in the model that are currently not commented out.

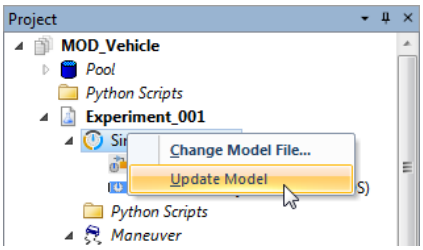
All the ASMSignalInterface blocks are updated. This also changes the model, which you must update in ModelDesk.

4 Start a new code generation process.

You must update the simulation model in ModelDesk to make all changes in the ASMSignalBus visible to ModelDesk.

Make sure your experiment is linked with the correct real-time model SDF file. ModelDesk shows the active model in bold.

5 In ModelDesk's Project Manager, right-click the Simulation model element and select Update Model.




**Result**

The plot configuration is updated.

**Related topics**

Basics

Collecting Signals for Plotting in the ASM Model (ModelDesk Plotting 🕮)

References

Signal Selector (ModelDesk Plotting 🕮)

# Changing the Simulation Platform

**Introduction**

You can change the simulation platform and reuse ControlDesk experiments. For how to change from VEOS to SCALEXIO, for example, refer to Switching the Simulation Platform and Reusing Experiment Parts (ControlDesk Platform Management 🕮).

Changing between other platforms works similarly.

# How to Use the ModelDesk Project for the Real-Time Simulation

**Objective**

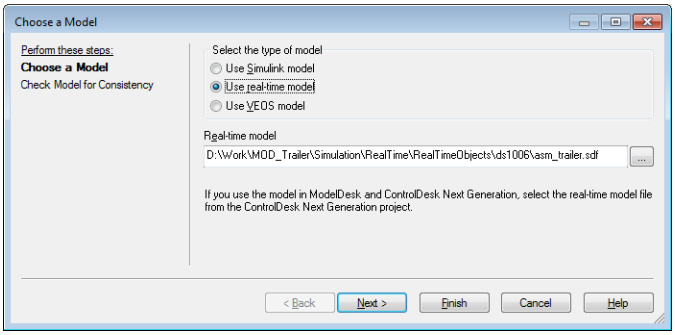
You can use a ModelDesk project for real-time simulation.

For this, you must make different settings in ModelDesk.

**Method**

**To use the ModelDesk project for the real-time simulation**

1. Open the ModelDesk project in:`<Projectfolder>\Parameterization\MOD_Vehicle\MOD_Vehicle.CDP`.
2. In the ModelDesk **Project Navigator**, open the context menu of the **Simulation Model** node and choose **Change Model File**. This has to be done to select the real-time simulation for ModelDesk, so that ModelDesk knows that the parameters should be loaded to the real-time hardware.
3. In the dialog, select **Use real-time model** and enter the path of the corresponding system description file.




4. Click **Finish** to end this process.
5. Parameterize your model in ModelDesk and switch to the experiment main page and click the **Download all** button to download the currently activated parameters of the ModelDesk project.
6. Switch to ControlDesk and observe the results on the different layouts.

**Result**

You parameterizes your ModelDesk project for real-time simulation.

**Related topics**

HowTos

How to Choose a Model and Initialize the Consistency Check (ModelDesk Parameterizing 🕮)

# How to Change a Parameter in ModelDesk and Observe the Results in MotionDesk and ControlDesk

**Obective**

You can modify model parameters in ModelDesk and then see how the modification affects the animation in MotionDesk and ControlDesk.

**Method**

**To change a parameter in ModelDesk and observe the results in MotionDesk and ControlDesk**

1. In the ModelDesk Project Navigator, open the Parametersets/MidSizeCar/Vehicle Dynamics/Body and Wheels/Body Geometry and Mass node.
2. Change the mass of the vehicle to 2000 kg.
3. Change the position of the vehicle's center of gravity z-value to 0.2 m.
4. Save and download the settings via context menu or use the Download button on the experiment main page.
5. Switch to ControlDesk and start the maneuver.
6. Switch to MotionDesk and start the animation.
7. In ModelDesk, change the position of the vehicle's center of gravity z-value to 0.8 m.
8. Save and download the settings via context menu or use the Download button on the experiment main page.
9. Switch back to ControlDesk and start the maneuver.
10. Switch to MotionDesk and start the animation.

# Interaction Between Drivetrain Blocks and Stabilization

## Interaction Between Drivetrain Blocks and Stabilization

**Introduction**

In case of multi-instance use in an ASM model, it might be required to adapt the connections between flexible drivetrain blocks and the numerical stabilization.

For more information about drivetrain stabilization, refer to Stabilization (ASM Vehicle Dynamics Reference 🕮).

For more information about the multi-instance feature, refer to Multi-Instance (ASM User Guide 🕮).

**Interaction rules**

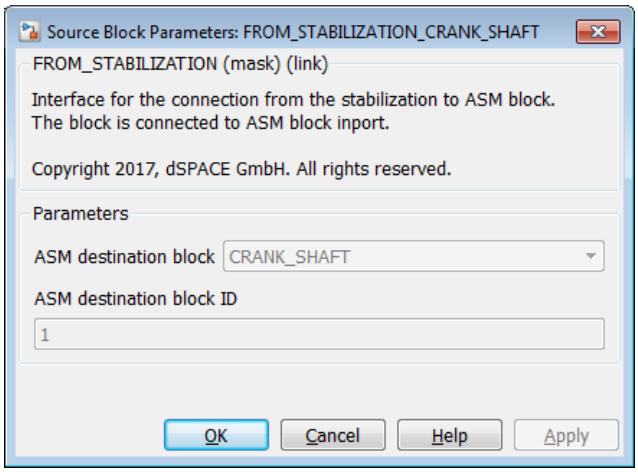
The connection between the stabilization and drivetrain is unique for each block. Because this connection is implemented by Goto/From blocks, the tags of these blocks are unique. To distinguish between the different drivetrain blocks, an identification name is used for each block. According to this identification name and the instance ID, the tag is set.

For the connection from the stabilization to the drivetrain block (FROM_STABILIZATION block) the `From_Stabi_<IdentificationName>_<InstanceID>` tag is used.

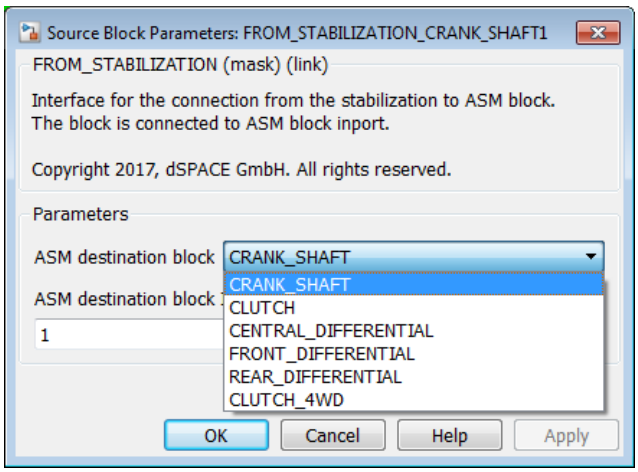
For the connection from the drivetrain block to the stabilization (TO_STABILIZATION block) the `To_Stabi_<IdentificationName>_<InstanceID>` tag is used.

These connection blocks, i.e., FROM_STABILIZATION and TO_STABILIZATION, detect a valid drivetrain block and adapt their tags automatically. This means, in case of a multi-instance block, no further action is needed.

As the blocks are correctly connected, their parameters are deactivated to protect them from unintended changes.



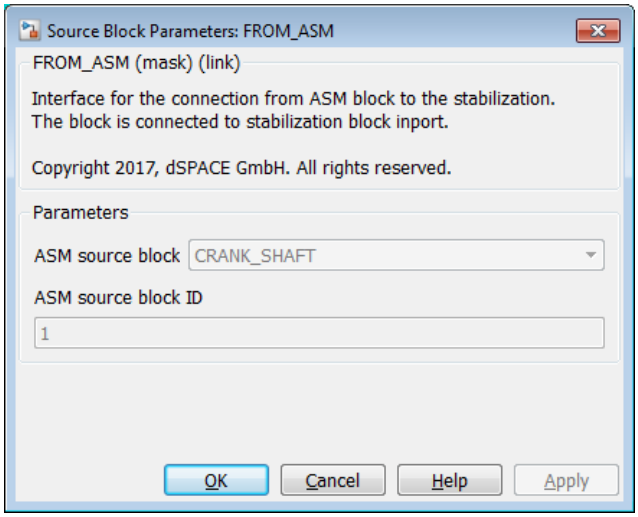

If the connected block is detected as invalid, a pop-up window appears and allows you to select the intended block.




The connection blocks (FROM_ASM and TO_ASM) that are connected to the stabilization block DRIVETRAIN_STABILIZATION cannot identify the source block automatically. Therefore, if a new connection is initialized, a pop-up menu

appears and allows you to select the source/destination drivetrain block as well as its instance ID.

In case of multi-instance blocks, it is therefore important to select the correct instance ID. Once the correct drivetrain source/destination block is selected, the block parameters are deactivated to prevent the block from unintended changes.




## Related topics

Basics

Multi-Instance (ASM User Guide 🕮)

References

Stabilization (ASM Vehicle Dynamics Reference 🕮)

# Troubleshooting

**Where to go from here**

Information in this section

Troubleshooting in MATLAB/Simulink.....67
These problems might occur when working with ASM and MATLAB/Simulink.

Troubleshooting in ModelDesk.....68
These problems might occur when working with ASM and ModelDesk.

Troubleshooting in ControlDesk.....69
These problems might occur when working with ASM and ControlDesk.

Troubleshooting in MotionDesk.....69
These are possible problems and solutions when you work in MotionDesk.

Troubleshooting with Rising Turnaround Times at Steady Operating Point.....70
Simulating the model on a real-time processor board at a steady state, might lead to a significant increase in turnaround time after several seconds of simulating this operating point.

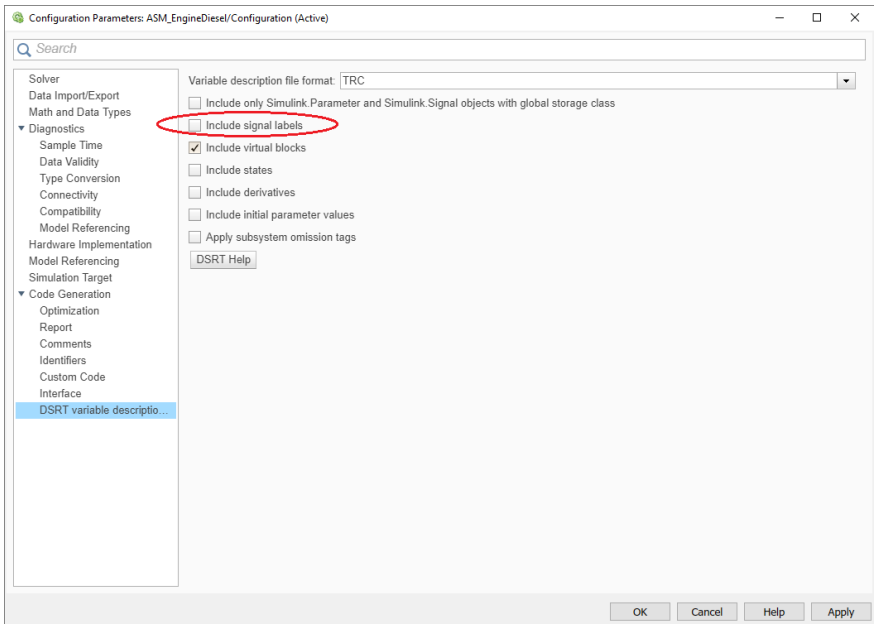
## Troubleshooting in MATLAB/Simulink

**Out of memory**

When the real-time application is built, an Out-of-Memory error at MATLAB might occur: for example, "TLC Compiler encountered an OUT_OF_MEMORY condition." This error message results from an insufficient memory allocation of the MATLAB session.

