ASM Vehicle Dynamics

Reference

For ASM Vehicle Dynamics Blockset 4.2.1 and ASM Vehicle Dynamics Operator Blockset 4.2.1

Release 2021-A – May 2021

dSPACE

How to Contact dSPACE

Mail:	dSPACE GmbH Rathenaustraße 26 33102 Paderborn 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 Reference 13

Overview of the Vehicle Dynamics Library 15

Vehicle Dynamics Library ... 15

Drivetrain 17

Clutch ... 18
- Clutch ... 18
- Clutch Rigid ... 22
- Clutch 4WD ... 24
- Clutch 4WD Rigid ... 27
- Lockup Clutch ... 29
- Lockup Clutch Rigid ... 30
- Torque Converter ... 31
- Torque Converter Rigid ... 34

Crankshaft and Starter ... 36
- Crankshaft ... 36
- Crankshaft Rigid ... 38
- Starter ... 40
- Test Bench ... 42

Differential ... 44
- Differentials ... 44
- Differentials Rigid ... 49

Gearbox ... 53
- Gearbox Automatic Transmission ... 53
- Gearbox Automatic Transmission Rigid ... 55
- Gearbox Manual Transmission ... 56
- Gearbox Manual Transmission Rigid ... 58
- Transfer Gearbox ... 60
- Transfer Gearbox Rigid ... 62

Shafts ... 64
- Shafts ... 64
- Rigid Shaft ... 66

Stabilization 68
Stabilization 68

Switches 71
Drivetrain Variant Switches 71

Engine Basic 73

Engine 73
Fuel Consumption 75

Soft ECU 79

Body 80
Engine Start Button 80

Engine Basic 84
Engine Operation Basic 84
ESP Fast Torque Set 86
Idle Speed Control Engine Basic 88
Shift Torque Set 89
Torque Intervention Basic 90

Steering 93
Soft ECU PowerSteering 93

Transmission 95
Clutch Engagement Control 95
Lockup Clutch Control 97
Shift Lock Control 99
Shift Strategy 100
Soft ECU Transmission Setup 103
TipShift Control 103
Torque Intervention Control 105

Vehicle Dynamics 107

Aerodynamics 108
Aerodynamics 108

Brake 112
Brake Booster 112
Brake Disc 117
Drum Brake 119

Drum Brake 2-D 120
Manifold Pressure 121

Common Paramter 124
Common Vehicle Parameters 124

Suspension 127

Base 127
Suspension Models 127

Common 128
Displacement Sensor 129
Rotation Switch 130
Subframe 131

Front 133

Common Suspension Blocks 134
Front Wheels Relative Velocities 135
Relative Position Front 137
Front Suspension Forces 138
Front Suspension Forces Active 141
Front Suspension Forces Ancillary Spring 143
Front Suspension Compliance 145
Suspension Compliance Rigid Axle (Symmetric) 148
Suspension Force Kinematics Front 152
Suspension Compliance 2-D (Symmetric) 155
Stabilizer Compliance (Symmetric) 160

Suspension Kinematics 162
Front Suspension Kinematics (Asymmetric) 163
Front Suspension Kinematics (Asymmetric, 3DOF) 169
Front Suspension Kinematics (Symmetric) 174
Front Suspension Kinematics MacPherson Strut 179
Front Suspension Kinematics (Symmetric, 3DOF) 183
Suspension Kinematics Rigid Truck Axle 189
Suspension Kinematics Rigid Axle (Symmetric) 194
Front Suspension Kinematics Ancillary Spring 199

Rear 202

Suspension Kinematics 202
Rear Suspension Kinematics (Asymmetric) 203
Rear Suspension Kinematics (Symmetric) 209
Rear Suspension Kinematics Semi-Trailing Arm 213
Rear Suspension Kinematics Rigid Axle 218
Rear Suspension Kinematics (Asymmetric, 3DOF) 222

Rear Suspension Kinematics (Symmetric, 3DOF)..... 222
Suspension Force Kinematics Rear..... 222
Rear Suspension Kinematics Ancillary Spring..... 225

Common Suspension Blocks..... 225
Rear Wheel Relative Velocities..... 227
Relative Position Rear..... 229
Rear Suspension Forces..... 230
Rear Suspension Compliance..... 233
Rear Suspension Forces Ancillary Spring..... 236
Rear Suspension Forces Active..... 236

Steering..... 237
Steering..... 237
Steering Tie Rod..... 243
Steering 3DOF Variable Ratio..... 244
Variable Steering Ratio..... 257

Tire..... 259
Additional CP Calculations..... 260
Contact Point Calculation..... 261
Tire Forces and Torques Vector..... 262
Tire Mode..... 264
Tire Model..... 265
Magic Formula Tire Model..... 266
Tire Model Magic Formula 6.1..... 268
Tire Model TMeasy..... 271
Tire Velocity Sampling Points..... 273

Vehicle Movement..... 275
Coordinate Transformation..... 275
Mass Forces and Torques..... 279
Sensor Motion..... 281
Vehicle and Wheel Velocities..... 283
Vehicle Mass and Additional Loads..... 284
Vehicle Motion..... 287
Vehicle Movement Info Car..... 291

Wheel Speed..... 293
Wheel Speed..... 293

Switches..... 296
Vehicle Dynamics Variant Switches..... 296

Demos 297

Vehicle Dynamics ... 298
Basics on the Vehicle Dynamics Demo Model ... 298
Overview of the VehicleDynamics Demo ... 299

ASM Modules ... 301
Driver Module ... 301
Driver Assistance Simple Module ... 305
Drivetrain Module ... 306
Electrical System Basic Module ... 309
Engine Basic Module ... 310
Environment Module ... 313
Start System Module ... 323
Torque Manager Basic Module ... 325
Vehicle Dynamics Module ... 327

Subsystems ... 336

Subsystems of the Driver Module ... 336
Lateral Controller Subsystem ... 336
Longitudinal Controller Subsystem ... 338
Reference Velocity Subsystem ... 341

Subsystems of the Drivetrain Module ... 341
Final Drive Assembly Subsystem ... 342
Transmission Subsystem ... 348
Automatic Transmission Subsystem ... 349
Manual Transmission Subsystem ... 352

Subsystems of the Engine Basic Module ... 355
Engine Basic Subsystem ... 355
Soft ECU Engine Subsystem ... 355

Subsystems of the Environment Module ... 358
Maneuver Submodule ... 358
Road Submodule ... 361

Subsystems of the Vehicle Dynamics Module ... 362
Brake Subsystem ... 363
Suspension Subsystem ... 364
Front Suspension Subsystem ... 365
Rear Suspension Subsystem ... 367
Vehicle Movement Subsystem ... 369

Blocks from Former Versions 373

Engine Basic (Version 2.0 or Earlier)..... 374
Engine Basic (Version 3.0)..... 374
ESP Interventions Fast (Version 3.0)..... 376
ESP Interventions Slow (Version 3.0)..... 377
Soft ECU Transmission (Version 8.0 or Earlier)..... 378
Soft ECU Transmission (Version 9.0 and 10.0)..... 382
Test Bench (Version 1.0)..... 389
Throttle (Version 2.0)..... 390

New Features/Migration History of the ASM Vehicle Dynamics Blockset 393

General Changes to the ASM Vehicle Dynamics Blockset..... 402
History of the ADDITIONAL_CP_CALCULATIONS Block..... 408
History of the AERODYNAMICS Block..... 408
History of the BASIC_ROADS Block..... 409
History of the BRAKE_BOOSTER Block..... 409
History of the BRAKE_DISC Block..... 410
History of the BRAKE_HYDRAULICS_BASIC Block..... 410
History of the BRAKING_CIRCUIT_BASIC Block..... 410
History of the CENTRAL_DIFFERENTIAL Block..... 410
History of the CENTRAL_DIFFERENTIAL_RIGID Block..... 412
History of the CLUTCH Block..... 412
History of the CLUTCH_ENGAGEMENT_CONTROL Block..... 413
History of the CLUTCH_RIGID Block..... 413
History of the CLUTCH_4WD Block..... 413
History of the CLUTCH_4WD_RIGID Block..... 414
History of the COMMON_VEHICLE_PARAMETERS Block..... 414
History of the CONTACT_POINT_CALCULATION Block..... 415
History of the CONTACT_POINT_CALCULATION_x Block..... 415
History of the COORDINATE_TRANSFORMATION Block..... 415
History of the CRANK_SHAFT Block..... 416
History of the CRANK_SHAFT_RIGID Block..... 417
History of the DISPLACEMENT_SENSOR Block..... 417
History of the DRIVETRAIN_VARIANT_SWITCHES Block..... 418
History of the DRUM_BRAKE Block..... 418
History of the DRUM_BRAKE_2D Block..... 418
History of the ENGINE Block..... 418
History of the ENGINE_BASIC Block..... 419

History of the ENGINE_OPERATION_BASIC Block..... 419
History of the ESP_TORQUE_INTERVENTION_FAST Block..... 420
History of the ESP_TORQUE_INTERVENTION_SLOW Block..... 420
History of the FRONT_DIFFERENTIAL Block..... 421
History of the FRONT_DIFFERENTIAL_RIGID Block..... 422
History of the FUEL_CONSUMPTION Block..... 422
History of the GEARBOX_AT Block..... 422
History of the GEARBOX_AT_RIGID Block..... 423
History of the GEARBOX_MT Block..... 424
History of the GEARBOX_MT_RIGID Block..... 425
History of the IDLE_SPEED_CONTROL_ENGINE_BASIC Block..... 426
History of the LATERAL_CONTROL1 Block..... 426
History of the LATERAL_CONTROL2 Block..... 427
History of the LOCKUP_CLUTCH Block..... 427
History of the LOCKUP_CLUTCH_CONTROL Block..... 428
History of the LOCKUP_CLUTCH_RIGID Block..... 428
History of the MANIFOLD_PRESSURE Block..... 428
History of the MASS_FORCES_AND_TORQUES Block..... 429
History of the MC_PHERSON_STRUT Block..... 429
History of the REAR_DIFFERENTIAL Block..... 429
History of the REAR_DIFFERENTIAL_RIGID Block..... 430
History of the RELATIVE_POSITION_x Block..... 431
History of the RELATIVE_VELOCITIES_x Block..... 431
History of the RIGID_AXLE Block..... 431
History of the SEMI_TRAILING_ARM Block..... 432
History of the SENSOR_MOTION Block..... 432
History of the SHAFT_RIGID Block..... 433
History of the SHAFT_TGCL4WD Block..... 433
History of the SHAFT_x Block..... 434
History of the SHIFT_STRATEGY Block..... 434
History of the SOFT_ECU_POWERSTEERING Block..... 434
History of the SOFT_ECU_TRANSMISSION Block..... 435
History of the SOFT_ECU_TRANSMISSION_10_0 Block..... 437
History of the STABI_COMP_FRONT Block..... 437
History of the STABILISATION Block..... 437
History of the STABILISATION_FF_FF4WD Block..... 437
History of the STABILISATION_TO_TRANSMISSION Block..... 438
History of the STARTER Block..... 438
History of the START_BUTTON Block..... 439
History of the STEERING Block..... 439
History of the SUSCOMP_2D_REAR Block..... 440
History of the STEERING_TIE_ROD Block..... 441

History of the STEERING_VARIABLE_RATIO Block..... 441
History of the STEERING_3DOF_VARIABLE_RATIO Block..... 442
History of the SUBFRAME Block..... 443
History of the SUSCOMP_RIGID_SYM_x Block..... 443
History of the SUSCOMP_2D_FRONT Block..... 444
History of the SUSFORCES_ANC_SPRING Block..... 444
History of the SUSKIN_ANC_SPRING Block..... 444
History of the SUSKIN_x_ASYM_3DOF Block..... 445
History of the SUSKIN_FRONT_ASYM_3DOF Block..... 445
History of the SUSKIN_FRONT_SYM_3DOF Block..... 445
History of the SUSKIN_REAR_ASYM_3DOF Block..... 446
History of the SUSKIN_REAR_SYM_3DOF Block..... 446
History of the SUSKIN_RIGID_SYM Block..... 446
History of the SUSKIN_RIGID_TRUCK_x Block..... 447
History of the SUSKIN_x_SYM_3DOF Block..... 447
History of the SUSPENSION_COMPLIANCE Block..... 447
History of the SUSPENSION_COMPLIANCE_FRONT Block..... 448
History of the SUSPENSION_COMPLIANCE_OPP_x Block..... 448
History of the SUSPENSION_COMPLIANCE_REAR Block..... 448
History of the SUSPENSION_COMPLIANCE_x Block..... 449
History of the SUSPENSION_FORCE_KINEMATICS Block..... 449
History of the SUSPENSION_FORCE_KINEMATICS_FRONT Block..... 450
History of the SUSPENSION_FORCE_KINEMATICS_REAR Block..... 450
History of the SUSPENSION_FORCES_ACTIVE Block..... 450
History of the SUSPENSION_FORCES_FRONT Block..... 451
History of the SUSPENSION_FORCES_REAR Block..... 451
History of the SUSPENSION_KINEMATICS_x_ASYMMETRIC Block..... 451
History of the SUSPENSION_KINEMATICS_x_SYMMETRIC Block..... 451
History of the SUSPENSION_KINEMATICS_FRONT_ASYMMETRIC Block..... 451
History of the SUSPENSION_KINEMATICS_FRONT_SYMMETRIC Block..... 452
History of the SUSPENSION_KINEMATICS_REAR_ASYMMETRIC Block..... 452
History of the SUSPENSION_KINEMATICS_REAR_SYMMETRIC Block..... 452
History of the TEST_BENCH Block..... 453
History of the THROTTLE Block..... 453
History of the TIRE_MF Block..... 453
History of the TIRE_MODEL_MAGIC_FORMULA Block..... 454
History of the TIRE_MODEL_MAGIC_FORMULA_DUMMY_x Block..... 454
History of the TIRE_MODEL_MAGIC_FORMULA_x Block..... 454
History of the TIRE_MODEL_TMEASY Block..... 455
History of the TORQUE_INTERVENTION_CONTROL Block..... 456
History of the TIRE_MODEL_TMEASY_DUMMY_x Block..... 456

History of the TIRE_MODEL_TMEASY_x Block..... 456
History of the TIRE_VEL_SAMPLING_POINT Block..... 457
History of the TORQUE_CONVERTER Block..... 458
History of the TORQUE_INTERVENTION_FAST_ENGINE_BASIC Block..... 458
History of the TORQUE_INTERVENTION_SLOW_ENGINE_BASIC Block..... 459
History of the TRANSFER_GEARBOX Block..... 459
History of the TRANSMISSION_TO_STABILISATION Block..... 460
History of the VEHICLE_DYNAMICS_VARIANT_SWITCHES Block..... 460
History of the VEHICLE_MASS_AND_ADDITIONAL_LOADS Block..... 460
History of the VEHICLE_MOTION_1_0 Block..... 461
History of the VEHICLE_MOTION_CAR Block..... 461
History of the VEHICLE_MOVEMENT_INFO_1_0 Block..... 462
History of the VEHICLE_MOVEMENT_INFO_CAR Block..... 462
History of the WHEEL_SPEED Block..... 463
History of the Vehicle Dynamics Demo Model..... 464

Appendix 467

Bibliography..... 467

Index 469

About This Reference

Content

This reference introduces you to the features provided by the ASM Vehicle Dynamics Model. It describes the structure and parts of the model, its physical background, and the data required for parameterization.

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.

Overview of the Vehicle Dynamics Library

Vehicle Dynamics Library

Introduction

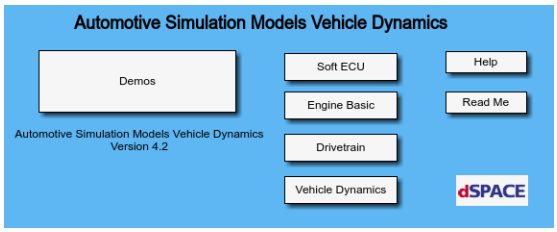
This topic gives you an overview of the ASM Vehicle Dynamics Library.

Opening the library

You can open the library in MATLAB/Simulink. Refer to How to Open an ASM Library (ASM Vehicle Dynamics Model Description 🕮).

Contents

The following illustration shows the first level of the library.



The library has the following main subsystems.

Soft ECU The Soft ECU subsystem contains all the subsystems that allow you to use the engine model offline or if a real ECU is not available. Refer to Soft ECU on page 79.

Engine Basic The Engnie Basic subsystem contains all the blocks needed to model a basic engine model. Refer to Engine Basic on page 73.

Drivetrain The Drivetrain subsystem contains all the blocks needed to model a vehicle drivetrain. Refer to Drivetrain on page 17.

Vehicle Dynamics The VehicleDynamics subsystem contains the blocks used to model the vehicle dynamics of a passenger car. Refer to Vehicle Dynamics on page 107.

Demos The Demos subsystem ready-to-use demo models that you can use to start modelling. Refer to Demos on page 297.

Drivetrain

Where to go from here

Information in this section

Clutch.....18
To describe the blocks of the subsystem.

Crankshaft and Starter.....36
To describe the blocks of the subsystem.

Differential.....44
To describe the blocks of the subsystem.

Gearbox.....53
To describe the blocks of the subsystem.

Shafts.....64
To describe the blocks of the subsystem.

Stabilization.....68
To describe the blocks of the subsystem.

Switches.....71
To describe the switches of the subsystem.

Clutch

Where to go from here

Information in this section

Clutch....18
Connects the engine to or disconnects it from the rest of the drivetrain in a vehicle equipped with a manual gearbox.

Clutch Rigid....22
Connects the engine to or disconnects it from the rest of the drivetrain in a vehicle equipped with a manual gearbox.

Clutch 4WD....24
The clutch 4WD is used to connect the transfer gearbox to the rear differential.

Clutch 4WD Rigid....27
The clutch 4WD is used to connect the transfer gearbox to the rear differential.

Lockup Clutch....29
The function of the lockup clutch is to mechanically connect and disconnect the engine to and from the rest of the drivetrain in a vehicle equipped with automatic transmission.

Lockup Clutch Rigid....30
The function of the lockup clutch is to mechanically connect and disconnect the engine to and from the rest of the drivetrain in a vehicle equipped with automatic transmission.

Torque Converter....31
The torque converter is used in automatic transmission drivetrains to transfer the engine torque to the transmission system.

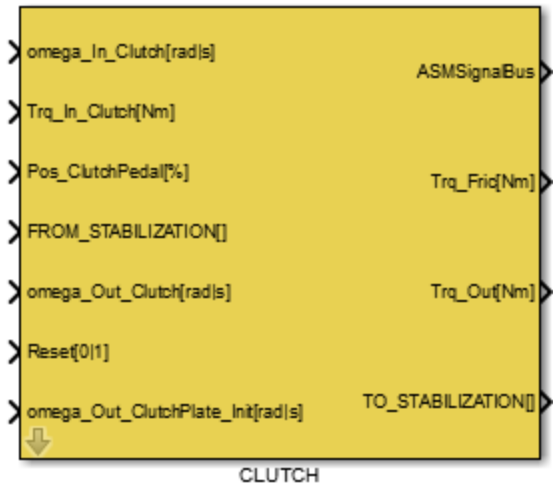
Torque Converter Rigid....34
The torque converter is used in automatic transmission drivetrains to transfer the engine torque to the transmission system.

Clutch

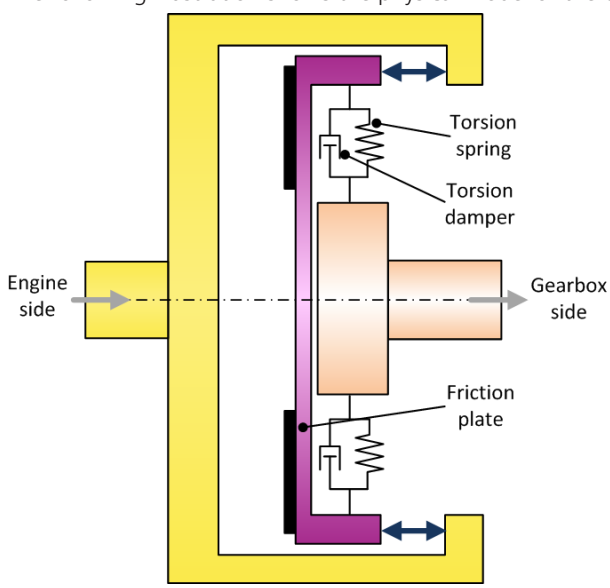
Description

The function of the clutch is to connect the engine to or disconnect it from the rest of the drivetrain in a vehicle equipped with a manual gearbox.

The clutch is modeled as two friction plates with an elastic element on its output side (torsion spring). The clutch torsion spring acts as isolation of the engine's high frequency oscillation from the rest of the drivetrain. The mass inertia of the engine side plate is added to the engine inertia.



The following illustration shows the physical model of the clutch:



The angular velocity of the clutch driven plate can be expressed as follows:

$$J_{Clutch} \cdot \dot{\omega}_{Clutch} = T_{Fric} - T_{Spring}$$

where:

J_{Clutch}	is the inertia of the clutch driven plate
$\dot{\omega}_{Clutch}$	is the angular speed of the clutch driven plate
T_{Fric}	is the clutch friction torque
T_{Spring}	is the torsion spring torque

The torsion spring torque is a function of the twist angle, the spring characteristics, and the damping coefficient of the torsion spring.

The resulting torque is calculated as the following:

$$T_{Spring} = f(\Delta\theta_{Clutch}) + d_{Spring}(\omega_{Clutch} - \omega_{Gearbox,In})$$

$$\Delta\theta_{Clutch} = \int \left(\omega_{Clutch} - \omega_{Gearbox,In}\right) dt$$

where:

$f(\Delta\theta_{Clutch})$ is the torsion spring characteristic function, which is described by a look-up table

d_{Spring} is the torsion equivalent damping coefficient of the torsion spring

$\omega_{Gearbox,In}$ is the angular speed of the gearbox input shaft

The clutch friction torque T_{Fric} is modeled with two components: the static and the dynamic torques as shown in the following:

$$T_{Fric} = T_{St} + T_{Dyn}$$

$$T_{Dyn} = D\left(\omega_{Engine} - \omega_{Clutch}\right)$$

where:

T_{St} is the static torque

T_{Dyn} is the dynamic torque

D is the change of the friction torque with the relative velocity

ω_{Engine} is the engine speed

The resulting friction torque is then limited by the clutch torque capacity as follows:

$$T_{St,Lim} = T_{St} \leq T_{Lim}$$

$$T_{Fric} = T_{St,Lim} + T_{Dyn} \leq T_{Lim}$$

where:

T_{St} is the clutch torque capacity

The clutch static torque is introduced to balance the engine and the clutch driven plate to run at almost the same speed. To allow the transition from a certain slip speed to the locked state, a dynamic term is added as a function of the relative velocity. The resulting friction torque is limited by the clutch torque capacity which depends on the clutch geometry parameters, the maximum transferable clutch torque and the clutch pedal position, i.e.:

$$T_{Lim} = f(100 - P_{Clutch}[\%]) \cdot T_{Max}$$

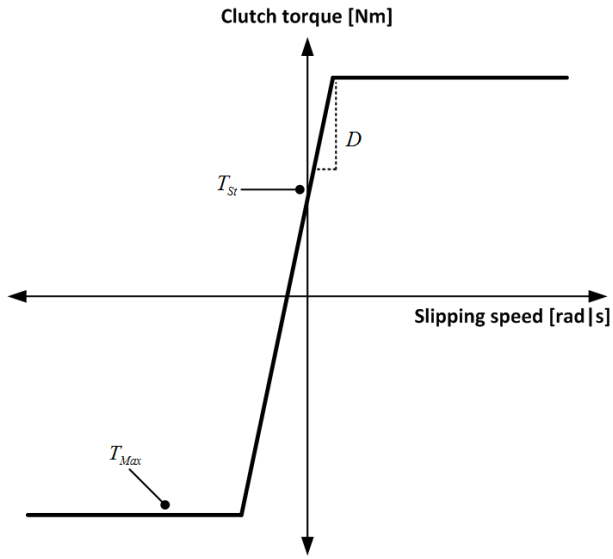
where:

$P_{Clutch}[\%]$ is the clutch pedal position

T_{Max} is the maximum transferable clutch torque

$f(\,)$ is a scaling function which describes the nominal clutch torque capacity

The following figure shows the clutch torque as a function of the slipping speed.



Inports

The following table shows the inports:

Name	Unit	Description
FROM_STABILIZATION	[]	Interface from drivetrain stabilization to ASM block. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
omega_In_Clutch	[rad/s]	Clutch input speed
omega_Out_Clutch	[rad/s]	Clutch output speed
omega_Out_ClutchPlate_Init	[rad/s]	Initial output speed of clutch plate
Pos_ClutchPedal	[%]	Clutch pedal position. 0[%] means fully closed, 100[%] means fully open.
Reset	[0|1]	Reset of states
Trq_In_Clutch	[Nm]	Clutch input torque

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
TO_STABILIZATION	[]	Interface from ASM block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
Trq_Fric	[Nm]	Clutch friction torque
Trq_Out	[Nm]	Output torque generated from the torsion spring

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Coef_DynamicClutchTrq	[Nm/(rad/s)]	Change of friction torque with relative velocity *D*
Const_Damping	[Nm/(rad/s)]	Damping coefficient of the torsion spring
Const_Inertia	[kg m^2]	Inertia of the clutch driven plate
Const_Torque_Max	[Nm]	Maximum transferable clutch friction torque
Map_Stiffenss	[deg]	Spring torque [Nm]=f(AngleSpring)
Map_Trq_Clutch_Red	[0_1]	Clutch nominal torque capacity
StepSize	[s]	Simulation step size

Related topics

References

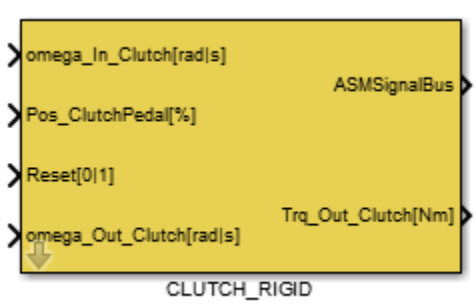
Clutch (ModelDesk Parameterizing 🕮)

Clutch Rigid

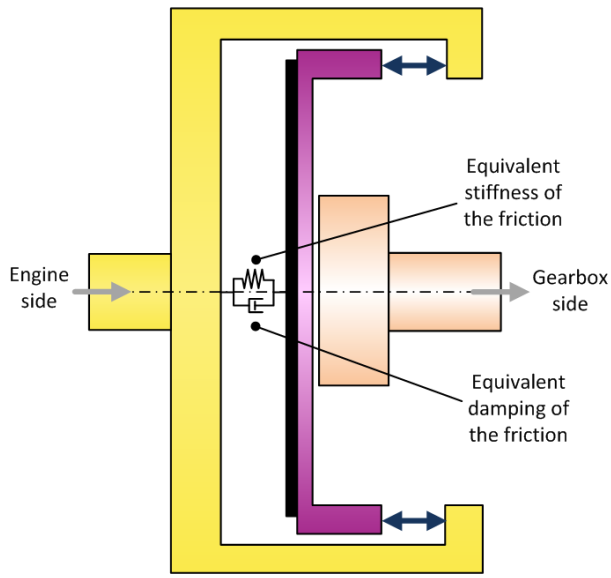
Description

The function of the clutch is to connect the engine to or disconnect it from the rest of the drivetrain in a vehicle equipped with a manual gearbox.

The rigid clutch is modeled as two plates. The torque is transferred via friction between them. In this model, no torsion spring is used. The idea of the so-called bristle friction model was used to model the friction torque which arises between the plates. The model is based on the assumption that the contact between the two surfaces acts like elastic bristles. As a result of the exerted torque, each bristle can deflect and the deflection torque can be described by a damper and a spring.



The following illustration shows the physical model of the clutch:



The friction torque can be formulated as follows:

$$T_{Fric} = K_{Fric}\,\Delta\,\theta_{Clutch} + C_{Fric}\,\Delta\,\omega_{Clutch}$$

where:

K_{Fric} is the equivalent stiffness coefficient of the friction model

C_{Fric} is the equivalent damping coefficient of the friction model

$\Delta\,\theta_{Clutch}$ is the twist angle difference between the input and output side of the clutch

$\Delta\,\omega_{Clutch}$ is the relative velocity between the input and output side of the clutch

The dependence of the friction torque on the normally exerted force or the command is considered a linear relationship.

The resulting friction torque is limited by the clutch torque capacity, which depends on the clutch geometry parameters, the maximum transferable clutch torque and the clutch pedal position, i.e.:

$$T_{Lim} = f(100 - P_{Clutch}[\%])T_{Max}$$

where:

$P_{Clutch}[\%]$ is the clutch pedal position

T_{Max} is the maximum transferable clutch torque

$f()$ is a scaling function which describes the nominal clutch torque capacity

Inports

The following table shows the inports:

Name	Unit	Description
omega_In_Clutch	[rad/s]	Clutch input speed
omega_Out_Clutch	[rad/s]	Clutch output speed

Name	Unit	Description
Pos_ClutchPedal	[%]	Clutch pedal position. 0[%] means fully closed, 100[%] means fully open.
Reset	[0|1]	Reset of states

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_Out_Clutch	[Nm]	Clutch output torque

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Damping_Fric_Clutch	[Nm/(rad/|s)]	Damping coefficient of clutch friction model
Const_Stiffness_Fric_Clutch	[Nm/rad]	Stiffness coefficient of clutch friction model
Const_Torque_Max	[Nm]	Maximum transferable clutch torque
Map_Trq_Clutch_Red	[0_1]	Clutch nominal torque capacity
StepSize	[s]	Simulation step size

Related topics

References

Clutch Rigid (ModelDesk Parameterizing 🕮)

Clutch 4WD

Description

The clutch 4WD is used to connect the transfer gearbox to the rear differential. When the clutch 4WD is active, a drivetrain configuration with 4-Wheel Drive is built.

The speed calculation for the output side is done by the rear differential. The torque on the output side and the inertia of the output side are therefore provided as output signals. The speed of the output side is received as input signal.

The speed of the input side is calculated in this block. Therefore it receives the input shaft torque.

The torques of the output shafts of the rear differential are required to calculate the static friction torque of the clutch model.

The approach to estimate the friction torque of the clutch is equal to the approach in the clutch of the manual transmission. For more information, refer to Clutch on page 18.

The maximum transferable torque depends on the clutch control signal and the geometry.

$$\widetilde{T}_{Max} = f\left(100 - Ctrl_{Clutch}[\%]\right) \cdot \dot{T}_{Max}$$

where:

$Ctrl_{Clutch}[\%]$	is the clutch 4WD control signal
T_{Max}	is the maximum transferable torque
$f()$	is a scaling function which describes the nominal clutch torque capacity

The saturated static friction torque is calculated as:

$$\widetilde{T}_{Fric,Sta} = \max\left(\min\left(\frac{T_{TGCL4WDShaft} + \frac{T_{RLShaft} + T_{RRShaft}}{i_{RearDifferential}}}{2}, \widetilde{T}_{Max}\right), -\widetilde{T}_{Max}\right)$$

where:

$\widetilde{T}_{Fric,Sta}$	is the saturated static friction torque
$T_{TGCL4WDShaft}$	is the torque in the shaft from transfer gearbox to the clutch 4WD
$T_{RLShaft}$	is the torque in the rear left shaft
$T_{RRShaft}$	is the torque in the rear right shaft
$i_{RearDifferential}$	is the main reduction of rear differential

The transferred clutch friction torque is:

$$T_{ClutchFric} = \max\left(\min\left(\left(\omega_{CL4WD_In} - \omega_{Cl4WD_Out}\right) \cdot d_{Clutch} + \widetilde{T}_{Fric,Sta}, \widetilde{T}_{Max}\right), -\widetilde{T}_{Max}\right)$$

where:

$T_{ClutchFric}$	is the transferred clutch friction torque
ω_{CL4WD_In}	is the speed of the clutch input side
ω_{Cl4WD_Out}	is the speed of the clutch output side
d_{Clutch}	is the change of lock clutch friction torque with relative velocity

The input speed of the clutch is calculated as:

$$\omega_{CL4WD_In} = \int \frac{T_{TGCL4WDShaft} - T_{ClutchFric}}{J_{Clutch_In}} dt$$

where:

ω_{CL4WD_In}	is the input speed of the clutch
$T_{TGL4WDShaft}$	is the torque in the front rear shaft
$T_{ClutchFric}$	is the friction torque of the clutch
J_{Clutch_In}	is the inertia of the clutch input side

Inports

The following table shows the inports:

Name	Unit	Description
Const_i_MainReduction_RearDifferential	[]	Main reduction gear ratio from the rear differential
Ctrl_LockClutch	[%]	Clutch control signal
FROM_STABILIZATION	[]	Interface from drivetrain stabilization to ASM block. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
omega_Clutch4WD_Out	[rad/s]	Output speed of the clutch
omega_Clutch_4WD_In_Init	[rad/s]	Initial clutch input speed
Reset	[0|1]	Resets all integrators in the block to their initial conditions
Trq_RL_Shaft	[Nm]	Rear left shaft torque
Trq_RR_Shaft	[Nm]	Rear right shaft torque
Trq_TGCL4WD_Shaft	[Nm]	Torque in the front rear shaft

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
omega_Clutch4WD_In	[rad/s]	Speed at input side of clutch
TO_STABILIZATION	[]	Interface from ASM block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
Trq_Lock	[Nm]	Friction torque of the clutch

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Coef_DynamicClutchTrq	[Nm/(rad/s)]	Change of lock clutch friction torque with relative velocity
Const_Inertia_InputSide	[kgm^2]	Inertia of the input side of the clutch
Map_Trq_Clutch_Red	[0_1]	Clutch-4WD nominal torque capacity
Const_Inertia_OutputSide	[kgm^2]	Inertia of the output side of the clutch
Const_Torque_Max	[Nm]	Maximum torque lock clutch
StepSize	[s]	Simulation step size

Related topics

References

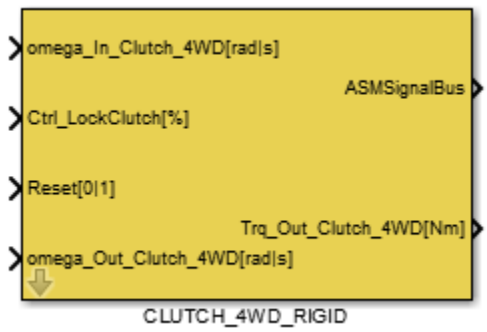
Clutch 4WD (ModelDesk Parameterizing 🕮)

Clutch 4WD Rigid

Description

The clutch 4WD is used to connect the transfer gearbox to the rear differential. When the clutch 4WD is active, a drivetrain configuration with 4-Wheel Drive is built.

The CLUCH_4WD_RIGID receives the velocities of the input and output sides and calculates the lock torque. The velocity of the output side is calculated by the rigid rear differential. The velocity of the input side is calculated by the rigid transfer gearbox.



The modeling approach for the CLUTCH_4WD_RIGID is the same as for the rigid clutch, refer to Rigid Clutch (refer to Clutch on page 18).

The resulting clutch lock torque is limited by the clutch torque capacity which depends on the clutch geometry parameters, the maximum transferable clutch torque, and the clutch control signal, i.e.:

$$T_{Lim} = (100 - Ctrl_{Clutch}[\%]) \cdot T_{Max}$$

$$T_{Clutch} = T_{Clutch} \leq T_{Lim}$$

where:

$Ctrl_{Clutch}[\%]$ is the clutch control signal. 0[%] means fully disengaged, 100[%] means fully engaged

T_{Max} is the maximum transferable clutch torque

T_{Clutch} is the clutch lock torque

$f()$ is the clutch lock torque

Inports

The following table shows the inports:

Name	Unit	Description
Ctrl_LockClutch	[%]	Offset of control signal of differential lock clutch [%]
omega_In_Clutch_4WD	[rad/s]	Input speed
omega_Out_Clutch_4WD	[rad/s]	Output speed of the clutch
Reset	[0|1]	Reset of states

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_Out_Clutch_4WD	[Nm]	Output torque of the clutch

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Damping_Fric_Clutch_4WD	[Nm/(rad/s)]	Damping coefficient of clutch 4WD friction model
Const_Stiffness_Fric_Clutch_4WD	[Nm/rad]	Stiffness coefficient of clutch 4WD friction model
Const_Torque_Max	[Nm]	Maximum transferable clutch 4WD torque
Map_Trq_Clutch_Red	[0_1]	Clutch-4WD nominal torque capacity
StepSize	[s]	Simulation step size

Related topics

References

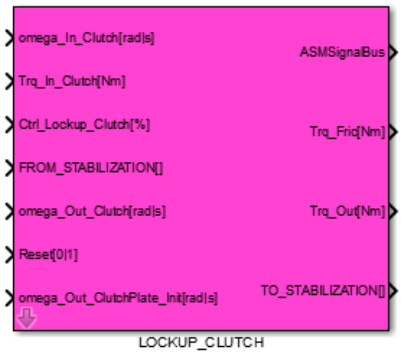
Clutch 4WD Rigid (ModelDesk Parameterizing 🕮)

Lockup Clutch

Description

The function of the lockup clutch is to mechanically connect and disconnect the engine to and from the rest of the drivetrain in a vehicle equipped with automatic transmission. It is used to increase the efficiency as it rigidly connects the engine with the drivetrain. The engagement and disengagement of the lockup clutch is controlled by the control unit, which can be a real ECU or a soft ECU. For more information about the transmission soft ECU, refer to Transmission on page 95.

The modeling approach of the **LOCKUP_CLUTCH** block is similar to the **Clutch** model. Refer to Clutch on page 18.



Inports

The following table shows the inports:

Name	Unit	Description
Ctrl_Lockup_Clutch	[%]	Lockup clutch control signal: 0% means fully closed and 100% means fully open
FROM_STABILIZATION	[]	Interface from drivetrain stabilization to ASM block. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
omega_In_Clutch	[rad/s]	Lockup clutch input speed
omega_Out_Clutch	[rad/s]	Lockup clutch output speed
omega_Out_ClutchPlate_Init	[rad/s]	Initial output speed of clutch plate
Reset	[0|1]	Reset of states
Trq_Engine	[Nm]	Engine torque

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).

Name	Unit	Description
TO_STABILIZATION	[]	Interface from ASM block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
Trq_Fric	[Nm]	Clutch friction torque
Trq_Out	[Nm]	Torque exerted by the torsion spring

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Coef_DynamicClutchTrq	[Nm/(rad/s)]	Change of friction torque with relative velocity
Const_Inertia	[kg m^2]	Inertia of the lockup clutch driven plate
Const_Torque_Max	[Nm]	Maximum transferable clutch friction torque
Map_Trq_Clutch_Red	[0_1]	Lockup clutch nominal torque capacity
StepSize	[s]	Simulation step size

Related topics

References

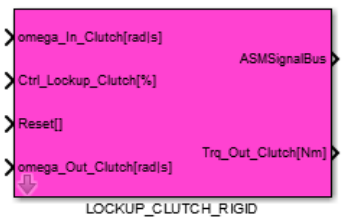
Torque Converter (ModelDesk Parameterizing 🕮)

Lockup Clutch Rigid

Description

The function of the lockup clutch is to mechanically connect and disconnect the engine to and from the rest of the drivetrain in a vehicle equipped with automatic transmission. It is used to increase the efficiency as it rigidly connects the engine with the drivetrain. The engagement and disengagement of the lockup clutch is controlled by the control unit, which can be a real ECU or a soft ECU. For more information about the transmission soft ECU, refer to Transmission on page 95.

The modeling approach of the LOCKUP_CLUTCH_RIGID is similar to the Rigid Clutch model. Refer to Clutch Rigid on page 22.



Inports

The following table shows the inports:

Name	Unit	Description
omega_In_Clutch	[rad/s]	Clutch input speed
omega_Out_Clutch	[rad/s]	Clutch output speed
Ctrl_Lockup_Clutch	[%]	Lockup clutch control signal: 0[%] means fully closed, 100[%] means fully open.
Reset	[0|1]	Reset of states

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_Out_Clutch	[Nm]	Clutch output torque

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Damping_Fric_LockupClutch	[Nm/(rad/s)]	Damping coefficient of lockup clutch friction model
Const_Stiffness_Fric_LockupClutch	[Nm/rad]	Stiffness coefficient of lockup clutch friction model
Const_Torque_Max	[Nm]	Maximum transferable lockup clutch torque
Map_Trq_Clutch_Red	[0_1]	Lockup clutch nominal torque capacity
StepSize	[s]	Simulation step size

Related topics

References

Torque Converter Rigid (ModelDesk Parameterizing 🕮)

Torque Converter

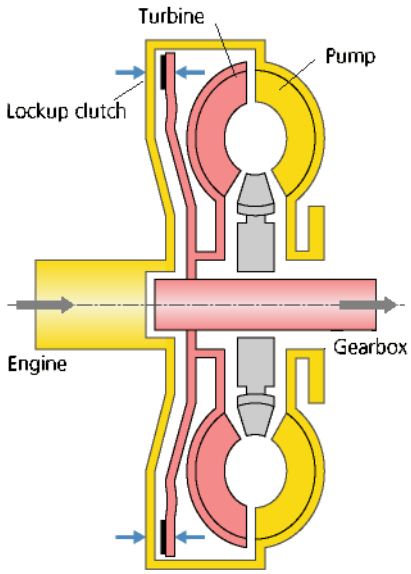
Description

The torque converter is used in automatic transmission drivetrains to transfer the engine torque to the transmission system.

The torque converter consists of the following elements

- Pump
- Turbine
- Stator

The next illustration is a schematic of a torque converter:



The pump is mounted on the engine crankshaft and accelerates the oil flowing to the turbine inside the converter. From the turbine, the oil flows through the stator back to the pump. If the pump speed is greater than the turbine speed, the torque converter increases the turbine torque. If the pump speed is nearly equal to the turbine speed, the converter works as a hydraulic clutch and transmits the pump torque to the turbine shaft.

The pump torque is modeled with the following equation, where the function *f* is a look-up table (Map_Eta_Slip):

$$T_p = f\left(\frac{\Omega_T}{\Omega_P}\right)\Omega_P{}^2$$

The turbine torque is calculated with the next equation, where the function *g* is a look-up table (Map_Rel_Trq_PumpTurb):

$$T_T = T_P \cdot g\left(\frac{\Omega_T}{\Omega_P}\right)$$

Inports

The following table shows the inports:

Name	Unit	Description
omega_Pump	[rad/s]	Pump shaft speed
omega_Turbine	[rad/s]	Turbine shaft speed

Outports

The following table shows the outports:

Name	Unit	Description
TO_STABILIZATION	[]	Interface from ASM block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
Trq_Pump	[Nm]	Pump torque
Trq_Turbine	[Nm]	Turbine torque

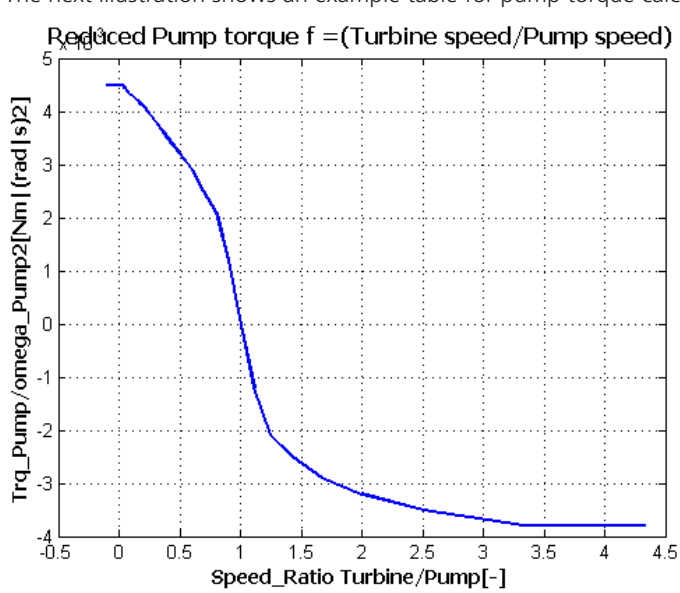
Parameters

The following table shows the parameters:

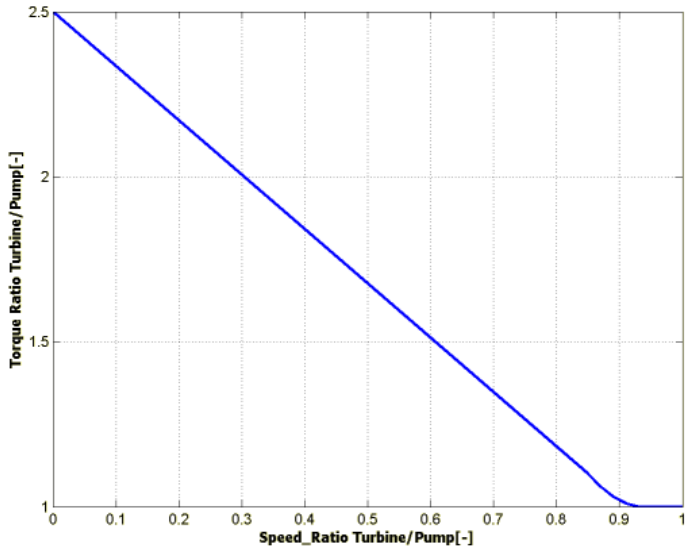
Name	Unit	Description
Map_Eta_Slip	[Nm/(rad/s)^2]	Reduced pump torque f = (turbine speed/pump speed)
Map_Rel_Trq_PumpTurb	[]	Torque ratio calculation f = (turbine speed/pump speed)

Processing information

The next illustration shows an example table for pump torque calculation.