To avoid this error, one of the following actions can help:

- Disable the **Include signal labels** option at the **RTI variable description** page, see the following illustration.




- In some cases, it is a temporary solution to close MATLAB, and continue with the new MATLAB session.

**Related topics**

References

Code Generation Dialog (Model Configuration Parameters Dialogs) (RTI and RTI-MP Implementation Reference 🕮)

# Troubleshooting in ModelDesk

**Wrong model**

If downloading new parameters from ModelDesk has no effect on the simulation behavior, check if the selected simulation model in ModelDesk points to the correct model (either Simulink model or real-time model).

In ModelDesk's Project Navigator, open the context menu of the Simulation model and choose **Change Model File**.

**Related topics**

References

Change Model File (ModelDesk Parameterizing 🕮)

# Troubleshooting in ControlDesk

**Overrun message**

If you get an overrun message in ControlDesk after the real-time application was loaded to the real-time platform, check the following points:

- The processor board must fit the minimum requirements (DS1006 or SCALEXIO).
- For a DS1006 board, the RTI task configuration options must be set correct. Refer to How to Generate a Real-Time Application for PHS-Bus Based Platforms on page 54.
- For a SCALEXIO system, the task priority, step size and offset must be set correct. Refer to Generating the Real-Time Application for SCALEXIO on page 54.

# Troubleshooting in MotionDesk

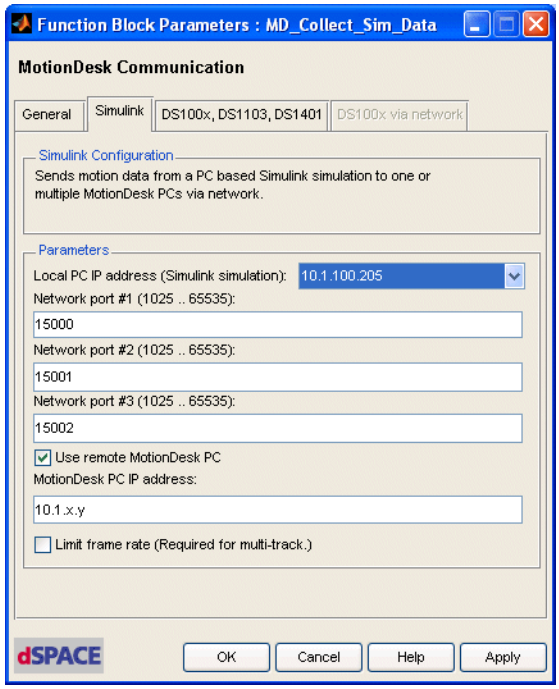
**MotionDesk does not open, error message displayed**

If you get an error message that your graphics adapter does not fit the MotionDesk requirements, check that the adapter fulfills the requirements described in Additional Requirements for 3-D Online Animation via MotionDesk (Installing dSPACE Software 🕮).

**MotionDesk animation does not work but no error message is displayed**

If the MotionDesk animation does not get new animation data (for example, no vehicle movement during a maneuver), do the following:

- Check whether the firewall prevents MotionDesk from sending data via the network.
- In the model, check whether the settings of the **MD_Collect_Sim_Data** block are correct (IP address for local PC and MotionDesk PC): **MDLUserInterface / MotionDeskInterface / MOTION_DESK_INTERFACE / MD_Communication**.




- In MotionDesk, check whether Network: UDP: Connection is set in Home – Platform – Source.
- Check whether the animation was started in MotionDesk.
- Check whether the correct source is selected in MotionDesk (Home – Platform – Customize).

**Related topics**

HowTos

How to Select the Data Source (MotionDesk Scene Animation 🕮)

References

Block Description (MD_Communication) (MotionDesk Calculating and Streaming Motion Data 🕮)
Simulation Data Sources (MotionDesk Scene Animation 🕮)

# Troubleshooting with Rising Turnaround Times at Steady Operating Point

**Description**

Simulating the model on a real-time processor board at a steady state, might lead to a significant increase in turnaround time after several seconds of simulating this operating point. When changing to another operating point, the turnaround time decreases to the original lower value. For example, in case of an

engine model this behavior might occur when you drag the engine to test bench mode at a fixed operating point.

For information on possible reasons and workarounds, refer to Troubleshooting with Rising Turnaround Times at Steady Operating Points (ASM User Guide 🕮).

# Appendix

**Where to go from here**

Information in this section

General Information..... 74
Provides additional information.

Roads..... 75

Scenarios..... 119

# General Information

## Additional Information

**OpenStreetMap**

This dSPACE Software may show contents from OpenStreetMap® during run time, that is, open data, licensed under the Open Data Commons Open Database License (ODbL) by the OpenStreetMap Foundation (OSMF). If you want to use the content in the public domain, e.g., an image that is derived from said data, you must add a credit to OpenStreetMap and the Open Database License as described on: http://www.openstreetmap.org/copyright/en.

# Roads

**Where to go from here**

Information in this section

ABSTrack Road....76
Shows information on the ABSTrack road.

FourwayJunction_2Lanes Road....78
Shows information on the FourwayJunction_2Lanes road.

FourwayJunction_4Lanes Road....80
Shows information on the FourwayJunction_4Lanes road.

HandlingTrack Road....82
Shows information on the HandlingTrack road.

Highway_A33_Paderborn Road....84
Shows information on the Highway_A33_Paderborn road.

HockenheimringTrack Road....86
Shows information on the HockenheimringTrack road.

LaneChange Road....87
Shows information on the LaneChange road.

MueSplit Road....89
Shows information on the MueSplit road.

MueSplitCurve Road....91
Shows information on the MueSplitCurve road.

NCAP_AEB Road....93
Shows information on the NCAP_AEB road.

NCAP_LSS Road....94
Shows information on the NCAP_LSS road.

OvalWide Road....96
Shows information on the OvalWide road.

RoadNetwork_Highway_Interchange_PB Road....97
Shows information on the RoadNetwork_Highway_Interchange_PB road.

RoadNetworkBasic Road....99
Shows information on the RoadNetworkBasic road.

RoadNetworkRural Road....101
Shows information on the RoadNetworkRural road.

RoadNetwork_Urban Road....103
Shows information on the RoadNetwork_Urban road.

RoadWithParkingLane Road....104
Shows information on the RoadWithParkingLane road.

RoadWorks_Basic Road..... 106
Shows information on the RoadWorks_Basic road.

RoadWorks_Highway Road..... 108
Shows information on the RoadWorks_Highway road.

Slalom Road..... 110
Shows information on the Slalom road.

StraightLong Road..... 111
Shows information on the StraightLong road.

TestGround Road..... 113
Shows information on the TestGround road.

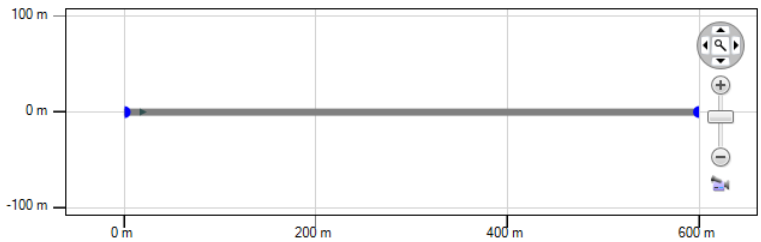
ThreeWayJunction_2Lanes Road..... 115
Shows information on the ThreeWayJunction_2Lanes road.

TrafficDriverBasic Road..... 117
Shows information on the TrafficDriverBasic road.

# ABSTrack Road

**Description**

The following table shows information on the ABSTrack road:

| | **ABSTrack Road** |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk | |
| Purpose | ▪ Example of how you can perform a braking maneuver on a slippery road.<br>▪ Example of how you can set different surface conditions. |
| Demo scenarios using this road | ▪ ABSBrake Scenario |

| | ABSTrack Road |
|---|---|
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 1 |
| Number of road elements | 1 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.6 km<br>▪ Width: 6 m |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the ABSTrack road:

| | ABSTrack Road |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

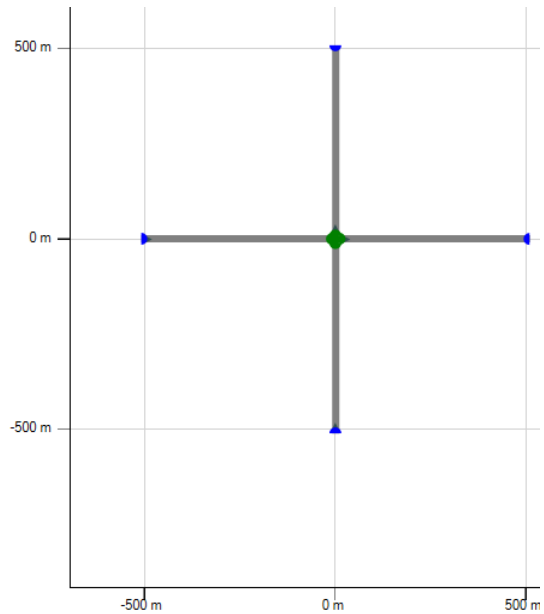
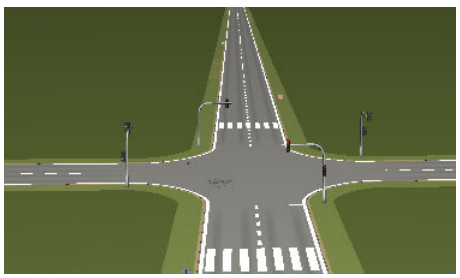
**Related topics**

References

ABSBrake Scenario..... 120

# FourwayJunction_2Lanes Road

**Description** The following table shows information on the FourwayJunction_2Lanes road:

| **FourwayJunction_2Lanes Road** | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk |  |
| Purpose | Example of how you can model a four-way junction with traffic lights. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 6 |
| Number of road elements | 4 |
| Number of junctions | 1 |
| Use of traffic lights | Yes |
| Use of traffic signs | Yes |
| Use of position markers | Yes |

| FourwayJunction_2Lanes Road | |
|---|---|
| Number of traffic lights/signs | 0 |
| Real map data[1)] | Yes |
| Network dimension | ▪ Length: 1 km<br>▪ Width: 1 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the FourwayJunction_2Lanes road:

| FourwayJunction_2Lanes Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

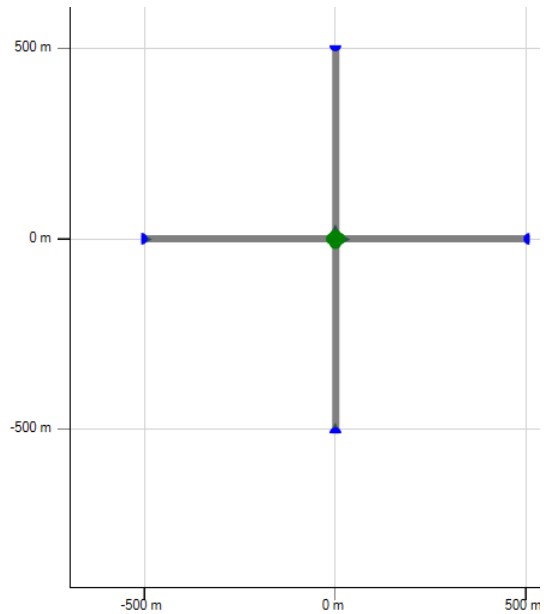
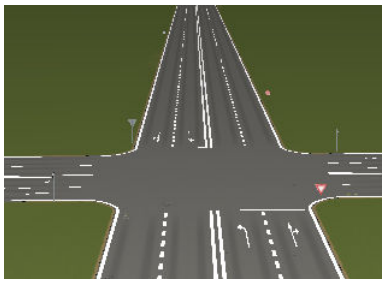
**Related topics**

References

FollowRoadEndless Scenario..... 129

# FourwayJunction_4Lanes Road

**Description** The following table shows information on the FourwayJunction_4Lanes road:

| **FourwayJunction_4Lanes Road** | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk |  |
| Purpose | Example of how you can model a four-way junction with four lanes. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 6 |
| Number of road elements | 4 |
| Number of junctions | 1 |
| Use of traffic lights | No |
| Use of traffic signs | Yes |
| Use of position markers | Yes |

| FourwayJunction_4Lanes Road | |
|---|---|
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 1 km<br>▪ Width: 1 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the FourwayJunction_4Lanes road:

| FourwayJunction_4Lanes Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

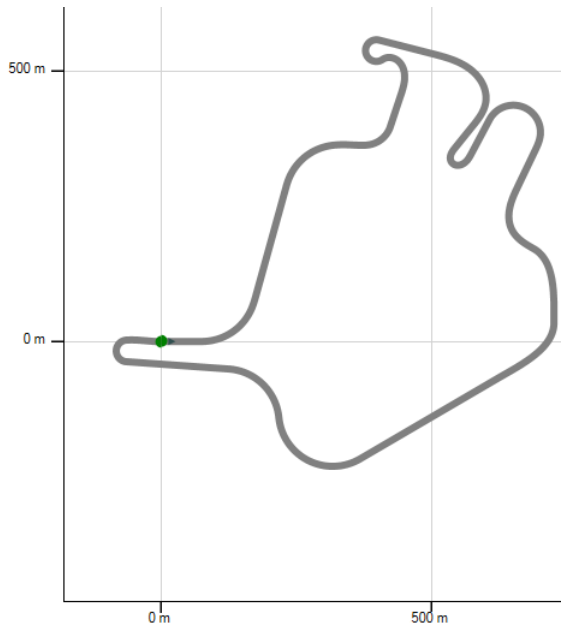
**Related topics**

References

FollowRoadEndless Scenario..... 129

# HandlingTrack Road

**Description** The following table shows information on the HandlingTrack road:

| **HandlingTrack Road** | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk | |
| Purpose | Demonstration of the vehicle dynamics characteristics of the ASM ego vehicle and its parameterization. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | Yes |

| HandlingTrack Road | |
|---|---|
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 1 |
| Number of road elements | 1 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.9 km<br>▪ Width: 0.8 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the HandlingTrack road:

| HandlingTrack Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

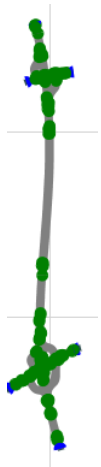
**Related topics**

References

FollowRoadEndless Scenario..... 129

# Highway_A33_Paderborn Road

**Description** The following table shows information on the Highway_A33_Paderborn road:

| **Highway_A33_Paderborn Road** | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk | |
| Purpose | ▪ Example of how you can model a road network with a highway.<br>▪ Example of a road network with a considerable number of road and junction elements.<br>▪ Real map data[1)] of roads and junctions was used. In addition, you can connect a ControlDesk Map instrument with a GPS map source, such as OpenStreetMap®. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | Yes |
| Used scenery types | Country Road<br>Bridge |

| Highway_A33_Paderborn Road | |
|---|---|
| Euro NCAP road | No |
| Number of routes | 5 |
| Number of road elements | 101 |
| Number of junctions | 44 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | Yes |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | Yes |
| Network dimension | ▪ Length: 4.2 km<br>▪ Width: 0.7 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the Highway_A33_Paderborn road:

| Highway_A33_Paderborn Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

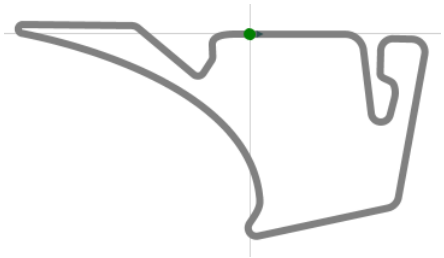
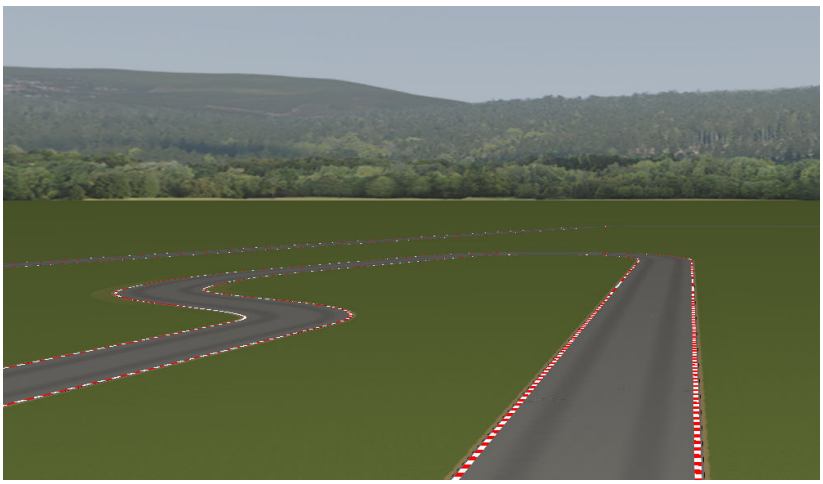
**Related topics**

References

FollowRoadEndless Scenario..... 129

# HockenheimringTrack Road

**Description** The following table shows information on the HockenheimringTrack road:

| **HockenheimringTrack Road** | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk |  |
| Purpose | ▪ Example of how you can model a road network with a race track.<br>▪ Example of a road network with road segments only.<br>▪ The road network exists in reality, but this road network is a smaller copy with a scale of 0.56. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 1 |
| Number of road elements | 1 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 0 |

| HockenheimringTrack Road | |
|---|---|
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.8 km<br>▪ Width: 0.4 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the HockenheimringTrack road:

| HockenheimringTrack Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

**Related topics**

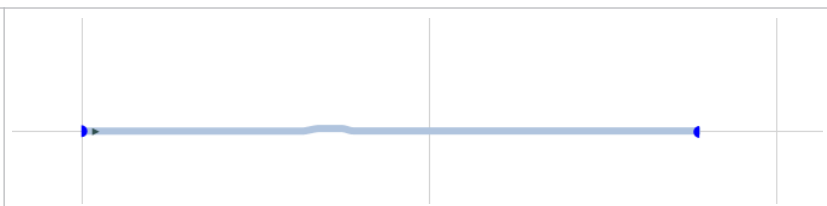
References

FollowRoadEndless Scenario..... 129

# LaneChange Road

**Description**

The following table shows information on the LaneChange road:

| LaneChange Road | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |

| | **LaneChange Road** |
|---|---|
| Overview MotionDesk | |
| Purpose | ▪ Example of a road with several road segments and different segment types.<br>▪ Example of a horizontal profile that changes depending on location, so that you can simulate a double lane change. |
| Demo scenarios using this road | LaneChange Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 1 |
| Number of road elements | 1 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.89 km<br>▪ Width: 0.01 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the LaneChange road:

| | Road |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

**Related topics**

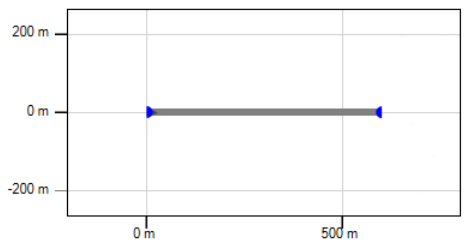
References

LaneChange Scenario..... 132

# MueSplit Road

**Description**

The following table shows information on the MueSplit road:

| | MueSplit Road |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |

| | MueSplit Road |
|---|---|
| Overview MotionDesk | |
| Purpose | ▪ Example of a road with different surface conditions on the left and right sides.<br>▪ Example of how you can model a mue-split road. |
| Demo scenarios using this road | MueSplit Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 1 |
| Number of road elements | 1 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.6 km<br>▪ Width: 6 m |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the MueSplit road:

| | MueSplit Road |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |

| | MueSplit Road |
|---|---|
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

**Related topics**

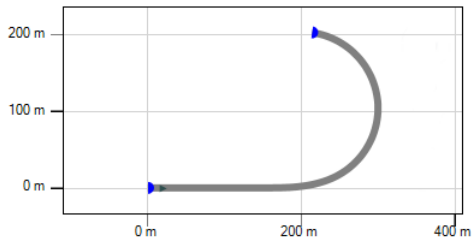
References

MueSplit Scenario..... 135

# MueSplitCurve Road

**Description**

The following table shows information on the MueSplitCurve road:

| | MueSplitCurve Road |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk | |
| Purpose | ▪ Example of a curved road with different surface conditions on the left and right sides.<br>▪ Example of how you can model a mue-split curved road. |
| Demo scenarios using this road | MueSplitCurve Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 1 |

| MueSplitCurve Road | |
|---|---|
| Number of road elements | 1 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.3 km<br>▪ Width: 0.23 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the MueSplitCurve road:

| MueSplitCurve Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

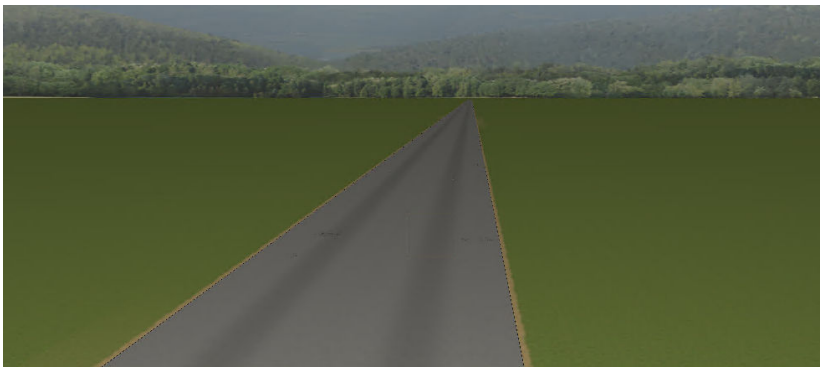
**Related topics**

References

MueSplitCurve Scenario..... 137

# NCAP_AEB Road

**Description** The following table shows information on the NCAP_AEB road:

| **NCAP_AEB Road** | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk |  |
| Purpose | ▪ Example of a possible test track for testing AEB systems according to Euro NCAP test protocols.<br>▪ Appropriate for testing AEB systems according to the *Euro NCAP AEB VRU* and *Euro NCAP AEB C2C* Euro NCAP test protocols. |
| Demo scenarios using this road | None |
| Usage of height profiles | No |
| Used scenery types | Batter |
| EuroNCAP road | Yes |
| Number of routes | 1 |
| Number of road elements | 1 |
| Number of junctions | 0 |
| Usage of traffic lights | No |
| Usage of traffic signs | No |
| Usage of position markers | No |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 2 km<br>▪ Width: 10 m |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the NCAP_AEB road:

| | NCAP_AEB Road |
|---|---|
| **ModelDesk Shapes** | |
| Usage of trajectories | ▪ Road: No<br>▪ Junction: No |
| Usage of lines | ▪ Road: No<br>▪ Junction: No |
| Usage of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Usage of repeating objects | ▪ Road: No<br>▪ Junction: No |