The next illustration shows an example table for turbine torque calculation.



Related topics

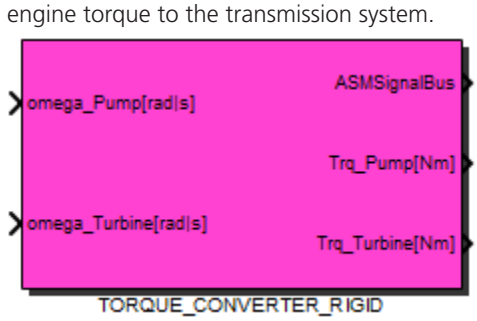
References

Torque Converter (ModelDesk Parameterizing 🕮)

Torque Converter Rigid

Description

The torque converter is used in automatic transmission drivetrains to transfer the engine torque to the transmission system.



The modeling approach is similar to the **Torque Converter**, see Torque Converter on page 31.

Inports

The following table shows the inports:

Name	Unit	Description
omega_Pump	[rad/s]	Pump speed
omega_Turbine	[rad/s]	Turbine speed

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_Pump	[Nm]	Pump torque
Trq_Turbine	[Nm]	Turbine torque

Parameters

The following table shows the parameters:

Name	Unit	Description
Map_Eta_Slip	[]	Pump torque calculation = f(Turbine speed/Pump speed)
Map_Rel_Trq_PumpTurb	[]	Torque ratio calculation = f(Turbine speed/Pump speed)

Related topics

References

Torque Converter Rigid (ModelDesk Parameterizing 🕮)

Crankshaft and Starter

Where to go from here

Information in this section

Crankshaft.....36
The crankshaft calculates the engine speed by integrating the different drive torques, e.g., clutch torque and mean effective torque of the engine.

Crankshaft Rigid.....38
The CRANK_SHAFT_RIGID block is used to simulate a rigid drivetrain.

Starter.....40
An external torque has to be generated to accelerate the crankshaft. This is done by the starter, which brings the engine up to its starting speed.

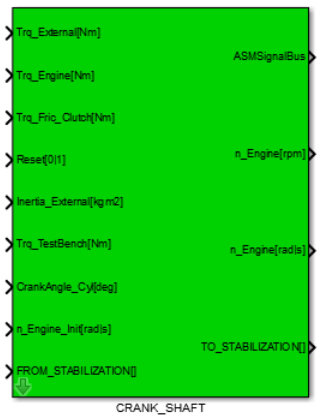
Test Bench.....42
The block generates torque to follow a reference engine speed.

Crankshaft

Crankshaft

The engine cylinders produce a mean torque on the crankshaft, which is connected to the clutch from the other side. In addition to the engine and the clutch torques, other torques can also act on the crankshaft, such as the starter and test bench torque. Moreover, the mass torque, which arises due to acceleration and deceleration of the pistons, is also considered.

The crankshaft calculates the engine speed by integrating the different drive torques, e.g., clutch torque and mean effective torque of the engine.



The engine speed is calculated with the following equation:

$$(J_{ICE} + \sum J_i) \cdot \dot{\omega}_{ICE} = (T_{ICE} - T_{clutch} + \sum T_i)$$

where:

J_{ICE} is the crankshaft equivalent inertia

$\sum J_i$ is the sum of other external inertias connected to the crankshaft

$\dot{\omega}_{ICE}$ is the angular acceleration of the crankshaft

T_{ICE} is the effective internal combustion engine torque exerted on the crankshaft

T_{clutch} is the clutch torque

$\sum T_i$ is the sum of all other torques acting on the crankshaft

In this particular case, the sum of all other torques acting on the crankshaft is given by:

$$\sum T_i = T_{starter} + T_{testbench} + T_{mass}$$

where:

$T_{starter}$ is the starter torque

$T_{testbench}$ is the test bench torque

T_{mass} is the modulated mass torque

The effect of moving pistons and conrod is included as an additional mass torque. This effect is implemented as a look-up table that depends on the crank angle. The cylinder-specific crank angle is calculated separately and provided by a signal line. The model uses vector summation to calculate the mass torque for the entire crankshaft according to the cylinder.

The mass torque look-up table is calculated with the following equation:

$$T_{mass} = {\omega_{ICE}}^2 \cdot f(\varphi)$$

$f(\varphi)$ is calculated as shown in the following:

$$f(\varphi) = m_{osz} \cdot r^2 \cdot \left[\frac{1}{4} \cdot \lambda \cdot \sin(\varphi) - \frac{1}{2} \cdot \lambda \cdot \sin(2\varphi) - \frac{3}{4} \cdot \lambda \cdot \sin(3\varphi) - \frac{1}{4} \cdot \lambda^2 \cdot \sin(4\varphi)\right]$$

where:

r is the crankshaft radius

λ is the ratio of crankshaft radius to conrod length

m_{osz} is the oscillating mass

φ is the crankshaft angle

Inports

The following table shows the inports:

Name	Unit	Description
CrankAngle_Cyl	[deg]	Individual cylinder crank angle
FROM_STABILIZATION	[]	Interface from drivetrain stabilization to ASM block. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
Inertia_External	[kgm^2]	Additional external inertia connected to the crankshaft, such as starter inertia
n_Engine_Init	[rad/s]	Initial engine speed

Name	Unit	Description
Reset	[0|1]	Reset of states
Trq_Engine	[Nm]	Engine torque
Trq_External	[Nm]	Additional external torque acting on the crankshaft, for example, starter torque
Trq_Fric_Clutch	[Nm]	Clutch friction torque
Trq_test bench	[Nm]	Test bench torque

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
n_Engine	[rpm]	Engine speed in [rpm]
n_Engine	[rad/s]	Engine speed in [rad/s]
TO_STABILIZATION		Interface from ASM block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Inertia	[kg m^2]	Crankshaft equivalent inertia
Map_Trq_Mass	[Nm/$(rad/s)^2$]	Modulated mass torque = f(Crank_Angle)
StepSize	[s]	Simulation step size
Sw_Trq_Mass_Mode	[1|2]	Switch modulated mass torque mode: ▪ 1: On ▪ 2: Off

Related topics

References

Crankshaft (ModelDesk Parameterizing 🕮)

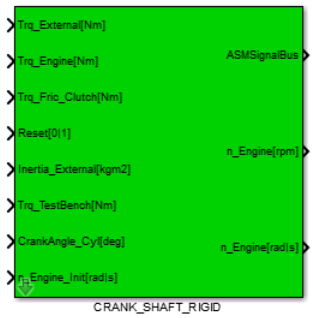
Crankshaft Rigid

Description

The engine cylinders produce a mean torque on the crankshaft, which is connected to the clutch from the other side. In addition to the engine and the clutch torques, other torques can also act on the crankshaft, such as the starter

and test bench torque. Moreover, the mass torque, which arises due to acceleration and deceleration of the pistons, is also considered.

The CRANK_SHAFT_RIGID block is used to simulate a rigid drivetrain. The modeling approach is similar to the Crankshaft model, refer to Crankshaft on page 36. The only difference is that the Rigid Crankshaft model does not have stabilization. For information on stabilizing the drivetrain solver, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).



Inports

The following table shows the inports:

Name	Unit	Description
CrankAngle_Cyl	[deg]	Individual cylinder crank angle
Inertia_External	[kg m^2]	Additional external inertia connected to the crankshaft, e.g., starter inertia
n_Engine_Init	[rad/s]	Initial engine speed
Reset	[0|1]	Reset of states
Trq_Engine	[Nm]	Engine torque
Trq_External	[Nm]	Additional external torque acting on the crankshaft, e.g., starter torque
Trq_Fric_Clutch	[Nm]	Clutch friction torque
Trq_test bench	[Nm]	Test bench torque

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
n_Engine	[rad/s]	Engine speed in [rad/s]
n_Engine	[rpm]	Engine speed in [rpm]

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Inertia	[kg m^2]	Crankshaft equivalent inertia
Map_Trq_Mass_Mod	[Nm / $(rad/s)^2$]	Modulated mass torque = f(CrankAngle)
StepSize	[s]	Simulation step size
Sw_Trq_Mass_Mode	[1|2]	Switch modulated mass torque: ▪ 1: On ▪ 2: Off

Related topics

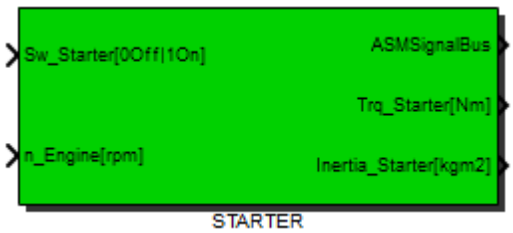
References

Crankshaft Rigid (ModelDesk Parameterizing 🕮)

Starter

Description

A combustion engine requires an initial crankshaft speed before the combustion process can be started.

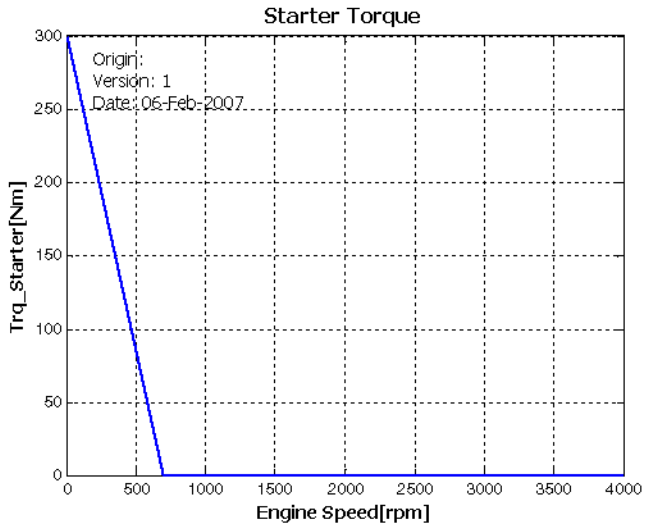


Therefore an external torque has to be generated to accelerate the crankshaft. This is done by the starter, which brings the engine up to its starting speed.

Tip

If the engine must always run up to the idle speed automatically, the Sw_Starter input signal must be set to one.

The starter model is as simple as possible, as it is not the main task of this model. Simple interpolation is carried out between the specified parameters. The illustration below shows the starter torque vs. engine speed.



Inports

The following table shows the inports:

Name	Unit	Description
n_Engine	[rpm]	Engine speed
Sw_Starter	[0|1]	Switch for starter activation: ▪ 0: Off ▪ 1: On

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Inertia_Starter	[kgm^2]	Starter inertia reduced to the crankshaft
Trq_Starter	[Nm]	Starter torque

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Inertia_Starter	[kg m^2]	Starter inertia reduced to the crankshaft
Const_n_Trq0	[rpm]	Engine speed at zero starter torque
Const_Trq_n0	[Nm]	Starter torque at zero speed

Related topics

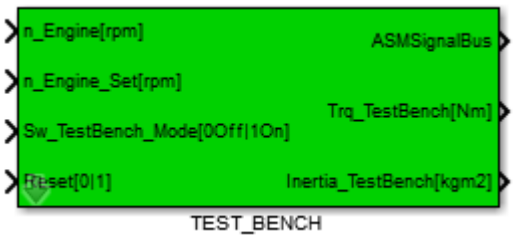
References

Engine Starter (ModelDesk Parameterizing 🕮)

Test Bench

Description

The block can be used to simulate an engine test bench. It generates torque to follow a reference engine speed. This is comparable to an engine test bench and can be used to examine the engine at a fixed engine speed.



The test bench model is implemented as a PI controller.

Inports

The following table shows the inports:

Name	Unit	Description
n_Engine	[rpm]	Engine speed
n_Engine_Set	[rpm]	Engine speed setpoint
Reset	[0|1]	Reset
Sw_test bench_Mode	[0|1]	Switch to enable or disable test bench model ▪ 0: Off ▪ 1: On

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_test bench	[Nm]	Test bench torque
Inertia_test bench	[kgm^2]	Test bench inertia

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_I_Gain	[1|s]	I gain of the dynamometer controller
Const_Inertia_test bench	[kgm^2]	Dynamometer inertia
Const_P_Gain	[]	P gain of the dynamometer controller
Const_Trq_Max_test bench	[Nm]	Maximum dynamometer torque

Related topics

References

Test Bench VD (ModelDesk Parameterizing 🕮)

Differential

Where to go from here

Information in this section

Differentials.....44
The differential transfers the torque from the input side to two output sides, which are allowed to spin at different speeds.

Differentials Rigid.....49
The differential transfers the torque from the input side to two output sides, which are allowed to spin at different speeds.

Differentials

Description

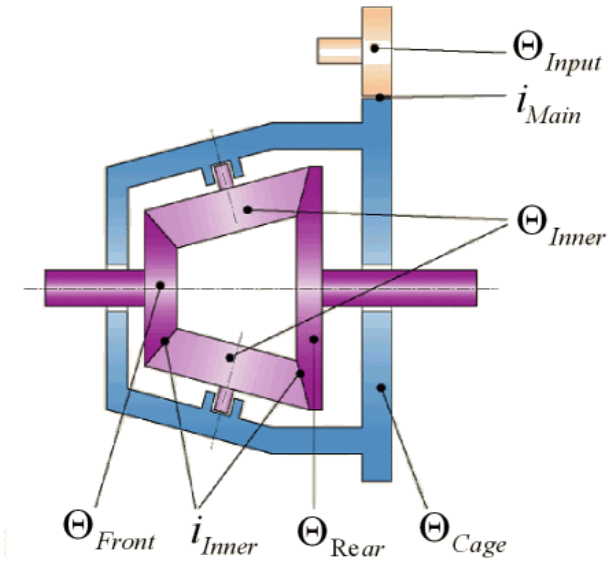
The differential transfers the torque from the input side to two output sides, which are allowed to spin at different speeds.

The differential has three tasks:

- To transfer the engine power to the wheels
- To act as the final gear reduction in the vehicle, e.g. by slowing the rotational speed of the transmission before it reaches the wheels
- To transmit the power to the wheels while allowing them to rotate at different speeds; this is mainly needed for driving in a curve with the outer wheels having to run faster than the inner wheels.

The following explanation uses the central differential as an example. The front and rear differential can be derived from this by setting the torque distribution ratio to 0.5. The left and right output shafts are equal to the front and rear

output shafts respectively. The following figure shows the physical model of the central differential.



where:

Θ_{Front} is the inertia of the front gear

Θ_{Rear} is the inertia of the rear gear

Θ_{Cage} is the inertia of the cage

Θ_{Inner} is the inertia of the inner gears

Θ_{Input} is the inertia of the input gear

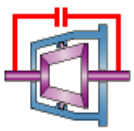
i_{Main} is the transmission ratio of the differential main reduction gear

i_{Inner} is the transmission ratio of the inner gear. This is calculated with:

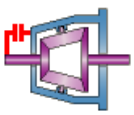
$$i_i = \frac{teeth\ front\ gear + teeth\ rear\ gear}{2 \cdot teeth\ inner\ gear}$$

The modeled differential in ASM Vehicle Dynamics is a lockable, open-type differential. Differential locking is implemented by a clutch, which can be controlled externally. The clutch mode can be selected with the Sw_DifferentialLock_Mode parameter. There are three different modes that can be selected:

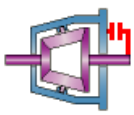
The clutch connects the output shafts
Sw_DifferentialLock_Mode = 1



The clutch connects the front output shaft to the cage
Sw_DifferentialLock_Mode = 2



The clutch connects the rear output shaft to the cage
Sw_DifferentialLock_Mode = 3



The differential has two degrees of freedom, namely the output speed at both sides. The input shaft speed can be calculated as:

$$\Omega_{Input} = i_{Main}[\rho\,\Omega_{Front} + (1-\rho)\Omega_{Rear}]$$

where:

Ω_{Input}	is the speed of the central differential input shaft
Ω_{Front}	is the speed of the central differential front
Ω_{Rear}	is the speed of the central differential rear
ρ	is the torque distribution to the front

The torque equations are:

$$T_{Input2Front} = \rho\, i_{Main} T_{Input}$$

$$T_{Input2Rear} = (1-\rho) i_{Main} T_{Input}$$

$$T_{Cage2Front} = \rho\, T_{LockCage}$$

$$T_{Cage2Rear} = \left(1-\rho\right) T_{LockCage}$$

where:

T_{Input}	is the torque on the central differential input shaft
$T_{Input2Front}$	is the torque on the front shaft of the differential caused by the input shaft
$T_{Input2Rear}$	is the torque on the rear shaft of the differential caused by the input shaft
$T_{LockCage}$	is the torque on the cage from the lockup clutch (depending on the clutch mode, this might be zero)
$T_{Cage2Front}$	is the torque on the front shaft of the differential caused by the lockup clutch torque on the cage
$T_{Cage2Rear}$	is the torque on the rear shaft of the differential caused by the lockup clutch torque on the cage

The differential lock is implemented as a friction clutch. The modeling approach is similar to the **Clutch** model, refer to Clutch on page 18. The maximum locking friction torque is defined as a parameter and can be scaled by defining the **Const_Ctrl_Offset_LockClutch** parameter or externally via the **Ctrl_LockClutch[%]** input. The locking friction torque limit can then be calculated as follows:

$$T_{Lock,limit} = T_{Lock,max}\left(\frac{Const_Ctrl_Offset_LockClutch}{100} + \left(1 - \frac{Const_Ctrl_Offset_LockClutch}{100}\right)\frac{Const_Ctrl_LockClutch}{100}\right)$$

The differential has two degrees of freedom, namely the output speed at both sides. Its equation of motion can be generally written as follows:

$$\begin{pmatrix} \Theta_{CD11} & \Theta_{CD12} \\ \Theta_{CD21} & \Theta_{CD22} \end{pmatrix} \begin{pmatrix} \dot{\Omega}_{CDF} \\ \dot{\Omega}_{CDR} \end{pmatrix} = \begin{pmatrix} T_{Input2Front} - T_{CFShaft} + T_{LockFront} + T_{Cage2Front} \\ T_{Input2Rear} - T_{CRShaft} + T_{LockRear} + T_{Cage2Rear} \end{pmatrix}$$

with

$$\Theta_{CD11} = \left(\left(\Theta_{GearOut} + \Theta_{Inner}\right) {i_{Main}}^2 + \Theta_{Cage} + \Theta_{Inner}\, {i_{Inner}}^2\right)\rho^2 + \Theta_{Front}$$

$$\Theta_{CD22} = \left((\Theta_{GearOut} + \Theta_{Inner})\, i_{Main}{}^2 + \Theta_{Cage} + \Theta_{Inner}\, i_{Inner}{}^2\right)(1-\rho)^2 + \Theta_{Rear}$$

$$\Theta_{CD12} = \left((\Theta_{GearOut} + \Theta_{Inner})\, i_{Main}{}^2 + \Theta_{Cage} - \Theta_{Inner}\, i_{Inner}{}^2\right)\left(\rho - \rho^2\right)$$

$$\Theta_{CD21} = \left((\Theta_{GearOut} + \Theta_{Inner})\, i_{Main}{}^2 + \Theta_{Cage} - \Theta_{Inner}\, i_{Inner}{}^2\right)\left(\rho - \rho^2\right)$$

where:

$T_{CFShaft}$ is the torque exerted on the drive shaft from central differential to *front* differential

$T_{CRShaft}$ is the torque exerted on the drive shaft from central differential to *rear* differential

$T_{LockFront}$ is the locking torque acting on the *front* side (depending on the clutch mode, this might be zero)

$T_{LockRear}$ is the locking torque acting on the *rear* side (depending on the clutch mode, this might be zero)

Θ_{CD} is the differential inertia matrix, which depends on the inertia of the differential gears

$\Theta_{GearOut}$ is the transmission inertia reduced to the output shaft

Central differential

Note

The drivetrain mode can be switched between four-, front-, and rear-wheel drive by setting the SW_DrivetrainMode switch to the corresponding value, see the inports definition.

The drivetrain mode is changed by adapting the torque distribution ratio. If a front-wheel drive is selected, the torque distribution ratio is set to one. If a rear-wheel drive is selected, the torque distribution ratio is set to zero.

A part-time four-wheel drive can be simulated by setting the drivetrain mode to four-wheel drive and then the setting of the torque distribution to one (front-wheel drive if part-time four-wheel drive is switched off) or zero (rear-wheel drive if part-time four-wheel drive is switched off). The part-time four-wheel drive can be switched on with the lockup clutch control input Ctrl_LockClutch[%].

Park pawl The parking pawl is simulated by the differential. If the input Ctrl_Parking_Pawl[0|1] is active, the speed integrators are set to zero. The parking pawl should always be simulated in the differential which is directly attached to the transmission.

Inports

The following table shows the inports:

Name	Unit	Description
Ctrl_LockClutch	[%]	Controlling signal to control the clutch lockup torque
Ctrl_Park_Pawl	[0|1]	Controlling signal for the parking pawl of an automatic transmission (only central differential)

Name	Unit	Description
FROM_STABILIZATION	[]	Interface from drivetrain stabilization to ASM block. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
Inertia_Ext_Cage	[kgm^2]	Additional Inertia of parts connected to the cage
Inertia_Trm (central differential) Inertia_Ext_InputShaft (front and rear differential)	[kgm^2]	Equivalent inertia of parts connected to the input shaft, for example, the transmission.
omega_Out_Front_Init	[rad/s]	Initial output speed of differential front shaft
omega_Out_Rear_Init	[rad/s]	Initial output speed of differential rear shaft
Reset	[]	Resets all integrators in the block to their initial conditions
Sw_DiffMode	[1|2|3]	Switch for differential mode: ▪ 1: Torque distributed to the front and rear shafts ▪ 2: The whole torque transmitted to the front shaft ▪ 3: The whole torque transmitted to the rear shaft
Trq_CF_Shaft	[Nm]	Torque exerted on the shaft between central differential and front differential
Trq_CR_Shaft	[Nm]	Torque exerted on the shaft between central differential and rear differential
Trq_Ext_Cage	[Nm]	External torque introduced on the Differential cage
Trq_Input_Shaft	[Nm]	Torque on the input shaft of the differential

Outports

The following table shows the outports:

Name	Unit	Description
omega_Cage	[rad/s]	Speed of the central differential cage
omega_in	[rad/s]	Speed of the differential input shaft
omega_out_Front	[rad/s]	Speed of the differential front output shaft
omega_out_Rear	[rad/s]	Speed of the differential rear output shaft
TO_STABILIZATION	[]	Interface from ASM block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Coef_DynamicClutchTrq	[Nm/(rad/s)]	Change of lock clutch friction torque with relative velocity
Const_Ctrl_Offset_LockClutch	[%]	Offset of clutch lockup torque as a percent of the maximum lockup torque, i.e., previous parameter
Const_i_InnerGear	[]	Ratio of inner gear
Const_i_MainReductionGear	[]	Transmission ratio of differential main reduction gear

Name	Unit	Description
Const_Inertia_FrontSideGear	[kgm^2]	Inertia of the front gear
Const_Inertia_InnerGear	[kgm^2]	Inertia of the inner gear
Const_Inertia_InputGear	[kgm^2]	Inertia of the input gear
Const_Inertia_MainReductionGear	[kgm^2]	Inertia of the main reduction gear and the cage gear
Const_Inertia_RearSideGear	[kgm^2]	Inertia of the rear gear
Const_TrqDistribution (central differential)	[]	Torque distribution ratio
Const_Trq_Max_LockClutch	[Nm]	Maximum torque lock clutch
Sw_DifferentialLock_Mode	[1|2|3]	Position of the lock clutch: ▪ 1: Front rear for central differential or left right for front and rear differential ▪ 2: Cage front for central differential or cage left for front and rear differential ▪ 3: Cage rear for central differential or cage right for front and rear differential
StepSize	[s]	Simulation step size

Related topics

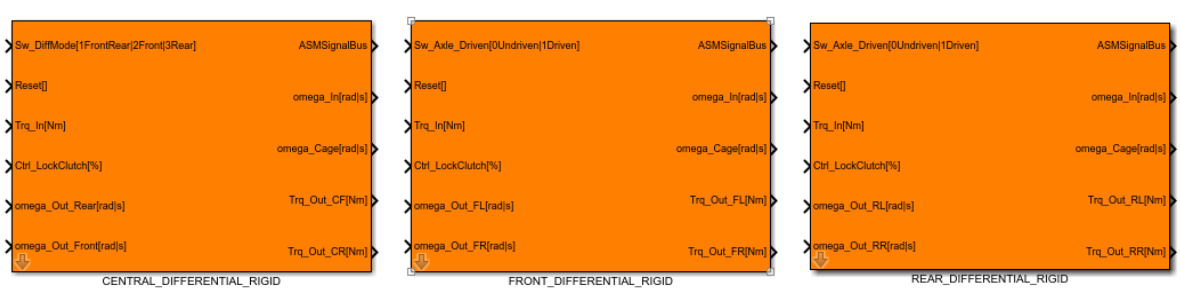
References

Central Differential (ModelDesk Parameterizing 🕮)
Front Differential (ModelDesk Parameterizing 🕮)
Rear Differential (ModelDesk Parameterizing 🕮)

Differentials Rigid

Description

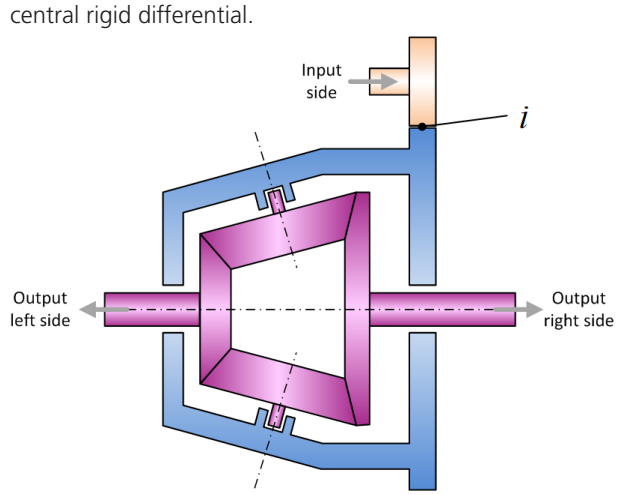
The differential transfers the torque from the input side to two output sides, which are allowed to spin at different speeds.



The differential has three tasks:

- To transfer the engine power to the wheels
- To act as the final gear reduction in the vehicle, e.g. by slowing the rotational speed of the transmission before it reaches the wheels
- To transmit the power to the wheels while allowing them to rotate at different speeds; this is mainly needed for driving in a curve with the outer wheels having to run faster than the inner wheels.

The following explanation uses the central rigid differential as an example. The front and rear rigid differentials can be derived by setting the torque distribution ratio to 0.5. The left and right output sides are equal to the front and the rear output sides, respectively. The following figure shows the physical model of the central rigid differential.



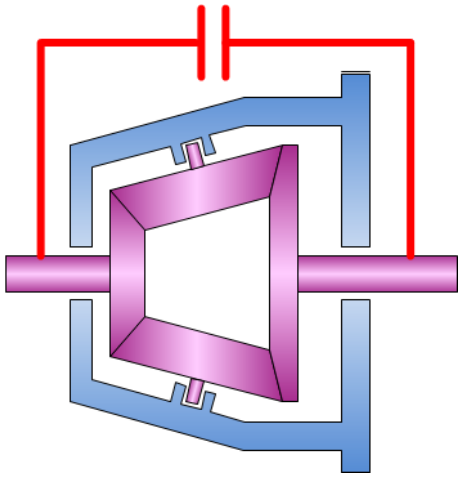
For a differential with a constant torque distribution ratio, the following equations apply:

$$T_R = \rho\, i\, T_{In}$$
$$T_L = (1-\rho) i\, T_{In}$$
$$\Omega_{in} = i(\rho\, \Omega_R + (1-\rho)\Omega_L)$$

where:

T_{In}	is the torque exerted on the input side of the differential
T_R, T_L	are the torques exerted on the right and left sides of the differential
Ω_{In}	is the input speed of the differential
Ω_R, Ω_L	is the speed of the right and left sides of the differential
i	is the transmission ratio of differential main reduction gear
ρ	is the torque distribution to the front side

Moreover, the modeled differential is a lockable open type. Differential locking is implemented by using a clutch model which can be controlled externally. For this type of differentials, just one clutch mode can be simulated. If the clutch is activated, it connects both output sides (as shown in the figure below) and tries to compensate the velocity difference.



If the lock clutch is applied, the calculation of the differential torques is changed to:

$$T_R = \rho\, i\, T_{In} + T_{lock}$$
$$T_L = (1-\rho) i\, T_{In} - T_{lock}$$

where

T_{lock} is the lock clutch torque

The lock clutch is implemented as a friction clutch. The modeling approach is similar to that of the rigid clutch, refer to Rigid Clutch (refer to Clutch on page 18).

You can configure the ratio of the transmitted torque to the right and left side by changing the value of the torque distribution. For example, if the differential is used as a central differential and the torque distribution is changed from 1 to 0, a drivetrain configuration of front wheel as well as rear wheel can be obtained.

A part-time four-wheel drive can be simulated by setting the drivetrain mode to four-wheel drive and the torque distribution to one (front-wheel drive if part-time four-wheel drive is switched off) or zero (rear-wheel drive if part-time four-wheel drive is switched off). The part-time four-wheel drive can be switched on with the lockup clutch control input **Ctrl_LockClutch[%]**.

For this type of differential, the parking pawl cannot be simulated.

Inports

The following table shows the inports:

Name	Unit	Description
Ctrl_LockClutch	[%]	Offset of control signal of differential lock clutch [%]
omega_Out_Front	[rad/s]	Output speed of the front shaft
omega_Out_Rear	[rad/s]	Output speed of the rear shaft
Reset	[]	Reset of states
Sw_DiffMode	[1|2|3]	Switch for differential mode: ▪ 1: Torque distributed to the front and rear sides

Name	Unit	Description
Trq_In	[Nm]	▪ 2: The whole torque transmitted to the front side ▪ 3: The whole torque transmitted to the rear side Input torque

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
omega_Cage	[rad/s]	Cage rotational speed
omega_In	[rad/s]	Input shaft speed
Trq_Out_CF	[Nm]	Output torque on the front side of the central differential
Trq_Out_CR	[Nm]	Output torque on the rear side of the central differential

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Ctrl_Offset_LockClutch	[%]	Offset of control signal of differential lock clutch
Const_Damping_Fric_CentralDiff	[Nm/(rad/s)]	Damping coefficient of lock clutch friction model
Const_Stiffness_Fric_CentralDiff	[Nm/rad]	Stiffness coefficient of lock clutch friction model
Const_TrqDistribution	[]	Torque distribution between the two output sides
Const_Trq_Max_LockClutch	[Nm]	Maximum torque of differential lock clutch
Const_i_MainReductionGear	[]	Ratio of main reduction gear
StepSize	[s]	Simulation step size

Related topics

References

Central Differential Rigid (ModelDesk Parameterizing 🕮)
Front Differential Rigid (ModelDesk Parameterizing 🕮)
Rear Differential Rigid (ModelDesk Parameterizing 🕮)

Gearbox

Where to go from here

Information in this section

Gearbox Automatic Transmission.....53
The function of the gearbox is to change the gear ratio from the engine to the rest of the drivetrain, thereby extending the working range. There are two automatic transmission gearbox models.

Gearbox Automatic Transmission Rigid.....55
The function of the gearbox is to change the gear ratio from the engine to the rest of the drivetrain, thereby extending the working range. There are two automatic transmission gearbox models.

Gearbox Manual Transmission.....56
The function of the gearbox is to change the gear ratio from the engine to the rest of the drivetrain, thereby extending the working range. There are two manual transmission gearbox models.

Gearbox Manual Transmission Rigid.....58
The function of the gearbox is to change the gear ratio from the engine to the rest of the drivetrain, thereby extending the working range. There are two manual transmission gearbox models.

Transfer Gearbox.....60
The transfer gearbox is used to connect different drivetrain components.

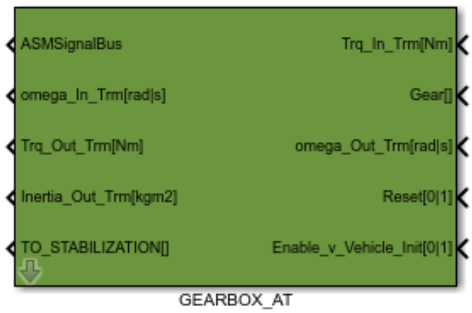
Transfer Gearbox Rigid.....62
The transfer gearbox is used to connect different drivetrain components.

Gearbox Automatic Transmission

Description

The function of the gearbox is to change the gear ratio from the engine to the rest of the drivetrain, thereby extending the working range.

For the automatic transmission gearbox, the same modeling approach is used as for the manual transmission gearbox. Refer to Gearbox Manual Transmission on page 56. The important difference between the models is that for the automatic transmission, the control unit selects the gear. This control unit can be a real ECU or a soft ECU. For more information on the soft ECU, refer to Transmission on page 95.



Inports

The following table shows the inports:

Name	Unit	Description
Enable_v_Vehicle_Init	[0|1]	Drivetrain initialization flag for scenarios with non-zero vehicle initial velocity
Gear	[]	Gear
omega_Out_Trm	[rad/s]	Transmission output speed
Reset	[0|1]	Reset of states
Trq_In_Trm	[Nm]	Transmission input speed

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Inertia_Out_Trm	[kg m^2]	Gearbox equivalent inertia reduced to the output shaft
omega_In_Trm	[rad/s]	Transmission input speed
TO_STABILIZATION	[]	Interface from ASM block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
Trq_Out_Trm	[Nm]	Transmission output torque

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Damping_InputGear	[Nm/(rad/|s)]	Damping coefficient of the gearbox input shaft
Const_Inertia_InputGear	[kg m^2]	Inertia of the input gear for neutral gear
Const_SyncTime	[s]	Synchronization time constant
Map_Forward_Efficiency	[]	Forward efficiency of transmission = f(Gear[])
Map_GearRatio	[]	Gear ratio[] = f(Gear[-])
Map_Inertia_Out	[kg m^2]	Map of transmission inertia reduced to output shaft = f(Gear[])

Name	Unit	Description
Map_Reverse_Efficiency	[]	Reverse efficiency of transmission = f(Gear[])
StepSize	[s]	Simulation step size

Related topics

References

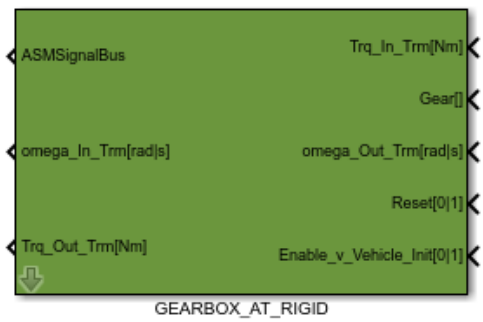
Gearbox Stepped Automatic (Version 8.0 or Earlier) (ModelDesk Parameterizing 🕮)

Gearbox Automatic Transmission Rigid

Description

The function of the gearbox is to change the gear ratio from the engine to the rest of the drivetrain, thereby extending the working range.

The rigid gearbox for automatic transmission is modeled similarly to the rigid gearbox for manual transmission. Refer to Gearbox Manual Transmission Rigid on page 58. The difference is that for the automatic transmission, the control unit selects the gear. This control unit can be a real ECU or a soft ECU. For more information on the soft ECU, refer to Transmission on page 95.



Inports

The following table shows the inports:

Name	Unit	Description
Enable_v_Vehicle_Init	[0|1]	Drivetrain initialization flag for scenarios with non-zero vehicle initial velocity
Gear	[]	Gear
omega_Out_Trm	[rad/s]	Transmission output speed
Reset	[0|1]	Reset of states
Trq_In_Trm	[Nm]	Transmission input torque

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
omega_In_Trm	[rad/s]	Transmission input speed
Trq_Out_Trm	[Nm]	Transmission output torque

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Damping_InputGear	[Nm/(rad/s)]	Damping coefficient of the gearbox input shaft
Const_Inertia_InputGear	[kg m^2]	Input gear inertia for neutral gear
Const_SyncTime	[s]	Synchronization time constant
Map_Forward_Efficiency	[]	Transmission forward efficiency = f(Gear[])
Map_GearRatio	[]	Transmission gear ratio = f(Gear[])
Map_Reverse_Efficiency	[]	Transmission reverse efficiency = f(Gear[])
StepSize	[s]	Simulation step size

Related topics

References

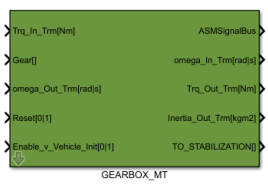
Gearbox Automatic Rigid (ModelDesk Parameterizing 🕮)

Gearbox Manual Transmission

Description

The function of the gearbox is to change the gear ratio from the engine to the rest of the drivetrain, thereby extending the working range.

The gearbox inertia at the output shaft is parameterized with a look-up table which has the current gear as input. The gear efficiency is also modeled using look-up tables. A simple consideration of the gearbox efficiency is proposed as a function of the selected gear. This efficiency is only considered for the torque calculation, i.e., speed-dependent efficiency is neglected.



There are two tables which were used to differentiate between forward and reverse efficiency. One for a positive transferred torque (the engine drives the vehicle) and one for a negative transferred torque (the vehicle drags the engine). The following relations are used to calculate the torque, speed, and inertia of the gearbox:

$$T_{Out} = \eta \cdot i \cdot T_{In}$$

$$\omega_{In} = i \cdot \omega_{Out}$$

$$\Theta_{Out} = f(gear)$$

where:

η	is the gearbox transmission efficiency
i	is the gearbox transmission ratio
T_{In}, T_{Out}	are the gearbox input and output torques, respectively
ω_{In}, ω_{Out}	are the gearbox input and output angular velocities, respectively
Θ_{Out}	is the gearbox inertia reduced to the output side

The synchronization during the shift process is represented by a first-order delay element. This can be described using the following equation:

$$t_{sync} \cdot \dot{i} + i = i_{unsync}$$

where:

t_{sync}	is the synchronization time constant
i_{unsync}	is the unsynchronized gearbox transmission ratio

If the neutral gear is selected, the gearbox input speed is calculated as a new degree of freedom. For this special handling, the input shaft inertia of the gearbox is used as a parameter.

Inports

The following table shows the inports:

Name	Unit	Description
Enable_v_Vehicle_Init	[0|1]	Drivetrain initialization flag for scenarios with non-zero vehicle initial velocity
Gear	[]	Gear requested by the driver
omega_Out_Trm	[rad/s]	Transmission output speed
Reset	[0|1]	Reset of states
Trq_Out_Clutch	[Nm]	Clutch output torque

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Inertia_Out_Trm	[kgm^2]	Gearbox equivalent inertia reduced to the output shaft
omega_In_Trm	[rad/s]	Transmission input speed

Name	Unit	Description
TO_STABILIZATION	[]	Interface from ASM block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮)
Trq_Out_Trm	[Nm]	Transmission output torque

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Damping_InputGear	[Nm/(rad/s)]	Damping coefficient of the gearbox input shaft
Const_Inertia_InputGear	[kgm^2]	Inertia of the input gear for neutral gear
Const_SyncTime	[s]	Synchronization time constant
Map_Forward_Efficiency	[]	Forward efficiency of the transmission []= f(Gear[-])
Map_GearRatio	[]	Gear ratio [] = f(Gear[-])
Map_Inertia_Out	[kgm^2]	Map of transmission inertia reduced to output shaft [kgm^2] = f(Gear[-])
Map_Reverse_Efficiency	[]	Reverse efficiency of the transmission []= f(Gear[-])
StepSize	[s]	Simulation step size

Related topics

References

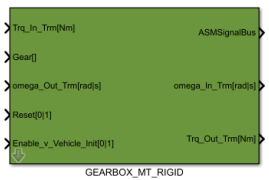
Manual Transmission Basic (ModelDesk Parameterizing 🕮)

Gearbox Manual Transmission Rigid

Description

The function of the gearbox is to change the gear ratio from the engine to the rest of the drivetrain, thereby extending the working range.

The implementation of the rigid gearbox is similar to that of the gearbox (refer to Gearbox Manual Transmission on page 56). However, in this model there is no calculated inertia.



Inports

The following table shows the inports:

Name	Unit	Description
Enable_v_Vehicle_Init	[0|1]	Drivetrain initialization flag for scenarios with non-zero vehicle initial velocity
Gear	[]	Gear requested by the driver
omega_Out_Trm	[rad/s]	Transmission output speed
Reset	[0|1]	Reset of states
Trq_In_Trm	[Nm]	Transmission input torque

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
omega_In_Trm	[rad/s]	Transmission input speed
Trq_Out_Trm	[Nm]	Transmission output torque

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Inertia_InputGear	[kg m^2]	Input gear inertia for neutral gear
Const_Damping_InputGear	[Nm/(rad/s)]	Damping coefficient of the gearbox input shaft
Const_SyncTime	[s]	Synchronization time constant
Map_Forward_Efficiency	[]	Transmission forward efficiency = f(Gear[-])
Map_GearRatio	[]	Transmission gear ratio = f(Gear[-])
Map_Reverse_Efficiency	[]	Transmission reverse efficiency = f(Gear[-])
StepSize	[s]	Simulation step size

Related topics

References

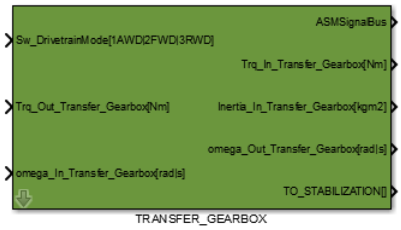
Gearbox Manual Rigid (ModelDesk Parameterizing 🕮)

Transfer Gearbox

Transfer gearbox

The transfer gearbox is used to connect different drivetrain components. It changes the speed and torque of the connected parts according to its transmission ratio.

The gear ratio is used to calculate the output speed and input torque. Moreover, the block considers the inertia of the input and output gear.



The input torque is calculated as follows:

$$T_{In} = \frac{T_{Out}}{i}$$

where:

T_{In}, T_{Out}	are the input and output transfer gearbox torques
i	is the transfer gearbox ratio

The output speed is calculated using the following equation:

$$\omega_{Out} = \frac{\omega_{In}}{i}$$

where:

ω_{Out}, ω_{In}	are the input and output transfer gearbox speeds
i	is the transfer gearbox ratio

Beside the torque and speed, the inertia reduced to the transfer gearbox input is calculated. For the calculation the inertias of both gears, which are provided as parameters, are required. The following equation is used for the calculation:

$$J_{Red} = J_{Gear_in} + \frac{J_{Gear_out}}{i^2}$$

where:

J_{Gear_in}, J_{Gear_out}	are the inertias of the input and output transfer gearbox gears
i	is the transfer gearbox ratio

The mentioned calculation above depends, however, on the current active drivetrain mode. It is only valid if the drivetrain mode is not set to 2FWD (Front-Wheel Drive). In case the current active drivetrain mode equals 2FWD, no inertia is calculated.

Inports

The following table shows the inports:

Name	Unit	Description
omega_In_Transfer_Gearbox	[rad/s]	Input speed
Sw_DrivetrainMode	[1|2|3]	Switch of the drivetrain mode: ▪ 1: AWD (All-wheel drive) ▪ 2: FWD (Front-wheel drive) ▪ 3: RWD (Rear-wheel drive)
TO_STABILIZATION		Interface from ASM block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
Trq_Out_Transfer_Gearbox	[Nm]	Output torque

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Inertia_In_TransferGearbox	[kg m^2]	Inertia reduced to the transfer gearbox input side
omega_Out_Transfer_Gearbox	[rad/s]	Output speed
Trq_In_Transfer_Gearbox	[Nm]	Input torque

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_GearRatio	[]	Gear ratio
Const_Inertia_Input_Gear	[kg m^2]	Inertia of input gear
Const_Inertia_Output_Gear	[kg m^2]	Inertia of the output gear and shaft

Related topics

References

Transfer Gearbox (ModelDesk Parameterizing 🕮)

Transfer Gearbox Rigid

Description

The transfer gearbox is used to connect different drivetrain components. It changes the speed and torque of the connected parts according to its transmission ratio.

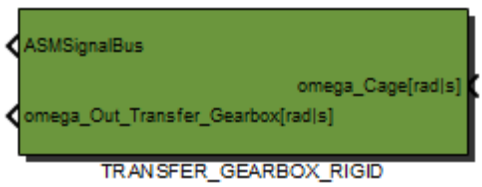
The relationship between the input and output velocity is used without taking the inertia of the connected parts into account. Therefore, the following equation is used for calculation:

$$\omega_{In} = i\,\omega_{Out}$$

where:

i is the transfer gearbox ratio

$\omega_{In}, \omega_{Out}$ are the transfer gearbox input and output angular velocities



Inports

The following table shows the inports:

Name	Unit	Description
omega_Cage	[rad/s]	Cage rotational speed

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
omega_Out_Transfer_Gearbox	[rad/s]	Output speed of the transfer gearbox

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_GearRatio	[]	Gear ratio

Related topics

References

Transfer Gearbox Rigid (ModelDesk Parameterizing 🕮)

Shafts

Where to go from here

Information in this section

Shafts.....64
The shafts are modeled as elastic rods with a damping coefficient.

Rigid Shaft.....66
The block can be used as a general shaft. It calculates the shaft speed from the shaft inertia and the applied torques.

Shafts

Description

The following explanation uses the front left shaft as an example. The implementation of other shafts is the same.

The shafts are modeled as elastic rods with a damping coefficient. The transferred torque is proportional to the relative velocity and twist angle between its terminals, i.e.:

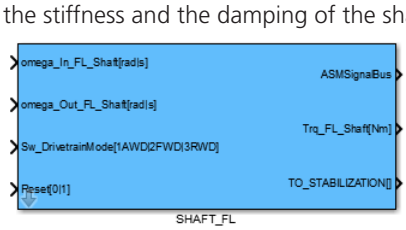
$$T_{Shaft} = k_{Shaft}\,\Delta\theta_{Shaft} + d_{Shaft}\,\Delta\Omega_{Shaft}$$

$$\Delta\theta_{Shaft} = \int \Delta\Omega_{Shaft}\,dt$$

where:

T_{Shaft} is the torque exerted on the shaft
k_{Shaft} is the torsion stiffness of the shaft
d_{Shaft} is the torsion damping of the shaft
$\Delta\Omega_{Shaft}$ is the relative velocity between input and output side of the shaft
$\Delta\theta_{Shaft}$ is the angle of twist between input and output side of the shaft

Depending on the selected drivetrain mode, different parameter sets are used for the stiffness and the damping of the shafts.



Inports

The following table shows the inports:

Name	Unit	Description
omega_In_FL_Shaft	[rad/s]	Shaft input speed (from driving component)
omega_Out_FL_Shaft	[rad/s]	Shaft output speed (to driven component)
Reset	[]	Resets all integrators in the block to their initial conditions
SW_DrivetrainMode	[1|2|3]	Switch of the drivetrain mode: ▪ 1: AWD (All-wheel drive) ▪ 2: FWD (Front-wheel drive) ▪ 3: RWD (Rear-wheel drive)
TO_STABILIZATION	[]	Interface from ASM block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).

Outports

The following table shows the outports:

Name	Unit	Description
Trq_FL_Shaft	[Nm]	Torque exerted on the shaft

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Damping_FourWheelDrive	[Nm/(rad/s)]	Torsional damping coefficient in AWD mode
Const_Damping_FrontWheelDrive	[Nm/(rad/s)]	Torsional damping coefficient in FWD mode
Const_Stiffenss_FourWheelDrive	[Nm/rad]	Torsional stiffness coefficient in AWD mode
Const_Stiffenss_FrontWheelDrive	[Nm/rad]	Torsional stiffness coefficient in FWD mode
StepSize	[s]	Simulation step size

Related topics

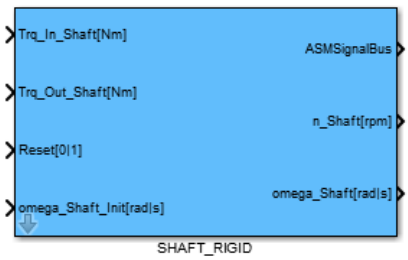
References

Shafts (Front Wheel Front Engine with 4WD Extension) (ModelDesk Parameterizing 🕮)
Shafts (Front Wheel Front Engine) (ModelDesk Parameterizing 🕮)
Shafts All-Wheel Drive (ModelDesk Parameterizing 🕮)
Shafts Front-Wheel Drive (ModelDesk Parameterizing 🕮)
Shafts Rear-Wheel Drive (ModelDesk Parameterizing 🕮)

Rigid Shaft

Description

The block can be used as a general shaft. It calculates the shaft speed from the shaft inertia and the applied torques.



The shaft speed is calculated by integrating the following equation:

$$J_{Shaft} \cdot \dot{\omega}_{Shaft} = (T_{In} - T_{Out}) - D \cdot \omega$$

where:

J_{Shaft}	is the shaft inertia
$\dot{\omega}_{Shaft}$	is the angular acceleration of the shaft
D	is the damping coefficient of the shaft rotation
T_{In}	is the applied torque on the shaft input side
T_{Out}	is the applied torque on the shaft output side

Inports

The following table shows the inports:

Name	Unit	Description
omega_Shaft_Init	[rad/s]	Initial shaft speed
Reset	[0|1]	Reset of states
Trq_In_Shaft	[Nm]	Applied torque on the shaft input side
Trq_Out_Shaft	[Nm]	Applied torque on the shaft output side

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
n_Shaft	[rpm]	Shaft speed in [rpm]
omega_Shaft	[rad/s]	Shaft speed in [rad/s]

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Damping_Coeff_Shaft	[Nm/(rad/s)]	Damping coefficient of shaft rotation
Const_Inertia_Shaft	[kg m^2]	Shaft inertia
StepSize	[s]	Simulation step size

Related topics

References

Shaft Rigid Basic (ModelDesk Parameterizing 🕮)

Stabilization

Stabilization

Description

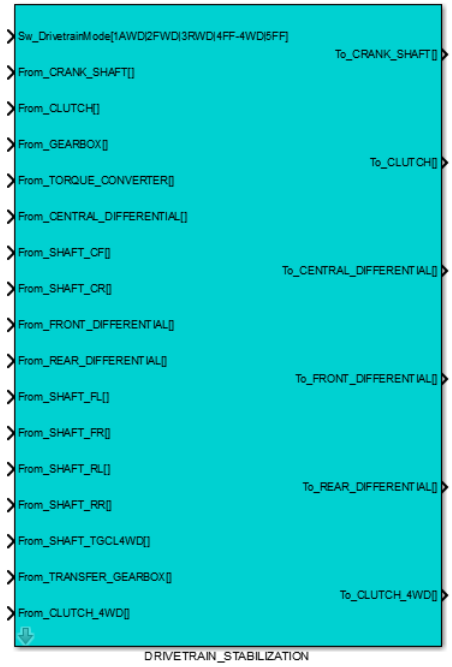
Due to the high stiffness and relatively low inertia of the simulated parts, the flexible drivetrain can result in a stiff system of equations. Therefore, the standard forward Euler solver might lead to numerical problems and instabilities. To avoid these problems, the whole drivetrain system of equations is stabilized using the semi-implicit Euler integration method. For further information on the stabilization equations, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).

The implementation of the drivetrain stabilization can be divided into two categories:

- Stabilization matrices: These are the required matrices for the semi-implicit Euler integration method. The matrices are calculated based on the system information and parameters.
- Stabilization interfaces: The interfaces create the interconnections between the stabilization matrices and the ASM blocks.

Stabilization matrices

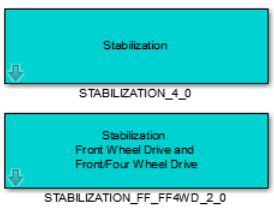
One DRIVETRAIN_STABILIZATION block is used to cover the supported drivetrain configurations. The calculations inside the block are based on the drivetrain system information and parameters, which are provided via inports. Each inport corresponds to a connection from an ASM drivetrain block to the stabilization matrices. The block distinguishes between the different drivetrain configurations using the first inport (Sw_DrivetrainMode[1AWD|2FWD|3RWD|4FF-4WD|5FF]).



The block result is the stabilized acceleration of ASM drivetrain blocks which are integrated inside the corresponding blocks.

Since the DRIVETRAIN_STABILIZATION block supports all the standard variants, the following blocks are obsolete and should not be used with new demo models:

- STABILIZATION_4_0
- STABILIZATION_FF_FF4WD_2_0

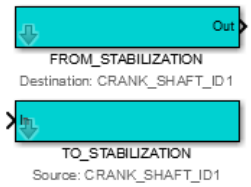


Stabilization interfaces

Interfaces are mainly systems with Goto and From Simulink blocks, which facilitate the information exchange between the stabilization matrices and the ASM blocks. They are connected to the stabilization matrices and to the ASM blocks.

Each ASM block which participates in the stabilization has the corresponding inport and/or outport for signals exchange with the stabilization matrices. The following blocks are used for this purpose:

- FROM_STABILIZATION
- TO_STABILIZATION



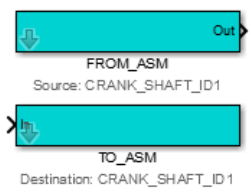
When these interface blocks are connected, they detect a valid ASM block and automatically configure themselves, i.e., source or destination ASM block and its ASMInstanceID. For more information on multi-instances, refer to Multi-Instance (ASM User Guide 🕮).

If an invalid connection is detected, you can set the right configuration.

Similarly, the stabilization matrices block has inports and outports used for signal exchange with ASM blocks.

The exchange is performed by the following blocks:

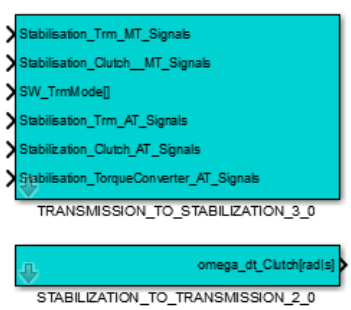
- FROM_ASM
- TO_ASM



These stabilization interfaces are the standard interface blocks.

Therefore, the following blocks are obsolete and should not be used with new demo models from Release 2017-A on:

- TRANSMISSION_TO_STABILIZATION_3_0
- STABILIZATION_TO_TRANSMISSION_2_0



Related topics

References

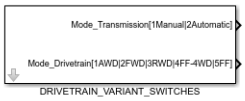
History of the STABILISATION_FF_FF4WD Block..... 437
History of the STABILISATION_TO_TRANSMISSION Block..... 438

Switches

Drivetrain Variant Switches

Description

This block contains switches and parameters to specify the correct model setup.



Parameters

The following tables shows the parameters:

Name	Unit	Description
SW_Drivetrain_Mode	[1|2|3|4|5]	Switch for drivetrain mode: ▪ 1: AWD ▪ 2: FWD ▪ 3: RWD ▪ 4: FF -4WD ▪ 5: FF
SW_Transmission_Mode	[1|2]	Switch for transmission mode: ▪ 1: Manual ▪ 2: Automatic

Engine Basic

Where to go from here

Information in this section

Engine.....73
The engine model is implemented as a 2-D look-up table with engine speed and accelerator pedal position as inputs and the engine torque as an output.

Fuel Consumption.....75
The fuel consumption model computes the fuel consumption and the carbon dioxide emissions.

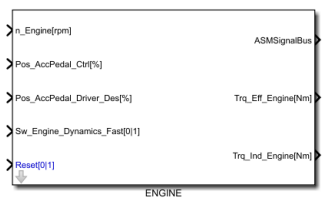
Engine

Description

As the focus of ASM Vehicle Dynamics is on the vehicle dynamics, the engine modeling details, i.e. combustion process, is not considered in this engine model. A 2-D look-up table with engine speed and accelerator pedal position as inputs and the engine torque as an output is implemented instead.

The engine dynamics is approached by a first-order delay element. The engine dynamics for a torque increase differs from the torque decrease dynamics. Two different time constants for the two cases can therefore be set. A third time constant describes the dynamic behavior of a fast torque decrease or increase that is activated with the Sw_Engine_Dynamics_Fast inport.

The following illustration shows the ENGINE block.



Inport

The following table shows the inports.

Name	Unit	Description
n_Engine	[rpm]	Engine speed
Pos_AccPedal_Driver_Des	[%]	Driver accelerator pedal position
Pos_AccPedal_Ctrl	[%]	Accelerator pedal position influenced by engine control
Reset	[0|1]	Reset of states
Sw_Engine_Dynamics_Fast	[0|1]	Switch to activate the fast engine dynamics: ▪ 0: Standard dynamics ▪ 1: Fast dynamics

Outport

The following table shows the outports.

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_Eff_Engine	[Nm]	Mean effective engine torque
Trq_Ind_Engine	[Nm]	Mean indicated engine torque

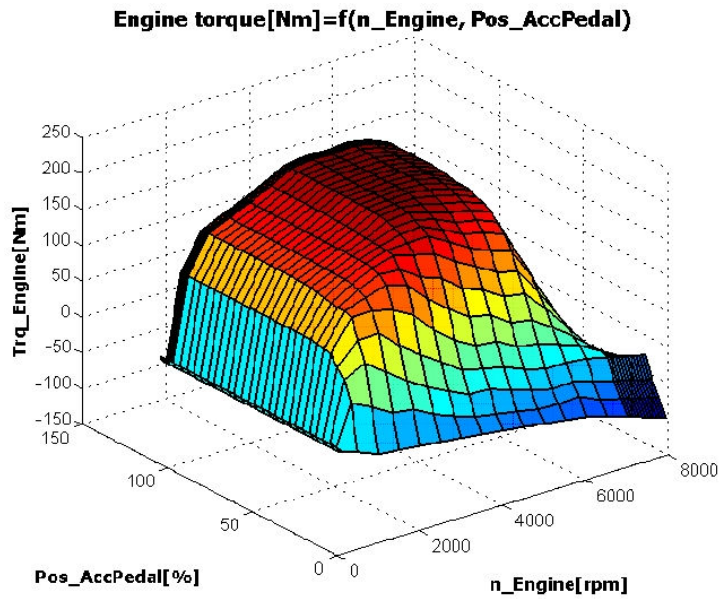
Parameters

The following table shows the parameters.

Name	Unit	Description
Const_TimeConst_Decrease	[s]	Engine dynamic time constant for torque decrease
Const_TimeConst_Fast	[s]	Engine dynamic time constant fast used with engine dynamic fast flag
Const_TimeConst_Increase	[s]	Engine dynamic time constant for torque increase
Map_Torque	[Nm]	Map of effective engine torque [Nm]=f(n_Engine[rpm], Pos_AccPedal[%])
StepSize	[s]	Simulation step size

Processing information

The following illustration shows an example of an engine torque look-up table.



Related topics

References

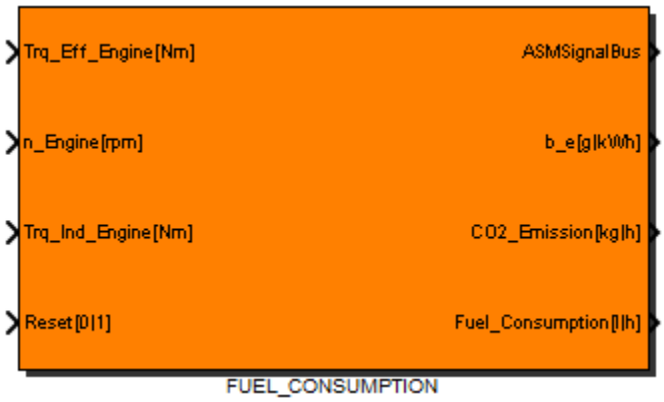
Engine (ModelDesk Parameterizing 🕮)

Fuel Consumption

Description

The model for fuel consumption computes fuel consumption and carbon dioxide emissions. Because ASM Vehicle Dynamics focuses on vehicle dynamics, the computation of fuel consumption is only an approximation.

The following illustration shows the FUEL_CONSUMPTION block.



The calculation is based on the brake-specific fuel consumption, which is a measure of fuel efficiency. However, this measure is only valid for positive values of the effective engine torque. From the lower heating value of the fuel, the minimum brake-specific fuel consumption is calculated using the following formula:

$$b_{e,min} = \frac{T_{ind}}{T_{eff} \cdot Q_{LHV}}$$

where:

Q_{LHV} is the lower heating value of the fuel at reference temperature

T_{ind} is the indicated engine torque

T_{eff} is the effective engine torque

The fuel mass flow rate is then computed by multiplying the brake-specific fuel consumption with the effective engine power:

$$\dot{m}_{driving} = T_{eff} \cdot n \cdot b_e$$

For coasting, i.e., non-positive values of the effective engine torque, the fuel mass flow is extrapolated based on the indicated engine torque:

$$\dot{m}_{coasting} = \frac{T_{ind}}{T_{fric,idle}} \cdot \dot{m}_{idle}$$

where:

$\dot{m}_{idle}$ is the fuel mass flow rate at engine idle speed

$T_{fric,idle}$ is the engine friction torque at engine idle speed

From the current fuel mass flow rate and the fuel properties, the resulting carbon dioxide emissions can be calculated:

$$CO2_{em} = \frac{\dot{m}}{\rho} \cdot CO2_{em,sp}$$

where:

$\dot{m}$ is the current fuel mass flow rate

ρ is fuel density

$CO2_{em,sp}$ is the specific carbon dioxide emissions of the fuel

Fuel consumption is the fuel flow divided by fuel density. By integrating this value over time, you can obtain the total fuel consumption. The total carbon dioxide emissions are calculated in a similar way.

Inports

The following table shows the inports:

Name	Unit	Description
n_Engine	[rpm]	Engine speed
Reset	[0|1]	Reset of states
Trq_Eff_Engine	[Nm]	Effective engine torque
Trq_Ind_Engine	[Nm]	Indicated engine torque

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
b_e	[g/kWh]	Brake-specific fuel consumption
CO2_Emission	[kg/h]	Carbon dioxide emission
Fuel_Consumption	[l/h]	Fuel consumption

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Q_LHV	[J/kg]	Lower heating value of the fuel at reference temperature
Const_Specif_Emission_CO2	[kg/l]	Specific carbon dioxide emissions of the fuel
Const_Trq_Fric_Engine_Idle	[Nm]	Engine friction torque at engine idle speed
Const_V_Displ	[m^3]	Engine displacement
Const_i	[]	Factor for a four stroke engine (0.5) or two stroke engine (1)
Const_mdot_Fuel_Idle	[g/h]	Fuel mass flow rate at engine idle speed
Const_rho_Fuel	[kg/m^3]	Fuel density
Map_Specif_FuelConsumption	[bar]	Map of brake-specific fuel consumption [g/kWh] = n_Engine[rpm], p_Eff_Mean[bar])
StepSize	[s]	Simulation step size

Related topics

References

Fuel Consumption (ModelDesk Parameterizing 🕮)

Soft ECU

Where to go from here

Information in this section

Body.....80
To describe the blocks of the subsystem.

Engine Basic.....84
To describe the blocks of the subsystem.

Steering.....93
To describe the blocks of the subsystem.

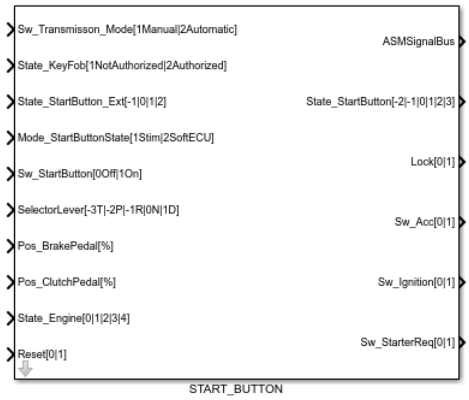
Transmission.....95
To describe the blocks of the subsystem.

Body

Engine Start Button

Description

The START_BUTTON block simulates the engine start button ECU that starts and shuts down the engine. It uses a state machine to simulate the different states. An external switch is used to simplify the key authorization process.



You can use the block with vehicles equipped with manual or automatic transmissions. According to the transmission type, the engine starts when the start button is pressed and held while the start operator (driver) conditions are fulfilled. These conditions are shown in the following table:

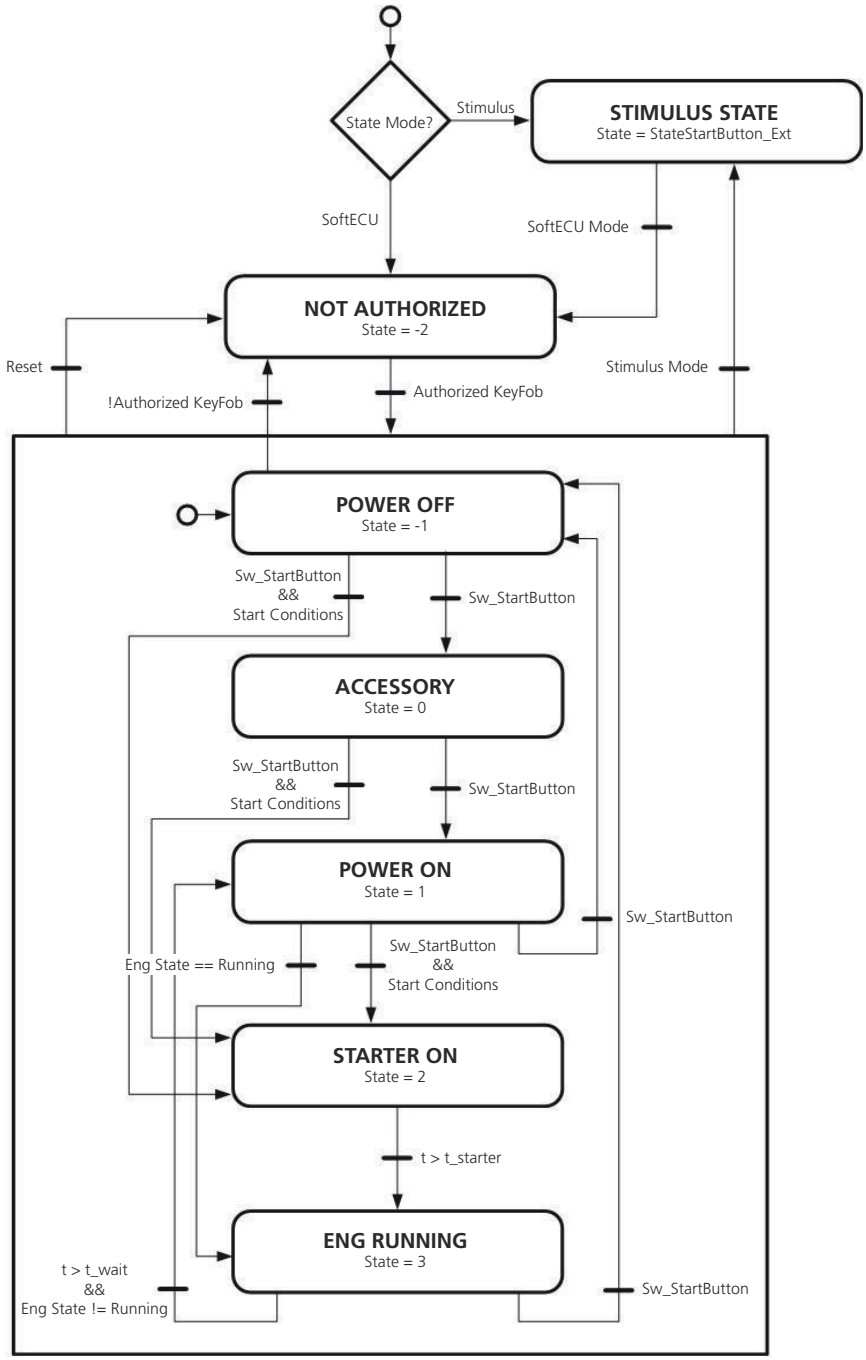
Transmission Type	Conditions
Manual	▪ Authorized key fob ▪ Clutch pedal pressed
Automatic	▪ Authorized key fob ▪ Selector lever in N or P Position ▪ Brake pedal pressed

In addition to the conditions stated above, the system responds only if you press the button and hold it for a time longer than a specified value (`Const_t_Min_Active` parameter). This means short button actuations are ignored.

If the engine is running and you press the button and hold it for a while, the engine switches off.

If the operator conditions above are not fulfilled and you press the button and hold it for a while, this leads to other states in which the engine is not started but other control signals are enabled, such as the accessory signal. You can also stimulate the different states and completely bypass the state machine.

The following state machine shows the working principle of the ECU:



In each state, the ECU calculates different control signals. The following table describes the states and their meanings:

State	Description
Not Authorized	Unauthorized key fob. The engine cannot start.

State	Description
Stimulus	Stimulus state. In this state, the ECU calculates the control signals according to the input: `State_StartButton_Ext[-1|0|1|2]`
Power Off	The default state with an authorized key fob.
Accessory	Accessory is on, for example, you can switch the radio on.
Power On	Power is on, for example, you can control car windows.
Starter On	The ECU sends a starter request to the engine ECU, which can then activate the engine starter.
Engine Running	The ECU detects the engine running state.

Inports

The following table shows the inports:

Name	Unit	Description
Mode_StartButtonState	[1|2]	Start button state mode: ▪ 1: Stimulus ▪ 2: Soft ECU
Pos_BrakePedal	[%]	Brake pedal position
Pos_ClutchPedal	[%]	Clutch pedal position
Reset	[0|1]	Reset of states
SelectorLever	[-3|-2|-1|0|1]	Selector lever position: ▪ -3: TipShift ▪ -2: Park ▪ -1: Reverse ▪ 0: Neutral ▪ 1: Drive
State_Engine	[0|1|2|3|4]	Engine state: ▪ 0: Engine off ▪ 1: Ignition on ▪ 2: Ignition on and starter activated ▪ 3: Engine is running ▪ 4: Ignition is switched off, shutdown active
State_KeyFob	[1|2]	An input that simplifies the key fob authentication process: ▪ 1: Not authorized ▪ 2: Authorized
State_StartButton_Ext	[-1|0|1|2]	Stimulated start button state if the `Mode_StartButtonState` input is set to `Stimulus`: ▪ -1: Power off ▪ 0: Accessory on ▪ 1: Power on ▪ 2: Starter on
Sw_StartButton	[0|1]	Start button actuation input
Sw_Transmission_Mode	[1|2]	Transmission type switch: ▪ 1: Manual

Name	Unit	Description
		▪ 2: Automatic

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Lock	[0|1]	Button lock signal (in case of unauthorized key fob or power off): ▪ 0: Start button is not locked ▪ 1: Start button is locked
State_StartButton	[-2|-1|0|1|2|2]	Start button state: ▪ -2: Unauthorized key fob ▪ -1: Power off ▪ 0: Accessory on ▪ 1: Power on ▪ 2: Starter on ▪ 3: Engine running
Sw_Acc	[0|1]	State of accessories: ▪ 0: Accessories are off ▪ 1: Accessories are on
Sw_Ignition	[0|1]	Ignition signal (terminal 15): ▪ 0: Off ▪ 1: On
Sw_StarterReq	[0|1]	Starter request signal (usually sent to the engine ECU): ▪ 0: Off ▪ 1: On

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Pos_BrakePedal_LowLim	[%]	Minimum value for pressed brake pedal
Const_Pos_ClutchPedal_LowLim	[%]	Minimum value for pressed clutch pedal
Const_t_Min_Active	[s]	Minimum pushing time for active button
Const_t_Min_Starter	[s]	Minimum starter request activation time
StepSize	[s]	Simulation step size

Related topics

References

Start Button (ModelDesk Parameterizing 🕮)

Engine Basic

Where to go from here

Information in this section

Engine Operation Basic.....84
The ENGINE_OPERATION_BASIC block detects the engine operation state from the engine speed and the ignition as well as the starter request switch.

ESP Fast Torque Set.....86
This subsystem is used to realize a fast torque intervention.

Idle Speed Control Engine Basic.....88
The model contains the idle speed control. The controller aims to keep the engine speed at a preset value.

Shift Torque Set.....89
The SHIFT_TORQUE_SET block calculates the desired indicated engine torque depending on the driver's wish and an external intervention.

Torque Intervention Basic.....90
The TORQUE_INTERVENTION_BASIC block is used to realize a torque intervention.

Engine Operation Basic

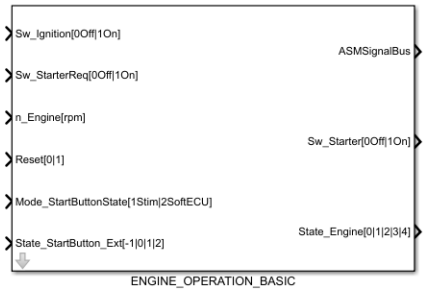
Description

The ENGINE_OPERATION_BASIC block detects the engine operation state from the engine speed and the ignition as well as the starter request switch. It activates the starter and gives information about the engine state.

The following engine states are possible:

- 0: Engine off/Error
- 1: Ignition on
- 2: Starter on
- 3: Engine running
- 4: Engine shutdown

The following illustration shows the Simulink implementation of the block:



Inports

The following table shows the inports:

Name	Unit	Description
Mode_StartButtonState	[1|2]	Mode of start button state: ▪ 1: Stimulus ▪ 2: Soft ECU
n_Engine	[rpm]	Engine speed
Reset	[0|1]	Reset of states
State_StartButton_Ext	[-1|0|1|2]	External start button state in case of stimulus: ▪ -1: Power off ▪ 0: Power on ▪ 1: Accessory ▪ 2: Starter on
Sw_Ignition	[0|1]	Switch for ignition signal (terminal 15): ▪ 0: Off ▪ 1: On
Sw_StarterReq	[0|1]	Switch for starter request signal (usually sent to engine ECU): ▪ 0: Off ▪ 1: On

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
State_Engine	[0|1|2|3|4]	Engine state ▪ 0: Engine off ▪ 1: Ignition on ▪ 2: Ignition on and starter activated ▪ 3: Engine is running ▪ 4: Ignition is switched off, shutdown active
Sw_Starter	[0|1]	Starter activate switch (output of ECU) ▪ 0: Off

Name	Unit	Description
		▪ 1: On

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_n_Engine_LowLim_Shutdown	[rpm]	Lower engine speed limit for shutdown state
Const_n_Engine_Min_EngineOn	[rpm]	Minimum engine speed when engine is on
Const_t_Max_Starter	[s]	Maximum duration for switching the starter on
Const_t_Shutdown	[s]	Engine shutdown duration

Related topics

References

Engine Operation Basic (ModelDesk Parameterizing 🕮)

ESP Fast Torque Set

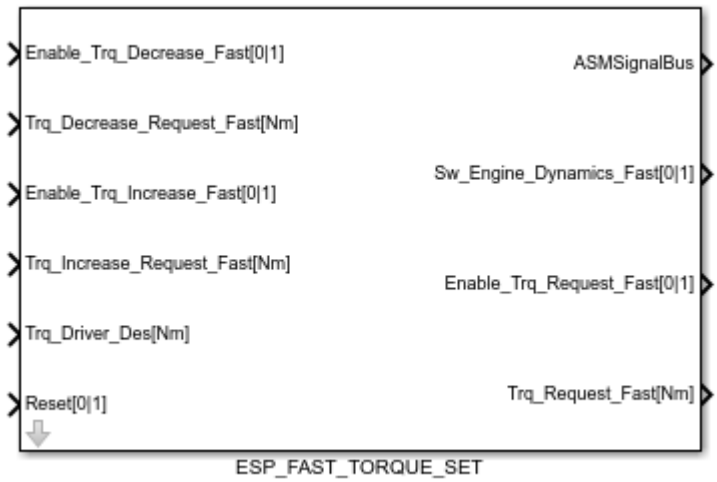
Description

This subsystem is used to realize a fast torque intervention. Because the engine basic model (see Engine on page 73) does not simulate the functionality of a fast torque intervention, for example, via ignition or injection control, a switch flag (Sw_Engine_Dynamics_Fast) is used. This flag signal is routed to the engine basic and is used there to increase the engine dynamic.

If the enable flag signals are set, the corresponding requested torque is applied as output torque. The transition between the requested torques is carried out via a first-order delay.

The driver desired torque has also an influence to the torque requests. At a torque increase situation the maximum value of the torque request and the driver desired torque is used. At a torque decrease situation the minimum value of the torque request and the driver desired torque is used.

The following illustration shows the ESP_FAST_TORQUE_SET block.



Inports

The following table shows the inports.

Name	Unit	Description
Enable_Trq_Decrease_Fast	[0|1]	Signal from ESP ECU to enable or disable the fast torque decreasing request
Enable_Trq_Increase_Fast	[0|1]	Signal from ESP ECU to enable or disable the fast torque increasing request
Reset	[]	Reset all integrators in the block to their initial condition
Trq_Decrease_Request_Fast	[Nm]	The torque request from ESP ECU, fast decreasing
Trq_Driver_Des	[Nm]	Driver desired torque
Trq_Increase_Request_Fast	[Nm]	The torque request from ESP ECU, fast increasing

Outports

The following table shows the outports.

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Enable_Trq_Intervention_Fast	[0|1]	Signal to enable or disable a fast torque request
Sw_Engine_Dynamics_Fast	[0|1]	Flag, routed to engine basic, to increase the engine dynamic ▪ 0: Standard ▪ 1: Fast
Trq_Request_Fast	[Nm]	The torque request fast

Parameters

The following table shows the parameters.

Name	Unit	Description
StepSize	[s]	Simulation step size

Related topics

References

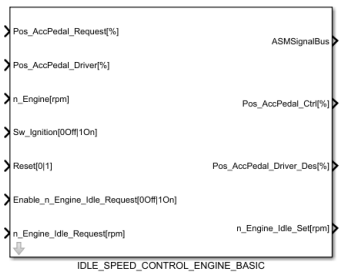
History of the TORQUE_INTERVENTION_FAST_ENGINE_BASIC Block..... 458

Idle Speed Control Engine Basic

Description

The IDLE_SPEED_CONTROL_ENGINE_BASIC block builds the idle speed controller used to keep the engine speed at a certain value while the engine is idling. The controller is implemented as a PI controller. In addition, the model accepts an external idle speed request, i.e., external idle speed request from the transmission control unit (TCU). This functionality can be activated with the Sw_n_Engine_Idle_Request and n_Engine_Idle_Request inports.

The following illustration shows the Simulink representation of the block:



Inports

The following table shows the inports.

Name	Unit	Description
Enable_Ignition	[0|1]	Ignition signal (terminal 15) ▪ 0: Off ▪ 1: On
n_Engine	[rpm]	Engine speed
n_Engine_Idle_Request	[rpm]	Engine idle speed request
Pos_AccPedal_Driver	[%]	Driver accelerator pedal position
Pos_AccPedal_Request	[%]	Accelerator pedal position influenced by torque requests
Reset	[]	Reset of states

Name	Unit	Description
Sw_n_Engine_Idle_Request	[0|1]	Switch to activate engine idle speed request ▪ 0: Off ▪ 1: On

Outports

The following table shows the outports.

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
n_Engine_Idle_Set	[rpm]	Set point for idle speed of combustion engine
Pos_AccPedal_Driver_Des	[%]	Driver desired accelerator pedal position without controller intervention
Pos_AccPedal_Ctrl	[%]	Accelerator pedal position influenced by the controller

Parameters

The following table shows the parameters.

Name	Unit	Description
Const_I_Idle	[1/s]	I gain of the idle speed controller
Const_P_Idle	[]	P gain of the idle speed controller
Const_n_EngineIdle_Set	[rpm]	Engine idle speed setpoint
StepSize	[s]	Simulation step size

Related topics

References

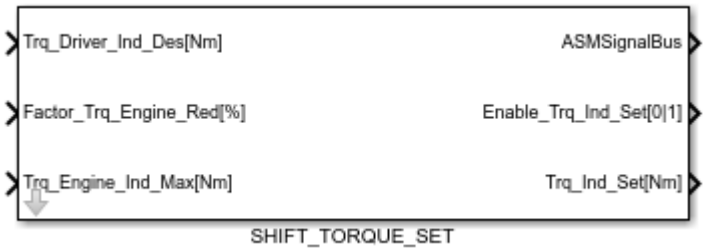
Idle Speed Control Basic (ModelDesk Parameterizing 🕮)

Shift Torque Set

Description

The SHIFT_TORQUE_SET block calculates the desired indicated engine torque depending on the driver's wish and the transmission control unit intervention.. The block offers a coupling with the Soft ECU Transmission and allows a torque reduction during the gearshift process.

The following illustration shows the Simulink representation of the engine model.



Inports

The following table shows the inports:

Name	Unit	Description
Factor_Trq_Engine_Red[%]	[%]	Engine torque reduction factor
Trq_Driver_Ind_Des	[Nm]	Induced engine torque setpoint of driver
Trq_Engine_Ind_Max	[Nm]	Maximum induced engine torque

Outports

The following table shows the outports:

Name	Unit	Description
Enable_Trq_Ind_Set	[0|1]	Activation flag of engine torque intervention
Trq_Ind_Set	[Nm]	Induced engine torque setpoint

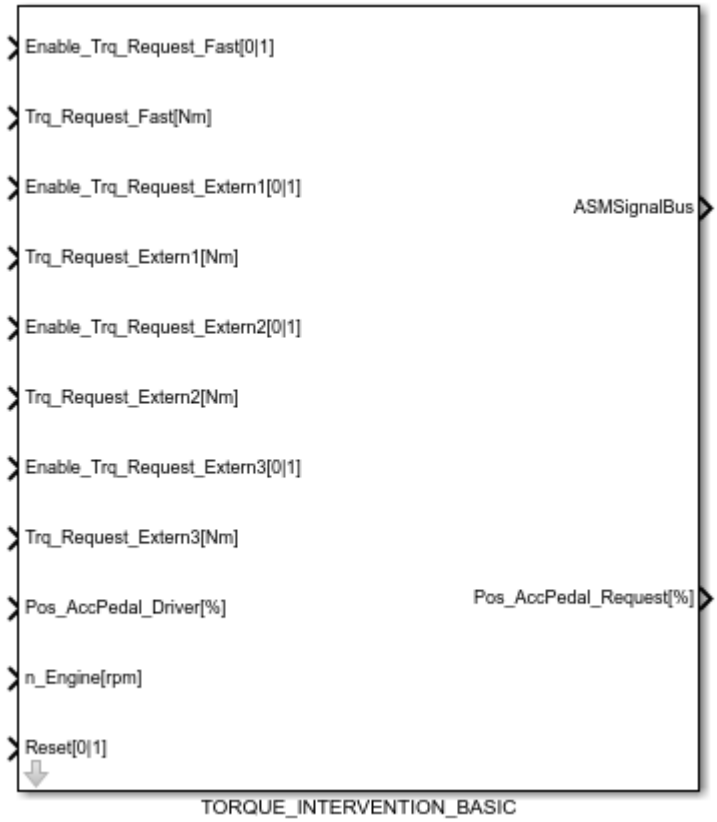
Torque Intervention Basic

Description

The TORQUE_INTERVENTION_BASIC block is used to implement a torque intervention, i.e., an external torque request from the transmission control unit (TCU). For this, an accelerator pedal control is implemented in this model. To get a certain torque from the engine, the accelerator pedal position must be calculated at the current engine speed. This is achieved by using an inversed engine torque map.

Depending on different enable flags and priorities, the corresponding accelerator pedal position is output. In this case, the external torque request has the highest priority and is followed by the driver torque request.

The following illustration shows the Simulink implementation of the block:



Inports

The following table shows the inports:

Name	Unit	Description
Enable_Trq_Request_Extern1	[0|1]	Signal to enable or disable the 1st external torque request
Enable_Trq_Request_Extern2	[0|1]	Signal to enable or disable the 2nd external torque request
Enable_Trq_Request_Extern3	[0|1]	Signal to enable or disable the 3rd external torque request
Enable_Trq_Request_Fast	[0|1]	Signal to enable or disable the fast torque request
n_Engine	[rpm]	Engine speed
Pos_AccPedal_Driver	[%]	Driver accelerator pedal position
Reset	[0|1]	Reset of states
Trq_Request_Extern1	[Nm]	1st external torque request
Trq_Request_Extern2	[Nm]	2nd external torque request
Trq_Request_Extern3	[Nm]	3rd external torque request
Trq_Request_Fast	[Nm]	Fast torque request

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Pos_AccPedal_Request	[%]	Requested accelerator pedal position

Parameter

The following table shows the parameters:

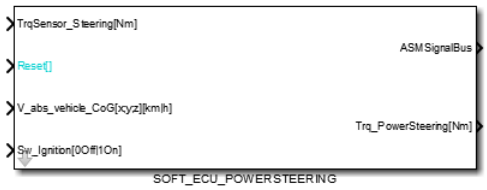
Name	Unit	Description
Map_Trq_Engine_Inv	[%]	Inverse engine map Pos_AccPedal = f(n_Engine[rpm], Trq_Engine[Nm])

Steering

Soft ECU PowerSteering

Description

The SOFT_ECU_POWERSTEERING block simulates a power steering ECU. The block controls the assistive torque provided in the steering system by the EPS system. The desired assistive torque output provided depends on vehicle velocity and steering column spring torque. When the soft ECU power steering model is used, the block's Trq_PowerSteering[Nm] output must be connected to the Trq_PowerSteering[Nm] input signal of the STEERING_3DOF_VARIABLE_RATIO block.



Inports

The following table shows the inports:

Name	Unit	Description
Reset	[]	Resets all integrators in the block to their initial condition
Sw_Ignition	[0|1]	Switch for activating assistive torque for power steering motor ▪ 0: Assistive torque inactive ▪ 1: Assistive torque active
TrqSensor_Steering	[Nm]	Sensor torque at steering column
TrqSpring_SteeringColumn	[Nm]	Spring torque at steering column
V_abs_vehicle_CoG[x;y;z]	[km/h]	Absolute velocity of the vehicle center of gravity

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_PowerSteering	[Nm]	Torque at power steering motor (internal EPS simulation)

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_t_PowerSteering	[]	Time constant of power steering dynamics [s] Default : 0.07s
Map_Trq_PowerSteering	[Nm]	Torque power steering [Nm] = f(Torque_SteeringColumn[Nm], VehicleSpeed[km|h])

Related topics

References

Soft ECU PowerSteering (ModelDesk Parameterizing 🕮)

Transmission

Where to go from here

Information in this section

Clutch Engagement Control.....95
The CLUTCH_ENGAGEMENT_CONTROL block controls the clutch actuation according to the requested gear.

Lockup Clutch Control.....97
The LOCKUP_CLUTCH_CONTROL block controls the lockup clutch in automatic transmissions.

Shift Lock Control.....99
The SHIFT_LOCK_CONTROL block activates the parking lock when the selector lever is in P position.

Shift Strategy.....100
The SHIFT_STRATEGY block controls the gearshift request when the selector lever is in (D) or (R) position.

Soft ECU Transmission Setup.....103
The SOFTECU_TRANSMISSION_SETUP block provides dimension and general setup parameters for the Soft ECU Transmission.

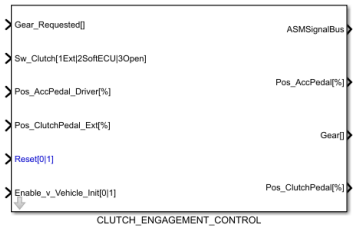
TipShift Control.....103
The TIP_SHIFT_CONTROL block allows external shift request when the selector lever is in T position.

Torque Intervention Control.....105
The TORQUE_INTERVENTION_CONTROL block builds the torque intervention during the gearshift as a scaling of the engine torque in percentage.

Clutch Engagement Control

Description

The CLUTCH_ENGAGEMENT_CONTROL block controls the clutch actuation according to the requested gear. It can be used for different types of transmission, e.g., with an automated manual transmission (AMT). Such a clutch control is also needed in parallel hybrid topologies where the clutch is located between the electric machine and the transmission.



The block does not provide feedback control. Instead, only user-defined clutch disengagement and engagement is used. Moreover, you can use external stimulus signals to specify the clutch pedal position.

During clutch actuation, the accelerator pedal can be scaled with a predefined factor. The gear request is passed to the output once the clutch is completely open.