# NCAP_LSS Road

**Description**

The following table shows information on the NCAP_LSS road:

| | NCAP_LSS Road |
|---|---|
| **General Information** | |
| Overview ModelDesk | |
| Overview MotionDesk | |
| Purpose | ▪ Example of a possible test track with solid lines, derived from the Euro NCAP Lane Support systems test protocol (Version 2.0.1). |

| | NCAP_LSS Road |
|---|---|
| | ▪ Appropriate for testing lane support systems according to Euro NCAP Lane Support systems test protocol (Version 2.0.1).<br>▪ Includes 12 free trajectories or NCAP test paths that can be used to demonstrate unintentional and intentional lane departures to the right- and the left-hand side and as a result potential accidents involving oncoming or overtaking traffic. |
| Demo scenarios using this road | None |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | Yes |
| Number of routes | 13 |
| Number of road elements | 1 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 1.5 km<br>▪ Width: 0.008 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

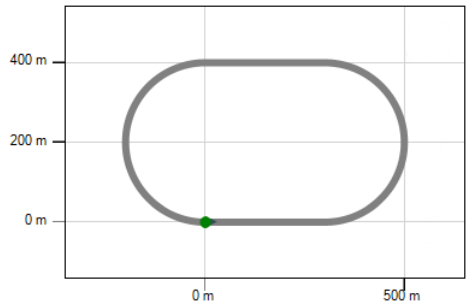
**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the NCAP_LSS road:

| | NCAP_LSS Road |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: Yes<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

# OvalWide Road

**Description** The following table shows information on the OvalWide road:

| **OvalWide Road** | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk | |
| Purpose | Example of a closed road with five lanes. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 1 |
| Number of road elements | 1 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.6 km<br>▪ Width: 0.4 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the OvalWide road:

| OvalWide Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

**Related topics**

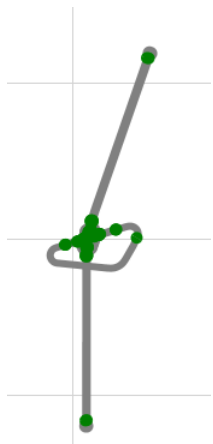
References

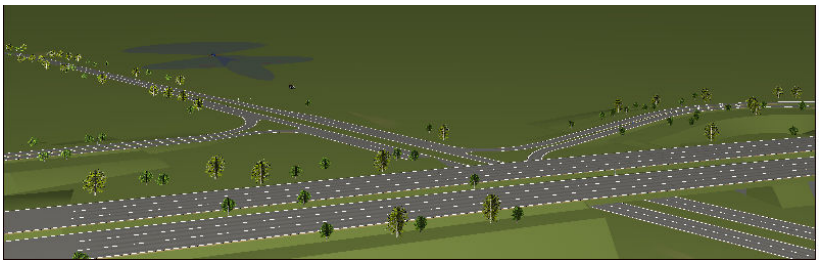
FollowRoadEndless Scenario..... 129

# RoadNetwork_Highway_Interchange_PB Road

**Description**

The following table shows information on the RoadNetwork_Highway_Interchange_PB road:

| RoadNetwork_Highway_Interchange_PB Road | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |

| RoadNetwork_Highway_Interchange_PB Road | |
|---|---|
| Overview MotionDesk |  |
| Purpose | ▪ Example of how you can model a road network with a highway.<br>▪ Example of a road network with multiple road and junction elements. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | Yes |
| Used scenery types | ▪ Country Road<br>▪ Bridge |
| Euro NCAP road | No |
| Number of routes | 4 |
| Number of road elements | 32 |
| Number of junctions | 19 |
| Use of traffic lights | No |
| Use of traffic signs | Yes |
| Use of position markers | Yes |
| Number of traffic lights/signs | 12 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 1 km<br>▪ Width: 4.5 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the RoadNetwork_Highway_Interchange_PB road:

| RoadNetwork_Highway_Interchange_PB Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

**Related topics**

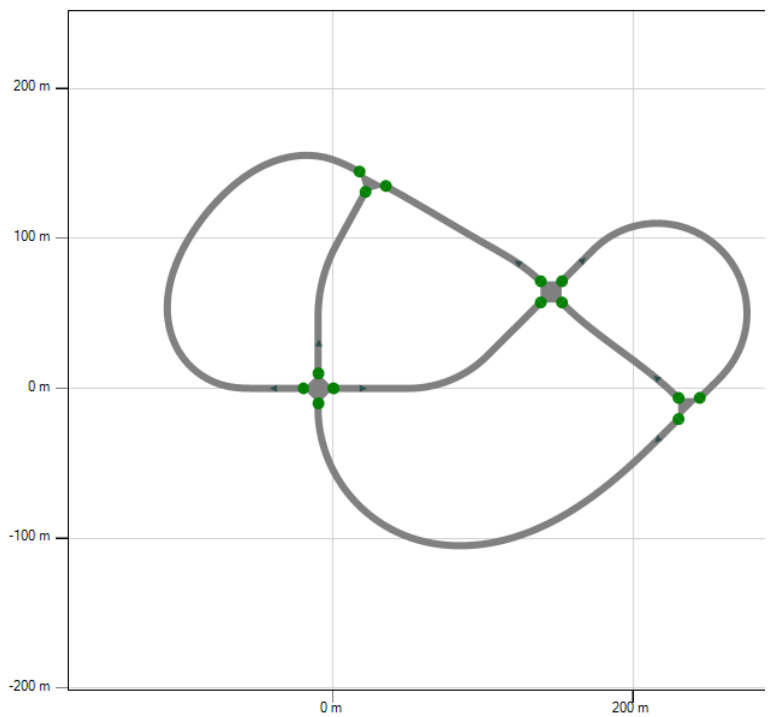
References

FollowRoadEndless Scenario..... 129

# RoadNetworkBasic Road

**Description**

The following table shows information on the RoadNetworkBasic road:

| RoadNetworkBasic Road | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |

| RoadNetworkBasic Road | |
|---|---|
| Overview MotionDesk | |
| Purpose | Example of a small basic road network. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 8 |
| Number of road elements | 7 |
| Number of junctions | 4 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.4 km<br>▪ Width: 0.25 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the RoadNetworkBasic road:

| RoadNetworkBasic Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

**Related topics**

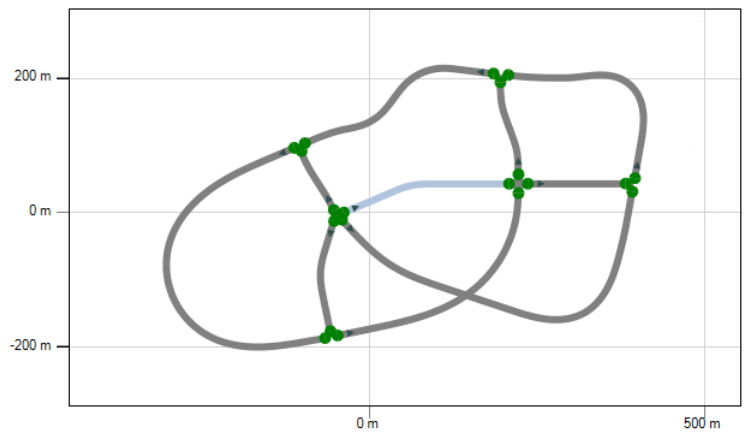
References

FollowRoadEndless Scenario..... 129

# RoadNetworkRural Road

**Description**

The following table shows information on the RoadNetworkRural road:

| | **RoadNetworkRural Road** |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk | |
| Purpose | Example of a road network with street lamps, traffic signs, and trees. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | Yes |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 8 |

| RoadNetworkRural Road | |
|---|---|
| Number of road elements | 10 |
| Number of junctions | 6 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 12 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.7 km<br>▪ Width: 0.4 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the RoadNetworkRural road:

| RoadNetworkRural Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

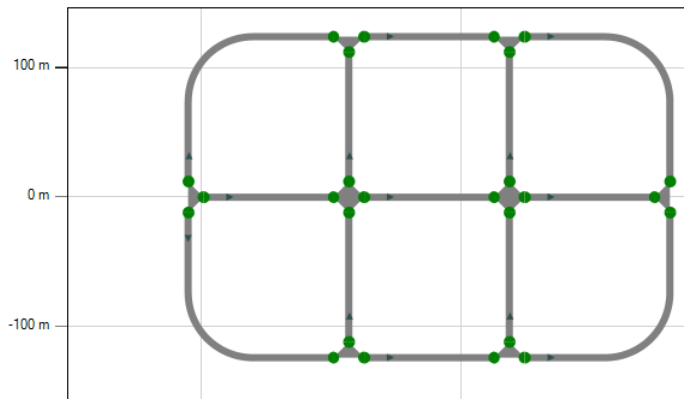
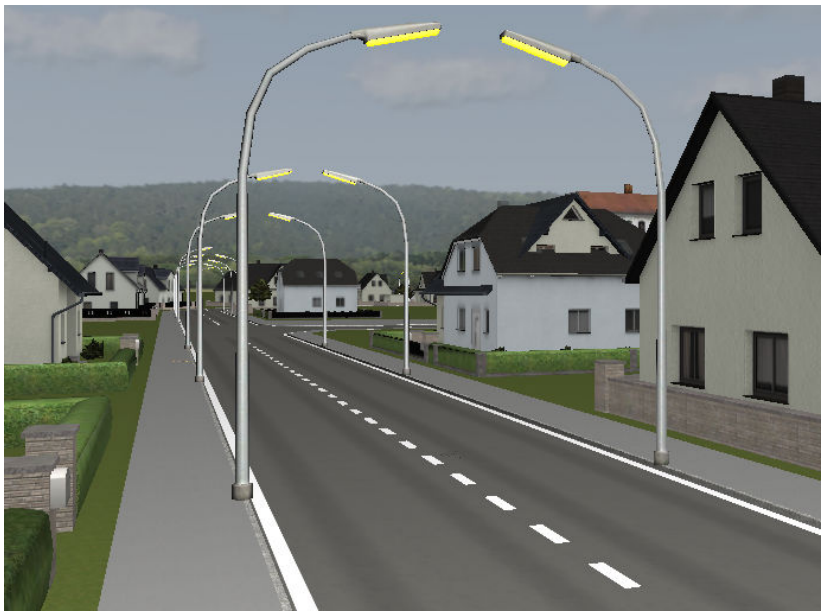
**Related topics**

References

FollowRoadEndless Scenario..... 129

# RoadNetwork_Urban Road

**Description** The following table shows information on the RoadNetwork_Urban road:

| **RoadNetwork_Urban Road** | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk |  |
| Purpose | Example of a road network with street lamps, sideways, and buildings. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 9 |
| Number of road elements | 13 |
| Number of junctions | 8 |

| RoadNetwork_Urban Road | |
|---|---|
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.7 km<br>▪ Width: 0.4 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the RoadNetwork_Urban road:

| RoadNetwork_Urban Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

**Related topics**

References

FollowRoadEndless Scenario..... 129

# RoadWithParkingLane Road

**Description**

The following table shows information on the RoadWithParkingLane road:

| RoadWithParkingLane Road | |
|---|---|
| **General Information** | |
| Overview ModelDesk | |

| RoadWithParkingLane Road | |
|---|---|
| Overview MotionDesk | |
| Purpose | ▪ Example of how you can model a road network for parking maneuvers.<br>▪ Example of a road network with a different number of lanes along the road.<br>▪ Example of how you can use a marker in a road network. |
| Demo scenarios using this road | None |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 1 |
| Number of road elements | 1 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | Yes |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 1 km<br>▪ Width: 0.01 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the RoadWithParkingLane road:

| RoadWithParkingLane Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |

| RoadWithParkingLane Road | |
|---|---|
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