Inports

The following table shows the inports:

Name	Unit	Description
Enable_v_Vehicle_Init	[0|1]	Drivetrain initialization flag for scenarios with non-zero vehicle initial velocity
Gear_Requested	[]	Requested gear
Pos_AccPedal_Driver	[%]	Accelerator pedal position from driver
Pos_ClutchPedal_Ext	[%]	External stimulus clutch pedal position
Reset	[0|1]	Reset of states
Sw_Clutch	[1|2|3]	Clutch pedal position mode: ▪ 1: External ▪ 2: Soft ECU ▪ 3: Open

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Gear	[]	Requested gear after clutch is completely disengaged
Pos_AccPedal	[%]	Scaled accelerator pedal position as a function of the requested gear time
Pos_ClutchPedal	[%]	Controlled clutch pedal position as a function of the requested gear time

Parameters

The following table shows the parameters:

Name	Unit	Description
Map_Factor_Pos_AccPedal	[%]	Percentage of accelerator pedal intervention as a function of the requested gear time

Name	Unit	Description
Map_Pos_ClutchPedal_Disengage	[%]	Clutch pedal disengagement position as a function of the requested gear time
Map_Pos_ClutchPedal_Engage	[%]	Clutch pedal engagement position after clutch disengagement finished
StepSize	[s]	Simulation step size

Related topics

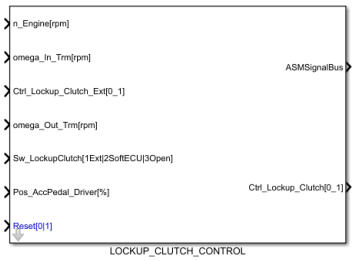
References

Clutch Engagement Control (ModelDesk Parameterizing 🕮)

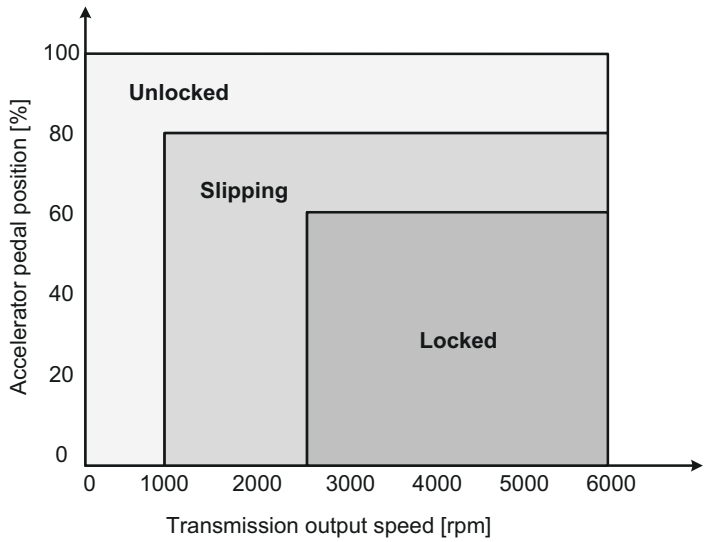
Lockup Clutch Control

Description

The LOCKUP_CLUTCH_CONTROL block controls the lockup clutch in automatic transmissions. The main function of the lockup clutch is to mechanically connect and disconnect the engine from the rest of the transmission units to increase the efficiency and avoid oscillations in the drivetrain.



The control strategy depends on the lockup state (unlock, slip, or lock), which is determined according to the transmission output speed and the accelerator pedal position as shown in the following illustration.



The lockup clutch is unlocked at the lowest transmission output speed and the highest accelerator pedal position to enable vehicle startup and isolate the engine oscillations, respectively.

The lockup clutch is locked at relatively high transmission speeds to increase the efficiency of the drivetrain by avoiding losses in the torque converter.

Between the lock and unlock states, slipping is allowed. Lockup clutch slipping is controlled by using a PI controller with the speed ratio as a feedback signal. The actual speed ratio is calculated as the ratio of the transmission input speed to the engine speed and can be used as an indicator of slipping in the torque converter.

Moreover, it is possible to use external stimulus signals for the lockup clutch control.

Inports

The following table shows the inports:

Name	Unit	Description
Ctrl_Lockup_Clutch_Ext	[0_1]	External stimulus lockup clutch control
n_Engine	[rpm]	Engine speed
omega_In_Trm	[rpm]	Transmission input speed
omega_Out_Trm	[rpm]	Transmission output speed
Pos_AccPedal_Driver	[%]	Accelerator pedal position from the driver
Reset	[0|1]	Reset of states
Sw_LockupClutch	[1|2|3]	Lockup clutch control mode: ▪ 1: External ▪ 2: Soft ECU ▪ 3: Open

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Ctrl_Lockup_Clutch	[0_1]	Lockup clutch control signal

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_P_LockupClutch	[]	P gain of lockup clutch controller
Const_I_LockupClutch	[1/s]	I gain of lockup clutch controller
Map_State_LockupClutch	[0_1]	Lockup clutch state = f(Pos_AccPedal,omega_Out_Trm)
StepSize	[s]	Simulation step size

Related topics

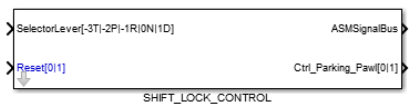
References

Lockup Clutch Control (ModelDesk Parameterizing 🕮)

Shift Lock Control

Description

The SHIFT_LOCK_CONTROL block activates the parking lock when the selector lever is in the P position.



Inports

The following table shows the inports:

Name	Unit	Description
Reset	[0|1]	Reset of states
SelectorLever	[-3|-2|-1|0|1]	Selector lever position for automatic transmission: ▪ -3 : TipShift ▪ -2 : Park ▪ -1 : Reverse ▪ 0 : Neutral ▪ 1 : Drive

Outports

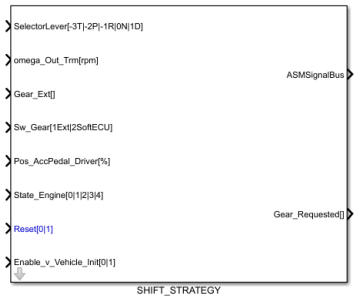
The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Ctrl_Parking_Pawl	[0|1]	Parking pawl control

Shift Strategy

Description

The SHIFT_STRATEGY block controls the gearshift request when the selector lever is in the D or R position.

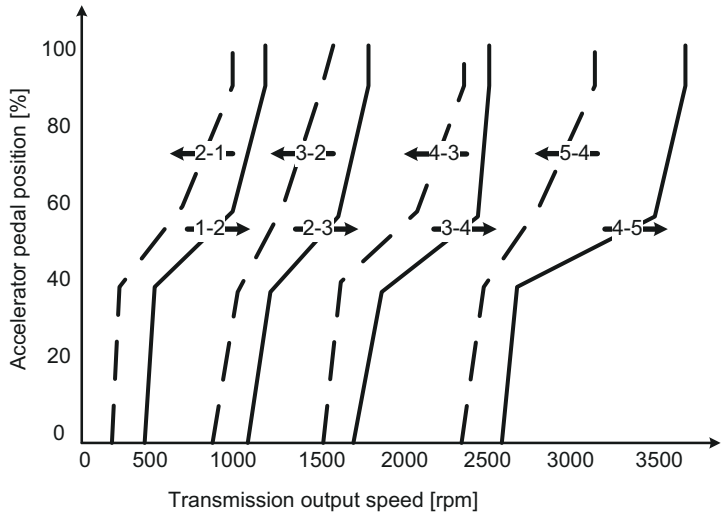


The shift request is controlled according to predefined shift patterns, which depend on the transmission output speed and the accelerator pedal position.

Two shift patterns for the up shift and down shift are parameterized in the block. Each shift pattern calculates the shift thresholds of the transmission output speed for several possible gears. These thresholds are then compared with the actual transmission output speed to request an up shift or down shift.

Shift patterns

The following figure explains the shift patterns:



To illustrate the shift strategy, up shift 1-2 and down shift 2-1 are clarified in the following:

As the driver shifts the selector lever to the D position, the first gear is requested and engaged at the same time. Meanwhile, the transmission output speed is increased to some point between curves 2-1 and 1-2. If the driver demands more acceleration by pressing the accelerator pedal, the transmission output speed increases accordingly and curve 1-2 is crossed. After curve1-2 is crossed, the second gear is requested.

Down shift 2-1 can be understood in a similar way. As the vehicle decelerates, the transmission output speed is decreased to a point where curve 2-1 is crossed and the first gear is requested.

To prevent the phenomenon of gear hunting, i.e., frequent up shifting and down shifting, a minimum time between two successive shifts is observed (**Const_t_Min_Shift[s]** parameter). In this way, and after successfully shifting gears, no additional shift request occurs within the specified time. If the user sets a sufficient time, gears can be skipped. However, this strongly depends on the shift patterns and the driving situation.

The gear request can be reset at any time by using the reset of states. Moreover, the gear request is reset automatically if the engine was stalled.

You can also use external stimulus gear requests instead of the automatic control.

Inports

The following table shows the inports:

Name	Unit	Description
Enable_v_Vehicle_Init	[0|1]	Drivetrain initialization flag for scenarios with non-zero vehicle initial velocity
Gear_Ext	[]	External stimulus gear

Name	Unit	Description
omega_Out_Trm	[rpm]	Transmission output speed
Pos_AccPedal_Driver	[%]	Accelerator pedal position from driver
Reset	[0|1]	Reset of states
SelectorLever	[-3|-2|-1|0|1]	Selector lever position for automatic transmission: ▪ -3 : TipShift ▪ -2 : Park ▪ -1 : Reverse ▪ 0 : Neutral ▪ 1 : Drive
State_Engine	[0|1|2|3|4]	Switch for state of engine: ▪ 0: Engine off / Error ▪ 1: Ignition on ▪ 2: Starter on ▪ 3: Engine running ▪ 4: Engine shutdown
Sw_Gear	[1|2]	Gear mode: ▪ 1: External ▪ 2: Soft ECU

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Gear_Requested	[]	Requested gear

Parameters

The following table shows the parameters:

Name	Unit	Description
Map_omega_Down shift	[rpm]	Transmission output speed for gear down shift schedule = f(Pos_AccPedal,Gear_Current)
Map_omega_Up shift	[rpm]	Transmission output speed for gear up shift schedule = f(Pos_AccPedal,Gear_Current)
StepSize	[s]	Simulation step size

Related topics

References

Shift Strategy (ModelDesk Parameterizing 🕮)

Soft ECU Transmission Setup

Description

The SOFTECU_TRANSMISSION_SETUP block provides dimension and general setup parameters for the Soft ECU Transmission model. For example, it provides the highest possible requested gear. The behavior can also be influenced by an external request.

Inports

The following table shows the inports:

Name	Unit	Description
Gear_Max_Ext	[1_n]	External highest gear

Outports

The following table shows the outports:

Name	Unit	Description
Const_Max_num_Gear	[1_n]	Highest requested gear

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Max_num_Gear	[]	Highest requested gear

Related topics

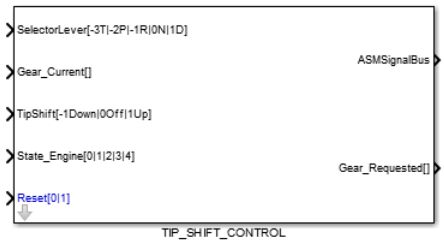
References

SoftECU Transmission Setup (ModelDesk Parameterizing 🕮)

TipShift Control

Description

The TIP_SHIFT_CONTROL block allows external shift requests when the selector lever is in the T position. The shift requests are triggered via the TipShift lever.



If the driver shifts the selector lever to the T position, the TipShift signal is used for gearshifting.

This signal has three values:

- -1: down shift
- 0: off
- 1: up shift

Each time the driver tips the TipShift up or down, a gear up shift or down shift is requested, respectively.

However, two conditions have to be fulfilled:

- The lowest possible gear is -1.
- A gearshift request can only be considered if the previous requested gear was successfully engaged.

Inports

The following table shows the inports:

Name	Unit	Description
Gear_Current	[]	Current gear
Reset	[0|1]	Reset of states
SelectorLever	[-3|-2|-1|0|1]	Selector lever position for automatic transmission: ▪ -3 : TipShift ▪ -2 : Park ▪ -1 : Reverse ▪ 0 : Neutral ▪ 1 : Drive
TipShift	[-1|0|1]	TipShift request signal for TipTronic function: ▪ -1 : Down shift ▪ 0 : Off ▪ 1 : Up shift
State_Engine	[0|1|2|3|4]	Engine state: ▪ 0: Engine off/Error ▪ 1: Ignition on ▪ 2: Starter on ▪ 3: Engine running ▪ 4: Engine shutdown

Outports

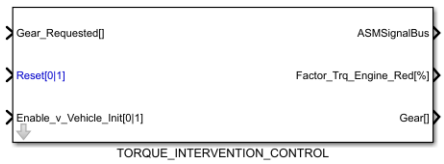
The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Gear_Requested	[]	Requested gear

Torque Intervention Control

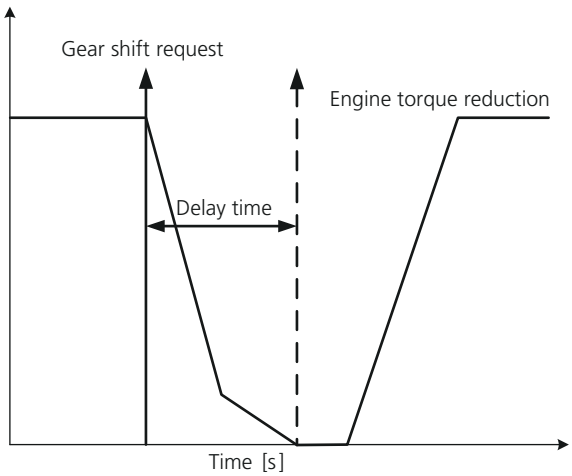
Description

The TORQUE_INTERVENTION_CONTROL block specifies the torque intervention while shifting gears as a scaling of the engine torque in percent. The requested gear is passed to the output after a predefined delay time.



This block connects the transmission ECU and the engine ECU, because an engine torque reduction is usually requested while shifting gears. The request from the transmission is converted to a set value for the engine ECU.

The following illustration shows an example of the torque intervention request. The torque is reduced once a gear is requested. This gear request is then delayed by a user-defined time:



The torque is reduced once a gear is requested. This gear request is then delayed by a user-defined time.

Inports

The following table shows the inports:

Name	Unit	Description
Enable_v_Vehicle_Init	[0|1]	Drivetrain initialization flag for scenarios with non-zero vehicle initial velocity
Gear_Requested	[]	Requested gear
Reset	[0|1]	Reset of states

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Factor_Trq_Engine_Red	[%]	Percentage factor of engine torque
Gear	[]	Requested gear after engine torque intervention

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_t_Gear_Offs	[s]	Requested gear delay time
Map_Factor_Trq_Engine_Red	[%]	Percentage of engine torque intervention as a function of the requested gear time
StepSize	[s]	Simulation step size

Related topics

References

Torque Intervention Control (ModelDesk Parameterizing 🕮)

Vehicle Dynamics

Where to go from here

Information in this section

Aerodynamics..... 108
To describe the blocks of the subsystem.

Brake..... 112
To describe the blocks of the subsystem.

Common Paramter..... 124
To describe the blocks of the subsystem.

Suspension..... 127
To describe the blocks of the subsystem.

Steering..... 237
To describe the blocks of the subsystem.

Tire..... 259
To describe the blocks of the subsystem.

Vehicle Movement..... 275
To describe the blocks of the subsystem.

Wheel Speed..... 293
To describe the blocks of the subsystem.

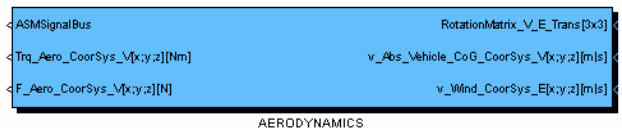
Switches..... 296
To describe the switches of the subsystem.

Aerodynamics

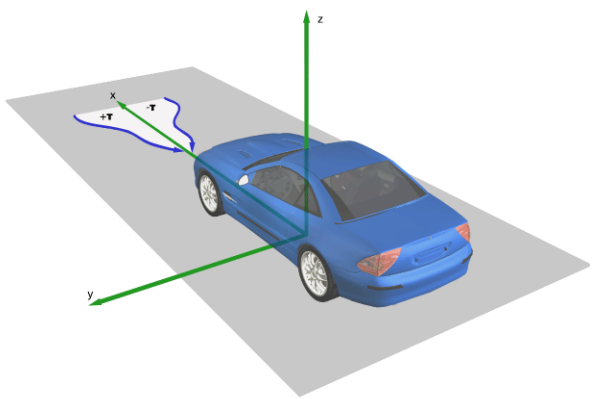
Aerodynamics

Description

The effects of aerodynamics are represented by aerodynamics forces and torques acting on the vehicle sprung mass.



The aerodynamics forces and torques are applied in the aerodynamics coordinate system. This body-fixed system is normally located in the ground plane (x-y-plane) at the middle of the wheelbase and track width. The following illustration shows the aerodynamics reference coordinate system and angle of incidence.



The forces and torques are derived from wind tunnel tests and can be calculated as follows:

$$F_{Aero,x} = -\frac{1}{2}\rho_{Air} \cdot v^2 \cdot C_x(\tau) \cdot A_x$$

$$F_{Aero,y} = -\frac{1}{2}\rho_{Air} \cdot v^2 \cdot C_y(\tau) \cdot A_x$$

$$F_{Aero,z} = \frac{1}{2}\rho_{Air} \cdot v^2 \cdot C_z(\tau) \cdot A_x$$

$$Trq_{Aero,x} = \frac{1}{2}\rho_{Air} \cdot v^2 \cdot C_{Mx}(\tau) \cdot A_x \cdot L_{char}$$

$$Trq_{Aero,y} = -\frac{1}{2}\rho_{Air} \cdot v^2 \cdot C_{My}(\tau) \cdot A_x \cdot L_{char}$$

$$Trq_{Aero,z} = -\frac{1}{2}\rho_{Air} \cdot v^2 \cdot C_{Mz}(\tau) \cdot A_x \cdot L_{char}$$

where:

$V = (V_{Vehicle} - V_{Wind})_{xy}$

$V_{Vehicle}$	is the velocity of vehicle [m/s]
V_{Wind}	is the velocity of wind [m/s]
ρ_{Air}	is the density of air [kg/m^3]
A_x	is the longitudinal shadow area of the vehicle [m^2]
L_{char}	is the characteristic length for torque calculation [m] (mostly vehicle wheelbase)
C_τ	is the aerodynamics coefficients [-]
τ	is the angle of incidence [deg]

Note

Only the longitudinal shadow area is used in the equations.

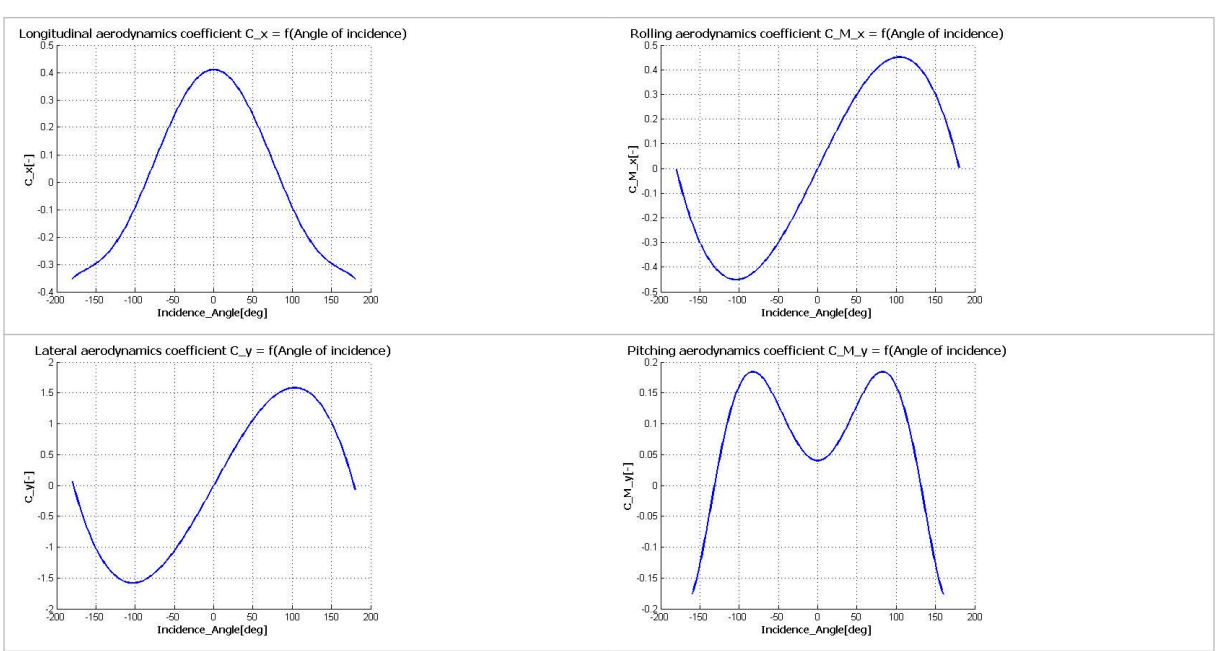
The imposed torque arising from the aerodynamics forces acting on the origin of the vehicle coordinate system is calculated from the position vector to the aerodynamics reference point:

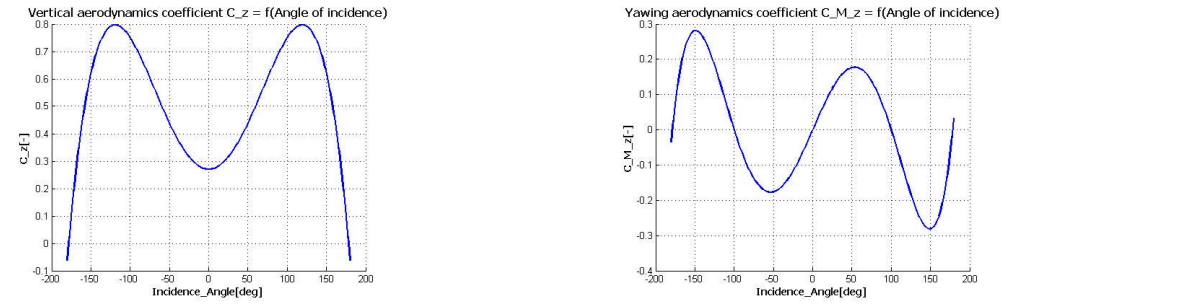
$$Trq_{Aero,Total} = Trq_{Aero} + r_{VA} \times F_{Aero}$$

where:

r_{VA}	is the position vector from the vehicle coordinate system to aerodynamics coordinate system [m]

As can be seen from above, the aerodynamics coefficients are dependent on the angle of incidence. This dependency is defined during wind tunnel measurements. The following illustrations show some characteristic curves:





Inports

The following table shows the inports:

Name	Unit	Description
RotationMatrix_V_E_Trans[3x3]	[]	Rotation matrix from earth coordinate system to vehicle coordinate system
v_Abs_Vehicle_CoG_CoorSys_V[x,y,z]	[m/s]	Absolute velocity of the vehicle CoG in the vehicle coordinate system
v_Wind_CoorSys_E[x,y,z]	[m/s]	Velocity of the wind in the earth coordinate system

Outports

The following table shows the outports:

Name	Unit	Description
F_Aero_CoorSys_V[x,y,z]	[N]	Aerodynamic forces represented in vehicle reference coordinate system
Trq_Aero_CoorSys_V[x,y,z]	[Nm]	Aerodynamic torques represented in vehicle reference coordinate system

Parameters

The following table shows the parameters:

Name	Unit	Description
Map_C_M_x	[]	Rolling aerodynamics coefficient C_M_x [-] = f(Angle of incidence [deg])
Map_C_M_y	[]	Pitching aerodynamics coefficient C_M_y [-] = f(Angle of incidence [deg])
Map_C_M_z	[]	Yawing aerodynamics coefficient C_M_z [-] = f(Angle of incidence [deg])
Map_C_x	[]	Longitudinal aerodynamics coefficient C_x [-] = f(Angle of incidence [deg])
Map_C_y	[]	Lateral aerodynamics coefficient C_y [-] = f(Angle of incidence [deg])
Map_C_z	[]	Vertical aerodynamics coefficient C_z [-] = f(Angle of incidence [deg])
Const_A_x	[m^2]	Longitudinal shadow area of the vehicle
Const_Char_Length	[m]	Characteristic length for aerodynamics torque calculation
Const_Dens_Air	[kg/m^3]	Air density
Const_PosVec_VtoRefAero	[m]	Position vector from vehicle coordinate system to the aerodynamics reference point [x;y;z]

Related topics

References

Aerodynamics (ModelDesk Parameterizing 🕮)

Brake

Where to go from here

Information in this section

Brake Booster 112
On a vacuum brake booster the force applied at the brake pedal regulates the amount of outside air which is applied to a diaphragm, while vacuum remains on the diaphragm's other side.

Brake Disc 117
A brake torque is calculated at each wheel according to the brake pressure and parking braking force at each brake disc.

Drum Brake 119
The drum brake subsystem calculates the braking torque from the acting brake pressure by means of Simulink look-up tables.

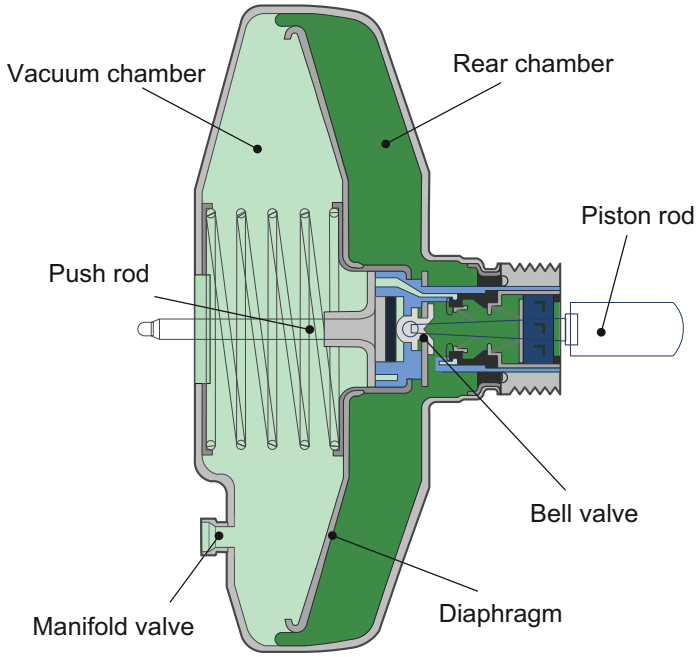
Drum Brake 2-D 120
The **DRUM_BRAKE_2D** subsystem calculates the braking torque from the acting brake pressure and vehicle speed by means of Simulink 2-D look-up tables.

Manifold Pressure 121
The manifold pressure is calculated from the accelerator pedal position.

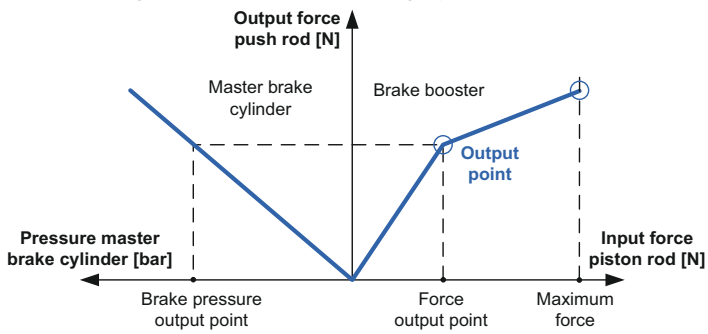
Brake Booster

Description

On a vacuum brake booster the force applied at the brake pedal (piston rod) regulates the amount of outside air which is applied to a diaphragm, while vacuum remains on the diaphragm's other side. The pressure differential at the diaphragm generates force to supplement that applied at the pedal [Bos96].



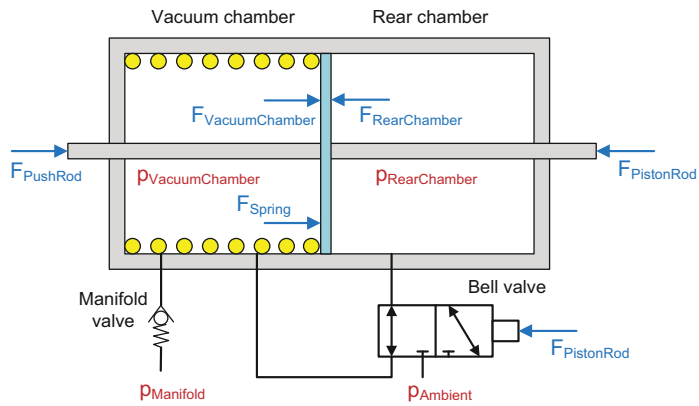
The following schematic illustrates the graph of the brake booster functionality.



The brake pressure is a result of a combination of the force at the pedal and an auxiliary assist. The proportion represented by the assist increases steadily up to full boost (output point); the designed-in boost factor determines the precise rate. At full boost the maximum pressure difference between outside air and vacuum was reached. Additional augmentation of the output force is only possible via an unaccustomed increase in the force applied at the pedal.

The position of the output point and the gradient of the force graph depend on different design parameters of the system like the master brake cylinder surface area, the diaphragm surface area, pedal leverage, vacuum level etc. Normally, the data sheet of the system shows the parameters of the graph for a nominal vacuum chamber pressure.

The following graphic shows the schematics of the model approach.



The model of the system consists out of the vacuum chamber with spring, the rear chamber, the manifold and bell valve and a diaphragm with piston rod and push rod.

The valves are modeled as an isentropic adiabatic flow through an orifice. The bell valve is implemented with a variable cross section and the manifold valve with a fixed cross section and leakage.

The mass flow through the valves can be written with the following equation:

$$\dot{m}_{Valve} = A \cdot p_{in} \cdot \sqrt{\frac{2}{R \cdot T_{in}}} \cdot \Psi\left(\frac{p_{out}}{p_{in}}\right)$$

where:

$\dot{m}_{Valve}$	is the mass flow
p_{in}	is the input pressure
p_{out}	is the output pressure
A	is the orifice area
R	is the ideal gas constant
T_{in}	is the input temperature
Ψ	is the flow function

The flow function Ψ is defined as follows and is not implemented analytically but as a look-up table:

$$\Psi\left(\frac{p_{out}}{p_{in}}\right) = \begin{cases} \sqrt{\frac{\kappa}{\kappa-1}\left[\left(\frac{p_{out}}{p_{in}}\right)^{\frac{2}{\kappa}} - \left(\frac{p_{out}}{p_{in}}\right)^{\frac{\kappa+1}{\kappa}}\right]} & \frac{p_{out}}{p_{in}} \geq \left(\frac{2}{\kappa+1}\right)^{\frac{\kappa}{\kappa-1}} \\ \left(\frac{2}{\kappa+1}\right)^{\frac{1}{\kappa}} \cdot \sqrt{\frac{\kappa}{\kappa+1}} & \frac{p_{out}}{p_{in}} < \left(\frac{2}{\kappa+1}\right)^{\frac{\kappa}{\kappa-1}} \end{cases}$$

κ Polytropic gas index

The pressure of the rear and vacuum chamber is calculated with the ideal gas law:

$$p_{Chamber} = \frac{RT}{V}\int(\dot{m}_{in} - \dot{m}_{out})dt$$

where:

$p_{Chamber}$	is the chamber pressure
$\dot{m}_{in}$	is the mass flow into the chamber
$\dot{m}_{out}$	is the mass flow out of the chamber
T	is the chamber temperature
V	is the chamber volume

The brake booster push rod force is the input force for the master brake cylinder and can be written with the following equation:

$$F_{PushRod} = (p_{RearChamber} - p_{VacuumChamber})A_{Diaphragm} - F_{Spring} + F_{PistonRod}$$

where:

$F_{PushRod}$	is the force push rod
$p_{RearChamber}$	is the pressure rear chamber
$p_{VacuumChamber}$	is the pressure vacuum chamber
$A_{Diaphragm}$	is the area diaphragm
F_{Spring}	is the spring force
$F_{PistonRod}$	is the force piston rod (pedal)

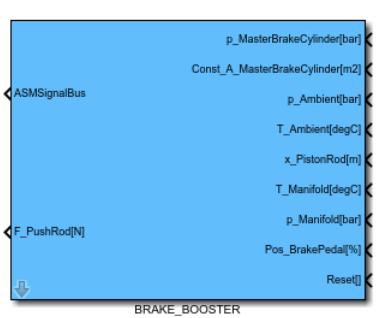
To set the output point of the system, the required diaphragm area is calculated from the master brake cylinder piston area and the nominal vacuum chamber pressure. For the output point the following equation can be written:

$$A_{Diaphragm} = \frac{p_{BrakeCyl,OP}\,A_{BrakeCyl} - F_{PistonRod,OP}}{(p_{Ambient} - p_{VacuumChamber,nom})}$$

where:

$p_{BrakeCyl,OP}$	is the pressure master brake cylinder in output point
$A_{BrakeCyl}$	is the piston area master brake cylinder
$F_{PistonRod,OP}$	is the force piston rod (pedal) in output point
$p_{Ambient}$	is the ambient pressure
$p_{VacuumChamber,nom}$	is the nominal pressure vacuum chamber from output point data sheet

The next illustration shows the BRAKE_BOOSTER block.



Inports

The following table shows the inports.

Name	Unit	Description
Const_A_MasterBrakeCylinder	[m^2]	Piston area master brake cylinder
p_Ambient	[bar]	Ambient pressure
p_Manifold	[bar]	Manifold pressure
p_MasterBrakeCylinder	[bar]	Pressure master brake cylinder
Pos_BrakePedal	[%]	Position brake pedal
Reset	[]	Resets all integrators in the block to their initial conditions
T_Ambient	[°C]	Ambient temperature
T_Manifold	[°C]	Manifold temperature
x_PistonRod	[m]	Displacement piston rod (diaphragm)

Outports

The following table shows the outports.

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
F_PushRod	[N]	Push rod output force

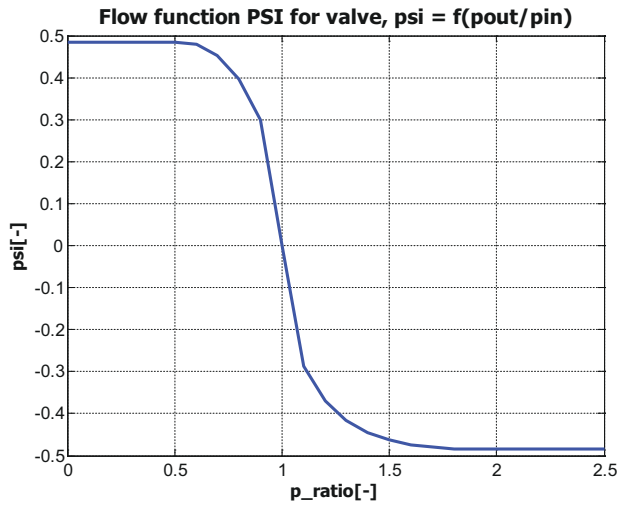
Parameters

The following table shows the parameters.

Name	Unit	Description
Const_A_Boost_BellValve	[m^2]	Flow area bell valve for ambient connection (boost)
Const_A_Connect_BellValve	[m^2]	Flow area bell valve for chamber connection
Const_A_Leakage_MainfoldValve	[m^2]	Leakage flow area manifold valve
Const_A_MainfoldValve	[m^2]	Flow area manifold valve
Const_c_spring	[N/m]	Spring constant
Const_F_Max_PistonRod	[N]	Maximal force piston rod
Const_F_OutputPoint	[N]	Force output point
Const_p_nom_VaccumChamber	[bar]	Nominal pressure vacuum chamber for output point parameterization
Const_p_OutputPoint	[bar]	Brake pressure master brake cylinder output point
Const_V_RearChamber	[m^3]	Volume rear chamber
Const_V_VacuumChamber	[m^3]	Volume vacuum chamber

Processing information

The next illustration shows the flow function curve.



Related topics

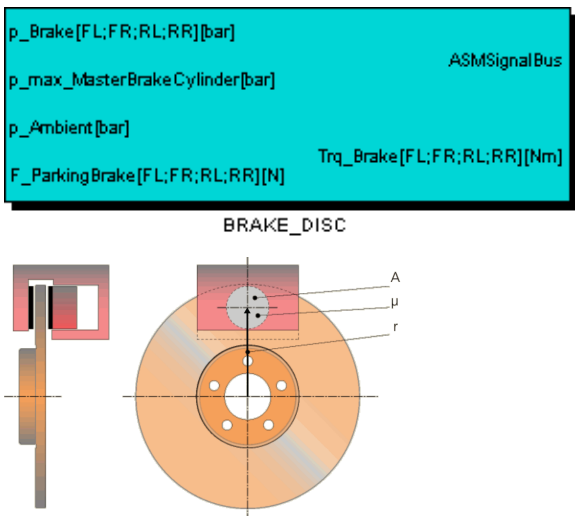
References

Brake Booster (ModelDesk Parameterizing 🕮)

Brake Disc

Description

A brake torque is calculated at each wheel according to the brake pressure and parking braking force at each brake disc.



According to the illustration above, the brake torque is calculated as follows:

$$T_{Br_i} = 2\,\mu_{Br}\,r_{Br}\left(p_{Br}\,A_{Br} + F_{ParkingBrake}\right)$$

Note

The calculated torque in the previous equation is the maximum brake torque, which corresponds to a certain brake pressure. The effect of this brake torque on the wheel movement is included in the WHEEL_SPEED block as a friction torque.

Inports

The following table shows the inports:

Name	Unit	Description
F_ParkingBrake[FL,FR,RL,RR]	[N]	Additional force from parking brake at each brake disc
p_Ambient	[bar]	Ambient pressure
p_Brake[FL,FR,RL,RR]	[bar]	Brake pressure at each brake disc
p_max_MasterBrakeCylinder	[bar]	Maximum brake pressure in the main cylinder

Outports

The following table shows the outports:

Name	Unit	Description
Trq_Brake[FL,FR,RL,RR]	[Nm]	Brake torques on the wheels

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Coef_Fric_FL	[]	Friction coefficient, brake disc, front left
Const_Coef_Fric_FR	[]	Friction coefficient, brake disc, front right
Const_Coef_Fric_RL	[]	Friction coefficient, brake disc, rear left
Const_Coef_Fric_RR	[]	Friction coefficient, brake disc, rear right
FL.Const_A_Fric	[m^2]	Friction area, brake disc, front left
FL.Const_r	[m]	Radius, brake disc, front left
FR.Const_A_Fric	[m^2]	Friction area, brake disc, front right
FR.Const_r	[m]	Radius, brake disc, front right
RL.Const_A_Fric	[m^2]	Friction area, brake disc, rear left
RL.Const_r	[m]	Radius, brake disc, rear left
RR.Const_A_Fric	[m^2]	Friction area, brake disc, rear right
RR.Const_r	[m]	Radius, brake disc, rear right

Related topics

References

Brake Disc (ModelDesk Parameterizing 🕮)
Brake Disc Rear 2nd (ModelDesk Parameterizing 🕮)
Brake Disc Rear 3rd (ModelDesk Parameterizing 🕮)
Trl Brake Disc Front (ModelDesk Parameterizing 🕮)
Trl Brake Disc Rear (ModelDesk Parameterizing 🕮)
Trl Brake Disc Rear 2nd (ModelDesk Parameterizing 🕮)
Trl Brake Disc Rear 3rd (ModelDesk Parameterizing 🕮)

Drum Brake

Description

A brake torque is calculated at each wheel according to the brake pressure at each drum brake.

DRUM_BRAKE_FRONT

The drum brake subsystem calculates the braking torque from the acting brake pressure by means of Simulink look-up tables. You can therefore configure nonlinear braking torque characteristics.

$$Trq_{Brake} = f(p_{Brake}) + Trq_{ParkingBrake}$$

A parking brake torque is superimposed on the braking torque.

Tip

To easily change the brake disc with the drum brake subsystems, copy the preconfigured DrumBrake subsystem from the `ASM_VehicleDynamics_lib/VehicleDynamics/Brake` library and replace the BRAKE_DISC subsystem in your model with the copied one. For accurate inclusion of the parking brake, a gain block has to be added before connecting the Trq_ParkingBrake inport of the DrumBrake subsystem.

Inports

The following tables show the inports:

Name	Unit	Description
p_Ambient	[bar]	Ambient pressure
p_Brake[Left;Right]	[bar]	Brake pressure at left and right drum brake
p_max_MasterBrakeCylinder	[bar]	Maximum brake pressure in the main cylinder

Name	Unit	Description
Trq_ParkingBrake[Left;Right]	[Nm]	Additional torque from parking brake at left and right drum brake

Outports

The following tables show the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_Brake[Left;Right]	[Nm]	Brake torques on left and right wheels

Parameters

The following tables show the parameters:

Name	Unit	Description
Map_BrakeTorqueLeft	[Nm]	Left drum brake torque [Nm] = f(pressure [bar])
Map_BrakeTorqueRight	[Nm]	Right drum brake torque [Nm] = f(pressure [bar])

Related topics

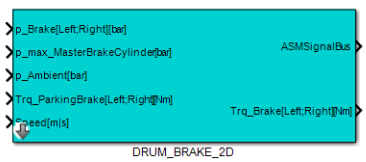
References

Drum Brake Front (ModelDesk Parameterizing 🕮)
Drum Brake Rear (ModelDesk Parameterizing 🕮)
Drum Brake Rear 2nd (ModelDesk Parameterizing 🕮)
Drum Brake Rear 3rd (ModelDesk Parameterizing 🕮)

Drum Brake 2-D

Description

The DRUM_BRAKE_2D subsystem calculates the braking torque from the acting brake pressure and vehicle speed by means of Simulink 2-D look-up tables. In this subsystem, you can define nonlinear braking torque characteristics. Furthermore, a parking brake torque is superimposed on the braking torque.



$$Trq_{brake} = f(p_{brake}, v_{veh}) + Trq_{ParkingBrake}$$

Inports

The following table shows the inports:

Name	Unit	Description
p_Ambient	[bar]	Ambient pressure
p_Brake[Left;Right]	[bar]	Brake pressure at left and right drum brake
p_max_MasterBrakeCylinder	[bar]	Maximum brake pressure in the main cylinder
Speed	[m/s]	Longitudinal vehicle speed
Trq_ParkingBrake[Left;Right]	[Nm]	Additional torque from parking brake at left and right drum brake

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_Brake[Left;Right]	[Nm]	Brake torques on left and right wheels

Parameters

The following table shows the parameters:

Name	Unit	Description
Map_BrakeTorqueLeft	[Nm]	Left drum brake torque [Nm] = f(pressure [bar], v [m/s])
Map_BrakeTorqueRight	[Nm]	Right drum brake torque [Nm] = f(pressure [bar], v [m/s])

Related topics

References

Drum Brake 2D (ModelDesk Parameterizing 🕮)

Manifold Pressure

Description

The manifold pressure depends on the engine operating point and has an influence on the vacuum chamber of the brake booster. The modeled system based on a look-up table that calculates the manifold pressure from the accelerator pedal position.

The following illustration shows the MANIFOLD_PRESSURE block.

Inports

The following table shows the inports.

Name	Unit	Description
Pos_AccPedal	[%]	Accelerator pedal position

Outports

The following table shows the outports.

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
P_Manifold	[bar]	Manifold pressure

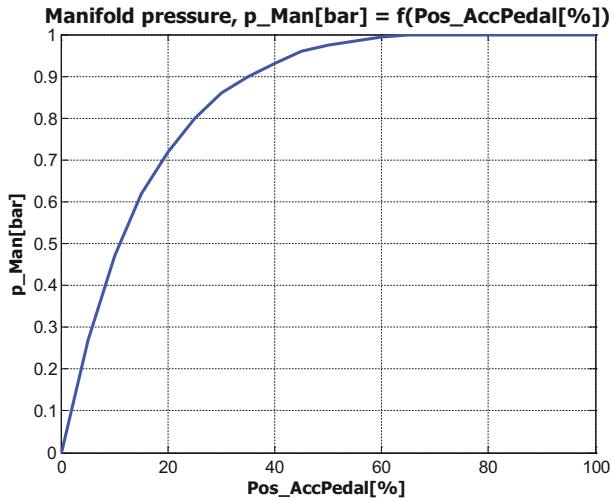
Parameters

The following table shows the parameters.

Name	Unit	Description
Map_p_Man	[bar]	Manifold pressure, p_Man[bar] = f(Pos_AccPedal[%])

Processing information

The following illustration shows an example of a look-up table that describes the manifold pressure.