## RoadWorks_Basic Road

**Description** The following table shows information on the RoadWorks_Basic road:

| RoadWorks_Basic Road | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk | |
| Purpose | ▪ Example of how you can model a road network with roadworks.<br>▪ Example of how you can use free trajectories.<br>▪ Example of how you can use several lines, repeating objects, or static traffic objects. |
| Demo scenarios using this road | ▪ FollowRoadEndless Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |

| RoadWorks_Basic Road | |
|---|---|
| Number of routes | 2 |
| Number of road elements | 1 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 2 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.7 km<br>▪ Width: 0.01 km |

1) Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the RoadWorks_Basic road:

| RoadWorks_Basic Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: Yes<br>▪ Junction: No |
| Use of lines | ▪ Road: Yes<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: Yes<br>▪ Junction: No |

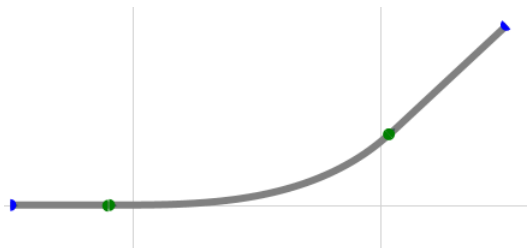
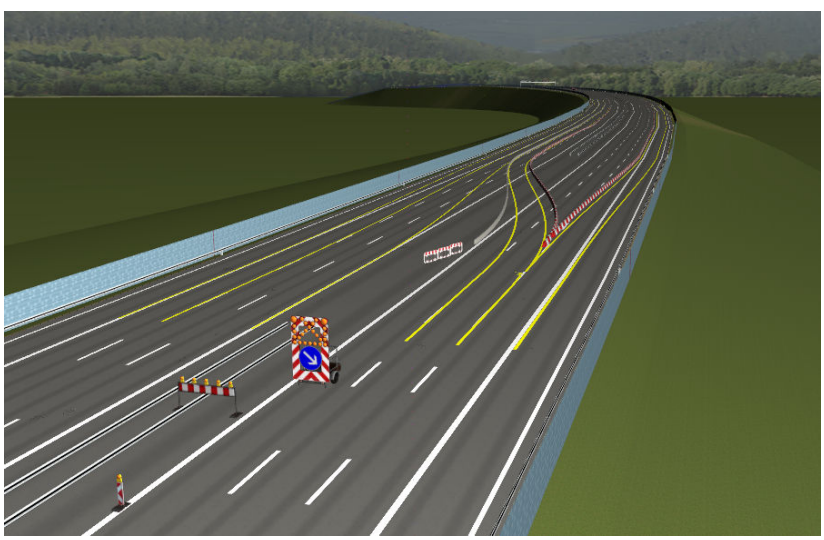
**Related topics**

References

FollowRoadEndless Scenario..... 129

# RoadWorks_Highway Road

**Description** The following table shows information on the RoadWorks_Highway road:

| **RoadWorks_Highway Road** | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk |  |
| Purpose | ▪ Example of how you can model roadworks on a highway.<br>▪ Example on how you can use free trajectories on roads.<br>▪ Example of a road with non-drivable lanes.<br>▪ Example of a road with a considerable number of static and continuous objects. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | Yes |
| Used scenery types | Highway |
| Euro NCAP road | No |
| Number of routes | 4 |
| Number of road elements | 3 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | Yes |

| RoadWorks_Highway Road | |
|---|---|
| Use of position markers | No |
| Number of traffic lights/signs | 36 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 1.51 km<br>▪ Width: 0.73 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the RoadWorks_Highway road:

| RoadWorks_Highway Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: Yes<br>▪ Junction: No |
| Use of lines | ▪ Road: Yes<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: Yes<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

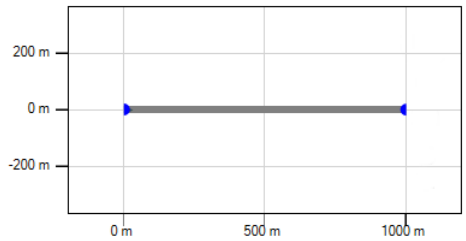
**Related topics**

References

FollowRoadEndless Scenario..... 129

# Slalom Road

**Description** The following table shows information on the Slalom road:

| | **Slalom Road** |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk | |
| Purpose | ▪ Example of how you can model a slalom maneuver with several pylons.<br>▪ Example of how you can use free trajectories. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 2 |
| Number of road elements | 1 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 2 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.7 km |

| | Slalom Road |
|---|---|
| | ▪ Width: 0.01 km |

[1] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the Slalom road:

| | Slalom Road |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: Yes<br>▪ Junction: No |
| Use of lines | ▪ Road: Yes<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: Yes<br>▪ Junction: No |

**Related topics**

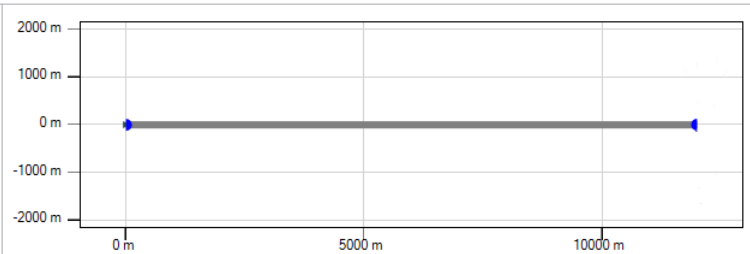
References

FollowRoadEndless Scenario..... 129

# StraightLong Road

**Description**

The following table shows information on the StraightLong road:

| | StraightLong Road |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |

| | **StraightLong Road** |
|---|---|
| Overview MotionDesk | |
| Purpose | Example of a long road with two lanes. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 1 |
| Number of road elements | 1 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 13 km<br>▪ Width: 15 m |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the StraightLong road:

| | **StraightLong Road** |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |

| | StraightLong Road |
|---|---|
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

**Related topics**

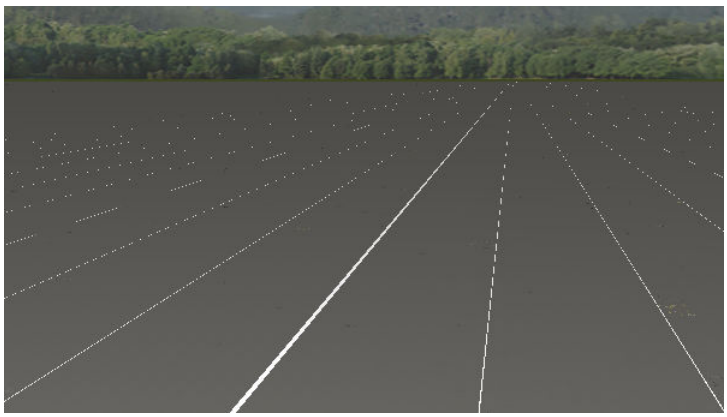
References

FollowRoadEndless Scenario..... 129

# TestGround Road

**Description**

The following table shows information on the TestGround road:

| | TestGround Road |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk |  |
| Purpose | Example of a road with a flat and wide surface that does not include objects or obstacles in the surrounding area. |
| Demo scenarios using this road | ▪ FMVSS_126_SIS Scenario<br>▪ FMVSS_126_SWD Scenario<br>▪ SineSteer Scenario<br>▪ SteadyStateCornering Scenario<br>▪ StepSteer Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |

| TestGround Road | |
|---|---|
| Number of routes | 1 |
| Number of road elements | 2 |
| Number of junctions | 0 |
| Use of traffic lights | No |
| Use of traffic signs | No |
| Use of position markers | No |
| Number of traffic lights/signs | 0 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 10 km<br>▪ Width: 10 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

## ModelDesk shapes

The following table shows information on ModelDesk shapes of the TestGround road:

| TestGround Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

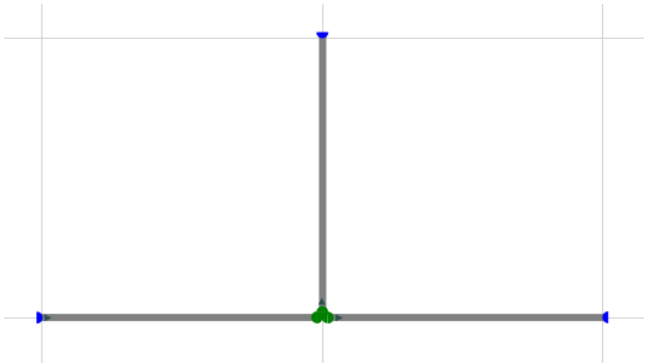
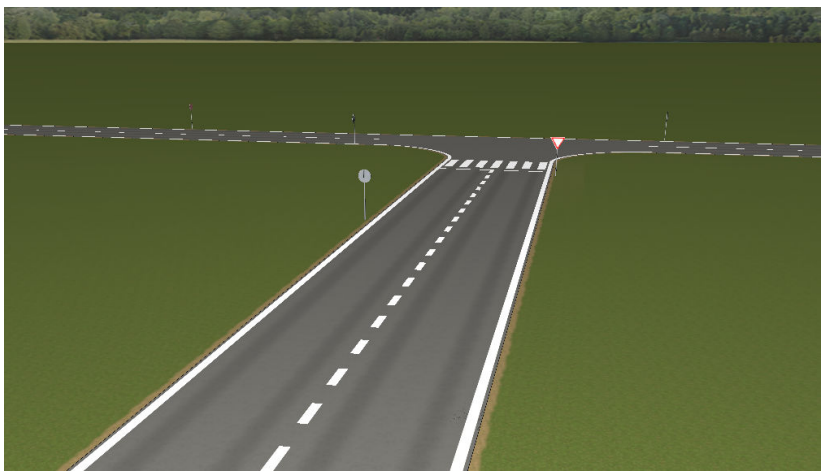
## Related topics

References

FMVSS_126_SIS Scenario..... 126
FMVSS_126_SWD Scenario..... 128
SineSteer Scenario..... 140
SteadyStateCornering Scenario..... 144
StepSteer Scenario..... 146

# ThreeWayJunction_2Lanes Road

**Description** The following table shows information on the ThreeWayJunction_2Lanes road:

| **ThreeWayJunction_2Lanes Road** | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk |  |
| Purpose | Example of how you can model a three-way junction with traffic signs. |
| Demo scenarios using this road | FollowRoadEndless Scenario |
| Use of height profiles | No |
| Used scenery types | Batter |
| Euro NCAP road | No |
| Number of routes | 3 |
| Number of road elements | 3 |
| Number of junctions | 1 |
| Use of traffic lights | No |
| Use of traffic signs | Yes |
| Use of position markers | 3 |
| Number of traffic lights/signs | 0 |

| ThreeWayJunction_2Lanes Road | |
|---|---|
| Real map data[1)] | No |
| Network dimension | ▪ Length: 1 km<br>▪ Width: 0.5 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the ThreeWayJunction_2Lanes road:

| ThreeWayJunction_2Lanes Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

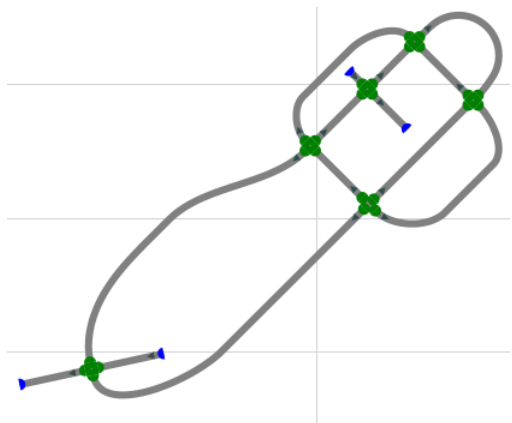
**Related topics**

References

FollowRoadEndless Scenario..... 129

# TrafficDriverBasic Road

**Description** The following table shows information on the TrafficDriverBasic road:

| **TrafficDriverBasic Road** | |
|---|---|
| **General Information** | |
| Overview ModelDesk |  |
| Overview MotionDesk | |
| Purpose | Example of a small road network that can be used to show the traffic driver features. |
| Demo scenarios using this road | TrafficDriver_30Fellows |
| Use of height profiles | No |
| Used scenery types | ▪ Country Road<br>▪ Bridge |
| Euro NCAP road | No |
| Number of routes | 2 |
| Number of road elements | 14 |
| Number of junctions | 7 |
| Use of traffic lights | Yes |
| Use of traffic signs | Yes |

| TrafficDriverBasic Road | |
|---|---|
| Use of position markers | No |
| Number of traffic lights/signs | 68 |
| Real map data[1)] | No |
| Network dimension | ▪ Length: 0.8 km<br>▪ Width: 0.3 km |

[1)] Real map data: The road network exists in the real world and real map data is imported to ModelDesk for network creation. The origin of the coordinate system matches the GPS coordinate system (required for using the ControlDesk Map instrument).

**ModelDesk shapes**

The following table shows information on ModelDesk shapes of the TrafficDriverBasic road:

| TrafficDriverBasic Road | |
|---|---|
| **ModelDesk Shapes** | |
| Use of free trajectories | ▪ Road: No<br>▪ Junction: No |
| Use of lines | ▪ Road: No<br>▪ Junction: No |
| Use of continuous objects | ▪ Road: No<br>▪ Junction: No |
| Use of repeating objects | ▪ Road: No<br>▪ Junction: No |

# Scenarios

**Where to go from here**

Information in this section

ABSBrake Scenario..... 120
Shows information on the ABSBrake scenario.

ACCPedalStimulus100 Scenario..... 122
Shows information on the ACCPedalStimulus100 scenario.

Fishhook Scenario..... 124
Shows information on the Fishhook scenario.

FMVSS_126_SIS Scenario..... 126
Shows information on the FMVSS_126_SIS scenario.

FMVSS_126_SWD Scenario..... 128
Shows information on the FMVSS_126_SWD scenario.

FollowRoadEndless Scenario..... 129
Shows information on the FollowRoadEndless scenario.

FTP75 Cycle Scenario..... 131
Shows information on the FTP75 Cycle scenario.

LaneChange Scenario..... 132
Shows information on the LaneChange scenario.

ManualDriving_WithRoad Scenario..... 134
Shows information on the ManualDriving_WithRoad scenario.

MueSplit Scenario..... 135
Shows information on the MueSplit scenario.

MueSplitCurve Scenario..... 137
Shows information on the MueSplitCurve scenario.

NCAP_LSS Scenario..... 138
Shows information on the NCAP_LSS scenario.

SineSteer Scenario..... 140
Shows information on the SineSteer scenario.

Slalom Scenario..... 142
Shows information on the Slalom scenario.

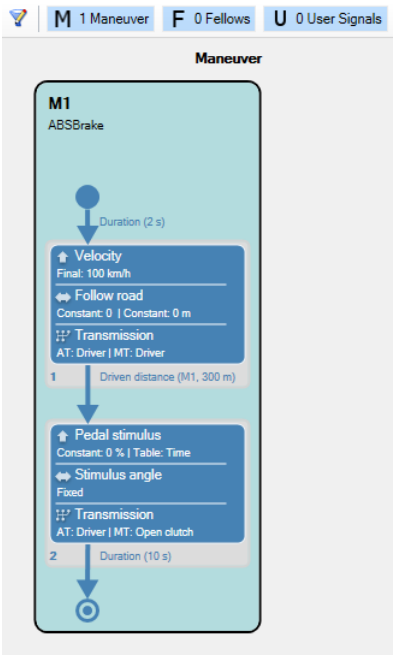
SteadyStateCornering Scenario..... 144
Shows information on the SteadyStateCornering scenario.

StepSteer Scenario..... 146
Shows information on the StepSteer scenario.

# ABSBrake Scenario

**Description**

The following table shows information on the ABSBrake scenario:

| ABSBrake Scenario | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | The ASM vehicle travels straight forward with a longitudinal speed of 100 km/h. After a traveled distance of 300 m, a full braking is performed for about 10 seconds. |
| Purpose | Demonstration of a full braking scenario on a slippery road. |
| Referenced road | The corresponding road is the ABSTrack.rd. |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the ABSBrake scenario:

| ABSBrake Scenario | |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | 1 |
| Number of segments | 1 |
| Use of maneuver user signals | No |
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |

| | ABSBrake Scenario |
|---|---|
| Used transition conditions | ▪ Duration<br>▪ Distance [Vehicle] |

**Parameters of the fellows**

The following table shows information on parameters of the fellows of the ABSBrake scenario:

| | ABSBrake Scenario |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

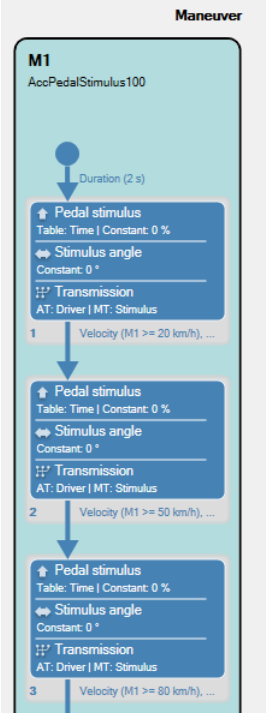
**Related topics**

References

ABSTrack Road ..... 76

# ACCPedalStimulus100 Scenario

**Description** The following table shows information on the ACCPedalStimulus100 scenario:

| **ACCPedalStimulus100 Scenario** | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | This maneuver stimulates all the driver inputs of the ASM vehicle.<br>To perform six-gear-switching operations for enabling acceleration from standstill to maximum speed, six maneuver segments specify the following tasks:<br>▪ Actuation of the gear shifter<br>▪ Acceleration<br>▪ Action of the brake pedal<br>▪ Action of the clutch pedal<br>The transmission type is manual.<br>The maneuver utilizes transition conditions of `Velocity` type to transition from one maneuver segment to the next. Each maneuver segment triggers the use of a different gear.<br>The steering wheel is also stimulated using a constant angle of zero. Therefore you can use only straight roads for this scenario. |
| Purpose | ▪ Demonstration of stimulus functionalities.<br>▪ Use of additional segment end conditions. |
| Referenced road | None |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the ACCPedalStimulus100 scenario:

| ACCPedalStimulus100 Scenario | |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | None |
| Number of segments | 6 |
| Use of maneuver user signals | Yes |
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | ▪ Duration<br>▪ Velocity |

**Parameters of the fellows**

The following table shows information on parameters of the fellows of the ACCPedalStimulus100 scenario:

| ACCPedalStimulus100 Scenario | |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

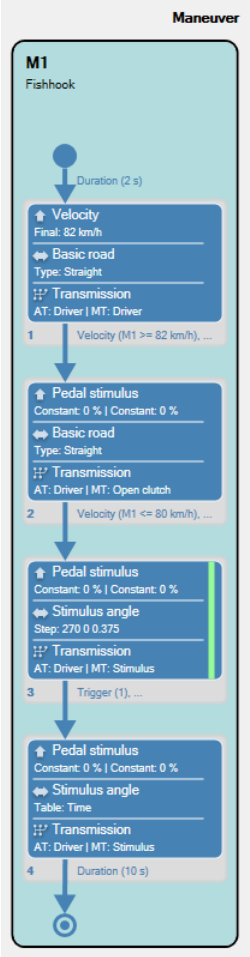
**Global user signals**

The following table shows information on the global user signals of the ACC_CutIn scenario:

| Global User Signals | |
|---|---|
| Number of ASM global user signals | 0 |

# Fishhook Scenario

**Description** The following table shows information on the Fishhook scenario:

| Fishhook Scenario | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | The ASM vehicle is accelerated to the velocity 82 km/h while traveling in a straight forward direction, which is slightly higher than the target velocity.<br>When the ASM vehicle reaches the velocity value, the maneuver releases the accelerator and the brake pedals, i.e., the positions are set to 0 %. As a result, the vehicle coasts.<br>When the velocity drops below 80 km/h, the steering angle increases by 720 deg/s. By means of two defined maneuver user signals, an external trigger excites the last segment, which includes steering in the opposite direction and afterwards back to the steering angle 0°. |
| Purpose | Demonstration of the fishhook maneuver. |
| Referenced road | None |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the Fishhook scenario:

| Fishhook Scenario | |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | 1 |
| Number of segments | 4 |
| Use of maneuver user signals | Yes |
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | ▪ Duration<br>▪ Velocity<br>▪ Trigger |

**Parameters of the fellows**

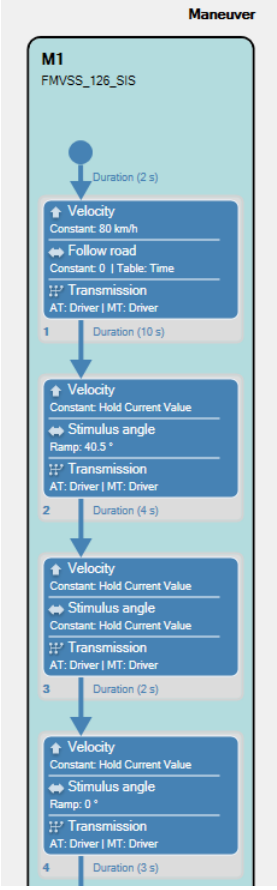
The following table shows information on parameters of the fellows of the Fishhook scenario:

| Fishhook Scenario | |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

# FMVSS_126_SIS Scenario

**Description**

The following table shows information on the FMVSS_126_SIS scenario:

| FMVSS_126_SIS Scenario | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | Implementation of the slowly increasing FMVSS 126 steering maneuver. |
| Purpose | Implementation of the slowly increasing FMVSS 126 steering maneuver. |
| Referenced road | None |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the FMVSS_126_SIS scenario:

| FMVSS_126_SIS Scenario | |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | None |
| Number of segments | 1 |

| FMVSS_126_SIS Scenario | |
|---|---|
| Use of maneuver user signals | No |
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | Duration |

**Parameters of the fellows**

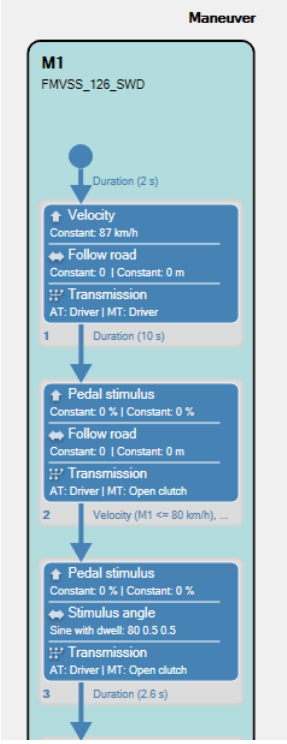
The following table shows information on parameters of the fellows of the FMVSS_126_SIS scenario:

| FMVSS_126_SIS Scenario | |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