Related topics

References

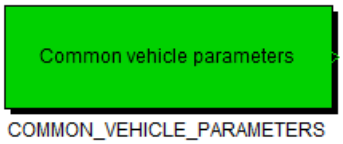
History of the MANIFOLD_PRESSURE Block..... 428

Common Paramter

Common Vehicle Parameters

Description

Parameters that are used several times in the vehicle model are collected in a subsystem called **Common Vehicle Parameters**. This provides central online access to all the parameters. If one of these parameters is modified, this affects all the parts of the model that use it online and offline. The common vehicle parameters are sent to the related sink blocks via GoTo-From blocks.



In the following, a detailed explanation is given on each parameter and on the parts of the vehicle which are taken into account for parameterization.

Basics

To give you a better understanding of the components that have to be taken into account for the parameterization, this section provides a short description of the ASM Vehicle Dynamics Model. For more details, refer to Vehicle Movement on page 275 and ASM Vehicle Dynamics Addendum 🕮.

ASM Vehicle Dynamics uses a multibody system (MBS) technique to write the equation of motion. There is one body for the chassis and four additional bodies, each for one wheel. The ordinary differential equations are set up for the whole system (chassis and wheels) and for the relative movements of the wheels towards the chassis. The wheel-related parameters refer to the wheel coordinate system, which is located in the midpoint of the wheel. Refer to Coordinate Transformation on page 275 and Coordinate Systems (ASM Vehicle Dynamics Addendum 🕮).

The sections below show what settings have to be made for the common vehicle parameters.

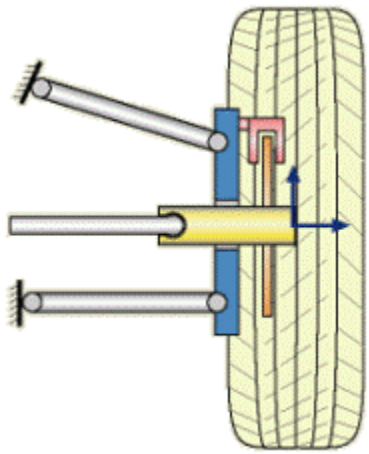
Wheel mass

To parameterize the wheel mass, you must take into account all of the components of one side of the axle that move vertically during jounce and rebound. The following masses have to be summed up:

- Wheel (including tire, rim and knuckle)
- Brake (disc and caliper)
- Half of the suspension control arms
- Axle drive shaft from wheel to differential

- Half of the spring
- Damper and stabilizer elements

The following illustration shows the related components (spring, damper, and stabilizer are not shown):



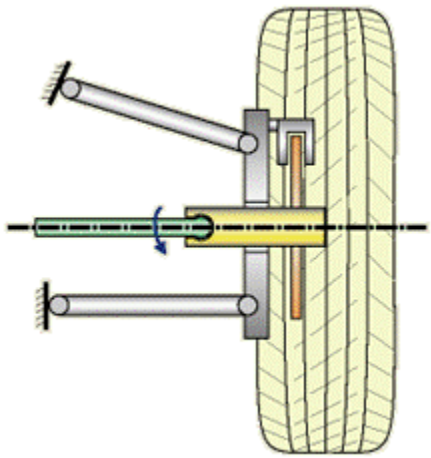
Inertia tensor of wheel

The 3 × 3 inertia tensor of the wheel contains the same components as for the wheel mass above. The inertia has to be given with respect to the wheel center. The wheel center is equal to the wheel coordinate system. The wheel coordinate system is defined by the wheelbase and track width in the configuration state. The figure above shows this (with small blue arrows).

Rotational inertia of wheel

The scalar rotational inertia of the wheel takes only the rotating parts about the wheel rotating axle into account: i.e. the wheel itself (including tire and rim), the brake disc and the axle drive shaft from the wheel to the differential.

The following illustration shows the related components:



For a detailed description of the transient friction coefficient for the braking torque, refer to Wheel Speed on page 293.

Outports

The following table shows the outports:

Name	Unit	Description
Common_Para	[]	Collection of common vehicle parameters that are used several times in the model

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Coef_DynamicBrakeTrq	[Nm/(rad/s)]	Change of braking torque with wheel speed (transient friction coefficient) of the wheels on the left and right side of each axle (see detailed parameter description above
Const_Inertia_Tensor	[kg m^2]	3x3 inertia tensor of the wheels on the left and right side of each axle (see detailed parameter description above)
Const_Inertia_y	[kg m^2]	Scalar inertia about the wheel rotation axis of the wheels on the left and right side of each axle (see detailed parameter description above)
Const_m	[kg]	Mass of the wheels on the left and right side of each axle (see detailed parameter description above)

Blocks affected by the common parameters

The following table gives an overview of the blocks affected by the common parameters:

Parameter	Block
Const_Coef_DynamicBrakeTrq	WHEEL_SPEED
Const_Inertia_Tensor	STEERING and VEHICLE_MOTION_CAR
Const_Inertia_y	MASS_FORCES_AND_TORQUES and WHEEL_SPEED
Const_m	MASS_FORCES_AND_TORQUES, STEERING, and VEHICLE_MOTION_CAR

Related topics

References

Wheel Mass and Inertia (ModelDesk Parameterizing 🕮)

Suspension

Where to go from here

Information in this section

Base.....127
To describe the blocks of the subsystem.

Common.....128
To describe the blocks of the subsystem.

Front.....133
To describe the blocks of the subsystem.

Rear.....202
To describe the blocks of the subsystem.

Base

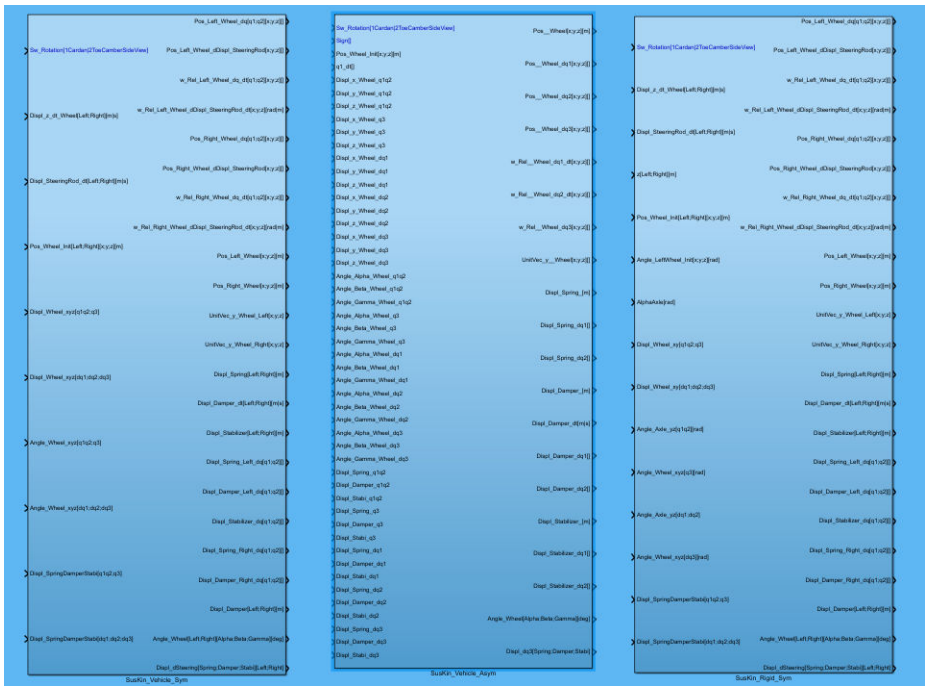
Suspension Models

Introduction

The ASM Vehicle Dynamics Library contains three Suspension models.

Models

The following illustration shows the models in the library:



The following suspension models are contained in the library:

- SusKin_Vehicle_Sym
- SusKin_Vehicle_Asym
- SusKin_Rigid_Sym

Common

Where to go from here

Information in this section

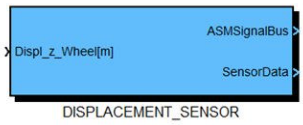
Displacement Sensor.....129
The DISPLACEMENT_SENSOR block can be used to simulate wheel displacement sensors.

Rotation Switch.....130
In ASM you can choose between two different rotation orders.

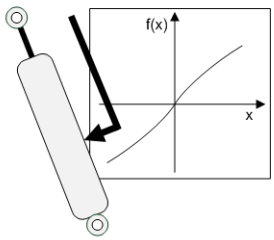
Subframe.....131
The SUBFRAME block describes the elastokinematics as functions of forces and torques.

Displacement Sensor

Description



The DISPLACEMENT_SENSOR block can be used to simulate wheel displacement sensors. The sensor characteristics are defined via a 1-D look-up table, as a function of the vertical wheel displacement. See the following illustration:



The physical unit of the sensor data can be chosen freely.

Inports

The following table shows the inports:

Name	Unit	Description
Displ_z_Wheel	[m]	Vertical displacement of wheel

Outports

The following table shows the outports:

Name	Unit	Descirption
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
SensorData	[]	Sensor output data, as a function of the vertical wheel displacement.

Parameters

The following table shows the parameters:

Name	Unit	Description
Map_SensorScaling	[]	Sensor characteristics as a function of the vertical wheel displacement

Related topics

References

Displacement Sensor (ModelDesk Parameterizing 🕮)

Rotation Switch

Description

Suspension orientation can be described with three rotation angles: alpha (rotation around the x-axis), beta (rotation around the y-axis) and gamma (rotation around the z-axis). The order of the rotation angles plays an important role in parameterizing values such as the suspension kinematics.

In ASM you can choose between two different rotation orders:

1. Gamma – Beta – Alpha
2. Toe – Camber – Sideview

Because a right-handed coordinate system is used in ASM, the left wheel toe angle is a negative rotation about the z-axis, the camber angle is a negative rotation about the x-axis, and the sideview angle is a negative rotation about the y-axis.

The switch signal will affect the parameterization of the suspension kinematics as well as suspension compliance. For example, if this switch is set to `Toe-Camber-Sideview`, you should parameterize the kinematics LUT and the compliance LUT of the left wheel regarding the definition of the `Toe-Camber-Sideview` of the left wheel.

For more information about the toe, camber and sideview angles, refer to Orientation Angles (ASM Vehicle Dynamics Addendum 🕮).

Outports

The following table shows the outports:

Name	Unit	Description
SW_Rotation	[]	Switch between two different rotation orders: 1. Gamma – Beta – Alpha 2. Toe – Camber – Sideview

Parameters

The following table shows the parameters:

Name	Unit	Description
Sw_Rotation	[]	Switch between two different rotation orders: 1. Gamma – Beta – Alpha 2. Toe – Camber – Sideview

Related topics

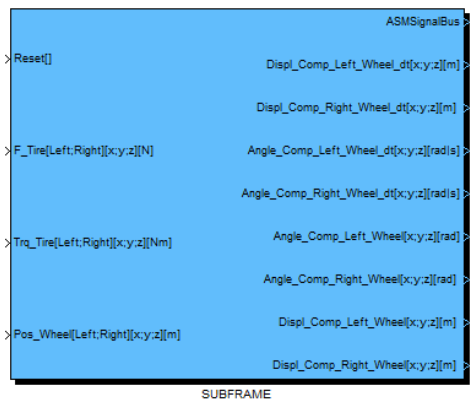
References

SusCommon (ModelDesk Parameterizing 🕮)

Subframe

Description

If the vehicle is equipped with a subframe, additional displacements and rotations are caused by the forces and torques acting on the subframe.



The SUBFRAME block describes these elastokinematics as functions of forces and torques:

$$\Delta x = f(F_x)$$

$$\Delta y = f(F_y)$$

$$\Delta \gamma = f(F_y)$$

$$\Delta \gamma = f(Trq_z)$$

The dynamics of these displacements are approximated by second-order dynamics with a natural frequency ω_n which is defined as a parameter.

For example, the dynamic x displacement can be calculated as follows:

$$\Delta x_{dyn} = \frac{{\omega_n}^2}{s^2 + 2\,\zeta\,\omega_n\,s + {\omega_n}^2}\Delta x$$

where
s is the Laplace operator
ζ is the damping ratio ($\zeta = 0{,}707$, set internally)

Note

The sample time of the model must be considered when defining ω_n, such that $(\omega_n/2\pi) \ll (1/ModelSampleTime)$.

As an option, the SUBFRAME block can be used in the model, in parallel to the SUSPENSION_COMPLIANCE block. The total wheel positions and orientations are the sum of the positions and orientations due to the kinematics and elastokinematics of the suspension and the elastokinematics of the subframe. For example, the total longitudinal position of the wheel is calculated as follows:

$$x_{Total} = x_0 + \Delta x_{Kinematics} + \Delta x_{Compliance} + \Delta x_{Subframe}$$

Inports

The following table shows the inports:

Name	Unit	Description
F_Tire[Left;Right][x;y;z]	[N]	Tire forces at wheel center of left and right wheel
Pos_Wheel[Left;Right][x;y;z]	[m]	Position of wheel center relative to vehicle coordinate system of left and right wheel
Reset	[]	Reset the integrators to their initial values
Trq_Tire[Left;Right][x;y;z]	[Nm]	Tire torques at wheel center of left and right wheel

Outports

The following table shows the outports:

Name	Unit	Description
Angle_Comp_Left_Wheel[x;y;z]	[rad]	Wheel orientation change due to subframe compliance of left wheel
Angle_Comp_Left_Wheel_dt[x;y;z]	[rad/s]	Relative wheel angular velocity due to subframe compliance of left wheel
Angle_Comp_Right_Wheel[x;y;z]	[rad]	Wheel orientation change due to subframe compliance of right wheel
Angle_Comp_Right_Wheel_dt[x;y;z]	[rad/s]	Relative wheel angular velocity due to subframe compliance of right wheel
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Displ_Comp_Left_Wheel[x;y;z]	[m]	Wheel position change due to subframe compliance of left wheel
Displ_Comp_Left_Wheel_dt[x;y;z]	[m]	Relative wheel translational velocity due to subframe compliance of left wheel
Displ_Comp_Right_Wheel[x;y;z]	[m]	Wheel position change due to subframe compliance of right wheel

Name	Unit	Description
Displ_Comp_Right_Wheel_dt[x;y;z]	[m]	Relative wheel translational velocity due to subframe compliance of right wheel

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_NaturalFrequency_Angle_Gamma	[1/s]	Natural frequency for the dynamics on the gamma angle
Const_NaturalFrequency_Displ_x	[1/s]	Natural frequency for the dynamics on the displacement in x direction
Const_NaturalFrequency_Displ_y	[1/s]	Natural frequency for the dynamics on the displacement in y direction
Const_Pos_Subframe	[m]	Position vector [x;y;z] from vehicle coordinate system to midpoint of subframe
Map_Angle_Gamma_Subframe_F_y	[deg]	Additional gamma angle depending on subframe lateral force: gamma[deg] = f(F[N])
Map_Angle_Gamma_Subframe_Trq_z	[deg]	Additional gamma angle depending on subframe vertical torque: gamma[deg] = f(Trq[Nm])
Map_Displ_x_Subframe_F_x	[mm]	Additional displacement in longitudinal direction depending on subframe longitudinal force: x[mm] = f(F[N])
Map_Displ_y_Subframe_F_y	[mm]	Additional displacement in lateral direction depending on subframe lateral force: y[mm] = f(F[N])

Related topics

References

Subframe (ModelDesk Parameterizing 🕮)

Front

Where to go from here

Information in this section

Common Suspension Blocks..... 134
To describe the blocks of the subsystem.

Suspension Kinematics..... 162
To describe the blocks of the subsystem.

Common Suspension Blocks

Where to go from here

Information in this section

Front Wheels Relative Velocities..... 135
This subsystem calculates the wheel relative translatory and angular velocities.

Relative Position Front..... 137
In this subsystem, the wheel relative position and orientation are calculated.

Front Suspension Forces..... 138
This block evaluates the forces acting between the wheel and vehicle body, which are generated by the spring, damper, and stabilizer.

Front Suspension Forces Active..... 141
This block evaluates the forces acting between the wheel and vehicle body, which are generated by the spring, damper, and stabilizer.

Front Suspension Forces Ancillary Spring..... 143
This block evaluates the forces acting between the wheel and vehicle body, which are generated by the ancillary spring.

Front Suspension Compliance..... 145
This block describes the wheel suspension elastokinematics for the front left and right wheels.

Suspension Compliance Rigid Axle (Symmetric)..... 148
Due to suspension elasticity, additional elastic displacements and rotation are caused by the forces and torques acting on the wheel. Suspension compliance kinematics describes these additional displacements and rotation as functions of the forces and torques.

Suspension Force Kinematics Front..... 152
This block projects the spring, damper and stabilizer forces in the direction of the axle degrees of freedom.

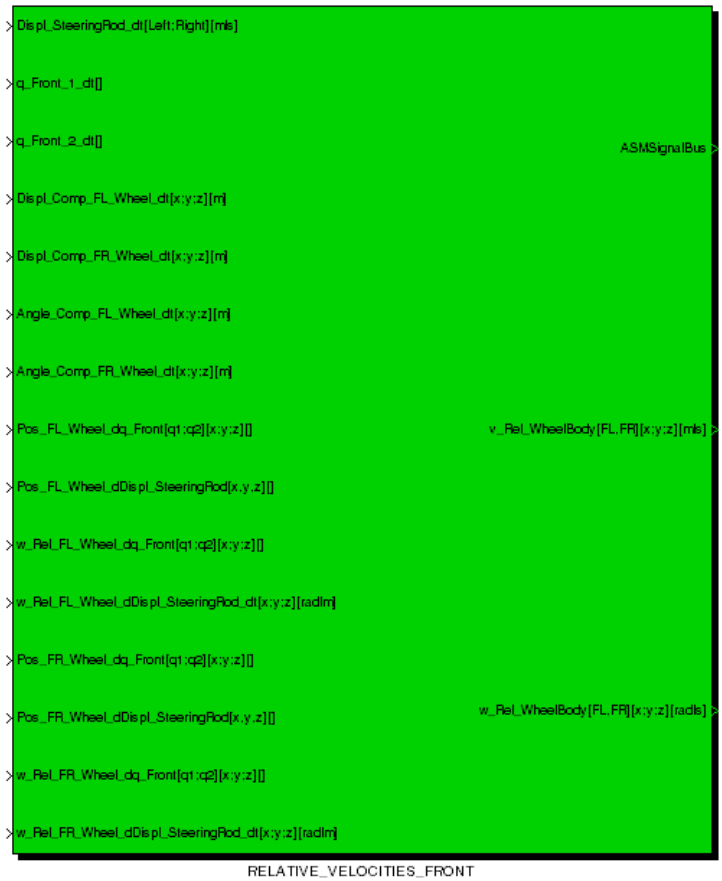
Suspension Compliance 2-D (Symmetric)..... 155
In this subsystem, the compliance effect of the left wheel can be defined as a nonlinear function of the left force/torque and the right force/torque as 2-D look-up table.

Stabilizer Compliance (Symmetric)..... 160
During vehicle roll movement, the difference between the left and right stabilizer endpoints will not be equal to zero. This causes a rotation of the stabilizer and the stabilizer force acts at the end-points.

Front Wheels Relative Velocities

Description

Due to suspension kinematics and elasticity, the wheels move in and rotate about the three directions (x, y, and z).



This subsystem calculates the wheel relative translatory and angular velocities. This is the calculation for the front left wheel:

$$v_{FL} = \frac{\partial v_{FL}}{\partial \dot{q}_{1_Front}} \dot{q}_{1_Front} + \frac{\partial v_{FL}}{\partial \dot{q}_{2_Front}} \dot{q}_{2_Front} + \frac{\partial v_{FL}}{\partial \dot{q}_{St}} \dot{q}_{St} + v_{FL_Compl}$$

$$\omega_{FL} = \frac{\partial \omega_{FL}}{\partial \dot{q}_{1_Front}} \dot{q}_{1_Front} + \frac{\partial \omega_{FL}}{\partial \dot{q}_{2_Front}} \dot{q}_{2_Front} + \frac{\partial \omega_{FL}}{\partial \dot{q}_{St}} \dot{q}_{St} + \omega_{FL_Compl}$$

The partial derivatives terms are calculated in the suspension kinematics subsystem. The compliance motion are calculated in the suspension compliance subsystem.

Inports

The following table shows the inports:

Name	Unit	Description
Angle_Comp_FL_Wheel_dt[x;y;z]	[rad/s]	Additional wheel angular velocity due to suspension compliance front left
Angle_Comp_FR_Wheel_dt[x;y;z]	[rad/s]	Additional wheel angular velocity due to suspension compliance front right
Displ_Comp_FL_Wheel_dt[x;y;z]	[m]	Additional wheel velocity due to suspension compliance front left
Displ_Comp_FR_Wheel_dt[x;y;z]	[m]	Additional wheel velocity due to suspension compliance front right
Displ_SteeringRod_dt[Left;Right]	[m/s]	Speed of steering rod displacement of left and right side
Pos_FL_Wheel_dDispl_SteeringRod[x,y,z]	[]	Change of wheel position front left with respect to displacement of steering rod
Pos_FL_Wheel_dq_Front[q1;q2][x;y;z]	[]	Change of wheel position front left with respect to changes of first and second generalized coordinates
Pos_FR_Wheel_dDispl_SteeringRod[x,y,z]	[]	Change of wheel position front right with respect to displacement of steering rod
Pos_FR_Wheel_dq_Front[q1;q2][x;y;z]	[]	Change of wheel position front right with respect to changes of first and second generalized coordinates
q_Front_1_dt	[]	Derivative of first generalized coordinate with respect to time
q_Front_2_dt	[]	Derivative of second generalized coordinate with respect to time
w_Rel_FL_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Change of wheel orientation front left with respect to displacement of steering rod
w_Rel_FL_Wheel_dq_Front[q1;q2][x;y;z]	[]	Change of wheel orientation front left with respect to changes of first and second generalized coordinates
w_Rel_FR_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Change of wheel orientation front right with respect to displacement of steering rod
w_Rel_FR_Wheel_dq_Front[q1;q2][x;y;z]	[]	Change of wheel orientation front right with respect to changes of first and second generalized coordinates

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
v_Rel_WheelBody[FL,FR][x;y;z]	[m/s]	Velocity of wheel body relative to vehicle body, front
w_Rel_WheelBody[FL,FR][x;y;z]	[rad/s]	Angular velocity of wheel body relative to vehicle body excluding wheel rotational speed, front

Parameters

With the above inputs and outputs, this subsystem requires no parameters to perform its calculation.

Related topics

References

History of the RELATIVE_VELOCITIES_x Block..... 431

Relative Position Front

Description

Due to suspension kinematics and elasticity, the wheels move in and rotate about the three directions (x, y and z). In this subsystem, the wheel relative position and orientation are calculated.

Inports

The following table shows the inports:

Name	Unit	Description
Angle_Comp_FL_Wheel[x;y;z]	[rad]	Additional wheel rotation angle due to suspension compliance front left
Angle_Comp_FR_Wheel[x;y;z]	[rad]	Additional wheel rotation angle due to suspension compliance front right
Displ_Comp_FL_Wheel[x;y;z]	[m]	Additional wheel displacement due to suspension compliance front left
Displ_Comp_FR_Wheel[x;y;z]	[m]	Additional wheel displacement due to suspension compliance front right
Pos_FL_Wheel[x;y;z]	[m]	Position vector of front left wheel
Pos_FR_Wheel[x;y;z]	[m]	Position vector of front right wheel

Name	Unit	Description
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (Gamma-Beta-Alpha) ▪ 2: ToeCamberSideview (Toe-Camber-Sideview)
UnitVec_y_FL_Wheel_CoorSys_V[x;y;z]	[m]	Unit vector in y direction of wheel coordinate system of front left wheel
UnitVec_y_FR_Wheel_CoorSys_V[x;y;z]	[m]	Unit vector in y direction of wheel coordinate system of front right wheel

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Collection of ASM signals
Pos_Wheel[FL;FR][x;y;z]	[m]	Position vector of front wheels including compliance effects
UnitVec_y_Wheel_CoorSys_V[FL;FR] [x;y;z]	[m]	Unit vector in y direction of wheel coordinate system of front wheels including compliance effects

Related topics

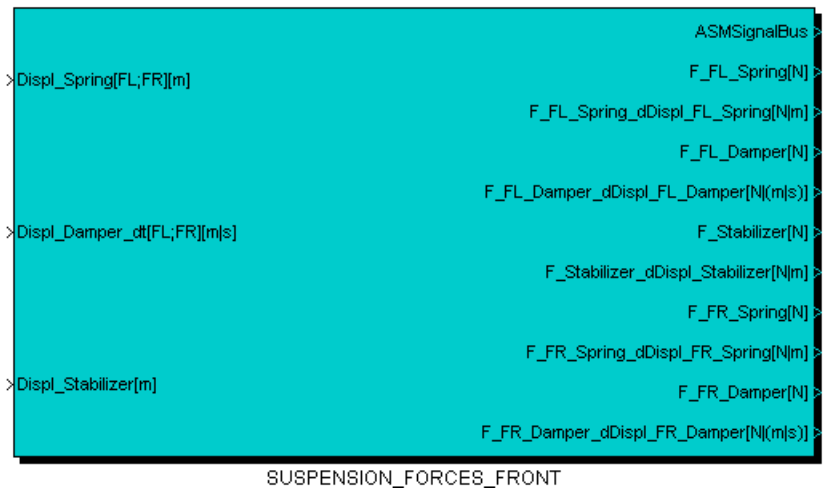
References

History of the RELATIVE_POSITION_x Block 431

Front Suspension Forces

Description

This block evaluates the forces acting between the wheel and vehicle body, which are generated by the spring, damper, and stabilizer.



These forces can be calculated by means of the spring, damper, and stabilizer characteristics and displacements, which were calculated in the kinematics subsystem.

$$F_{sp} = F_{sp}(z_{sp}),\ \ F_d = F_d(\dot{z}_d),\ \ F_{stab} = F_{stab}(z_{stab})$$

Note

The generalized suspension force in the direction of the wheel degrees of freedom is calculated in another block.

Inports

The following table shows the inports:

Name	Unit	Description
Displ_Damper_dt[FL;FR]	[m/s]	Vertical speed of displacement of the damper joint point, front left and front right
Displ_Spring[FL;FR]	[m]	Vertical displacement of the spring joint point, front left and front right
Displ_Stabilizer	[m]	Vertical displacement of the stabilizer left joint point relative to the right joint point

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Collection of ASM signals
F_FL_Damper	[N]	Force of front left damper
F_FL_Damper_dDispl_FL_Damper	[N/(m/s)]	Change of damper force front left with respect to vertical speed of displacement of damper joint point front left
F_FL_Spring	[N]	Force of front left spring
F_FL_Spring_dDispl_FL_Spring	[N/m]	Change of spring force front left with respect to vertical displacement of spring joint point front left
F_FR_Damper	[N]	Force of front right damper
F_FR_Damper_dDispl_FR_Damper	[N/(m/s)]	Change of damper force front right with respect to vertical speed of displacement of damper joint point front right
F_FR_Spring	[N]	Force of front right spring
F_FR_Spring_dDispl_FR_Spring	[N/m]	Change of spring force front right with respect to vertical displacement of spring joint point front right
F_Stabilizer	[N]	Force of stabilizer
F_Stabilizer_dDispl_Stabilizer	[N/m]	Change of stabilizer force with respect to vertical displacement of stabilizer joint point

Parameters

The following table shows the parameters:

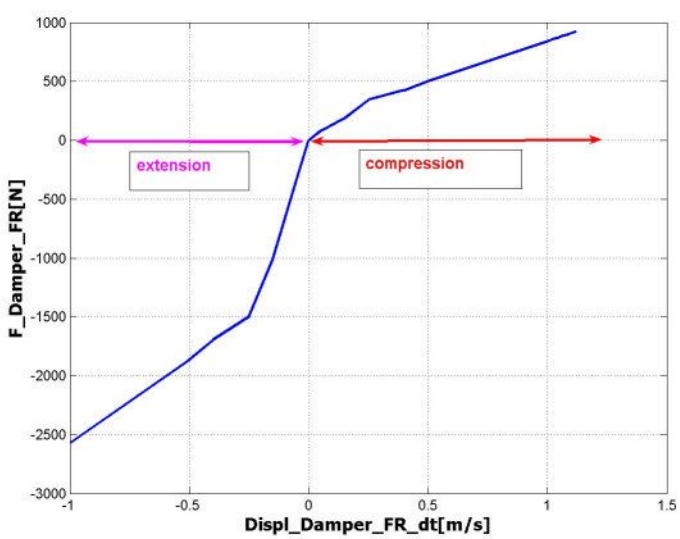
Name	Unit	Description
Map_Force_LeftDamper	[N]	Left damper force = f(damper speed[m/s])
Map_Force_LeftSpring	[N]	Left spring force = f(Spring displacement[m])
Map_Forc_RightDamper	[N]	Right damper force = f(damper speed[m/s])
Map_Force_RightSpring	[N]	Right spring force = f(spring displacement[m])
Map_Force_Stabilizer	[N]	Stabilizer force = f(stabilizer relative displacement of left joint point to right joint point[m])

Processing information

Wheel jounce (+z) is considered to be positive, wheel rebound (-z) is negative.

Note

Damper force is considered positive during compression (wheel jounce speed (+z)) and damper force is considered negative during extension. The following illustration shows a damper characteristic curve.



Note

The force obtained from the curve is in the damper, and the working point of this force is not the wheel center.

Related topics

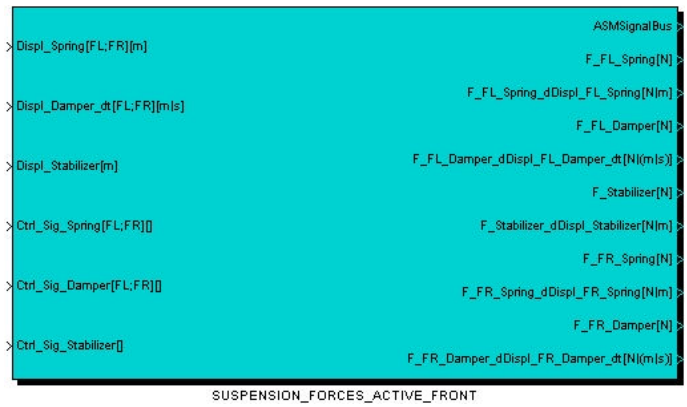
References

SusForces Front (ModelDesk Parameterizing 🕮)
SusForces Rear (ModelDesk Parameterizing 🕮)
SusForces Rear 2nd (ModelDesk Parameterizing 🕮)
SusForces Rear 3rd (ModelDesk Parameterizing 🕮)

Front Suspension Forces Active

Description

This block evaluates the forces acting between the wheel and vehicle body, which are generated by the spring, damper, and stabilizer.



These forces can be calculated by means of the spring, damper, and stabilizer characteristics and displacements, which were calculated in the kinematics subsystem. You can additionally define different curve characteristics as function of control signals.

$$F_{sp} = F_{sp}(z_{sp}, Ctrl_{sp}),\ \ F_d = F_{stab}(\dot{z}_d, Ctrl_d),\ \ F_{stab} = F_{stab}(z_{stab}, Ctrl_{stab})$$

By defining different damper curve characteristics, for example, an active damper functionality can be simulated. The control signal might be a control current from an electronic control unit (ECU), for example, from an active damper ECU.

Inports

The following table show the inports:

Name	Unit	Description
Ctrl_Sig_Damper[FL;FR]	[]	Control signal for changing damper characteristics, front left and front right
Ctrl_Sig_Spring[FL;FR]	[]	Control signal for changing spring characteristics, front left and front right
Ctrl_Sig_Stabilizer	[]	Control signal for changing stabilizer characteristics
Displ_Damper_dt[FL;FR]	[m/s]	Vertical speed of displacement of the damper joint point, front left and front right
Displ_Spring[FL;FR]	[m]	Vertical displacement of the spring joint point, rear left and rear right

Name	Unit	Description
Displ_Stabilizer	[m]	Vertical displacement of the stabilizer left joint point relative to the right joint point

Outports

The following table show the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
F_FL_Damper	[N]	Force of front left damper
F_FL_Damper_dDispl_FL_Damper_dt	[N/(m/s)]	Change of damper force front left with respect to vertical speed of displacement of damper joint point front left
F_FL_Spring	[N]	Force of front left spring
F_FL_Spring_dDispl_FL_Spring	[N/m]	Change of spring force front left with respect to vertical displacement of spring joint point front left
F_FR_Damper	[N]	Force of front right damper
F_FR_Damper_dDispl_FR_Damper_dt	[N/(m/s)]	Change of damper force front right with respect to vertical speed of displacement of damper joint point front right
F_FR_Spring	[N]	Force of front right spring
F_FR_Spring_dDispl_FR_Spring	[N/m]	Change of spring force front right with respect to vertical displacement of spring joint point front right
F_Stabilizer	[N]	Force of stabilizer
F_Stabilizer_dDispl_Stabilizer	[N/m]	Change of stabilizer force with respect to vertical displacement of stabilizer joint point

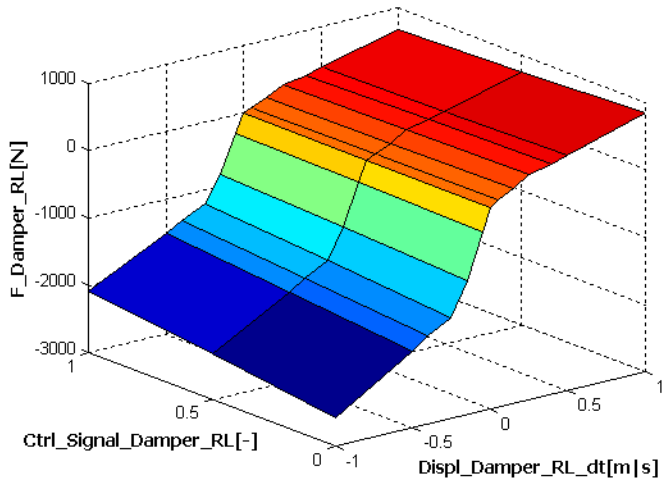
Parameters

The following table show the parameters:

Name	Unit	Description
Map_Force_Left_Damper	[N]	Left damper force = f(damper speed[m/s], Control signal damper)
Map_Force_Left_Spring	[N]	Left spring force = f(spring displacement[m], Control signal spring)
Map_Force_Right_Damper	[N]	Right damper force = f(damper speed[m/s], Control signal damper)
Map_Force_Right_Spring	[N]	Right spring force = f(spring displacement[m], Control signal spring)
Map_Force_Stabilizer	[N]	Stabilizer force = f(stabilizer relative displacement of left joint point to right joint point[m], Control signal stabilizer)

Processing information

The following table shows different damper characteristic curves defined in one Simulink 2-D look-up table:



Related topics

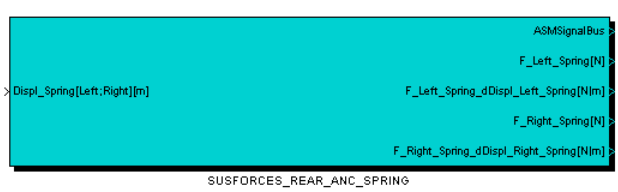
References

SusForcesAct Front (ModelDesk Parameterizing 🕮)
SusForcesAct Rear (ModelDesk Parameterizing 🕮)
SusForcesAct Rear 2nd (ModelDesk Parameterizing 🕮)
SusForcesAct Rear 3rd (ModelDesk Parameterizing 🕮)

Front Suspension Forces Ancillary Spring

Description

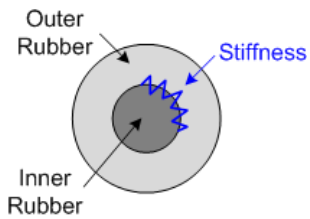
This block evaluates the forces acting between the wheel and vehicle body, which are generated by the ancillary spring.



These forces can be calculated by means of the ancillary spring characteristics and displacements, which were calculated in the related kinematics subsystem:

$$F_{sp} = F_{sp}(z_{sp})$$

The ancillary spring accounts for additional stiffnesses of real suspension joints, which are not made of ideal revolute joints but contain elastic parts.



The ancillary spring is used in parallel to the main spring. Thus, the total spring stiffness of an axle is increased when using ancillary spring in addition. The stiffness of the ancillary spring can be calculated from the total stiffness of one axle (for example, from KnC test bench) and the main spring characteristics. If the ancillary spring characteristic is once known, you can make variation studies of the main spring by simply changing its spring parameterization.

Inports

The following table shows the inports.

Name	Unit	Description
Displ_Spring[Left;Right]	[m]	Relative displacement of the ancillary spring joint points on left and right sides

Outports

The following table shows the outports.

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
F_Left_Spring	[N]	Force of left ancillary spring
F_Left_Spring_dDispl_Left_Spring	[N/m]	Change of ancillary spring force left with respect to relative displacement of ancillary spring left
F_Right_Spring	[N]	Force of right ancillary spring
F_Right_Spring_dDispl_Right_Spring	[N/m]	Change of ancillary spring force right with respect to relative displacement of ancillary spring right

Parameters

The following table shows the parameters.

Name	Unit	Description
Map_Force_Left_Spring	[N]	Ancillary spring force left = f(Ancillary spring displacement left [m])
Map_Force_Right_Spring	[N]	Ancillary spring force right = f(Ancillary spring displacement right [m])

Related topics

References

SusForces Front (ModelDesk Parameterizing 🕮)
SusForces Rear (ModelDesk Parameterizing 🕮)
SusForces Rear 2nd (ModelDesk Parameterizing 🕮)
SusForces Rear 3rd (ModelDesk Parameterizing 🕮)

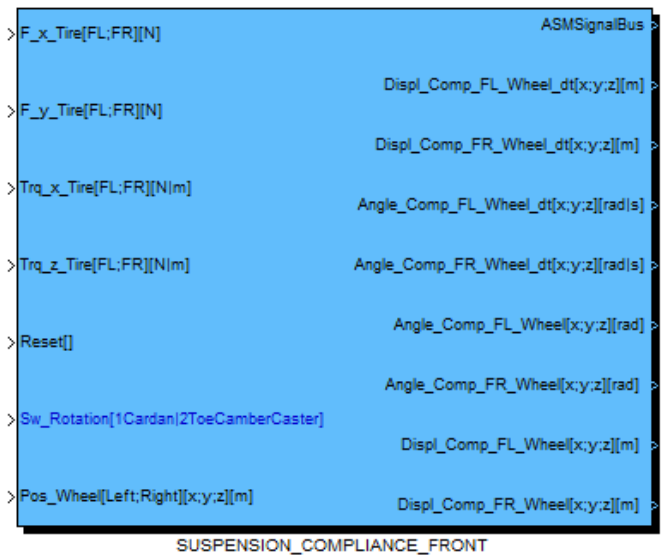
Front Suspension Compliance

Description

This block describes the wheel suspension elastokinematics for the front left and right wheels.

Due to suspension elasticity, additional elastic displacements are caused by the forces and torques acting on the wheel. Suspension compliance kinematics describes these additional displacements as functions of the forces and torques.

There are two suspension compliance blocks available. The SUSPENSION_COMPLIANCE subsystem takes the forces and torques acting on the current wheel center into account, whereas the SUSPENSION_COMPLIANCE_OPP subsystem takes the forces and torques acting on the opposite wheel center into account.



The additional displacements are considered in x and y directions and additional alpha (α) and gamma (γ) angles. These variables are defined as functions of the longitudinal force, lateral force, and self-aligning torque, i.e.:

$$\Delta x = \sum_i \Delta x_i(Q_i)$$

$$\Delta y = \sum_i \Delta y_i(Q_i)$$

$$\Delta\alpha = \sum_i \Delta\alpha_i(Q_i)'$$

$$\Delta\gamma = \sum_i \Delta\gamma_i(Q_i)$$

where

$i = 1, \ldots, 6$

$Q_i = F_x, F_y, T_x, T_y, F_{x_opp}, F_{y_opp}, T_{x_opp}, T_{y_opp}$

The total wheel position and orientation are the sum of the positions and orientations due to kinematics and elasto-kinematics.

The dynamics of these displacements are approximated by second-order dynamics with a natural frequency ω_n which is defined as a parameter. For example, the dynamic x displacement can be calculated as follows:

$$\Delta x_{dyn} = \frac{{\omega_n}^2}{s^2 + 2\,\zeta\,\omega_n\,s + {\omega_n}^2}\Delta x$$

where s is the Laplace operator and ζ is the damping ratio ($\zeta = 0{,}707$, set internally)

Note

The sample time of the model must be considered when defining ω_n, such that$(\omega_n/2\pi) \ll (1/ModelSampleTime)$.

Inports

The following table shows the inports:

Name	Unit	Description
F_x_Tire[FL;FR]	[N]	Longitudinal tire force, front left and front right
F_y_Tire[FL;FR]	[N]	Lateral tire force, front left and front right
Reset	[]	Resets all integrators in the block to their initial conditions
Trq_x_Tire[FL;FR]	[Nm]	Tire torque about the longitudinal axis, front left and front right
Trq_z_Tire[FL;FR]	[Nm]	Tire torque about the vertical axis, front left and front right

Outports

The following table shows the outports:

Name	Unit	Description
Angle_Comp_FL_Wheel[x,y,z]	[rad]	Wheel orientation change due to suspension compliance, front left
Angle_Comp_FL_Wheel_dt[x,y,z]	[rad/s]	Relative wheel angular velocity due to suspension compliance, front left
Angle_Comp_FR_Wheel[x,y,z]	[rad]	Wheel orientation change due to suspension compliance, front right
Angle_Comp_FR_Wheel_dt[x,y,z]	[rad/s]	Relative wheel angular velocity due to suspension compliance, front right
Displ_Comp_FL_Wheel[x,y,z]	[m]	Wheel position change due to suspension compliance, front left

Name	Unit	Description
Displ_Comp_FL_Wheel_dt[x,y,z]	[m/s]	Relative wheel translational velocity due to suspension compliance, front left
Displ_Comp_FR_Wheel[x,y,z]	[m]	Wheel position change due to suspension compliance, front right
Displ_Comp_FR_Wheel_dt[x,y,z]	[m/s]	Relative wheel translational velocity due to suspension compliance, front right

Parameters

The following table shows the parameters for the SUSPENSION_COMPLIANCE subsystem. The parameters for the SUSPENSION_COMPLIANCE_OPP subsystem are structurally identical, but refer to the opposite wheel each. Both subsystems must be parameterized for the left wheel only. The right wheel compliance effects are symmetrically mirrored from the left ones below the mask.

Example

- The `Map_Displ_y_Wheel_F_y` parameter describes the lateral displacement of the left wheel center as a function of the lateral force (F_y) acting on the left wheel center. This parameter is needed for independent and dependent suspension types.
- The `Map_Displ_y_Wheel_F_y_opp` parameter describes the lateral displacement of the left wheel center as a function of the lateral force (F_y_opp) acting on the right wheel center. This parameter is needed for dependent suspension types only.