# FMVSS_126_SWD Scenario

**Description**

The following table shows information on the FMVSS_126_SWD scenario:

| FMVSS_126_SWD Scenario | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | Implementation of the FMVSS 126 sine with dwell maneuver. |
| Purpose | Implementation of the FMVSS 126 sine with dwell maneuver. |
| Referenced road | None |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the FMVSS_126_SWD scenario:

| FMVSS_126_SWD Scenario | |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | None |
| Number of segments | 1 |
| Use of maneuver user signals | No |
| Reverse direction | No |

| FMVSS_126_SWD Scenario | |
|---|---|
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | Duration |

**Parameters of the fellows**

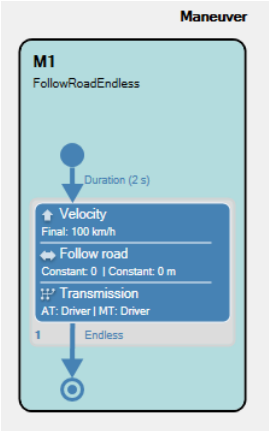
The following table shows information on parameters of the fellows of the FMVSS_126_SWD scenario:

| FMVSS_126_SWD Scenario | |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

# FollowRoadEndless Scenario

**Description**

The following table shows information on the FollowRoadEndless scenario:

| FollowRoadEndless Scenario | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | The ASM vehicle accelerates to 100 km/h. While the vehicle is traveling at this constant velocity, it follows the course of the road without stopping. |
| Purpose | Demonstration of an endless maneuver |
| Referenced road | The corresponding roads are:<br>▪ HandlingTrack.rd.<br>▪ FourwayJunction_2Lanes.rd<br>▪ FourwayJunction_4Lanes.rd<br>▪ HandlingTrack.rd<br>▪ Highway_A33_Paderborn.rd |

| FollowRoadEndless Scenario | |
|---|---|
| | ▪ HockenheimringTrack.rd<br>▪ OvalWide.rd (refer to OvalWide Road on page 96)<br>▪ RoadNetwork_Highway_Interchange_PB.rd<br>▪ RoadNetworkBasic.rd<br>▪ RoadNetworkRural.rd<br>▪ RoadNetwork_Urban.rd<br>▪ RoadWorks_Basic.rd<br>▪ RoadWorks_Highway<br>▪ Slalom.rd<br>▪ StraightLong.rd<br>▪ ThreeWayJunction_2Lanes.rd |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the FollowRoadEndless scenario:

| FollowRoadEndless Scenario | |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | 1 |
| Number of segments | 1 |
| Use of maneuver user signals | No |
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | ▪ Duration<br>▪ Endless |

**Parameters of the fellows**

The following table shows information on parameters of the fellows of the FollowRoadEndless scenario:

| FollowRoadEndless Scenario | |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

**Related topics**

References

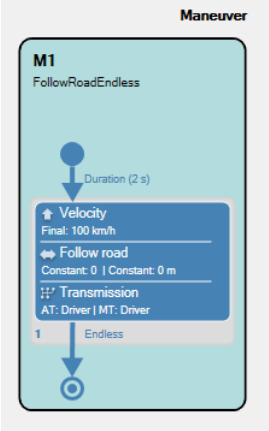
FourwayJunction_2Lanes Road.....78
FourwayJunction_4Lanes Road.....80
HandlingTrack Road.....82
Highway_A33_Paderborn Road.....84
HockenheimringTrack Road.....86
RoadNetwork_Highway_Interchange_PB Road.....97
RoadNetwork_Urban Road.....103

RoadNetworkBasic Road....99
RoadNetworkRural Road....101
RoadWorks_Basic Road....106
RoadWorks_Highway Road....108
Slalom Road....110
StraightLong Road....111
ThreeWayJunction_2Lanes Road....115

# FTP75 Cycle Scenario

**Description**

The following table shows information on the FTP75 Cycle scenario:

| FTP75 Cycle Scenario | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | The maneuver specifies the reference velocity of the ASM vehicle. In this case, the reference velocity is specified by a table and relates to the FTP75 cycle. |
| Purpose | ▪ Demonstration of the use of tables for setting the reference velocity of the ASM vehicle.<br>▪ Implementation of the FTP 75 cycle. |
| Referenced road | None |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the FTP75 Cycle scenario:

| FTP75 Cycle Scenario | |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | None |
| Number of segments | 1 |

| FTP75 Cycle Scenario | |
|---|---|
| Use of maneuver user signals | No |
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | Duration |

**Parameters of the fellows**

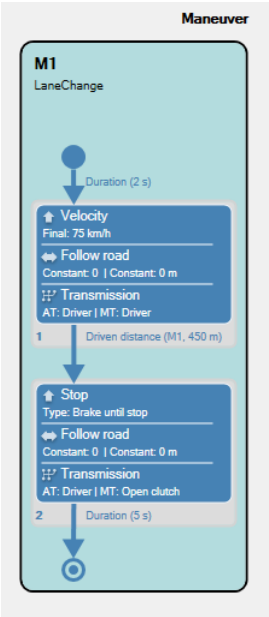
The following table shows information on parameters of the fellows of the FTP75 Cycle scenario:

| FTP75 Cycle Scenario | |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

# LaneChange Scenario

**Description**

The following table shows information on the LaneChange scenario:

| LaneChange Scenario | |
|---|---|
| **General Information** | |
| Overview |  |

| LaneChange Scenario | |
|---|---|
| Description | The ASM vehicle accelerates to 75 km/h. While the vehicle is traveling at this constant velocity, it follows the course of the road. Due to the changing horizontal profile of the road, the vehicle performs a lateral movement that corresponds to a double lane change with high lateral accelerations. After about 450 m, the vehicle stops. |
| Purpose | Demonstration of a vehicle movement that corresponds to a double lane change. |
| Referenced road | The corresponding road is the LaneChange.rd. |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the LaneChange scenario:

| LaneChange Scenario | |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | 1 |
| Number of segments | 2 |
| Use of maneuver user signals | No |
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | ▪ Duration<br>▪ Driven Distance |

**Parameters of the Fellows**

The following table shows information on parameters of the fellows of the LaneChange scenario:

| LaneChange Scenario | |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

**Related topics**

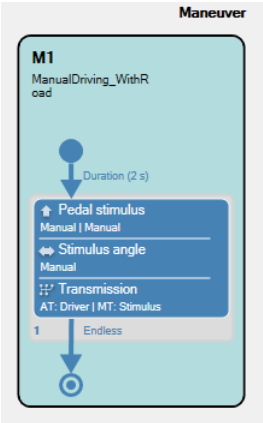
References

LaneChange Road ..... 87

# ManualDriving_WithRoad Scenario

**Description**

The following table shows information on the ManualDriving_WithRoad scenario:

| ManualDriving_WithRoad Scenario | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | The movement of the ASM vehicle is controlled by stimulus signals that you define manually and that affect the accelerator pedal, brake pedal, and the steering wheel. |
| Purpose | Demonstration of a manually controlled vehicle. |
| Referenced road | The corresponding road is the StraightLong.rd. |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the ManualDriving_WithRoad scenario:

| ManualDriving_WithRoad Scenario | |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | 1 |
| Number of segments | 1 |
| Use of maneuver user signals | No |
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | ▪ Duration<br>▪ Endless |

**Parameters of the fellows**

The following table shows information on parameters of the fellows of the ManualDriving_WithRoad scenario:

| ManualDriving_WithRoad Scenario | |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

**Related topics**

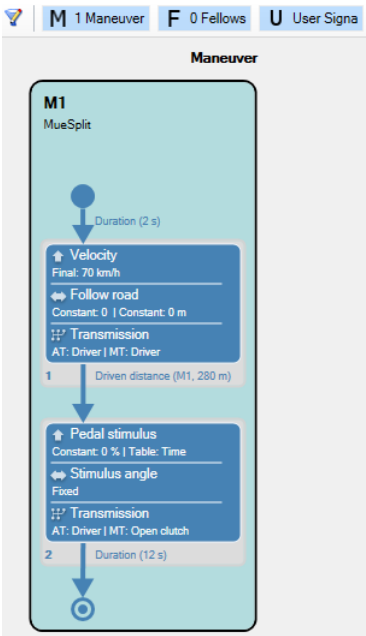
References

StraightLong Road..... 111

# MueSplit Scenario

**Description**

The following table shows information on the MueSplit scenario:

| MueSplit Scenario | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | The ASM vehicle travels straight ahead at a final velocity of 70 km/h. After traveling a distance of 280 m, a full braking action is performed for about 12 seconds. |

| | MueSplit Scenario |
|---|---|
| Purpose | Demonstration of a full braking scenario with different frictions on the left and right sides of the road. |
| Referenced road | The corresponding road is the MueSplit.rd. |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the MueSplit scenario:

| | MueSplit Scenario |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | 1 |
| Number of segments | 1 |
| Use of maneuver user signals | No |
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | ▪ Duration<br>▪ Distance [Vehicle] |

**Parameters of the fellows**

The following table shows information on parameters of the fellows of the MueSplit scenario:

| | MueSplit Scenario |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

**Related topics**

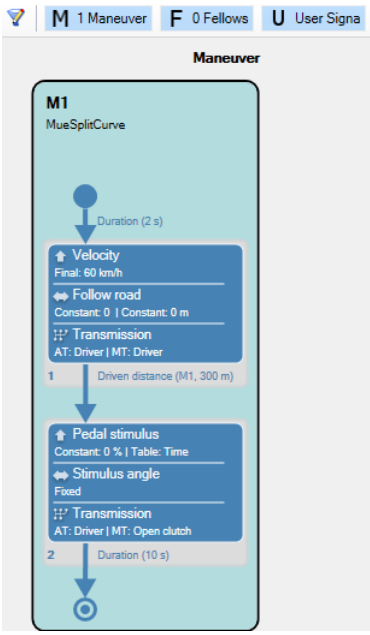
References

MueSplit Road..... 89

# MueSplitCurve Scenario

**Description**

The following table shows information on the MueSplitCurve scenario:

| MueSplitCurve Scenario | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | The ASM vehicle travels straight ahead and follows the road at a final velocity of 60 km/h. After traveling a distance of 300 m, a full braking action is performed for about 10 seconds. |
| Purpose | Demonstration of a full braking scenario with different frictions on the left and right sides of a curved road. |
| Referenced road | The corresponding road is the MueSplitCurve.rd. |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the MueSplitCurve scenario:

| MueSplitCurve Scenario | |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | 1 |
| Number of segments | 1 |
| Use of maneuver user signals | No |
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |

| | MueSplitCurve Scenario |
|---|---|
| Used transition conditions | ▪ Duration<br>▪ Distance [Vehicle] |

**Parameters of the fellows**

The following table shows information on parameters of the fellows of the MueSplitCurve scenario:

| | MueSplitCurve Scenario |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

**Related topics**

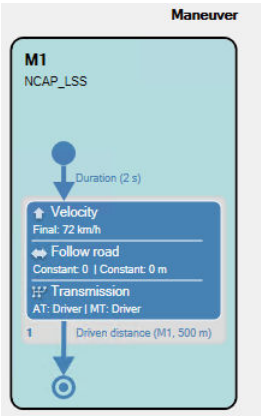
References

MueSplitCurve Road.....91

# NCAP_LSS Scenario

**Description**

The following table shows information on the NCAP_LSS scenario:

| | NCAP_LSS Scenario |
|---|---|
| **General Information** | |
| Overview |  |
| Description | The ASM vehicle travels along a defined Euro NCAP LSS test path or shape, which triggers a lane change to the left hand-side. The ASM vehicle accelerates to the target velocity of 72 km/h.<br>Afterwards, it continues driving with this constant target velocity, while it is following the Euro NCAP LSS test path or shape. |

| | NCAP_LSS Scenario |
|---|---|
| | During the lane change, the lateral velocity is 0.1 m/s.<br> |
| Purpose | ▪ Scenario according to the Euro NCAP Lane Support systems test protocol (Version 2.0.1), for testing lane support systems (LSS), such as lane keep assist (LKA) or lane departure warning (LDW).<br>▪ Demonstration of unintentional lane departure. |
| Referenced road | The corresponding road is the NCAP_LSS.rd. |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the NCAP_LSS scenario:

| | NCAP_LSS Scenario |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | 1 |
| Number of segments | 1 |
| Use of maneuver user signals | No |
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: Yes<br>▪ On junctions: No |
| Used transition conditions | ▪ Duration<br>▪ Driven distance |

**Parameters of the fellows**

The following table shows information on parameters of the fellows of the NCAP_LSS scenario:

| | NCAP_LSS Scenario |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

**Global user signals**

The following table shows information on the global user signals of the ACC_CutIn scenario:

| **Global User Signals** | |
|---|---|
| Number of ASM global user signals | 0 |

**Related topics**

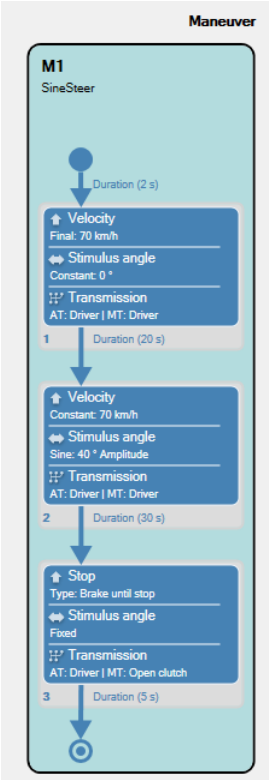
References

NCAP_LSS Road.....94

# SineSteer Scenario

**Description**

The following table shows information on the SineSteer scenario:

| **SineSteer Scenario** | |
|---|---|
| **General Information** | |
| Overview |  |

| | SineSteer Scenario |
|---|---|
| Description | The ASM vehicle accelerates to 70 km/h. In the second scenario segment, the maneuver uses a sine-sweep signal to stimulate the steering wheel angle movement for about 30 seconds. |
| Purpose | Demonstration of a sine-sweep maneuver. |
| Referenced road | The corresponding road is the TestGround.rd. |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the SineSteer scenario:

| | SineSteer Scenario |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | 1 |
| Number of segments | 1 |
| Use of maneuver user signals | No |
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | Duration |

**Parameters of the fellows**

The following table shows information on parameters of the fellows of the SineSteer scenario:

| | SineSteer Scenario |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

**Related topics**

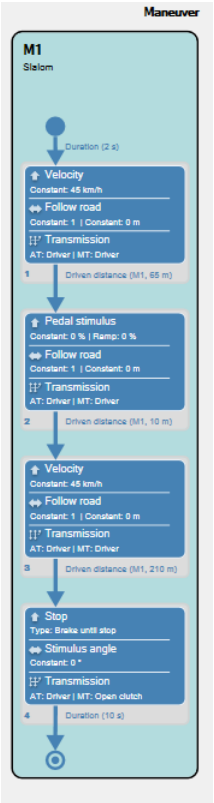
References

TestGround Road..... 113

# Slalom Scenario

**Description**

The following table shows information on the Slalom scenario:

| Slalom Scenario | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | The ASM vehicle accelerates to 45 km/h. After traveling a distance of 75 m, the driver follows the pre-defined trajectory to drive through the pylons. |
| Purpose | Demonstration of a slalom maneuver. |
| Referenced road | The corresponding road is the Slalom.rd. |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the Slalom scenario:

| Slalom Scenario | |
|---|---|
| **Parameters of the maneuver** | |
| Route index | 1 |
| Number of segments | 1 |
| Use of maneuver user signals | No |

| | Slalom Scenario |
|---|---|
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | ▪ Duration<br>▪ Distance [Vehicle] |

**Parameters of the fellows**

The following table shows information on parameters of the fellows of the Slalom scenario:

| | Slalom Scenario |
|---|---|
| **Parameters of the fellows** | |
| Number of fellows | 0 |

**Related topics**

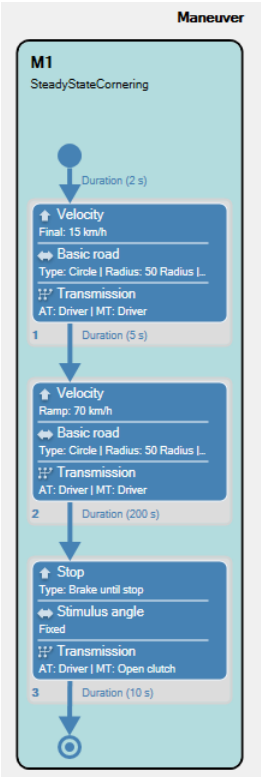
References

Slalom Road..... 110

# SteadyStateCornering Scenario

**Description**

The following table shows information on the SteadyStateCornering scenario:

| SteadyStateCornering Scenario | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | The ASM vehicle accelerates to 15 km/h. In the second scenario segment, the vehicle is accelerated to 70 km/h within 200 seconds and a constant curve radius of 50 m is maintained. |
| Purpose | Demonstration of a steady state cornering maneuver. |
| Referenced road | The corresponding road is the TestGround.rd. |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the SteadyStateCornering scenario:

| SteadyStateCornering Scenario | |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | 1 |
| Number of segments | 1 |
| Use of maneuver user signals | No |

| SteadyStateCornering Scenario | |
|---|---|
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | Duration |

**Parameters of the fellows**

The following table shows information on parameters of the fellows of the SteadyStateCornering scenario:

| SteadyStateCornering Scenario | |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

**Related topics**

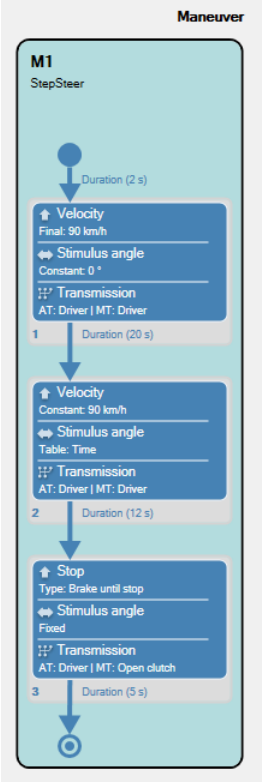
References

TestGround Road..... 113

# StepSteer Scenario

**Description**

The following table shows information on the StepSteer scenario:

| StepSteer Scenario | |
|---|---|
| **General Information** | |
| Overview |  |
| Description | The ASM vehicle accelerates to 90 km/h. In the second scenario segment, the maneuver uses a step signal to stimulate the steering wheel angle movement for about 12 seconds. |
| Purpose | Demonstration of a step-steer maneuver. |
| Referenced road | The corresponding road is the TestGround.rd. |

**Parameters of the maneuver**

The following table shows information on the parameters of the maneuver of the StepSteer scenario:

| StepSteer Scenario | |
|---|---|
| **Parameters of the Maneuver** | |
| Route index | 1 |
| Number of segments | 1 |
| Use of maneuver user signals | No |

| StepSteer Scenario | |
|---|---|
| Reverse direction | No |
| Use of free trajectories | ▪ On roads: No<br>▪ On junctions: No |
| Used transition conditions | Duration |

**Parameters of the fellows**

The following table shows information on parameters of the fellows of the StepSteer scenario:

| StepSteer Scenario | |
|---|---|
| **Parameters of the Fellows** | |
| Number of fellows | 0 |

**Related topics**

References

TestGround Road..... 113

## A

ABSBrake
  scenario 120
ABSTrack
  road 76
accelerator mode 29
ACCPedalStimulus100
  scenario 122
animating simulation results 35
ASM example model
  starting via MATLAB 14
ASM library
  opening 12

## C

changing parameters
  in ModelDesk 61
changing suspension kinematics model 39
code generation
  for SCALEXIO 54
  for VEOS 52
Common Program Data folder 8
ControlDesk 56

## D

Documents folder 8
drivetrain configuration 45

## F

Fishhook
  scenario 124
FMVSS_126_SIS
  scenario 126
FMVSS_126_SWD
  scenario 128
FollowRoadEndless
  scenario 129
FourwayJunction_2Lanes
  road 78
FourwayJunction_4Lanes
  road 80
FTP75 Cycle
  scenario 131

## G

generating code
  for PHS-bus based platforms 54

## H

handling
  Simulink plotters 30
handling multi-instances 25
HandlingTrack
  road 82
Highway_A33_Paderborn
  road 84
HockenheimringTrack
  road 86

## L

LaneChange
  road 87
  scenario 132
library types 25
Local Program Data folder 8

## M

ManualDriving_WithRoad
  scenario 134
ModelDesk
  changing parameters 37
  parameterizing 18
ModelDesk plotting 22
MotionDesk 55
MueSplit
  road 89
  scenario 135
MueSplitCurve
  road 91
  scenario 137
multi-instances 25

## N

NCAP_AEB
  road 93
NCAP_LSS
  road 94
  scenario 138

## O

opening
  ASM library 12
opening parameterization project 17
OvalWide
  road 96

## P

parameterizing using ModelDesk 18
plot configuration
  for real-time simulation 59
  for VEOS 59
  update 31
plotter
  in ModelDesk 33
plotting 22

## R

real-time simulation 60
road
  ABSTrack 76
  FourwayJunction_2Lanes 78
  FourwayJunction_4Lanes 80
  HandlingTrack 82
  Highway_A33_Paderborn 84
  HockenheimringTrack 86
  LaneChange 87
  MueSplit 89
  MueSplitCurve 91
  NCAP_AEB 93
  NCAP_LSS 94
  OvalWide 96
  RoadNetwork_Highway_Interchange_PB 97
  RoadNetwork_Urban 103
  RoadNetworkBasic 99
  RoadNetworkRural 101
  RoadWithParkingLane 104
  RoadWorks_Basic 106
  RoadWorks_Highway 108
  Slalom 110
  StraightLong 111
  TestGround 113
  ThreeWayJunction_2Lanes 115
  TrafficDriverBasic 117
RoadNetwork_Highway_Interchange_PB
  road 97
RoadNetwork_Urban
  road 103
RoadNetworkBasic
  road 99
RoadNetworkRural
  road 101
RoadWithParkingLane
  road 104
RoadWorks_Basic
  road 106
RoadWorks_Highway
  road 108

## S

SCALEXIO
  code generation 54
scenario
  ABSBrake 120
  ACCPedalStimulus100 122
  Fishhook 124
  FMVSS_126_SIS 126
  FMVSS_126_SWD 128
  FollowRoadEndless 129
  FTP75 Cycle 131
  LaneChange 132
  ManualDriving_WithRoad 134
  MueSplit 135
  MueSplitCurve 137
  NCAP_LSS 138
  SineSteer 140
  Slalom 142
  SteadyStateCornering 144
  StepSteer 146
scopes 29
Simulink plotters
  handling 30
SineSteer
  scenario 140
Slalom
  road 110
  scenario 142

starting via MATLAB
ASM example model 14
SteadyStateCornering
scenario 144
StepSteer
scenario 146
StraightLong
road 111
suspension kinematics model
changing 39
switching library types 25

## T

TestGround
road 113
ThreeWayJunction_2Lanes
road 115
TrafficDriverBasic
road 117
troubleshooting 67

## U

using the accelerator mode 29

## V

VEOS
code generation 52

## W

watching simulation results 29