Name	Unit	Description
Const_NaturalFrequency_Angle_Alpha	[1/s]	Natural frequency for the dynamics of the α angle
Const_NaturalFrequency_Angle_Gamma	[1/s]	Natural frequency for the dynamics of the γ angle
Const_NaturalFrequency_Displ_x	[1/s]	Natural frequency for the dynamics of the displacement in x direction
Const_NaturalFrequency_Displ_y	[1/s]	Natural frequency for the dynamics of the displacement in y direction
Map_Angle_Alpha_Wheel_F_x	[deg]	α angle as a function of tire longitudinal force
Map_Angle_Alpha_Wheel_F_y	[deg]	α angle as a function of tire lateral force
Map_Angle_Alpha_Wheel_Trq_x	[deg]	α angle as a function of tire torque about longitudinal axis
Map_Angle_Alpha_Wheel_Trq_z	[deg]	α angle as a function of tire torque about the vertical axis
Map_Angle_Gamma_Wheel_F_x	[deg]	γ angle as a function of tire longitudinal force
Map_Angle_Gamma_Wheel_F_y	[deg]	γ angle as a function of tire lateral force
Map_Angle_Gamma_Wheel_Trq_x	[deg]	γ angle as a function of tire torque about longitudinal axis
Map_Angle_Gamma_Wheel_Trq_z	[deg]	γ angle as a function of tire torque about the vertical axis
Map_Displ_x_Wheel_F_x	[mm]	x displacement of wheel center as a function of tire longitudinal force
Map_Displ_x_Wheel_F_y	[mm]	x displacement of wheel center as a function of tire lateral force

Name	Unit	Description
Map_Displ_x_Wheel_Trq_x	[mm]	x displacement of wheel center as a function of tire torque about the longitudinal axis
Map_Displ_x_Wheel_Trq_z	[mm]	x displacement of wheel center as a function of tire torque about the vertical axis
Map_Displ_y_Wheel_F_x	[mm]	y displacement of wheel center as a function of tire longitudinal force
Map_Displ_y_Wheel_F_y	[mm]	y displacement of wheel center as a function of tire lateral force
Map_Displ_y_Wheel_Trq_x	[mm]	y displacement of wheel center as a function of tire torque about the longitudinal axis
Map_Displ_y_Wheel_Trq_z	[mm]	y displacement of wheel center as a function of tire torque about the vertical axis

Related topics

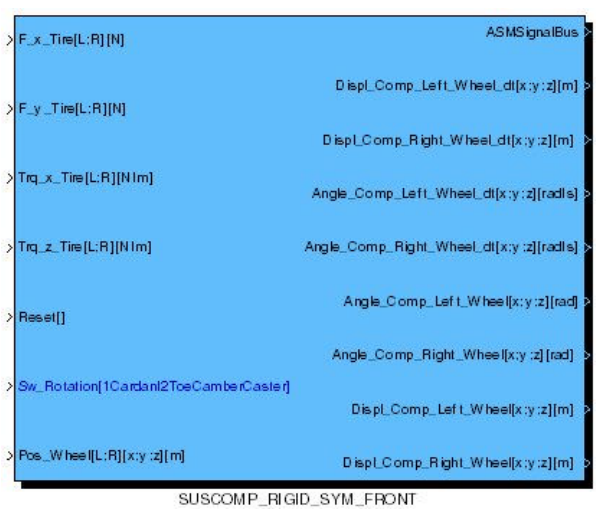
References

SusComp Front Front (ModelDesk Parameterizing 🕮)
SusCompOpp Front (ModelDesk Parameterizing 🕮)

Suspension Compliance Rigid Axle (Symmetric)

Description

Due to suspension elasticity, additional elastic displacements and rotation are caused by the forces and torques acting on the wheel. Suspension compliance kinematics describes these additional displacements and rotation as functions of the forces and torques.



For a rigid axle, two compliance effects are taken into account:

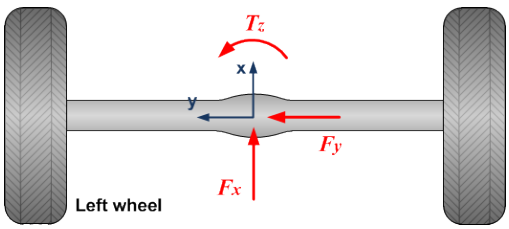
- Axle center movement due to forces and torques and
- Wheel rotation due to forces and torques

Axle Center Compliance

For the axle center, the following compliance effects can be parameterized:

- Lateral displacement of the axle center due to the sum of the left and right tire lateral force. To measure this effect, either two equal parallel lateral forces can be applied at the wheel center of the suspension or a single lateral force can be applied at the axle center. Axle center displacement almost always occurs in the same direction as the force.
- Longitudinal displacement of the axle center due to the sum of the left and right tire longitudinal force. This effect can be measured by applying two equal parallel longitudinal forces at the wheel center of the suspension or by applying a single longitudinal force at the axle center.
- Rotation around the z-axis of the axle center due to torque around the z-axis. This compliance effect can be measured by applying torque about the z-axis at the axle.

See the following illustration:

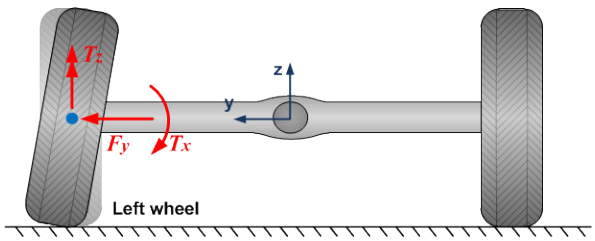


Left wheel compliance

For wheel compliance, the following compliance effects are considered:

- Alpha rotation wheel (rotation around the x-axis) of the wheel due to lateral, longitudinal forces and torque at the wheel center can be parameterized. To measure this effect, the lateral/longitudinal forces or torque about the z-axis are applied to the left wheel center.
- Gamma rotation wheel (rotation around the z-axis) of the wheel due to lateral, longitudinal forces and torque at the wheel center can be parameterized. This effect can be measured by applying the lateral/longitudinal forces or torque about the z-axis to the left wheel center.

The following figure shows an example of angle alpha compliance:



Note

The wheel compliance tables are defined only for the left side. The right side is mirrored.

The total wheel position and rotation are the sum of the positions and orientations due to kinematics and elasto-kinematics.

The dynamics of these displacements are approximated by second-order dynamics with a natural frequency ω_n which is defined as a parameter. For example, the dynamic *x* displacement can be calculated as follows:

$$\Delta x_{dyn} = \frac{{\omega_n}^2}{s^2 + 2\,\zeta\,\omega_n\,s + {\omega_n}^2}\,\Delta x$$

where:
s is the Laplace operator
ζ is the damping ratio ($\zeta = 0{,}707$, set internally)

Inports

The following table shows the inports:

Name	Unit	Description
F_x_Tire[L;R]	[N]	Longitudinal tire force, left and right side
F_y_Tire[L;R]	[N]	Lateral tire force, left and right side
Pos_Wheel[L;R][x;y;z]	[m]	Position vector of the left and right wheel
Reset	[]	Resets all integrators in the block to their initial conditions
Sw_Rotation[1|2]	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (gamma-beta-alpha) ▪ 2: Toe-camber-sideview
Trq_x_Tire[L;R]	[N m]	Tire torque about the longitudinal axis, left and right side
Trq_z_Tire[L;R]	[N m]	Tire torque about the vertical axis, left and right side

Note

The following rules apply to the left wheel:
- Toe = gamma
- Camber = -alpha
- Sideview = -beta

For more information about the toe, camber and sideview angles refer to Orientation Angles (ASM Vehicle Dynamics Addendum 🕮).

Outports

The following table shows the outports:

Name	Unit	Description
Angle_Comp_Left_Wheel[x,y,z]	[rad]	Wheel orientation change due to suspension compliance, left side

Name	Unit	Description
Angle_Comp_Left_Wheel_dt[x,y,z]	[rad/s]	Relative wheel angular velocity due to suspension compliance, left side
Angle_Comp_Right_Wheel[x,y,z]	[rad]	Wheel orientation change due to suspension compliance, right side
Angle_Comp_Right_Wheel_dt[x,y,z]	[rad/s]	Relative wheel angular velocity due to suspension compliance, right side
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Displ_Comp_Left_Wheel[x,y,z]	[m]	Wheel position change due to suspension compliance, left side
Displ_Comp_Left_Wheel_dt[x,y,z]	[m/s]	Relative wheel translational velocity due to suspension compliance, left side
Displ_Comp_Right_Wheel[x,y,z]	[m]	Wheel position change due to suspension compliance, right side
Displ_Comp_Right_Wheel_dt[x,y,z]	[m/s]	Relative wheel translational velocity due to suspension compliance, right side

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_NaturalFrequency_Displ_x	[1/s]	Natural frequency for the dynamics of the displacement in x direction
Const_NaturalFrequency_Displ_y	[1/s]	Natural frequency for the dynamics of the displacement in y direction
Const_NaturalFrequency_Angle_AlphaCamber	[1/s]	Natural frequency for the dynamics of the α angle
Const_NaturalFrequency_Angle_GammaToe	[1/s]	Natural frequency for the dynamics of the γ angle
Map_Angle_AlphaCamber_LeftWheel_F_x	[deg]	α angle of left wheel as a function of left tire longitudinal force
Map_Angle_AlphaCamber_LeftWheel_F_y	[deg]	α angle of left wheel as a function of left tire lateral force
Map_Angle_AlphaCamber_LeftWheel_Trq_x	[deg]	α angle of left wheel as a function of left tire torque around the x-axis
Map_Angle_AlphaCamber_LeftWheel_Trq_z	[deg]	α angle of left wheel as a function of left tire torque around the z-axis
Map_Angle_Gamma_Axle_F_y	[deg]	γ angle of axles as a function of lateral force
Map_Angle_Gamma_Axle_Trq_z	[deg]	γ angle of axle as a function of left and right tire torque around the z-axis
Map_Angle_GammaToe_LeftWheel_F_x	[deg]	γ angle of left wheel as a function of left tire longitudinal force
Map_Angle_GammaToe_LeftWheel_F_y	[deg]	γ angle of left wheel as a function of left tire lateral force
Map_Angle_GammaToe_LeftWheel_Trq_x	[deg]	γ angle of left wheel as a function of left tire torque around the x-axis
Map_Angle_GammaToe_LeftWheel_Trq_z	[deg]	γ angle of left wheel as a function of left tire torque around the z-axis
Map_Displ_x_Axle_F_x	[mm]	x displacement of axle center as a function of the sum of left and right tire longitudinal force

Name	Unit	Description
Map_Displ_y_Axle_F_y	[mm]	y displacement of axle center as a function of the sum of left and right tire lateral force

Suspension Force Kinematics Front

Description

This block projects the spring, damper and stabilizer forces in the direction of the axle degrees of freedom. The projection is carried out according to the kinematic relations described in the KINEMATICS block.

The evaluated forces in the Front Suspension Forces must be transformed in the direction of the relevant degrees of freedom.

For a spring deflection

$z_{Sp} = \left[z_{Sp_FL}(q_{1_Front}, q_{2_Front}) z_{Sp_FR}(q_{1_Front}, q_{2_Front})\right]^T$ and front suspension DOFs $q_{Front} = \left[q_{1_Front}\ q_{2_Front}\right]$ a Jacobian matrix is calculated as follows:

$$J_{Sp} = \frac{\partial z_{Sp}}{[\partial q_{Front}]^T}$$

From the spring forces

$$F_{Sp} = \left[F_{Sp_FL}(z_{Sp_FL})\ F_{Sp_FR}(z_{Sp_FR})\right]^T$$

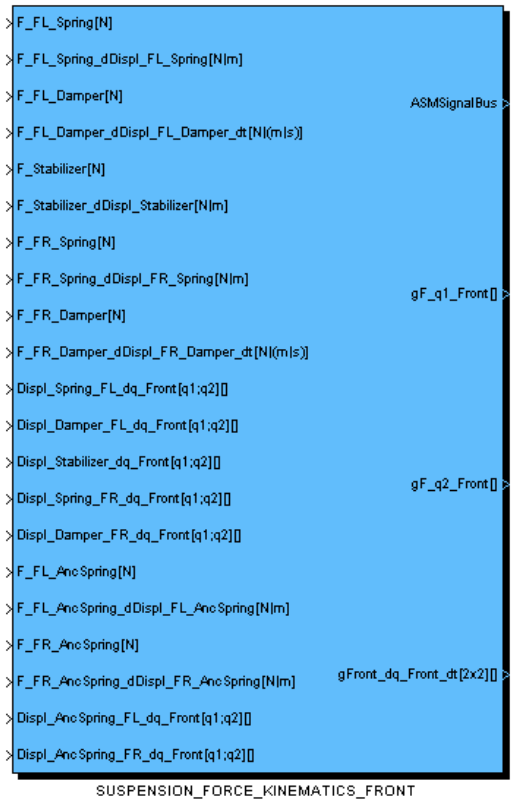
the generalized forces are calculated as follows:

$$gF_{Sp} = J^T{}_{Sp} F_{Sp}$$

The same calculations are carried out for the damper and the stabilizer. Finally, the generalized suspension forces are added.

$$[gF_q_{Front}]_{2 \times 1} = J^T{}_{Sp} F_{Sp} + J^T{}_{D} F_{D} + J^T{}_{Stabi} F_{Stabi}$$

To stabilize the equation of motion, the derivatives of the generalized forces with respect to the generalized speeds are required. It can be calculated using the Jacobian matrices and the derivatives of the spring, damper and stabilizer forces with respect to spring displacement, damper velocity, and stabilizer displacement respectively.



Inports

The following table shows the inports:

Name	Unit	Description
Displ_AncSpring_FL_dq_Front[q1;q2]	[]	Derivative of ancillary spring displacement of front left wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_AncSpring_FR_dq_Front[q1;q2]	[]	Derivative of ancillary spring displacement of front right wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_Damper_FL_dq_Front[q1;q2]	[]	Derivative of damper displacement of front left wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_Damper_FR_dq_Front[q1;q2]	[]	Derivative of damper displacement of front right wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_Spring_FL_dq_Front[q1;q2]	[]	Derivative of spring displacement of front left wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_Spring_FR_dq_Front[q1;q2]	[]	Derivative of spring displacement of front right wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel

Name	Unit	Description
Displ_Stabilizer_dq_Front[q1;q2]	[]	Derivative of stabilizer displacement of front left wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
F_FL_AncSpring	[N]	Force of front left ancillary spring
F_FL_AncSpring_dDispl_FL_AncSpring	[N/m]	Change of ancillary spring force front left with respect to vertical displacement of ancillary spring joint point front left
F_FL_Damper	[N]	Force of front left damper
F_FL_Damper_dDispl_FL_Damper_dt	[N/(m/s)]	Change of damper force front left with respect to vertical speed of displacement of damper joint point front left
F_FL_Spring	[N]	Force of front left spring
F_FL_Spring_dDispl_FL_Spring	[N/m]	Change of spring force front left with respect to vertical displacement of spring joint point front left
F_FR_AncSpring	[N]	Force of front right ancillary spring
F_FR_AncSpring_dDispl_FR_AncSpring	[N/m]	Change of ancillary spring force front right with respect to vertical displacement of ancillary spring joint point front right
F_FR_Damper	[N]	Force of front right damper
F_FR_Damper_dDispl_FR_Damper_dt	[N/(m/s)]	Change of damper force front right with respect to vertical speed of displacement of damper joint point front right
F_FR_Spring	[N]	Force of front right spring
F_FR_Spring_dDispl_FR_Spring	[N/m]	Change of spring force front right with respect to vertical displacement of spring joint point front right
F_Stabilizer	[N]	Force of stabilizer
F_Stabilizer_dDispl_Stabilizer	[N/m]	Change of stabilizer force with respect to vertical displacement of stabilizer joint point

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
gF_q1_Front	[N] or [Nm][1)]	Generalized force in direction of q1_Front
gF_q2_Front	[N] or [Nm][1)]	Generalized force in direction of q2_Front
gFront_dq_Front_dt[2x2]	[N] or [Nm][1)]	Derivatives of the generalized forces with respect to the generalized speeds

1) The unit of the generalized forces and/or torques depends on the degree of freedom. If q is a displacement, gF is a force and has a unit of [N]. If q is a rotation, gF is a torque and has a unit of [Nm].

Related topics

References

History of the SUSPENSION_FORCE_KINEMATICS_FRONT Block..... 450

Suspension Compliance 2-D (Symmetric)

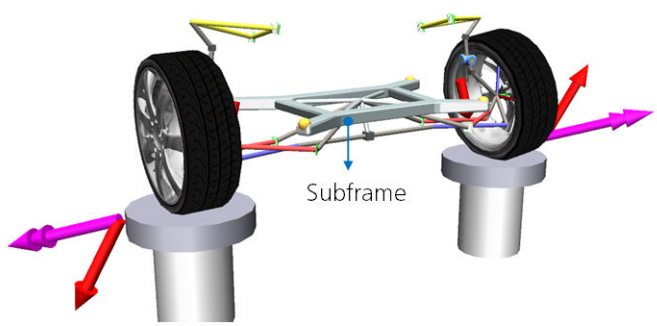
Description

Due to suspension elasticity, additional elastic displacements and rotation are caused by the forces and torques acting on the wheel. Suspension compliance describes these additional displacements and rotation as functions of the forces and torques.

For independent suspension, the compliance effect of the left wheel normally only depends on the force and torque applying to the left wheel.

For semi-independent suspension (e.g., twist-beam axle), dependent suspension, or independent suspension with subframe, the force and torque at the right wheel also cause additional displacements and rotation at the left wheel.

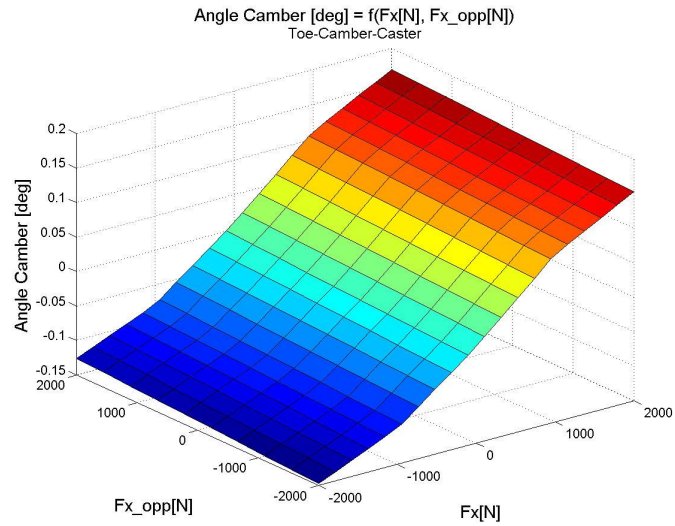
The following illustration shows an independent suspension (four link suspension) connected to a subframe. Since the subframe is elastically attached to the chassis, the left wheel will experience some movement due to force/torque on the right wheel.



In this subsystem, the compliance effect of the left wheel can be defined as a nonlinear function of the left force/torque and the right force/torque as 2-D look-up table (LUT):

- The x-axis defines the left force/torque
- The y-axis defines the right force/torque
- The z-axis defines the additional displacement/rotation of the left wheel

The following illustration shows an example of the compliance effect of the camber angle of the left wheel that depends on longitudinal forces:



Note

The wheel-compliance tables are defined for the left side. The right side is mirrored.

The total wheel position and rotation are the sum of the positions and orientations due to kinematics and elasto-kinematics.

The dynamics of these displacements are approximated by second-order dynamics with a natural frequency ω_n defined as a parameter. For example, the dynamic x displacement can be calculated as follows:

$$\Delta x_{dyn} = \frac{{\omega_n}^2}{s^2 + 2\,\zeta\,\omega_n\,s + {\omega_n}^2}\,\Delta x$$

where:

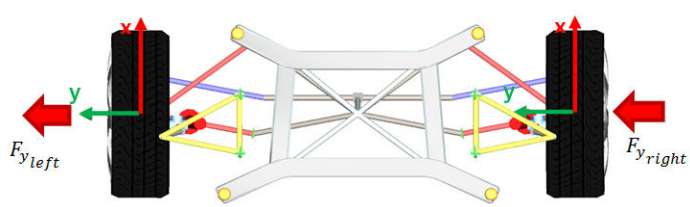
s is the Laplace operator

ζ is the damping ratio

$\zeta = 0{,}707$, set internally

Coordinate system

The underlying coordinate system is a right-hand coordinate system where x points forward, y points to the left, and z points upwards. The left and right wheels also have a right-hand coordinate system. For example, a positive lateral force for the left and right wheels acts in the positive y-axis direction. See the following illustration:



Inports

The following table shows the inports:

Name	Unit	Description
F_x_Tire[L;R]	[N]	Longitudinal tire force, left and right side
F_y_Tire[L;R]	[N]	Lateral tire force, left and right side
Pos_Wheel[L;R][x;y;z]	[m]	Position vector of the left and right wheel
Reset	[]	Resets all integrators in the block to their initial conditions
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Gamma-Beta-Alpha ▪ 2: Toe-Camber-Sideview
Trq_x_Tire[L;R]	[Nm]	Tire torque about the longitudinal axis, left and right side
Trq_y_Tire[L;R]	[Nm]	Tire torque about the lateral axis, left and right side
Trq_z_Tire[L;R]	[Nm]	Tire torque about the vertical axis, left and right side

Note

For the left wheel, the following rules are valid:

- toe = - gamma
- camber = - alpha
- sideview = -beta

For more information on the toe, camber, and sideview angles, refer to Orientation Angles (ASM Vehicle Dynamics Addendum 🕮).

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Angle_Comp_Left_Wheel[x,y,z]	[rad/s]	Wheel orientation change due to suspension compliance, left side
Angle_Comp_Left_Wheel_dt[x,y,z]	[rad/s]	Relative wheel angular velocity due to suspension compliance, left side
Angle_Comp_Right_Wheel[x,y,z]	[rad/s]	Wheel orientation change due to suspension compliance, right side
Angle_Comp_Right_Wheel_dt[x,y,z]	[rad/s]	Relative wheel angular velocity due to suspension compliance, right side
Displ_Comp_Left_Wheel[x,y,z]	[m]	Wheel position change due to suspension compliance, left side
Displ_Comp_Left_Wheel_dt[x,y,z]	[m/s]	Relative wheel translational velocity due to suspension compliance, left side
Displ_Comp_Right_Wheel[x,y,z]	[m]	Wheel position change due to suspension compliance, right side
Displ_Comp_Right_Wheel_dt[x,y,z]	[m/s]	Relative wheel translational velocity due to suspension compliance, right side

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_NaturalFrequency_Angle_AlphaCamber	[1/s]	Natural frequency for the dynamics of the α angle
Const_NaturalFrequency_Angle_GammaToe	[1/s]	Natural frequency for the dynamics of the γ angle
Const_NaturalFrequency_Displ_x	[1/s]	Natural frequency for the dynamics of the displacement in x direction
Const_NaturalFrequency_Displ_y	[1/s]	Natural frequency for the dynamics of the displacement in y direction
Map_Angle_Alpha_Wheel_F_x_2d	[deg]	α angle of left wheel as a function of left and right tire longitudinal force
Map_Angle_Alpha_Wheel_F_y_2d	[deg]	α angle of left wheel as a function of tire lateral force
Map_Angle_Alpha_Wheel_Trq_x_2d	[deg]	α angle of left wheel as a function of left and right tire torque about longitudinal axis
Map_Angle_Alpha_Wheel_Trq_y_2d	[deg]	α angle of left wheel as a function of left and right tire torque about lateral axis

Name	Unit	Description
Map_Angle_Alpha_Wheel_Trq_z_2d	[deg]	α angle of left wheel as a function of left and right tire torque about the vertical axis
Map_Angle_Gamma_Wheel_F_x_2d	[deg]	γ angle of left wheel as a function of left and right tire longitudinal force
Map_Angle_Gamma_Wheel_F_y_2d	[deg]	γ angle of left wheel as a function of left and right tire lateral force
Map_Angle_Gamma_Wheel_Trq_x_2d	[deg]	γ angle of left wheel as a function of left and right tire torque about the longitudinal axis
Map_Angle_Gamma_Wheel_Trq_y_2d	[deg]	γ angle of left wheel as a function of left and right tire torque about the lateral axis
Map_Angle_Gamma_Wheel_Trq_z_2d	[deg]	γ angle of left wheel as a function of left and right tire torque about the vertical axis
Map_Displ_x_Wheel_F_x_2d	[mm]	x displacement of left wheel center as a function of left and right tire longitudinal force
Map_Displ_x_Wheel_F_y_2d	[mm]	x displacement of left wheel center as a function of left and right tire lateral force
Map_Displ_x_Wheel_Trq_x_2d	[mm]	x displacement of left wheel center as a function of left and right tire torque about the longitudinal axis
Map_Displ_x_Wheel_Trq_y_2d	[mm]	x displacement of left wheel center as a function of left and right tire torque about the lateral axis
Map_Displ_x_Wheel_Trq_z_2d	[mm]	x displacement of left wheel center as a function of left and right tire torque about the vertical axis
Map_Displ_y_Wheel_F_x_2d	[mm]	y displacement of left wheel center as a function of left and right tire longitudinal force
Map_Displ_y_Wheel_F_y_2d	[mm]	y displacement of left wheel center as a function of tire lateral force
Map_Displ_y_Wheel_Trq_x_2d	[mm]	y displacement of left wheel center as a function of left and right tire torque about the longitudinal axis
Map_Displ_y_Wheel_Trq_y_2d	[mm]	y displacement of left wheel center as a function of left and right tire torque about the lateral axis
Map_Displ_y_Wheel_Trq_z_2d	[mm]	y displacement of left wheel center as a function of left and right tire torque about the vertical axis

Related topics

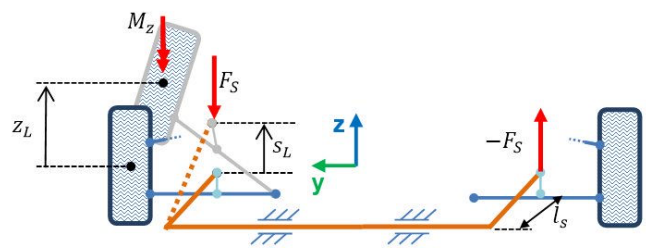
References

SusComp 2D Front (ModelDesk Parameterizing 🕮)
SusComp 2D Rear (ModelDesk Parameterizing 🕮)

Stabilizer Compliance (Symmetric)

Description

During vehicle roll movement, the difference between the left and right stabilizer endpoints will not be equal to zero. This causes a rotation of the stabilizer and the stabilizer force acts at the end-points.



Due to the geometry of the suspension, the stabilizer force will cause a torque at the wheel center (currently, only vertical torque M_z is considered) and this torque will produce an additional compliance effect on the wheel.

The following kinematic relationships for the stabilizer displacement s_L and the wheel rotation about the z-axis γ hold:

$$s_L = s_L(u, z)$$

$$\gamma = \gamma(u, z)$$

where:

u is the steering rod displacement

z is the vertical wheel displacement

u and z are the degrees of freedom of the suspension.

The kinetostatic equation for the stabilizer force F_{s_L} is as follows:

$$M_z \cdot \delta\gamma + \cdots = F_{s_L} \cdot \delta s_L$$

And the virtual displacement δs_L can be written as:

$$\delta s_L = \frac{\delta s_L}{\delta u} \cdot \left(\frac{\delta\gamma}{\delta u}\right)^{-1} \cdot \delta\gamma + \frac{\delta s_L}{\delta z} \cdot \delta z$$

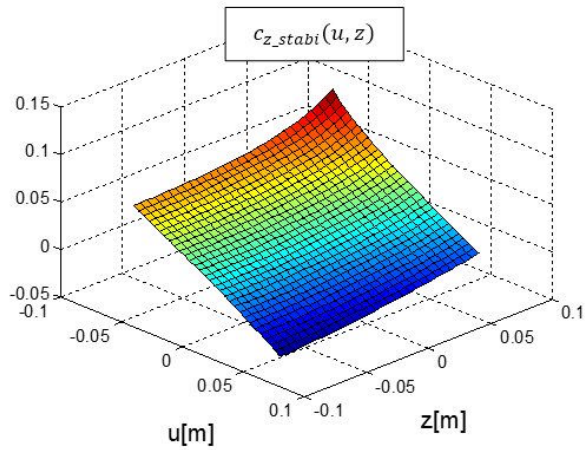
Combining these two equations leads to the following equation:

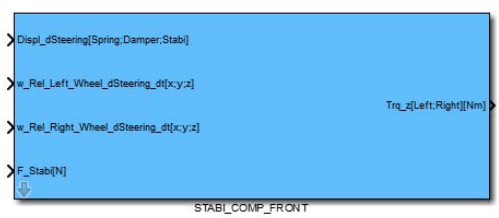
$$M_z \cdot \delta\gamma + \cdots = F_{s_L} \cdot \left(\frac{\delta s_L}{\delta u} \cdot \left(\frac{\delta\gamma}{\delta u}\right)^{-1} \cdot \delta\gamma + \cdots\right)$$

And finally you get the equation for M_z:

$$M_z = F_{s_L} \cdot \frac{\delta s_L}{\delta u} \cdot \left(\frac{\delta\gamma}{\delta u}\right)^{-1} = F_{sL} \cdot c_{z_{stabi}}\left(u, z\right)$$

The table for the stabilizer compliance can also be calculated automatically in ASM KnC. The following illustration shows an example of the parameterization using ASM KnC.





Inports

The following table shows the inports:

Name	Unit	Description
Displ_dSteering[Spring;Damper;Stabi]	[]	Derivative of the spring, damper and stabilizer displacement with respect to the steering rod displacement
F_Stabi	[N]	Stabilizer force
w_Rel_Left_Wheel_dSteering_dt[x;y;z]	[rad/m]	Derivative of the left wheel angular speed with respect to the steering rod speed
w_Rel_Right_Wheel_dSteering_dt[x;y;z]	[rad/m]	Derivative of the right wheel angular speed with respect to the steering rod speed

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_z[Left;Right]	[Nm]	Additional vertical torque at the wheel center

Related topics

Basics

Introduction to ASM KnC (ASM KnC Guide 🕮)

Suspension Kinematics

Where to go from here

Information in this section

Front Suspension Kinematics (Asymmetric)..... 163
In this subsystem, the wheel motion for each wheel is defined separately via a set of look-up tables.

Front Suspension Kinematics (Asymmetric, 3DOF)..... 169
The position of the wheel midpoint, the orientation of the wheel and displacements of the spring, damper and stabilizer are calculated in this block.

Front Suspension Kinematics (Symmetric)..... 174
This block describes the wheel suspension kinematics for the front left and right wheels. These kinematics depend on both the wheel lift and the displacement of the steering rod.

Front Suspension Kinematics MacPherson Strut..... 179
The MC_PHERSON_STRUT system is a geometric description of the suspension kinematics.

Front Suspension Kinematics (Symmetric, 3DOF)..... 183
This block describes the wheel suspension kinematics for the rear left and right wheels.These kinematics depend on both the wheel lift and the displacement of the steering rod.

Suspension Kinematics Rigid Truck Axle..... 189
With the SUSKIN_RIGID_TRUCK_FRONT block, the kinematics of a rigid (solid) axle suspension kinematics can be simulated.

Suspension Kinematics Rigid Axle (Symmetric)..... 194
With this block, rigid (solid) axle suspension kinematics can be simulated.

Front Suspension Kinematics Ancillary Spring..... 199
The displacements of the ancillary springs are calculated in these blocks.

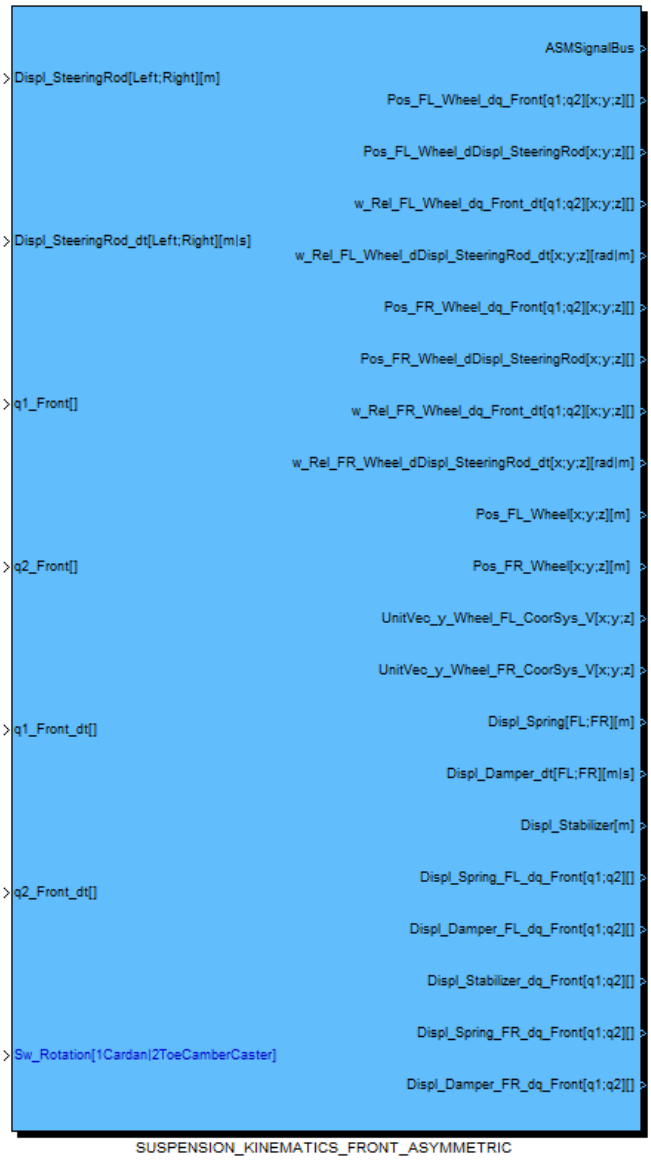
Front Suspension Kinematics (Asymmetric)

Description

In this subsystem, the wheel motion for each wheel is defined separately via a set of look-up tables. This is in contrast to the Front Suspension Kinematics (Symmetric) subsystem, where the tables define only the left wheel, and the right wheel is mirrored.

Note

The block is implemented in such a way that the degrees of freedom are configurable and generalized, for example, they do not need to be the wheel vertical motion and steering. In addition, both the front and the rear suspension can be configured to be steerable and dependent. These features are not yet supported by ModelDesk. Only the asymmetry functionality can be used via ModelDesk. In the standard configurations, q1_Front is z_FL_Wheel (i.e. front left wheel vertical displacement) and q2_Front is z_FR_Wheel (i.e. front right wheel vertical displacement).



These kinematics are described as 2-D look-up tables with two inputs:

- Steering rod position q_{St}
- Wheel self-vertical displacement q_{FL}, q_{FR}

These are the degrees of freedom used for the front suspension. The outputs of these look-up tables are:

- x displacement of wheel center
- y displacement of wheel center
- α angle, rotation around the x-axis
- β angle, rotation around the y-axis
- γ angle, rotation around the z-axis

- z_{sp} displacement of spring in z direction
- z_d displacement of damper in z direction
- z_{st} displacement of stabilizer in z direction

For example, for the front left wheel, the relative wheel position change can be calculated as follows:

$$\Delta r_{FL} = \Delta r_{FL}(q_{FL}, q_{St})$$

The total relative wheel position can be calculated as follows:

$$r_{FL} = r_{FL0} + \Delta r_{FL}$$

where r_{FL0} is the position of the wheel center at the zero position, at which the kinematical displacements are measured.

The equation for wheel orientation θ_{FL} is:

$$\theta_{FL} = \theta_{FL}(q_{FL}, q_{St})$$

where

$\theta_{FL} = [\alpha_{FL}, \beta_{FL}, \gamma_{FL}]$ are the cardan angles of the wheel.

Note

The rotation sequence is rotation about the z- then y- then x-axis, i.e., RotZY'X".

These angles are used to calculate the orientation of the wheel to the vehicle system:

$$^{V}T_{FL} = T_z(\gamma_{FL})T_y(\beta_{FL})T_x(\alpha_{FL})$$

The spring, damper, and stabilizer joint points also depend kinematically on the wheel center displacement and steering input. The main dependency is mostly on the wheel center vertical displacement.

$$z_{sp} = z_{sp}(q_{FL}, q_{St}),\ z_d = z_s(q_{FL,} q_{St}),\ z_{stab} = z_{stab}(q_{FL}, q_{St})$$

These displacements are used to calculate the spring, damper, and stabilizer forces.

To establish the equation of motion, the partial derivatives of the relation described above must be evaluated, i.e., changes in wheel position and orientation with respect to vertical wheel movement and steering input:

$$\frac{\partial r_{FL}}{\partial q_{1_Front}}, \frac{\partial r_{FL}}{\partial q_{2_Front}}, \frac{\partial r_{FL}}{\partial q_{St}}, \frac{\partial \omega_{FL}}{\partial \dot{q}_{1_Front}}, \frac{\partial \omega_{FL}}{\partial \dot{q}_{2_Front}}, \frac{\partial \omega_{FL}}{\partial \dot{q}_{St}}$$

In addition, to calculate the suspension equivalent force in the direction of wheel center motion, the following derivatives must be evaluated:

$$\frac{\partial z_{Sp_FL}}{\partial q_{q_Front}}, \frac{\partial z_{Sp_FL}}{\partial q_{2_Front}}, \frac{\partial z_{D_FL}}{\partial q_{1_Front}}, \frac{\partial z_{D_FL}}{\partial q_{2_Front}}, \frac{\partial z_{Stabi}}{\partial q_{1_Front}}, \frac{\partial z_{Stabi}}{\partial q_{2_Front}}$$

Inports

The following table shows the inports:

Name	Unit	Description
Displ_SteeringRod	[m]	Steering rod displacement
Displ_SteeringRod_dt	[m/s]	Speed of steering rod displacement
q1_Front	[]	First generalized coordinate for front suspension
q1_Front_dt	[]	Derivative of first generalized coordinate for front suspension with respect to time
q2_Front	[]	Second generalized coordinate for front suspension
q2_Front_dt	[]	Derivative of second generalized coordinate for front suspension with respect to time
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (Gamma-Beta-Alpha) ▪ 2: ToeCamberSideview (Toe-Camber-Sideview)

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Collection of ASM signals
Displ_Damper_dt[FL;FR]	[m/s]	Speed of vertical displacement of damper joint points of front left and front right wheel
Displ_Damper_FL_dq_Front[q1;q2]	[]	Derivative of damper displacement of front left wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_Damper_FR_dq_Front[q1;q2]	[]	Derivative of damper displacement of front right wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_Spring[FL;FR][m|s]	[m]	Vertical displacement of spring joint points of front left and front right wheel
Displ_Spring_FL_dq_Front[q1;q2]	[]	Derivative of spring displacement of front left wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_Spring_FR_dq_Front[q1;q2]	[]	Derivative of spring displacement of front right wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_Stabilizer	[m]	Vertical displacement of the stabilizer left joint point relative to the right joint point
Displ_Stabilizer_dq_Front[q1;q2]	[]	Derivative of stabilizer displacement of front left wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Pos_FL_Wheel[x,y,z]	[m]	Position vector of the front left wheel
Pos_FL_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of front left wheel with respect to steering rod displacement

Name	Unit	Description
Pos_FL_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of wheel position of front left wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Pos_FR_Wheel[x,y,z]	[m]	Position vector of the front right wheel
Pos_FR_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of front right wheel with respect to steering rod displacement
Pos_FR_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of wheel position of front right wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
UnitVec_y_Wheel_FL_CoorSys_V[x;y;z]	[]	Unit vector in y direction of front left wheel
UnitVec_y_Wheel_FR_CoorSys_V[x;y;z]	[]	Unit vector in y direction of front right wheel
w_Rel_FL_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of front left wheel angular velocity with respect to steering rod speed
w_Rel_FL_Wheel_dq_Front_dt[q1;q2][x;y;z]	[]	Derivative of front left wheel angular velocity with respect to vertical wheel speeds of front left (q1) and front right (q2) wheel
w_Rel_FR_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel orientation of front right wheel with respect to steering rod displacement
w_Rel_FR_Wheel_dq_Front_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of front right wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel

Parameters

Note

According to ASM naming conventions, parameter names that begin with "Map" are look-up tables. In Front Suspension Kinematics, these are 2-D look-up tables with steering rod displacement as the row index and vertical wheel displacement as the column index.

The following table shows the parameters:

Name	Unit	Description
Const_TrackWidth_Front	[m]	Track width front
Map_Angle_Alpha_Wheel_Left	[rad]	α angle of left wheel, i.e. left wheel rotation around the x-axis relative to vehicle
Map_Angle_Alpha_Wheel_Right	[rad]	α angle of right wheel, i.e. left wheel rotation around the x-axis relative to vehicle
Map_Angle_Beta_Wheel_Left	[rad]	β angle of left wheel, i.e. left wheel rotation around the y-axis relative to vehicle
Map_Angle_Beta_Wheel_Right	[rad]	β angle of right wheel, i.e. left wheel rotation around the y-axis relative to vehicle
Map_Angle_Gamma_Wheel_Left	[rad]	γ angle of left wheel, i.e. left wheel rotation around the z-axis relative to vehicle

Name	Unit	Description
Map_Angle_Gamma_Wheel_Right	[rad]	γ angle of right wheel, i.e. left wheel rotation around the z-axis relative to vehicle
Map_Displ_Damper_Left	[m]	Vertical displacement of the left damper joint
Map_Displ_Damper_Right	[m]	Vertical displacement of the right damper joint
Map_Displ_Spring_Left	[m]	Vertical displacement of the left spring joint
Map_Displ_Spring_Right	[m]	Vertical displacement of the right spring joint
Map_Displ_Stabilizer_Left	[m]	Vertical displacement of the left stabilizer joint
Map_Displ_Stabilizer_Right	[m]	Vertical displacement of the right stabilizer joint
Map_Displ_x_Wheel_Left	[m]	x displacement of left wheel center relative to vehicle
Map_Displ_x_Wheel_Right	[m]	x displacement of right wheel center relative to vehicle
Map_Displ_y_Wheel_Left	[m]	y displacement of left wheel center relative to vehicle
Map_Displ_y_Wheel_Right	[m]	y displacement of right wheel center relative to vehicle

Related topics

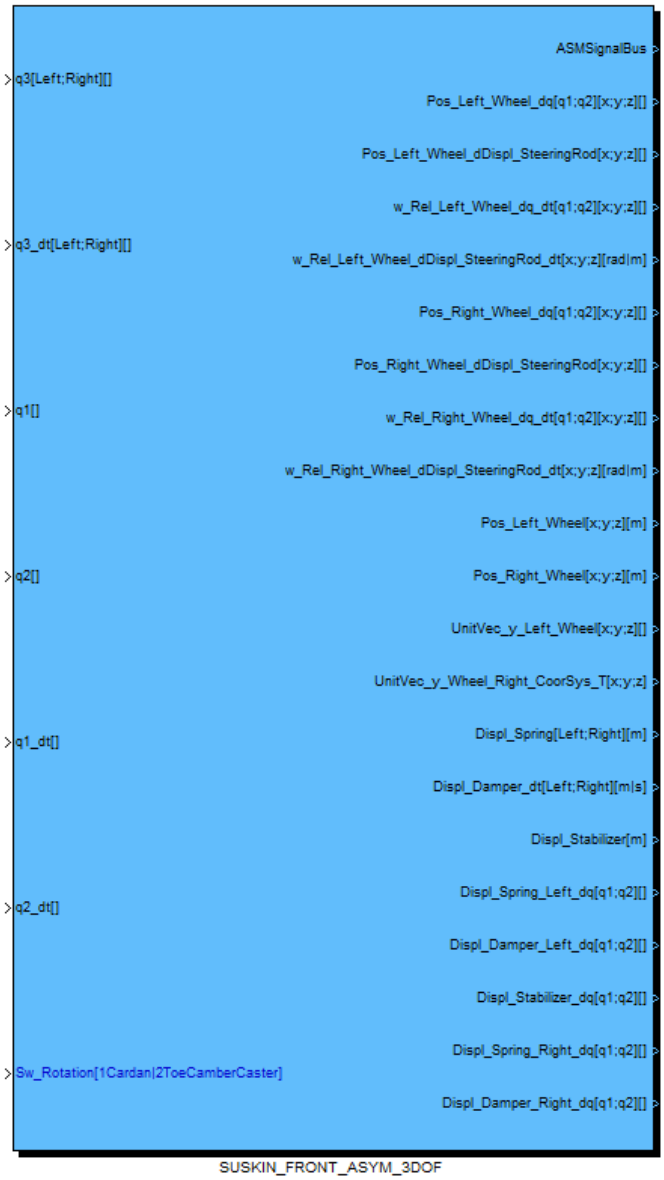
References

Front Suspension Kinematics (Symmetric)..... 174
SusKin Front Asym (ModelDesk Parameterizing 🕮)

Front Suspension Kinematics (Asymmetric, 3DOF)

Description

The position of the wheel midpoint, the orientation of the wheel and displacements of the spring, damper and stabilizer are calculated in this block.



These kinematics are described as 2-D and 1-D look-up tables with three inputs:

- Steering rod position q_{steer}
- Wheel vertical displacement on left q_{left}
- Wheel vertical displacement on right q_{right}

These are the three degrees of freedom for each suspension.

The outputs of these look-up tables are:

- x displacement of wheel center
- y displacement of wheel center
- α angle, rotation around the x-axis
- β angle, rotation about y'-axis
- γ angle, rotation about z''-axis
- z_{spring}, displacement of spring in z direction
- z_{damper}, displacement of damper in z direction
- z_{stabi}, displacement of stabilizer in z direction

For example, for the left wheel, the relative wheel position change can be calculated as follows:

$$\Delta r_{left} = \Delta r_{left}(q_{left}, q_{right}) + \Delta r_{left}(q_{steer})$$

Note

The 2-D look-up tables on the left side are functions of q_{left} and q_{right}, but the 2-D look-up tables on the right side depend on q_{right} and q_{left} $\Delta r_{left} = \Delta r_{left}(q_{left}, q_{right}) + \Delta r_{left}(q_{steer})$ (order is changed), for example:

$\Delta r_{right} = \Delta r_{right}(q_{right}, q_{left}) + \Delta r_{right}(q_{steer})$

The total relative wheel position can be calculated as follows:

$$r_{left} = r_{left,0} + \Delta r_{left}$$

where $r_{left,0}$ is the position of the wheel center at the zero position, at which the kinematical displacements are measured.

The equation for wheel orientation θ_{left} is:

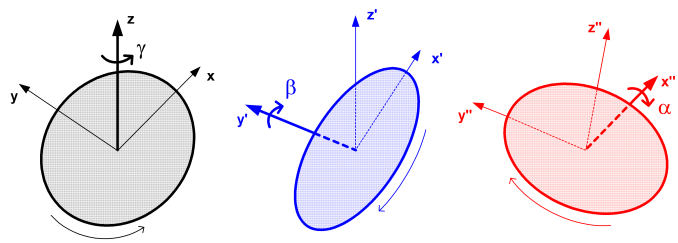
$$\theta_{left} = \theta_{left}(q_{left}, q_{right}) + \theta_{left}(q_{steer})$$

where

$\theta = [\alpha, \beta, \gamma]$ are the cardan angles of the wheel.

Note

The rotation sequence is rotation about the z- then y- then x-axis, i.e., RotZY'X". The following illustration shows the rotation sequence.



These angles are used to calculate the orientation of the wheel to the vehicle system:

$$^{V}T_{FL} = T_z(\gamma_{FL})T_y(\beta_{FL})T_x(\alpha_{FL})$$

The spring, damper, and stabilizer joint points also depend kinematically on the wheel center displacement and steering input. The main dependency is mostly on the wheel center vertical displacement and the steering rod displacement:

$$z_{spring} = z_{spring}(q_{left}, q_{right}) + z_{spring}(q_{steer})$$

$$z_{damper} = z_{damper}(q_{left}, q_{right}) + z_{damper}(q_{steer})$$

$$z_{stabi} = z_{stabi}(q_{left}, q_{right}) + z_{stabi}(q_{steer})$$

These displacements are used to calculate the spring, damper, and stabilizer forces.

To establish the equation of motion, the partial derivatives of the relation described above must be evaluated, i.e., changes in wheel position and orientation with respect to vertical wheel movement and steering input, i.e:

$$\frac{\partial r_{left}}{\partial q_{left}}, \frac{\partial r_{left}}{\partial q_{right}}, \frac{\partial r_{left}}{\partial q_{steer}}, \frac{\partial \omega_{left}}{\partial q_{left}}, \frac{\partial \omega_{left}}{\partial q_{right}}, \frac{\partial \omega_{left}}{\partial q_{steer}}$$

In addition, to calculate the suspension equivalent force in the direction of wheel center motion, the following derivatives must be evaluated:

$$\frac{\partial z_{spring}}{\partial q_{left}}, \frac{\partial z_{spring}}{\partial q_{right}}, \frac{\partial z_{spring}}{\partial q_{steer}}, \frac{\partial z_{damper}}{\partial q_{left}}, \frac{\partial z_{damper}}{\partial q_{right}}, \frac{\partial z_{damper}}{\partial q_{steer}}, \frac{\partial z_{stabi}}{\partial q_{left}}, \frac{\partial z_{stabi}}{\partial q_{right}}, \frac{\partial z_{stabi}}{\partial q_{steer}}$$

Inports

The following table shows the inports:

Name	Unit	Description
q1	[]	Generalized coordinate, e.g. vertical displacement of the left wheel
q1_dt	[]	Generalized speed, e.g. speed of vertical displacement of the left wheel
q2	[]	Generalized coordinate, e.g. vertical displacement of the right wheel
q2_dt	[]	Generalized speed, e.g. speed of vertical displacement of the right wheel
q3	[]	Generalized coordinate, e.g. steering rod displacement
q3_dt	[]	Generalized speed, e.g. speed of steering rod displacement

Name	Unit	Description
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (Gamma-Beta-Alpha) ▪ 2: ToeCamberSideview (Toe-Camber-Sideview)

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Displ_Damper_dt[Left;Right]	[m]	Speed of vertical displacement of damper joint points of left and right wheel
Displ_Damper_Left_dq[q1;q2]	[]	Derivative of damper displacement of left wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Damper_Right_dq[q1;q2]	[]	Derivative of damper displacement of right wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Spring[Left;Right]	[m/s]	Vertical displacement of spring joint points of left and right wheel
Displ_Spring_Left_dq[q1;q2]	[]	Derivative of spring displacement of left wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Spring_Right_dq[q1;q2]	[]	Derivative of spring displacement of right wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Stabilizer	[]	Vertical displacement of the stabilizer left joint point relative to the right joint point
Displ_Stabilizer_dq[q1;q2]	[]	Derivative of stabilizer displacement of wheels with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Pos_Left_Wheel[x,y,z]	[m]	Position vector of the left wheel
Pos_Left_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of left wheel with respect to steering rod displacement
Pos_Left_Wheel_dq[q1;q2][x;y;z]	[]	Derivative of wheel position of left wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Pos_Right_Wheel[x,y,z]	[m]	Position vector of the right wheel
Pos_Right_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of right wheel with respect to steering rod displacement
Pos_Right_Wheel_dq[q1;q2][x;y;z]	[]	Derivative of wheel position of right wheel with respect to vertical wheel speeds of left (q1) and right (q2) wheel
UnitVec_y_Wheel_Left	[x;y;z]	Unit vector in y direction of left wheel

Name	Unit	Description
UnitVec_y_Wheel_Right	[x;y;z]	Unit vector in y direction of right wheel
w_Rel_Left_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel angular speed of left wheel with respect to steering rod speed
w_Rel_Left_Wheel_dq_dt[q1;q2][x;y;z]	[rad/m]	Derivative of wheel angular speed of left wheel with respect to vertical wheel speeds of left (q1) and right (q2) wheel
w_Rel_Right_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel angular speed of right wheel with respect to steering rod speed
w_Rel_Right_Wheel_dq_dt[q1;q2][x;y;z]	[rad/m]	Derivative of wheel angular speed of right wheel with respect to vertical wheel speeds of left (q1) and right (q2) wheel

Parameters

Note

- According to ASM naming conventions, parameter names that begin with *Map* are look-up tables. In front suspension kinematics, these are 2-D look-up tables with vertical wheel lift on the left side as the row index and vertical wheel lift on the right side as the column index. The 1-D look-up table has steering rod displacement as the column index.
- The kinematics tables are defined only for the left side. The right side is mirrored.

The following table shows the parameters:

Name	Unit	Description
Const_Dist_LeftWheel	[m]	Position from vehicle coordinate system to left wheel
Const_Dist_RightWheel	[m]	Position from vehicle coordinate system to right wheel
Map_Angle_Alpha_LeftWheel_q1q2	[deg]	Angle_Alpha_LeftWheel[deg] = f(q1[], q2[])
Map_Angle_Alpha_LeftWheel_q3	[deg]	Angle_Alpha_LeftWheel[deg] = f(Displ_SteeringRod[m])
Map_Angle_Alpha_RightWheel_q1q2	[deg]	Angle_Alpha_RightWheel[deg] = f(q1[], q2[])
Map_Angle_Alpha_RightWheel_q3	[deg]	Angle_Alpha_RightWheel[deg] = f(Displ_SteeringRod[m])
Map_Angle_Beta_LeftWheel_q1q2	[deg]	Angle_Beta_LeftWheel[deg] = f(q1[], q2[])
Map_Angle_Beta_LeftWheel_q3	[deg]	Angle_Beta_LeftWheel[deg] = f(Displ_SteeringRod[m])
Map_Angle_Beta_RightWheel_q1q2	[deg]	Angle_Beta_RightWheel[deg] = f(q1[], q2[])
Map_Angle_Beta_RightWheel_q3	[deg]	Angle_Beta_RightWheel[deg] = f(Displ_SteeringRod[m])
Map_Angle_Gamma_LeftWheel_q1q2	[deg]	Angle_Gamma_LeftWheel[deg] = f(q1[], q2[])
Map_Angle_Gamma_LeftWheel_q3	[deg]	Angle_Gamma_LeftWheel[deg] = f(Displ_SteeringRod[m])
Map_Angle_Gamma_RightWheel_q1q2	[deg]	Angle_Gamma_RightWheel[deg] = f(q1[], q2[])
Map_Angle_Gamma_RightWheel_q3	[deg]	Angle_Gamma_RightWheel[deg] = f(Displ_SteeringRod[m])
Map_Displ_Damper_LeftWheel_q1q2	[m]	Displ_LeftDamper[m] = f(q1[], q2[])
Map_Displ_Damper_LeftWheel_q3	[m]	Displ_LeftDamper[m] = f(Displ_SteeringRod[m])

Name	Unit	Description
Map_Displ_Damper_RightWheel_q1q2	[m]	Displ_RightDamper[m] = f(q1[], q2[])
Map_Displ_Damper_RightWheel_q3	[m]	Displ_RightDamper[m] = f(Displ_SteeringRod[m])
Map_Displ_Spring_LeftWheel_q1q2	[m]	Displ_LeftSpring[m] = f(q1[], q2[])
Map_Displ_Spring_LeftWheel_q3	[m]	Displ_LeftSpring[m] = f(Displ_SteeringRod[m])
Map_Displ_Spring_RightWheel_q1q2	[m]	Displ_RightSpring[m] = f(q1[], q2[])
Map_Displ_Spring_RightWheel_q3	[m]	Displ_RightSpring[m] = f(Displ_SteeringRod[m])
Map_Displ_Stabi_LeftWheel_q1q2	[m]	Displ_LeftStabilizer[m] = f(q1[], q2[])
Map_Displ_Stabi_LeftWheel_q3	[m]	Displ_LeftStabilizer[m] = f(Displ_SteeringRod[m])
Map_Displ_Stabi_RightWheel_q1q2	[m]	Displ_RightStabilizer[m] = f(q1[], q2[])
Map_Displ_Stabi_RightWheel_q3	[m]	Displ_RightStabilizer[m] = f(Displ_SteeringRod[m])
Map_Displ_x_LeftWheel_q1q2	[m]	Displ_x_LeftWheel[m] = f(q1[], q2[])
Map_Displ_x_LeftWheel_q3	[m]	Displ_x_LeftWheel[m] = f(Displ_SteeringRod[m])
Map_Displ_x_RightWheel_q1q2	[m]	Displ_x_RightWheel[m] = f(q1[], q2[])
Map_Displ_x_RightWheel_q3	[m]	Displ_x_RightWheel[m] = f(Displ_SteeringRod[m])
Map_Displ_y_LeftWheel_q1q2	[m]	Displ_y_LeftWheel[m] = f(q1[], q2[])
Map_Displ_y_LeftWheel_q3	[m]	Displ_y_LeftWheel[m] = f(Displ_SteeringRod[m])
Map_Displ_y_RightWheel_q1q2	[m]	Displ_y_RightWheel[m] = f(q1[], q2[])
Map_Displ_y_RightWheel_q3	[m]	Displ_y_RightWheel[m] = f(Displ_SteeringRod[m])

Related topics

References

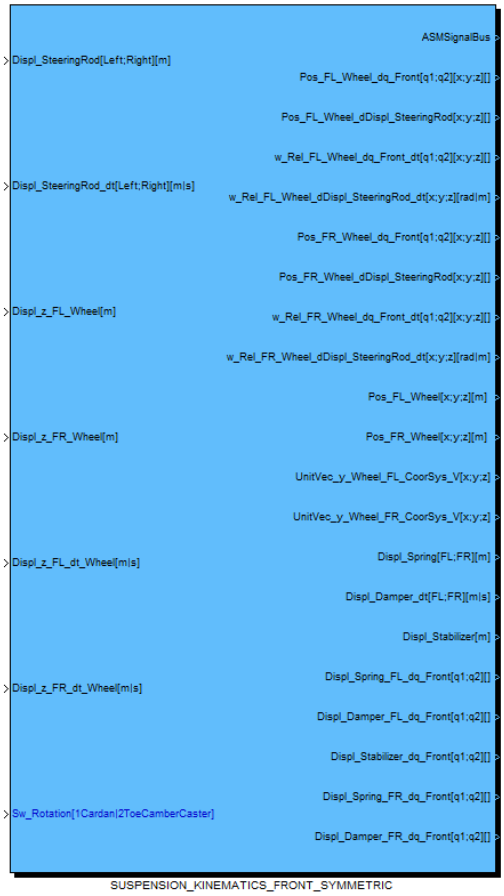
Front Suspension Kinematics (Symmetric, 3DOF)..... 183
SusGeo Front Asym (ModelDesk Parameterizing 🕮)
SusGeo Rear 2nd Asym (ModelDesk Parameterizing 🕮)
SusGeo Rear 3rd Asym (ModelDesk Parameterizing 🕮)
SusGeo Rear Asym (ModelDesk Parameterizing 🕮)
SusKin Front Asym 3DOF (ModelDesk Parameterizing 🕮)
SusKin Rear 2nd Asym 3DOF (ModelDesk Parameterizing 🕮)
SusKin Rear 3rd Asym 3DOF (ModelDesk Parameterizing 🕮)
SusKin Rear Asym 3DOF (ModelDesk Parameterizing 🕮)

Front Suspension Kinematics (Symmetric)

Description

This block describes the wheel suspension kinematics for the front left and right wheels. These kinematics depend on both the wheel lift and the displacement of the steering rod.

The position of the wheel midpoint, the orientation of the wheel and displacements of the spring, damper and stabilizer are calculated in this block.



These kinematics are described as 2-D look-up tables with two inputs:

- Steering rod position q_{St}
- Wheel self-vertical displacement q_{FL}, q_{FR}

These are the degrees of freedom used for the front suspension. The outputs of these look-up tables are:

- x displacement of wheel center
- y displacement of wheel center
- α angle, rotation around the x-axis
- β angle, rotation around the y-axis
- γ angle, rotation around the z-axis
- z_{sp} displacement of spring in z direction
- z_d displacement of damper in z direction
- z_{st} displacement of stabilizer in z direction

For example, for the front left wheel, the relative wheel position change can be calculated as follows:

$\Delta r_{FL} = \Delta r_{FL}(q_{FL}, q_{St})$

The total relative wheel position can be calculated as follows:

$r_{FL} = r_{FL0} + \Delta r_{FL}$

where r_{FL0} is the position of wheel center at the zero position, at which the kinematical displacements are measured.

The equation for wheel orientation θ_{FL} is:

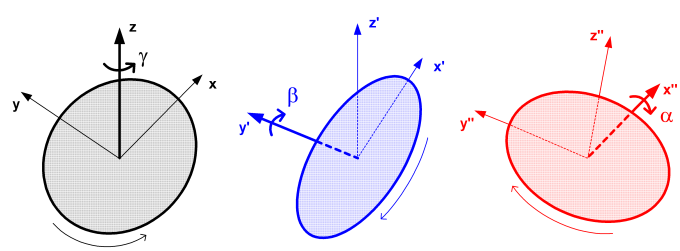
$\theta_{FL} = \theta_{FL}(q_{FL}, q_{St})$

where

$\theta_{FL} = [\alpha_{FL}, \beta_{FL}, \gamma_{FL}]$ are the cardan angles of the wheel.

Note

The rotation sequence is rotation about the z- then y- then x-axis, i.e., RotZY'X". The following illustration shows the rotation sequence.



These angles are used to calculate the orientation of the wheel to the vehicle system:

${}^{V}T_{FL} = T_z(\gamma_{FL})T_y(\beta_{FL})T_x(\alpha_{FL})$

The spring, damper, and stabilizer joint points also depend kinematically on the wheel center displacement and steering input. The main dependency is mostly on the wheel center vertical displacement.

$z_{sp} = z_{sp}(q_{FL}, q_{St})$, $z_d = z_s(q_{FL,} q_{St})$, $z_{stab} = z_{stab}(q_{FL}, q_{St})$

These displacements are used to calculate the spring, damper, and stabilizer forces.

To establish the equation of motion, the partial derivatives of the relation described above must be evaluated, i.e., changes in wheel position and orientation with respect to vertical wheel movement and steering input:

$$\frac{\partial r_{FL}}{\partial q_{1_Front}}, \frac{\partial r_{FL}}{\partial q_{2_Front}}, \frac{\partial r_{FL}}{\partial q_{St}}, \frac{\partial \omega_{FL}}{\partial \dot{q}_{1_Front}}, \frac{\partial \omega_{FL}}{\partial \dot{q}_{2_Front}}, \frac{\partial \omega_{FL}}{\partial \dot{q}_{St}}$$

Also to calculate the suspension equivalent force in the direction of wheel center motion, the following derivatives must be evaluated:

$$\frac{\partial z_{Sp_FL}}{\partial q_{q_Front}}, \frac{\partial z_{Sp_FL}}{\partial q_{2_Front}}, \frac{\partial z_{D_FL}}{\partial q_{1_Front}}, \frac{\partial z_{D_FL}}{\partial q_{2_Front}}, \frac{\partial z_{Stabi}}{\partial q_{1_Front}}, \frac{\partial z_{Stabi}}{\partial q_{2_Front}}$$

Inports

The following table shows the inports:

Name	Unit	Description
Displ_SteeringRod	[m]	Steering rod displacement
Displ_SteeringRod_dt	[m/s]	Speed of steering rod displacement
Displ_z_FL_Wheel	[m]	Vertical displacement of the front left wheel
Displ_z_FL_dt_Wheel	[m/s]	Speed of vertical displacement of front left wheel
Displ_z_FR_Wheel	[m]	Vertical displacement of the front right wheel
Displ_z_FR_dt_Wheel	[m/s]	Speed of vertical displacement of front right wheel
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (Gamma-Beta-Alpha) ▪ 2: ToeCamberSideview (Toe-Camber-Sideview)

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Collection of ASM signals
Displ_Damper_dt[FL;FR]	[m/s]	Speed of vertical displacement of damper joint points of front left and front right wheel
Displ_Damper_FL_dq_Front[q1;q2]	[]	Derivative of damper displacement of front left wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_Damper_FR_dq_Front[q1;q2]	[]	Derivative of damper displacement of front right wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_Spring[FL;FR][m|s]	[m]	Vertical displacement of spring joint points of front left and front right wheel
Displ_Spring_FL_dq_Front[q1;q2]	[]	Derivative of spring displacement of front left wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_Spring_FR_dq_Front[q1;q2]	[]	Derivative of spring displacement of front right wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Displ_Stabilizer	[m]	Vertical displacement of the stabilizer left joint point relative to the right joint point
Displ_Stabilizer_dq_Front[q1;q2]	[]	Derivative of stabilizer displacement of front wheels with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Pos_FL_Wheel[x,y,z]	[m]	Position vector of the front left wheel

Name	Unit	Description
Pos_FL_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of front left wheel with respect to steering rod displacement
Pos_FL_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of wheel position of front left wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Pos_FR_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of wheel position of front right wheel with respect to vertical wheel displacement of front left (q1) and front right (q2) wheel
Pos_FR_Wheel[x,y,z]	[m]	Position vector of the front right wheel
Pos_FR_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of front right wheel with respect to steering rod displacement
UnitVec_y_Wheel_FL_CoorSys_V[x;y;z]	[]	Unit vector in y direction of front left wheel
UnitVec_y_Wheel_FR_CoorSys_V[x;y;z]	[]	Unit vector in y direction of front right wheel
w_Rel_FL_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel angular speed of front left wheel with respect to steering rod speed
w_Rel_FL_Wheel_dq_Front_dt[q1;q2][x;y;z]	[]	Derivative of wheel angular speed of front left wheel with respect to vertical wheel speeds of front left (q1) and front right (q2) wheel
w_Rel_FR_Wheel_dq_Front_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of front right wheel with respect to vertical wheel speeds of front left (q1) and front right (q2) wheel
w_Rel_FR_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel orientation of front right wheel with respect to steering rod speed

Parameters

Note

- According to ASM naming conventions, parameter names that begin with "Map" are look-up tables. In Front Suspension Kinematics, these are 2-D look-up tables with steering rod displacement as the row index and vertical wheel displacement as the column index.
- The kinematics tables are defined only for the left side. The right side is mirrored.

The following table shows the parameters:

Name	Unit	Description
Const_TrackWidth_Front	[m]	Track width front
Map_Angle_Alpha_Wheel	[rad]	α angle of the wheel, i.e., wheel rotation around the x-axis relative to vehicle
Map_Angle_Beta_Wheel	[rad]	β angle of the wheel, i.e., wheel rotation around the y-axis relative to vehicle
Map_Angle_Gamma_Wheel	[rad]	γ angle of the wheel, i.e., wheel rotation around the z-axis relative to vehicle
Map_Displ_Damper	[m]	Vertical displacement of the damper joint point
Map_Displ_Spring	[m]	Vertical displacement of the spring joint point

Name	Unit	Description
Map_Displ_Stabilizer	[m]	Vertical displacement of the stabilizer joint point
Map_Displ_x_Wheel	[m]	x displacement of wheel center relative to vehicle
Map_Displ_y_Wheel	[m]	y displacement of wheel center relative to vehicle

Related topics

References

Body Geometry and Mass (ModelDesk Parameterizing 🕮)
Front Suspension Kinematics (Asymmetric)..... 163
SusKin Front V1 (ModelDesk Parameterizing 🕮)

Front Suspension Kinematics MacPherson Strut

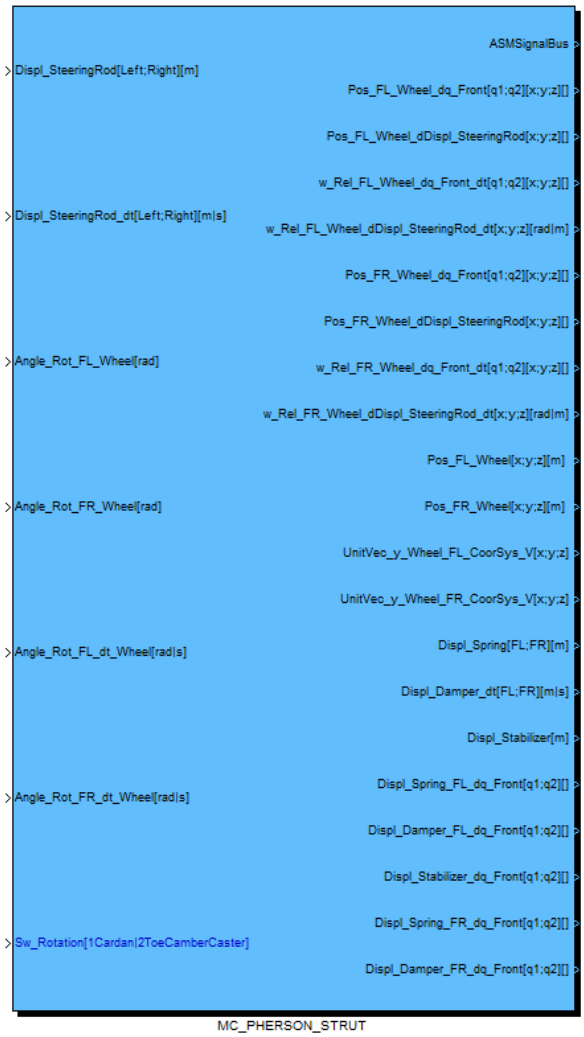
Description

The MC_PHERSON_STRUT system is a geometric description of the suspension kinematics. There are several linkage points of suspension to chassis and wheel carrier that have to be configured. The position of wheel midpoint, the orientation of the wheel, displacements of the spring, damper, and stabilizer, and several partial derivatives are calculated analytically within each simulation step. For a detailed description of calculated derivatives, refer to Front Suspension Kinematics (Symmetric) on page 174.

The MacPherson strut system can be used as an alternative to table-based suspension kinematics by replacing the table-based suspension systems with the MacPherson block. For further information on how to replace the table-based suspension kinematics system with the MacPherson strut and vice versa, refer to How to Change the Suspension Kinematics Model (ASM Vehicle Dynamics Model Description 🕮). The table-based compliance systems are superimposed on the analytically calculated MacPherson suspension.

Note

The suspension kinematics models are functions of two generalized coordinates, namely q_Front[q1,q2] for front suspension and q_Rear[q1,q2] for rear suspension. The table-based suspension kinematics are functions of the vertical wheel displacement of the left and right sides. The geometric MacPherson strut is a function of the rotation angle of the left and right sides. Thus, when replacing the table-based suspension kinematics by the geometric kinematics, you have to connect the q1 and q2 inports to Angle_Rot_FL and Angle_Rot_FR, respectively.



Tip

To visualize the geometrically suspension description, call `asm_plot_geosusp` in the MATLAB Command Window. This M-function plots each linkage point in the geometric suspension and shows the suspension as currently configured. 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 🕮).

Inports

The following table shows the inports:

Name	Unit	Description
Angle_Rot_FL_Wheel	[rad]	Rotation angle about rotation axis, i.e. axis between front connection of control arm to chassis and rear connection of control arm to chassis, of front left wheel
Angle_Rot_FL_dt_Wheel	[rad/s]	Speed of rotation angle about rotation axis, i.e. axis between front connection of control arm to chassis and rear connection of control arm to chassis, of front left wheel
Angle_Rot_FR_Wheel	[rad]	Rotation angle about rotation axis, i.e. axis between front connection of control arm to chassis and rear connection of control arm to chassis, of front right wheel
Angle_Rot_FR_dt_Wheel	[rad/s]	Speed of rotation angle about rotation axis, i.e. axis between front connection of control arm to chassis and rear connection of control arm to chassis, of front right wheel
Displ_SteeringRod	[m]	Steering rod displacement
Displ_SteeringRod_dt	[m/s]	Speed of steering rod displacement
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (Gamma-Beta-Alpha) ▪ 2: ToeCamberSideview (Toe-Camber-Sideview)

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Collection of ASM signals
Displ_Damper_dt[FL;FR]	[m/s]	Speed of vertical displacement of damper joint points of front left and front right wheel
Displ_Damper_FL_dq_Front[q1;q2]	[]	Derivative of damper displacement of front left wheel with respect to rotation angle around rotation axis of front left (q1) and front right (q2) wheel
Displ_Damper_FR_dq_Front[q1;q2]	[]	Derivative of damper displacement of front right wheel with respect to rotation angle around rotation axis of front left (q1) and front right (q2) wheel
Displ_Spring[FL;FR][m|s]	[m]	Vertical displacement of spring joint points of front left and front right wheel
Displ_Spring_FL_dq_Front[q1;q2]	[]	Derivative of spring displacement of front left wheel with respect to rotation angle around rotation axis of front left (q1) and front right (q2) wheel
Displ_Spring_FR_dq_Front[q1;q2]	[]	Derivative of spring displacement of front right wheel with respect to rotation angle around rotation axis of front left (q1) and front right (q2) wheel
Displ_Stabilizer	[m]	Vertical displacement of the stabilizer left joint point relative to the right joint point

Name	Unit	Description
Displ_Stabilizer_dq_Front[q1;q2]	[]	Derivative of stabilizer displacement of front wheels with respect to rotation angle around rotation axis of front left (q1) and front right (q2) wheel
Pos_FL_Wheel[x;y;z]	[m]	Position vector of front left wheel
Pos_FL_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of front left wheel with respect to steering rod displacement
Pos_FL_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of wheel position of front left wheel with respect to rotation angle around rotation axis of front left (q1) and front right (q2) wheel
Pos_FR_Wheel[x;y;z]	[m]	Position vector of front right wheel
Pos_FR_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of front right wheel with respect to steering rod displacement
Pos_FR_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of wheel position of front right wheel with respect to rotation angle around rotation axis of front left (q1) and front right (q2) wheel
UnitVec_y_Wheel_FL_CoorSys_V[x;y;z]	[]	Unit vector in y direction of front left wheel
UnitVec_y_Wheel_FR_CoorSys_V[x;y;z]	[]	Unit vector in y direction of front right wheel
w_Rel_FL_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel orientation of front left wheel with respect to speed of steering rod displacement
w_Rel_FL_Wheel_dq_Front_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of front left wheel with respect to rotation angle speed around rotation axis of front left (q1) and front right (q2) wheel
w_Rel_FR_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/s]	Derivative of wheel orientation of front right wheel with respect to speed of steering rod displacement
w_Rel_FR_Wheel_dq_Front_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of front right wheel with respect to rotation angle speed around rotation axis of front left (q1) and front right (q2) wheel

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Arm2Chassis_Front_Left	[m]	Position vector of connection of control arm to chassis on left side (front point)
Const_Arm2Chassis_Front_Right	[m]	Position vector of connection of control arm to chassis on right side (front point)
Const_Arm2Chassis_Rear_Left	[m]	Position vector of connection of control arm to chassis on left side (rear point)
Const_Arm2Chassis_Rear_Right	[m]	Position vector of connection of control arm to chassis on right side (rear point)
Const_Arm2Wheel_Left	[m]	Position vector of connection of control arm to wheel carrier on left side
Const_Arm2Wheel_Right	[m]	Position vector of connection of control arm to wheel carrier on right side

Name	Unit	Description
Const_Camber_Initial_Left	[deg]	Initial camber angle of left wheel (positive if wheel is inclined outward at the top)
Const_Camber_Initial_Right	[deg]	Initial camber angle of right wheel (positive if wheel is inclined outward at the top)
Const_StabiSelect	[]	Switch to select (1) or clear (0) stabilizer
Const_Stabi2Wheel_Left	[m]	Position vector of connection of stabilizer to wheel carrier on left side
Const_Stabi2Wheel_Right	[m]	Position vector of connection of stabilizer to wheel carrier on right side
Const_Steer2Steer_Left	[m]	Position vector of connection of steering rod to steering system on left side
Const_Steer2Steer_Right	[m]	Position vector of connection of steering rod to steering system on right side
Const_Steer2Wheel_Left	[m]	Position vector of connection of steering rod to wheel carrier on left side
Const_Steer2Wheel_Right	[m]	Position vector of connection of steering rod to wheel carrier on right side
Const_Strut2Chassis_Left	[m]	Position vector of connection of suspension strut to chassis on left side
Const_Strut2Chassis_Right	[m]	Position vector of connection of suspension strut to chassis on right side
Const_Strut2Wheel_Left	[m]	Position vector of connection of suspension strut to wheel carrier on left side
Const_Strut2Wheel_Right	[m]	Position vector of connection of suspension strut to wheel carrier on right side
Const_ToeIn_Initial_Left	[deg]	Initial toe-in angle of left wheel (positive if wheel front is turned towards the vehicle center line)
Const_ToeIn_Initial_Right	[deg]	Initial toe-in angle of right wheel (positive if wheel front is turned towards the vehicle center line)
Const_WheelCenter_Left	[m]	Position vector of front left wheel center
Const_WheelCenter_Right	[m]	Position vector of front right wheel center

Related topics

References

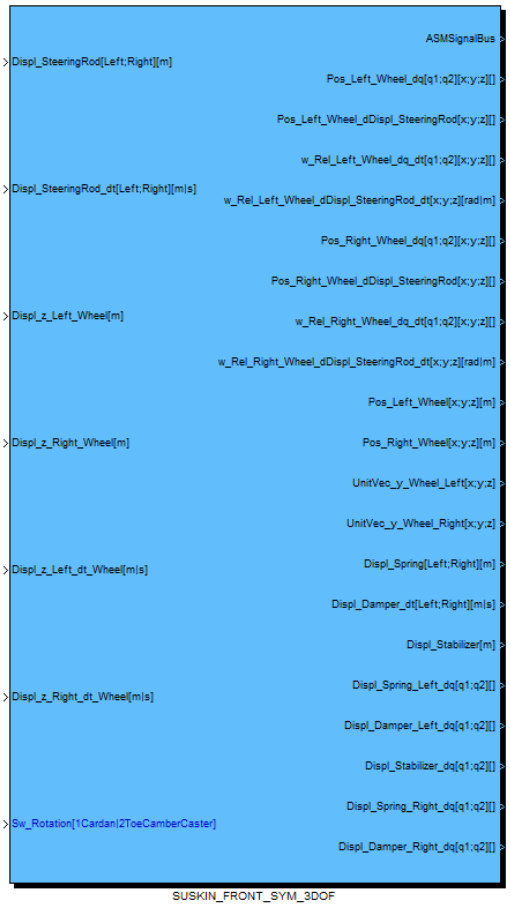
SusKin Front McPherson (ModelDesk Parameterizing 🕮)

Front Suspension Kinematics (Symmetric, 3DOF)

Description

This block describes the wheel suspension kinematics for the rear left and right wheels.These kinematics depend on both the wheel lift and the displacement of the steering rod.

The position of the wheel midpoint, the orientation of the wheel and displacements of the spring, damper and stabilizer are calculated in this block.



These kinematics are described as 2-D and 1-D look-up tables with three inputs:

- Steering rod position q_{steer}
- Wheel vertical displacement on left q_{left}
- Wheel vertical displacement on right q_{right}

These are the three degrees of freedom for each suspension.

The outputs of these look-up tables are:

- x displacement of wheel center
- y displacement of wheel center
- α angle, rotation around the x-axis
- β angle, rotation about y'-axis
- γ angle, rotation about z''-axis
- z_{spring}, displacement of spring in z direction
- z_{damper}, displacement of damper in z direction
- z_{stabi}, displacement of stabilizer in z direction

For example, for the left wheel, the relative wheel position change can be calculated as follows:

$$\Delta r_{left} = \Delta r_{left}(q_{left}, q_{right}) + \Delta r_{left}(q_{steer})$$

The total relative wheel position can be calculated as follows:

$$r_{left} = r_{left,0} + \Delta r_{left}$$

where $r_{left,0}$ is the position of the wheel center at the zero position, at which the kinematical displacements are measured.

The equation for wheel orientation θ_{left} is:

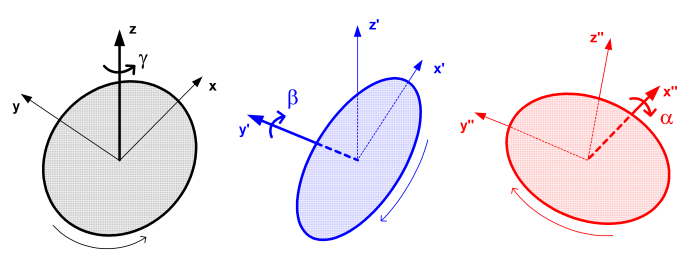
$$\theta_{left} = \theta_{left}(q_{left}, q_{right}) + \theta_{left}(q_{steer})$$

where

$\theta = [\alpha, \beta, \gamma]$ are the cardan angles of the wheel.

Note

The rotation sequence is rotation about the z- then y- then x-axis, i.e., RotZY'X". The following illustration shows the rotation sequence.



These angles are used to calculate the orientation of the wheel to the vehicle system:

$$^{V}T_{FL} = T_z(\gamma_{FL})T_y(\beta_{FL})T_x(\alpha_{FL})$$

The spring, damper, and stabilizer joint points also depend kinematically on the wheel center displacement and steering input. The main dependency is mostly on the wheel center vertical displacement and the steering rod displacement:

$$z_{spring} = z_{spring}(q_{left}, q_{right}) + z_{spring}(q_{steer})$$

$$z_{damper} = z_{damper}(q_{left}, q_{right}) + z_{damper}(q_{steer})$$

$$z_{stabi} = z_{stabi}(q_{left}, q_{right}) + z_{stabi}(q_{steer})$$

These displacements are used to calculate spring, damper and stabilizer forces.

To establish the equation of motion, the partial derivatives of the relation described above must be evaluated, i.e., changes in wheel position and orientation with respect to vertical wheel movement and steering input, i.e:

$$\frac{\partial r_{left}}{\partial q_{left}}, \frac{\partial r_{left}}{\partial q_{right}}, \frac{\partial r_{left}}{\partial q_{steer}}, \frac{\partial \omega_{left}}{\partial q_{left}}, \frac{\partial \omega_{left}}{\partial q_{right}}, \frac{\partial \omega_{left}}{\partial q_{steer}}$$

In addition, to calculate the suspension equivalent force in the direction of wheel center motion, the following derivatives must be evaluated:

$$\frac{\partial z_{spring}}{\partial q_{left}}, \frac{\partial z_{spring}}{\partial q_{right}}, \frac{\partial z_{spring}}{\partial q_{steer}}, \frac{\partial z_{damper}}{\partial q_{left}}, \frac{\partial z_{damper}}{\partial q_{right}}, \frac{\partial z_{damper}}{\partial q_{steer}}, \frac{\partial z_{stabi}}{\partial q_{left}}, \frac{\partial z_{stabi}}{\partial q_{right}}, \frac{\partial z_{stabi}}{\partial q_{steer}}$$

Inports

The following table shows the inports:

Name	Unit	Description
Displ_SteeringRod	[m]	Steering rod displacement
Displ_SteeringRod_dt	[m/s]	Speed of steering rod displacement
Displ_z_Left_Wheel	[m]	Vertical displacement of the left wheel
Displ_z_Right_Wheel	[m]	Vertical displacement of the right wheel
Displ_z_Left_dt_Wheel	[m/s]	Speed of vertical displacement of the left wheel
Displ_z_Right_dt_Wheel	[m/s]	Speed of vertical displacement of the right wheel
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (Gamma-Beta-Alpha) ▪ 2: ToeCamberSideview (Toe-Camber-Sideview)

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Collection of ASM signals
Displ_Damper_dt[Left;Right]	[m]	Speed of vertical displacement of damper joint points of left and right wheel
Displ_Damper_Left_dq[q1;q2]	[]	Derivative of damper displacement of left wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Damper_Right_dq[q1;q2]	[]	Derivative of damper displacement of right wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Spring[Left;Right]	[m-/s]	Vertical displacement of spring joint points of left and right wheel
Displ_Spring_Left_dq[q1;q2]	[]	Derivative of spring displacement of left wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Spring_Right_dq[q1;q2]	[]	Derivative of spring displacement of right wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel

Name	Unit	Description
Displ_Stabilizer	[]	Vertical displacement of the stabilizer left joint point relative to the right joint point
Displ_Stabilizer_dq[q1;q2]	[]	Derivative of stabilizer displacement of wheels with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Pos_Left_Wheel[x,y,z]	[m]	Position vector of the left wheel
Pos_Left_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of left wheel with respect to steering rod displacement
Pos_Left_Wheel_dq[q1;q2][x;y;z]	[]	Derivative of wheel position of left wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Pos_Right_Wheel[x,y,z]	[m]	Position vector of the right wheel
Pos_Right_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of right wheel with respect to steering rod displacement
Pos_Right_Wheel_dq[q1;q2][x;y;z]	[]	Derivative of wheel position of right wheel with respect to vertical wheel speeds of left (q1) and right (q2) wheel
UnitVec_y_Wheel_Left	[x;y;z]	Unit vector in y direction of left wheel
UnitVec_y_Wheel_Right	[x;y;z]	Unit vector in y direction of right wheel
w_Rel_Left_Wheel_dDispl_SteeringRod_dt[x;y;z]	[]	Derivative of wheel orientation of left wheel with respect to steering rod speed
w_Rel_Left_Wheel_dq_dt[q1;q2][x;y;z]	[rad/m]	Derivative of wheel angular speed of left wheel with respect to vertical wheel speeds of left (q1) and right (q2) wheel
w_Rel_Right_Wheel_dDispl_SteeringRod_dt[x;y;z]	[m]	Derivative of wheel orientation of right wheel with respect to steering rod speed
w_Rel_Right_Wheel_dq_dt[q1;q2][x;y;z]	[rad/m]	Derivative of wheel orientation of right wheel with respect to vertical wheel speeds of left (q1) and right (q2) wheel

Parameters

Note

- According to ASM naming conventions, parameter names that begin with "Map" are look-up tables. In Front Suspension Kinematics, these are 2-D look-up tables with vertical wheel lift on the left side as the row index and vertical wheel lift on the right side as the column index. The 1-D look-up table has steering rod displacement as the column index.
- The kinematics tables are defined only for the left side. The right side is mirrored.

The following table shows the parameters:

Name	Unit	Description
Const_Dist_LeftWheel	[m]	Position from vehicle coordinate system to left wheel
Map_Angle_Alpha_LeftWheel_q1q2	[deg]	Angle_Alpha_LeftWheel[deg]=f(Displ_z_Left_Wheel[m],Displ_z_Right_Wheel[m])
Map_Angle_Alpha_LeftWheel_q3	[deg]	Angle_Alpha_LeftWheel[deg]=f(Displ_SteeringRod[m])
Map_Angle_Beta_LeftWheel_q1q2	[deg]	Angle_Beta_LeftWheel[deg]=f(Displ_z_Left_Wheel[m],Displ_z_Right_Wheel[m])
Map_Angle_Beta_LeftWheel_q3	[deg]	Angle_Beta_LeftWheel[deg]=f(Displ_SteeringRod[m])
Map_Angle_Gamma_LeftWheel_q1q2	[deg]	Angle_Gamma_LeftWheel[deg]=f(Displ_z_Left_Wheel[m],Displ_z_Right_Wheel[m)
Map_Angle_Gamma_LeftWheel_q3	[deg]	Angle_Gamma_LeftWheel[deg]=f(Displ_Steering Rod[m])
Map_Displ_Damper_LeftWheel_q1q2	[m]	Displ_LeftDamper[m]=f(Displ_z_Left_Wheel[m],Displ_z_Right_Wheel[m)
Map_Displ_Damper_LeftWheel_q3	[m]	Displ_LeftDamper[m]=f(Displ_SteeringRod[m])
Map_Displ_Spring_LeftWheel_q1q2	[m]	Displ_LeftSpring[m]=f(Displ_z_Left_Wheel[m],Displ_z_Right_Wheel[m)
Map_Displ_Spring_LeftWheel_q3	[m]	Displ_LeftSpring[m]=f(Displ_SteeringRod[m])
Map_Displ_x_LeftWheel_q1q2	[m]	Displ_x_LeftWheel[m]=f(Displ_z_Left_Wheel[m], Displ_z_Right_Wheel[m])
Map_Displ_x_LeftWheel_q3	[m]	Displ_x_LeftWheel[m]=f(Displ_SteeringRod[m])
Map_Displ_y_LeftWheel_q1q2	[m]	Displ_y_LeftWheel[m]=f(Displ_z_Left_Wheel[m], Displ_z_Right_Wheel[m])
Map_Displ_y_LeftWheel_q3	[m]	Displ_y_LeftWheel[m]=f(Displ_SteeringRod[m])
Map_Displ_Stabi_LeftWheel_q1q2	[m]	Displ_LeftStabilizer[m]=f(Displ_z_Left_Wheel[m], Displ_z_Right_Wheel[m)
Map_Displ_Stabi_LeftWheel_q3	[m]	Displ_LeftStabilizer[m]=f(Displ_SteeringRod[m])

Related topics

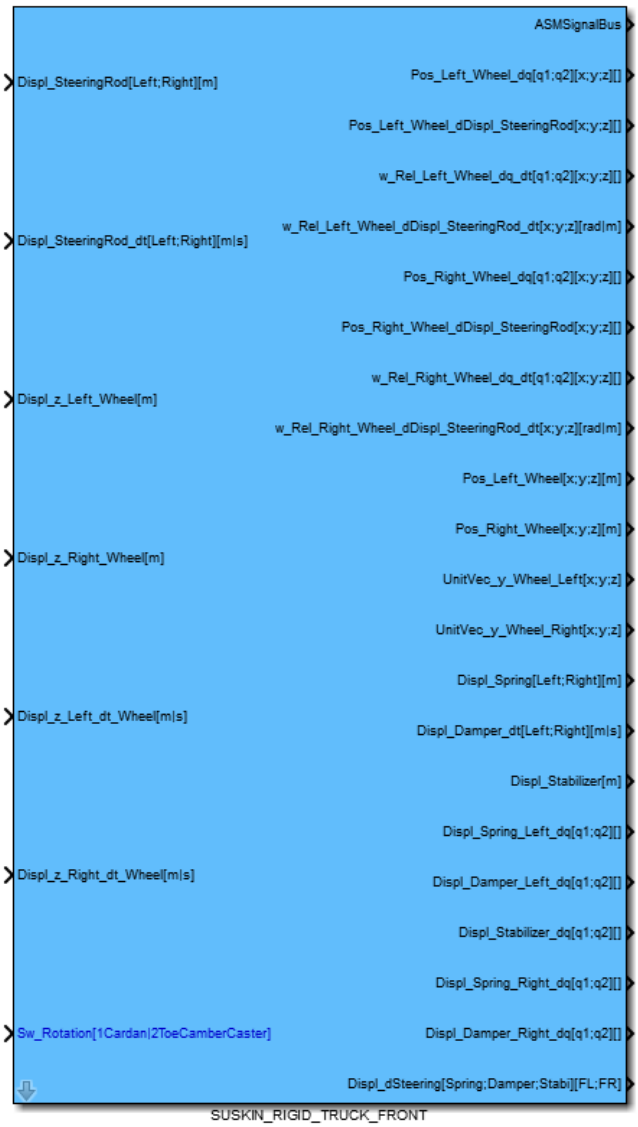
References

Front Suspension Kinematics (Asymmetric, 3DOF)..... 169
SusGeo Front Sym (ModelDesk Parameterizing 🕮)
SusGeo Rear 2nd Sym (ModelDesk Parameterizing 🕮)
SusGeo Rear 3rd Sym (ModelDesk Parameterizing 🕮)
SusGeo Rear Sym (ModelDesk Parameterizing 🕮)
SusKin Front Sym 3DOF (ModelDesk Parameterizing 🕮)
SusKin Rear 2nd Sym 3DOF (ModelDesk Parameterizing 🕮)
SusKin Rear 3rd Sym 3DOF (ModelDesk Parameterizing 🕮)
SusKin Rear Sym 3DOF (ModelDesk Parameterizing 🕮)

Suspension Kinematics Rigid Truck Axle

Description

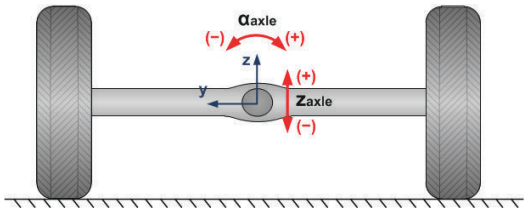
With the SUSKIN_RIGID_TRUCK_FRONT block, the kinematics of a rigid (solid) axle suspension kinematics can be simulated.



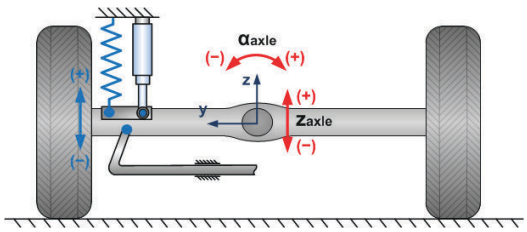
The rigid axle has two degrees of freedom (DOF) relative to the vehicle body:

- The first DOF is the vertical axle displacement.
- The second DOF is the axle roll (axle rotation around the x-axis).

The following illustration shows the rear view of the rigid axle. The two DOF (with sign convention) are indicated by the arrows at the top and at the right:



Because the main degree of freedom of the rigid axle is the vertical axle displacement and axle roll, the spring, damper, and stabilizer kinematics tables are defined as 2-D look-up tables depending on these two DOF.



Note

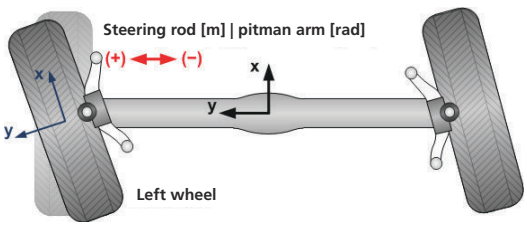
Sign convention: upward movement causes positive spring, damper and stabilizer displacement.

Note

The spring, damper and stabilizer kinematics tables are defined only for the left side. The right side is mirrored.

In addition to the mentioned two DOF (vertical axle displacement and axle roll), for a steerable rigid axle a third DOF is defined as one of the following:

- The steering rod displacement in [m] (if a rack-and-pinion steering system is used),
- A pitman arm rotation in [rad] (if a recirculating ball steering system is used).



Note

Sign convention: steering rod movement to the left (pitman arm rotation to the left) is positive displacement (rotation).

Inports

The following table shows the inports:

Name	Unit	Description
Displ_SteeringRod[Left;Right]	[m/s][rad/s]	Steering rod displacement [m] or pitman arm rotation [rad]
Displ_SteeringRod_dt[Left;Right]	[m/s][rad/s]	Steering rod velocity [m/s] or pitman arm rotation velocity [rad/s].
Displ_z_Left_Wheel	[m]	Vertical displacement of the left wheel
Displ_z_Left_dt_Wheel	[m/s]	Speed of vertical displacement of the left wheel
Displ_z_Right_Wheel	[m]	Vertical displacement of the right wheel
Displ_z_Right_dt_Wheel	[m/s]	Speed of vertical displacement of the right wheel
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Gamma-Beta-Alpha ▪ 2: Toe-Camber-Sideview

Note

The following rules apply for the left wheel:
- Toe = -gamma
- Camber = -alpha
- Sideview = -beta

Refer to Orientation Angles (ASM Vehicle Dynamics Addendum 🕮).

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Displ_Damper_dt[Left;Right]	[m/s]	Speed of vertical displacement of damper joint points of left and right wheel
Displ_Damper_Left_dq[q1;q2]	[]	Derivative of damper displacement of left wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Damper_Right_dq[q1;q2]	[]	Derivative of damper displacement of right wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Spring[Left;Right][m|s]	[m]	Vertical displacement of spring joint points of left and right wheel
Displ_Spring_Left_dq[q1;q2]	[]	Derivative of spring displacement of left wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Spring_Right_dq[q1;q2]	[]	Derivative of spring displacement of right wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel

Name	Unit	Description
Displ_Stabilizer	[m]	Vertical displacement of the stabilizer left joint point relative to the right joint point
Displ_Stabilizer_dq[q1;q2]	[]	Derivative of stabilizer displacement of wheels with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Pos_Left_Wheel[x,y,z]	[m]	Position vector of the left wheel
Pos_Left_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of wheel position of left wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Pos_Right_Wheel[x,y,z]	[m]	Position vector of the right wheel
Pos_Left_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of left wheel with respect to steering rod displacement
Pos_Right_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of wheel position of front right wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Pos_Right_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of right wheel with respect to steering rod displacement
UnitVec_y_Wheel_Left_CoorSys_V[x;y;z]	[]	Unit vector in y direction of left wheel
UnitVec_y_Wheel_Right_CoorSys_V[x;y;z]	[]	Unit vector in y direction of right wheel
w_Rel_Left_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of angular wheel speed of left wheel with respect to steering rod speed
w_Rel_Left_Wheel_dq_Front_dt[q1;q2][x;y;z]	[]	Derivative of angular wheel speed of left wheel with respect to vertical wheel speeds of left (q1) and right (q2) wheel
w_Rel_Right_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel orientation of right wheel with respect to steering rod speed
w_Rel_Right_Wheel_dq_Front_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of right wheel with respect to vertical wheel speeds of left (q1) and right (q2) wheel

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Angle_LeftWheel_Init	[deg]	Initial angle of the left wheel
Const_Pos_LeftWheel	[m]	Position from vehicle coordinate system to left wheel
Map_Angle_AlphaCamber_LeftWheel_q3	[deg]	α angle of left wheel as a function of steering rod displacement/pitman arm rotation
Map_Angle_AlphaCamber_RightWheel_q3	[deg]	α angle of right wheel as a function of steering rod displacement/pitman arm rotation
Map_Angle_Beta_Axle_q1q2	[deg]	β angle of axle center as a function of vertical axle displacement and axle α angle (rotation around the x-axis)

Name	Unit	Description
Map_Angle_BetaSideview_LeftWheel_q3	[deg]	β angle of left wheel as a function of steering rod displacement/pitman arm rotation
Map_Angle_BetaSideview_RightWheel_q3	[deg]	β angle of right wheel as a function of steering rod displacement/pitman arm rotation
Map_Angle_Gamma_Axle_q1q2	[deg]	γ angle of axle center as a function of vertical axle displacement and axle α angle (rotation around the x-axis)
Map_Angle_GammaToe_LeftWheel_q3	[deg]	γ angle of left wheel as a function of steering rod displacement/pitman arm rotation
Map_Angle_GammaToe_RightWheel_q3	[deg]	γ angle of right wheel as a function of steering rod displacement/pitman arm rotation
Map_Displ_Damper_AxleLeft_q1q2	[m]	Left damper displacement as a function of vertical axle displacement and axle α angle (rotation around the x-axis)
Map_Displ_Spring_AxleLeft_q1q2	[m]	Left spring displacement as a function of vertical axle displacement and axle α angle (rotation around the x-axis)
Map_Displ_Stabilizer_AxleLeft_q1q2	[m]	Left stabilizer displacement as a function of vertical axle displacement and axle α angle (rotation around the x-axis)
Map_Displ_x_Axle_q1q2	[m]	x displacement of axle center as a function of vertical axle displacement and axle α angle (rotation around the x-axis)
Map_Displ_x_LeftWheel_q3	[m]	x displacement of left wheel as a function of steering rod displacement/pitman arm rotation
Map_Displ_x_RightWheel_q3	[m]	x displacement of right wheel as a function of steering rod displacement/pitman arm rotation
Map_Displ_y_Axle_q1q2	[m]	y displacement of axle center as a function of vertical axle displacement and axle α angle (rotation around the x-axis)
Map_Displ_y_LeftWheel_q3	[m]	y displacement of left wheel as a function of steering rod displacement/pitman arm rotation
Map_Displ_y_RightWheel_q3	[m]	y displacement of right wheel as a function of steering rod displacement/pitman arm rotation

Related topics

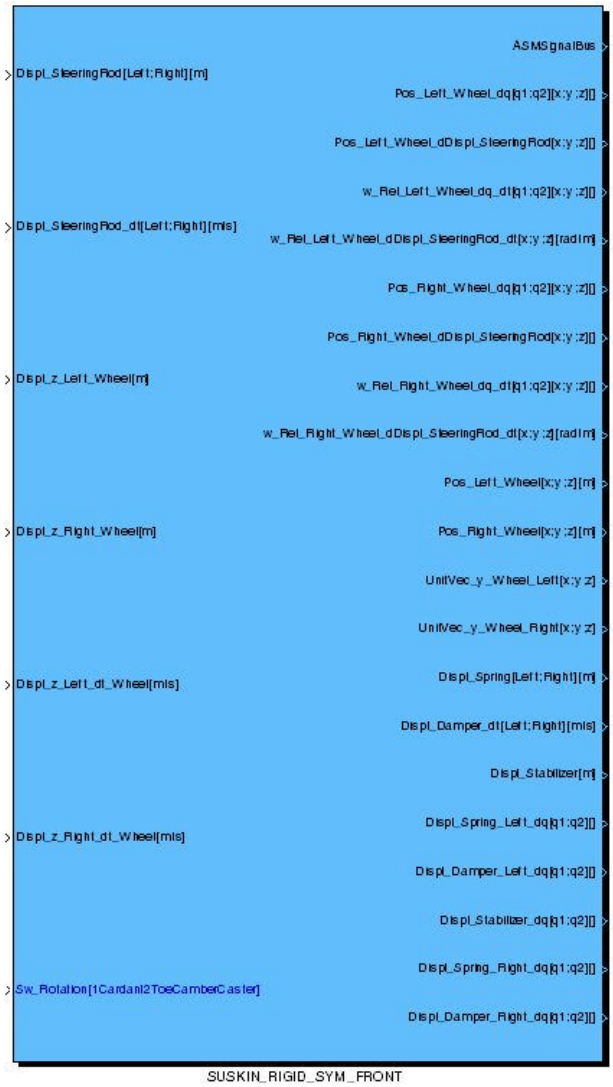
References

Trl SusGeo Rigid Truck Front (ModelDesk Parameterizing 🕮)
Trl SusGeo Rigid Truck Rear (ModelDesk Parameterizing 🕮)
Trl SusGeo Rigid Truck Rear 2nd (ModelDesk Parameterizing 🕮)
Trl SusGeo Rigid Truck Rear 3rd (ModelDesk Parameterizing 🕮)
Trl SusKin Rigid Truck Front (ModelDesk Parameterizing 🕮)
Trl SusKin Rigid Truck Rear (ModelDesk Parameterizing 🕮)
Trl SusKin Rigid Truck Rear 2nd (ModelDesk Parameterizing 🕮)
Trl SusKin Rigid Truck Rear 3rd (ModelDesk Parameterizing 🕮)

Suspension Kinematics Rigid Axle (Symmetric)

Description

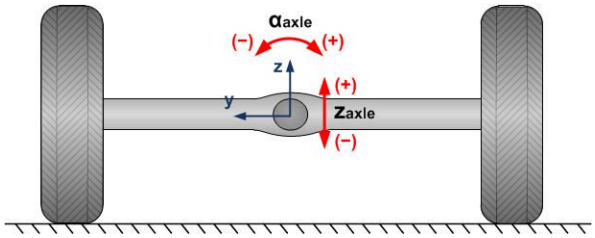
With this block, rigid (solid) axle suspension kinematics can be simulated.



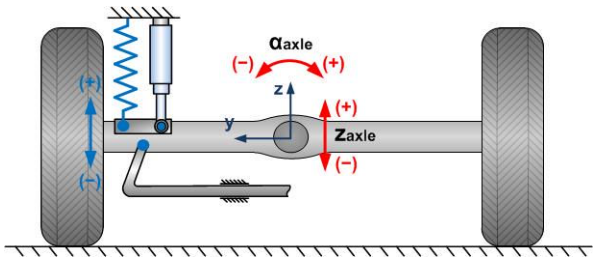
The rigid axle has two degrees of freedom (DOF) relative to the vehicle body.

- The axle vertical displacement
- The axle roll (axle rotation around the x-axis)

The following figure shows the rear view of the rigid axle. The two DOFs (with sign convention) are indicated by red arrows:



Since the main degrees of freedom of the rigid axle are the axle vertical displacement and axle roll, the spring, damper and stabilizer kinematics table are defined as functions of these DOFs in a 2-D look-up table.



Note

Sign convention: upward movement causes positive spring, damper and stabilizer displacement.

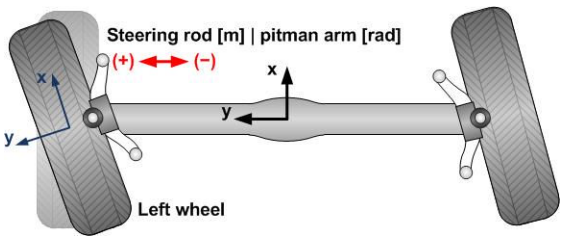
Note

The spring, damper and stabilizer kinematics tables are defined only for the left side. The right side is mirrored.

For a steerable rigid axle, a third DOF is defined in addition to the two DOFs mentioned (axle vertical displacement and axle roll):

- For a rack-and-pinion steering system: the steering rod displacement in [m]
- For a recirculating ball steering system: the pitman arm rotation in [rad]

See the following illustration:



Note

- Sign convention: steering rod movement to the left (pitman arm rotation to the left) is positive displacement (rotation).
- The wheel kinematics tables are defined only for the left side. The right side is mirrored.

Inports

The following table shows the inports:

Name	Unit	Description
Displ_SteeringRod[Left;Right]	[m]	Steering rod displacement
Displ_SteeringRod_dt[Left;Right]	[m/s]	Steering rod velocity
Displ_z_Left_Wheel	[m]	Vertical displacement of the left wheel
Displ_z_Left_dt_Wheel	[m/s]	Speed of vertical displacement of the left wheel
Displ_z_Right_Wheel	[m]	Vertical displacement of the right wheel
Displ_z_Right_dt_Wheel	[m/s]	Speed of vertical displacement of the right wheel
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (gamma-beta-alpha) ▪ 2: Toe-camber-sideview

Note

The following rules apply to the left wheel:

- Toe = -gamma
- Camber = -alpha
- Sideview = -beta

For more information about the toe, camber and sideview angles, refer to Orientation Angles (ASM Vehicle Dynamics Addendum 🕮).

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Displ_Damper_dt[Left;Right]	[m/s]	Speed of vertical displacement of damper joint points of left and right wheel
Displ_Damper_Left_dq[q1;q2]	[]	Derivative of damper displacement of left wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Damper_Right_dq[q1;q2]	[]	Derivative of damper displacement of right wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel

Name	Unit	Description
Displ_Spring[Left;Right][m|s]	[m]	Vertical displacement of spring joint points of left and right wheel
Displ_Spring_Left_dq[q1;q2]	[]	Derivative of spring displacement of left wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Spring_Right_dq[q1;q2]	[]	Derivative of spring displacement of right wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Displ_Stabilizer	[m]	Vertical displacement of the stabilizer left joint point relative to the right joint point
Displ_Stabilizer_dq[q1;q2]	[]	Derivative of stabilizer displacement of wheels with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Pos_Left_Wheel[x,y,z]	[m]	Position vector of the left wheel
Pos_Left_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of left wheel with respect to steering rod displacement
Pos_Left_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of wheel position of left wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
Pos_Right_Wheel[x,y,z]	[m]	Position vector of the right wheel
Pos_Right_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of right wheel with respect to steering rod displacement
Pos_Right_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of wheel position of front right wheel with respect to vertical wheel displacement of left (q1) and right (q2) wheel
UnitVec_y_Wheel_Left_CoorSys_V[x;y;z]	[]	Unit vector in y direction of left wheel
UnitVec_y_Wheel_Right_CoorSys_V[x;y;z]	[]	Unit vector in y direction of right wheel
w_Rel_Left_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel angular speed of left wheel with respect to steering rod speed
w_Rel_Left_Wheel_dq_Front_dt[q1;q2][x;y;z]	[]	Derivative of wheel angular speed of left wheel with respect to vertical wheel speeds of left (q1) and right (q2) wheel
w_Rel_Right_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel orientation of right wheel with respect to steering rod speed
w_Rel_Right_Wheel_dq_Front_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of right wheel with respect to vertical wheel speeds of left (q1) and right (q2) wheel

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Pos_LeftWheel	[m]	Position from vehicle coordinate system to left wheel
Const_Angle_LeftWheel_Init	[deg]	Initial angle of the left wheel
Map_Angle_AlphaCamber_LeftWheel_q3	[deg]	α angle of left wheel as a function of steering rod displacement/pitman arm rotation
Map_Angle_Beta_Axle_q1q2	[deg]	β angle of axle center as a function of axle vertical displacement and axle α angle (rotation around the x-axis)
Map_Angle_BetaSideview_LeftWheel_q3	[deg]	β angle of left wheel as a function of steering rod displacement/pitman arm rotation
Map_Angle_Gamma_Axle_q1q2	[deg]	γ angle of axle center as a function of axle vertical displacement and axle α angle (rotation around the x-axis)
Map_Angle_GammaToe_LeftWheel_q3	[deg]	γ angle of left wheel as a function of steering rod displacement/pitman arm rotation
Map_Displ_Damper_AxleLeft_q1q2	[m]	Left stabilizer displacement as a function of steering rod displacement/pitman arm rotation
Map_Displ_Damper_AxleLeft_q3	[m]	Left damper displacement as a function of steering rod displacement/pitman arm rotation
Map_Displ_Spring_AxleLeft_q1q2	[m]	Left spring displacement as a function of axle vertical displacement and axle α angle (rotation around the x-axis)
Map_Displ_Spring_AxleLeft_q3	[m]	Left spring displacement as a function of steering rod displacement/pitman arm rotation
Map_Displ_Stabilizer_AxleLeft_q1q2	[m]	Left stabilizer displacement as a function of axle vertical displacement and axle α angle (rotation around the x-axis)
Map_Displ_Stabilizer_AxleLeft_q3	[m]	Left stabilizer displacement as a function of steering rod displacement/pitman arm rotation
Map_Displ_x_Axle_q1q2	[m]	x displacement of axle center as a function of axle vertical displacement and axle α angle (rotation around the x-axis)
Map_Displ_x_LeftWheel_q3	[m]	x displacement of left wheel as a function of steering rod displacement/pitman arm rotation
Map_Displ_y_Axle_q1q2	[m]	y displacement of axle center as a function of axle vertical displacement and axle α angle (rotation around the x-axis)
Map_Displ_y_LeftWheel_q3	[m]	y displacement of left wheel as a function of steering rod displacement/pitman arm rotation

Related topics

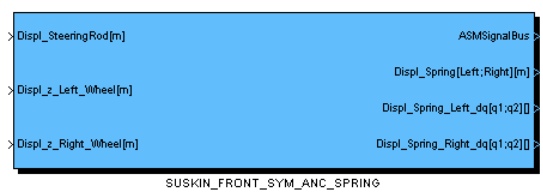
References

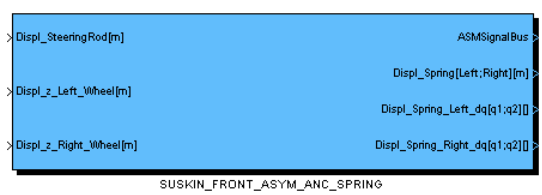
SusGeo Rigid Truck Front (ModelDesk Parameterizing 🕮)
SusGeo Rigid Truck Rear (ModelDesk Parameterizing 🕮)
SusGeo Rigid Truck Rear 2nd (ModelDesk Parameterizing 🕮)
SusKin Rigid Truck Front (ModelDesk Parameterizing 🕮)
SusKin Rigid Truck Rear (ModelDesk Parameterizing 🕮)
SusKin Rigid Truck Rear 2nd (ModelDesk Parameterizing 🕮)
SusKin Rigid Truck Rear 3rd (ModelDesk Parameterizing 🕮)

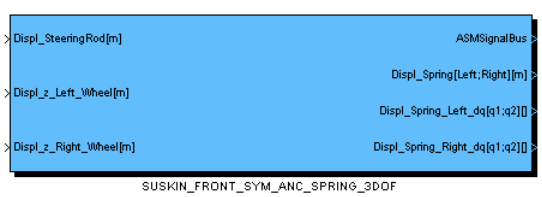
Front Suspension Kinematics Ancillary Spring

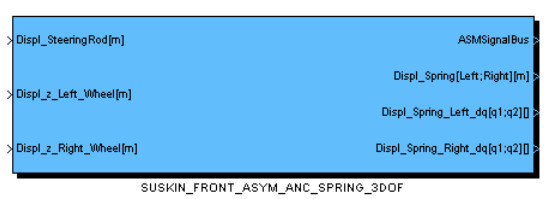
Description

The displacements of the ancillary springs are calculated in these blocks.









Depending on the used suspension type (symmetric/asymmetric, 3DOF) the kinematics is described as functions of 2-D look-up tables (depending on wheel

lift left/right side) or 2-D and 1-D look-up tables (depending on wheel lift left/right side and steering rod displacement), respectively:

- Symmetric:

  $$z_{AncSpring,\,Left} = f\left(u_{SteeringRod}, q_{Left}\right)$$

- Asymmetric:

  $$z_{AncSpring,\,Left} = f\left(u_{SteeringRod}, q_{Left}\right)$$

  $$z_{AncSpring,\,Right} = f\left(u_{SteeringRod}, q_{Right}\right)$$

- 3DOF symmetric:

  $$z_{AncSpring,\,Left} = f\left(q_{Left}, q_{Right}\right) + f\left(u_{SteeringRod}\right)$$

- 3DOF asymmetric:

  $$z_{AncSpring,\,Left} = f\left(q_{Left}, q_{Right}\right) + f\left(u_{SteeringRod}\right)$$

  $$z_{AncSpring,\,Right} = f\left(q_{Right}, q_{Left}\right) + f\left(u_{SteeringRod}\right)$$

Inports

The following table shows the inports.

Name	Unit	Description
Displ_SteeringRod	[m]	Steering rod displacement (uS)
Displ_z_Left_Wheel	[m]	Vertical displacement of the left wheel (q1)
Displ_z_Right_Wheel	[m]	Vertical displacement of the right wheel (q2)

Outports

The following table shows the outports.

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Displ_Spring[Left;Right]	[m]	Relative displacement of the ancillary spring joint points on left and right sides
Displ_Spring_Left_dq[q1;q2]	[]	Change of ancillary spring force left with respect to vertical wheel displacement of left (q1) and right (q2) wheels
Displ_Spring_Right_dq[q1;q2]	[]	Change of ancillary spring force right with respect to vertical wheel displacement of left (q1) and right (q2) wheels

Parameters for symmetric type

The following table shows the parameters for the symmetric type.

Name	Unit	Description
Map_Displ_LeftSpring	[m]	Displ_LeftSpring[m] = f(uS[m], q1[m])

Parameters for asymmetric type

The following table shows the parameters for the asymmetric type.

Name	Unit	Description
Map_Displ_LeftSpring	[m]	Displ_LeftSpring[m] = f(uS[m], q1[m])
Map_Displ_RightSpring	[m]	Displ_RightSpring[m] = f(uS[m], q2[m])

Parameters for 3DOF symmetric type

The following table shows the parameters for the 3DOF symmetric type.

Name	Unit	Description
Map_Displ_AncSpring_LeftWheel_q1q2	[m]	Displ_LeftSpring[m] = f(q1[], q2[])
Map_Displ_AncSpring_LeftWheel_q3	[m]	Displ_LeftSpring[m] = f(uS[m])

Parameters for 3DOF asymmetric type

The following table shows the parameters for the 3DOF asymmetric type.

Name	Unit	Description
Map_Displ_AncSpring_LeftWheel_q1q2	[m]	Displ_LeftSpring[m] = f(q1[], q2[])
Map_Displ_AncSpring_LeftWheel_q3	[m]	Displ_LeftSpring[m] = f(uS[m])
Map_Displ_AncSpring_RightWheel_q1q2	[m]	Displ_RightSpring[m] = f(q2[], q1[])
Map_Displ_AncSpring_RightWheel_q3	[m]	Displ_RightSpring[m] = f(uS[m])

Related topics

References

SusKin Front Asym (ModelDesk Parameterizing 🕮)
SusKin Front Asym 3DOF (ModelDesk Parameterizing 🕮)
SusKin Front Sym 3DOF (ModelDesk Parameterizing 🕮)
SusKin Front V1 (ModelDesk Parameterizing 🕮)
SusKin Rear Asym (ModelDesk Parameterizing 🕮)
SusKin Rear Asym 3DOF (ModelDesk Parameterizing 🕮)
SusKin Rear Sym (ModelDesk Parameterizing 🕮)
SusKin Rear Sym 3DOF (ModelDesk Parameterizing 🕮)

Rear

Where to go from here

Information in this section

Suspension Kinematics.....202
To describe the blocks of the subsystem.

Common Suspension Blocks.....225
To describe the blocks of the subsystem.

Suspension Kinematics

Where to go from here

Information in this section

Rear Suspension Kinematics (Asymmetric).....203
The position of the wheel midpoint, the orientation of the wheel and displacement of the spring, damper and stabilizer are calculated in this block for each wheel separately via a set of look-up tables.

Rear Suspension Kinematics (Symmetric).....209
The position of the wheel midpoint, the orientation of the wheel and displacement of the spring, damper and stabilizer are calculated in this block.

Rear Suspension Kinematics Semi-Trailing Arm.....213
This block describes the wheel suspension kinematics for a Semi-Trailing-Arm suspension. These kinematics depend on the rotation angle of the Semi-Trailing-Arm.

Rear Suspension Kinematics Rigid Axle.....218
The RIGID_AXLE system describes suspension kinematics in geometrical way. There are several linkage points of suspension to chassis and wheel carrier that have to be configured.

Rear Suspension Kinematics (Asymmetric, 3DOF).....222
The position of the wheel midpoint, the orientation of the wheel and displacements of the spring, damper and stabilizer are calculated in this block.

Rear Suspension Kinematics (Symmetric, 3DOF).....222
The position of the wheel midpoint, the orientation of the wheel and displacements of the spring, damper and stabilizer are calculated in this block.

Suspension Force Kinematics Rear..... 222
This block projects the spring, damper and stabilizer forces in the direction of the axle degrees of freedom. The projection is carried out according to the kinematic relations described in the KINEMATICS block.

Rear Suspension Kinematics Ancillary Spring..... 225
The displacements of the ancillary springs are calculated in these blocks.

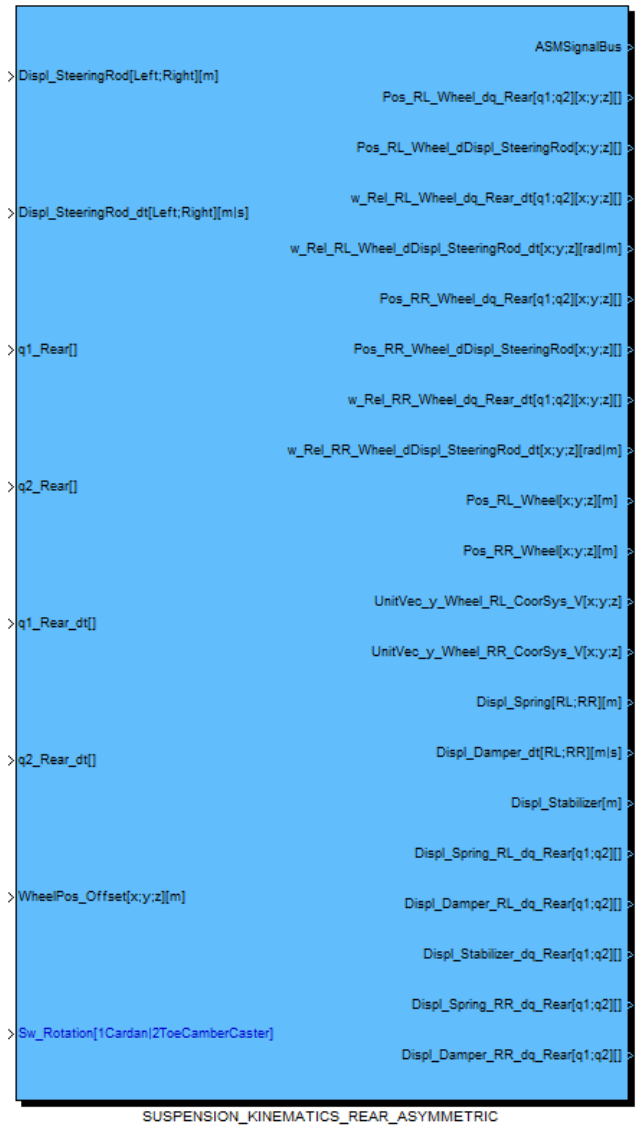
Rear Suspension Kinematics (Asymmetric)

Description

In the Rear Suspension Kinematics (Symmetric) subsystem, the wheel motion is defined via look-up tables. The tables are only defined for the left wheel. The right wheel is mirrored. In this subsystem, however, each wheel is defined separately via a set of look-up tables.

Note

The block is implemented in such a way that the degrees of freedom are configurable and generalized, for example, they need not to be the wheel vertical motion and steering. These features are not yet supported via ModelDesk. Only the asymmetry functionality can be used via ModelDesk. In the standard configurations q1_Rear is z_RL_Wheel (i.e. rear left wheel vertical displacement) and q2_Rear is z_RR_Wheel (i.e. rear left wheel vertical displacement).



These dependencies are described as 2-D look-up tables with two inputs:

- Wheel vertical displacement q_W
- Wheel vertical displacement at the opposite side $q_{W_Opposite}$

These three inputs are the degrees of freedom used for the front suspension. The output of look-up tables are:

- x displacement of wheel center
- y displacement of wheel center
- α angle, rotation around the x-axis
- β angle, rotation around the y-axis
- γ angle, rotation around the z-axis

- z_{sp} displacement of spring in z direction
- z_d displacement of damper in z direction
- z_{st} displacement of stabilizer in z direction

For example, for the rear left wheel and its relative wheel displacement

$$\Delta r_{RL} = \Delta r_{RL}(q_{RL}, q_{RR})$$

the wheel position can be calculated as follows:

$$r_{RL} = r_{RL0} + \Delta r_{RL}$$

where r_{RL0} is the position of wheel center at the zero position, at which the kinematical displacements are measured.

The equation for wheel orientation θ_{RL} is:

$$\theta_{RL} = \theta_{RL}(q_{RL}, q_{RR})$$

where

$\theta_{RL} = [\alpha_{RL}, \beta_{RL}, \gamma_{RL}]$ are the cardan angles of the wheel

Note

The rotation sequence is rotation about the z- then y- then x-axis, i.e., RotZY'X".

The orientation of the wheel to the vehicle system can be calculated as follows:

$$^{V}T_{RL} = T_z(\gamma_{RL})\, T_y(\beta_{RL})\, T_x(\alpha_{RL})$$

The spring, damper, and stabilizer joint points also depend kinematically on the wheel center displacement and steering input. The main dependency is mostly on the wheel center vertical displacement.

$$z_{sp} = z_{sp}(q_{RL}, q_{RR})$$

$$z_d = z_d(q_{RL}, q_{RR})$$

$$z_{stab} = z_{stab}(q_{RL}, q_{RR})$$

These displacements are used to calculate spring, damper, and stabilizer forces.

To establish the equation of motion, the partial derivatives of the relations described above must be evaluated, i.e., changes in wheel position and orientation with respect to the vertical wheel movement and steering input.

$$\frac{\partial r_{RL}}{\partial q_{1_Rear}}, \frac{\partial r_{RL}}{\partial q_{2_Rear}}, \frac{\partial \omega_{Rl}}{\partial \dot{q}_{1_Rear}}, \frac{\partial \omega_{RL}}{\partial \dot{q}_{2_Rear}}$$

To calculate the suspension equivalent force in the direction of wheel center motion, the following derivative must be evaluated:

$$\frac{\partial z_{Sp_RL}}{\partial q_{1_Rear}}, \frac{\partial z_{Sp_RL}}{\partial q_{2_Rear}}, \frac{\partial z_{D_RL}}{\partial q_{1_Rear}}, \frac{\partial z_{D_RL}}{\partial q_{2_Rear}} \frac{\partial z_{Stabi}}{\partial q_{1_Rear}}, \frac{\partial z_{Stabi}}{\partial q_{2_Rear}}$$

Tip

The tables can also be defined to depend only on the wheel's own vertical movement.

Inports

The following table shows the inports:

Name	Unit	Description
Displ_SteeringRod	[m]	Steering rod displacement
Displ_SteeringRod_dt	[m/s]	Speed of steering rod displacement
q1_Rear	[]	First generalized coordinate for rear suspension
q1_Rear_dt	[]	Derivative of first generalized coordinate for rear suspension with respect to time
q2_Rear	[]	Second generalized coordinate for rear suspension
q2_Rear_dt	[]	Derivative of second generalized coordinate for rear suspension with respect to time
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (Gamma-Beta-Alpha) ▪ 2: ToeCamberSideview (Toe-Camber-Sideview)
Wheel_Pos_Offset	[m]	Additional offset for initial wheel position for left wheel

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Collection of ASM signals
Displ_Damper_dt[RL;RR]	[m/s]	Speed of vertical displacement of damper joint points of rear left and rear right wheel
Displ_Damper_RL_dq_Rear[q1;q2]	[]	Derivative of damper displacement of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Damper_RR_dq_Rear[q1;q2]	[]	Derivative of damper displacement of rear right wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Spring[RL;RR][m|s]	[m]	Vertical displacement of spring joint points of rear left and rear right wheel
Displ_Spring_RL_dq_Rear[q1;q2]	[]	Derivative of spring displacement of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Spring_RR_dq_Rear[q1;q2]	[]	Derivative of spring displacement of rear right wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel

Name	Unit	Description
Displ_Stabilizer	[m]	Vertical displacement of the stabilizer left joint point relative to the right joint point
Displ_Stabilizer_dq_Rear[q1;q2]	[]	Derivative of stabilizer displacement of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Pos_RL_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of rear left wheel with respect to steering rod displacement
Pos_RL_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Derivative of wheel position of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Pos_RR_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of rear right wheel with respect to steering rod displacement
Pos_RR_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Derivative of wheel position of rear right wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Pos_RL_Wheel[x;y;z]	[m]	Position vector of rear left wheel
Pos_RR_Wheel[x;y;z]	[m]	Position vector of rear right wheel
UnitVec_y_Wheel_RL_CoorSys_V[x;y;z]	[]	Unit vector in y direction of rear left wheel
UnitVec_y_Wheel_RR_CoorSys_V[x;y;z]	[]	Unit vector in y direction of rear right wheel
w_Rel_RL_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel orientation of rear left wheel with respect to steering rod displacement
w_Rel_RL_Wheel_dq_Rear_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
w_Rel_RR_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/s]	Derivative of wheel orientation of rear right wheel with respect to steering rod displacement
w_Rel_RR_Wheel_dq_Rear_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of rear right wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel

Parameters

Note

- According to ASM naming conventions, parameter names that begin with "Map" are look-up tables. In Rear Suspension Kinematics, these are 2-D with vertical wheel displacement at the opposite side as the row index and the vertical wheel's own displacement as the column index. In cases where that the kinematics depends only on the wheel's own vertical displacement, you should define a 2-D look-up table with no dependency in the direction of the opposite wheel.
- The kinematics tables are defined only for the left side. The right side is mirrored.

The following table shows the parameters:

Name	Unit	Description
Const_Pos_z0_Rear	[m]	Position offset vector, i.e. the zero position at which the kinematics tables are measured
Const_TrackWidth_Rear	[m]	Track width rear
Const_WheelBase	[m]	Distance from front to rear axle
Map_Angle_Alpha_Wheel_Left	[rad]	α angle of left wheel, i.e. left wheel rotation around the x-axis relative to vehicle
Map_Angle_Alpha_Wheel_Right	[rad]	α angle of right wheel, i.e. right wheel rotation around the x-axis
Map_Angle_Beta_Wheel_Left	[rad]	β angle of left wheel, i.e. left wheel rotation around the y-axis relative to vehicle
Map_Angle_Beta_Wheel_Right	[rad]	β angle of right wheel, i.e. right wheel rotation around the y-axis
Map_Angle_Gamma_Wheel_Left	[rad]	γ angle of left wheel, i.e. left wheel rotation around the z-axis relative to vehicle
Map_Angle_Gamma_Wheel_Right	[rad]	γ angle of right wheel, i.e. right wheel rotation around the z-axis
Map_Displ_Damper_Left	[m]	Vertical displacement of left damper joint
Map_Displ_Damper_Right	[m]	Vertical displacement of right damper joint
Map_Displ_Spring_Left	[m]	Vertical displacement of left spring joint
Map_Displ_Spring_Right	[m]	Vertical displacement of right spring joint
Map_Displ_Stabilizer_Left	[m]	Vertical displacement of left stabilizer joint
Map_Displ_Stabilizer_Right	[m]	Vertical displacement of right stabilizer joint
Map_Displ_x_Wheel_Left	[m]	X displacement of left wheel center relative to vehicle
Map_Displ_x_Wheel_Right	[m]	X displacement of right wheel center
Map_Displ_y_Wheel_Left	[m]	Y displacement of left wheel center relative to vehicle
Map_Displ_y_Wheel_Right	[m]	Y displacement of right wheel center

Related topics

References

Rear Suspension Kinematics (Symmetric)..... 209
SusKin Rear Asym (ModelDesk Parameterizing 🕮)
SusKin Rear V1 (ModelDesk Parameterizing 🕮)

Rear Suspension Kinematics (Symmetric)

Description

The position of the wheel midpoint, the orientation of the wheel and displacement of the spring, damper and stabilizer are calculated in this block.

These dependencies are described as 2-D look-up tables with two inputs:

- Wheel vertical displacement q_W
- Wheel vertical displacement at the opposite side $q_{W_Opposite}$

These three inputs are the degrees of freedom used for the front suspension. The output of look-up tables are:

- x displacement of wheel center
- y displacement of wheel center
- α angle, rotation around the x-axis
- β angle, rotation around the y-axis
- γ angle, rotation around the z-axis
- z_{sp} displacement of spring in z direction
- z_d displacement of damper in z direction
- z_{st} displacement of stabilizer in z direction

For example, for the rear left wheel and its relative wheel displacement

$$\Delta r_{RL} = \Delta r_{RL}(q_{RL}, q_{RR})$$

the wheel position can be calculated as follows:

$$r_{RL} = r_{RL0} + \Delta r_{RL}$$

where r_{RL0} is the position of wheel center at the zero position, at which the kinematical displacements are measured.

The equation for wheel orientation θ_{RL} is:

$$\theta_{RL} = \theta_{RL}(q_{RL}, q_{RR})$$

where

$\theta_{RL} = [\alpha_{RL}, \beta_{RL}, \gamma_{RL}]$ are the cardan angles of the wheel

Note

The rotation sequence is rotation about the z- then y- then x-axis, i.e., RotZY'X".

The orientation of the wheel to the vehicle system can be calculated as follows:

$${}^{V}T_{RL} = T_z(\gamma_{RL})\, T_y(\beta_{RL})\, T_x(\alpha_{RL})$$

The spring, damper, and stabilizer joint points also depend kinematically on the wheel center displacement and steering input. The main dependency is mostly on the wheel center vertical displacement.

$$z_{sp} = z_{sp}(q_{RL}, q_{RR})$$

$$z_d = z_d(q_{RL}, q_{RR})$$

$$z_{stab} = z_{stab}(q_{RL}, q_{RR})$$

These displacements are used to calculate spring, damper, and stabilizer forces.

To establish the equation of motion, the partial derivatives of the relations described above must be evaluated, i.e., changes in wheel position and orientation with respect to the vertical wheel movement and steering input.

$$\frac{\partial r_{RL}}{\partial q_{1_Rear}}, \frac{\partial r_{RL}}{\partial q_{2_Rear}}, \frac{\partial \omega_{Rl}}{\partial \dot{q}_{1_Rear}}, \frac{\partial \omega_{RL}}{\partial \dot{q}_{2_Rear}}$$

To calculate the suspension equivalent force in the direction of wheel center motion, the following derivative must be evaluated:

$$\frac{\partial z_{Sp_RL}}{\partial q_{1_Rear}}, \frac{\partial z_{Sp_RL}}{\partial q_{2_Rear}}, \frac{\partial z_{D_RL}}{\partial q_{1_Rear}}, \frac{\partial z_{D_RL}}{\partial q_{2_Rear}} \frac{\partial z_{Stabi}}{\partial q_{1_Rear}}, \frac{\partial z_{Stabi}}{\partial q_{2_Rear}}$$

Tip

The tables can also be defined to depend only on the wheel's own vertical movement.

Inports

The following table shows the inports:

Name	Unit	Description
Displ_SteeringRod	[m]	Steering rod displacement
Displ_SteeringRod_dt	[m/s]	Speed of steering rod displacement
Displ_z_RL_Wheel	[m]	Vertical displacement of rear left wheel
Displ_z_RL_dt_Wheel	[m/s]	Speed of vertical displacement of rear left wheel
Displ_z_RR_dt_Wheel	[m/s]	Speed of vertical displacement of rear right wheel
Displ_z_RR_Wheel	[m]	Vertical displacement of rear right wheel
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (Gamma-Beta-Alpha) ▪ 2: ToeCamberSideview (Toe-Camber-Sideview)
Wheel_Pos_Offset	[m]	Additional offset for initial wheel position for left wheel

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Collection of ASM signals
Displ_Damper_dt[RL;RR]	[m/s]	Speed of vertical displacement of damper joint points of rear left and rear right wheel
Displ_Damper_RL_dq_Rear[q1;q2]	[]	Derivative of damper displacement of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Damper_RR_dq_Rear[q1;q2]	[]	Derivative of damper displacement of rear right wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Spring[RL;RR]	[m]	Vertical displacement of spring joint points of rear left and rear right wheel

Name	Unit	Description
Displ_Spring_RL_dq_Rear[q1;q2]	[]	Derivative of spring displacement of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Spring_RR_dq_Rear[q1;q2]	[]	Derivative of spring displacement of rear right wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Stabilizer	[m]	Vertical displacement of the stabilizer left joint point relative to the right joint point
Displ_Stabilizer_dq_Rear[q1;q2]	[]	Derivative of stabilizer displacement of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Pos_FL_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of rear left wheel with respect to steering rod displacement
Pos_RL_Wheel[x;y;z]	[m]	Position vector of rear left wheel
Pos_RL_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Derivative of wheel position of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Pos_RR_Wheel[x;y;z]	[m]	Position vector of rear right wheel
Pos_RR_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of rear right wheel with respect to steering rod displacement
Pos_RR_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Derivative of wheel position of rear right wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
UnitVec_y_Wheel_RL_CoorSys_V[x;y;z]	[]	Unit vector in y direction of rear left wheel
UnitVec_y_Wheel_RR_CoorSys_V[x;y;z]	[]	Unit vector in y direction of rear right wheel
w_Rel_RL_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel orientation of rear left wheel with respect to steering rod displacement
w_Rel_RL_Wheel_dq_Rear_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
w_Rel_RR_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel orientation of rear right wheel with respect to steering rod displacement
w_Rel_RR_Wheel_dq_Rear_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of rear right wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel

Parameters

Note

- According to ASM naming conventions, parameter names that begin with "Map" are look-up tables. In Rear Suspension Kinematics, these are 2-D with vertical wheel displacement at the opposite side as the row index and the vertical wheel's own displacement as the column index. In cases where that the kinematics depends only on the wheel's own vertical displacement, you should define a 2-D look-up table with no dependency in the direction of the opposite wheel.
- The kinematics tables are defined only for the left side. The right side is mirrored.

The following table shows the parameters:

Name	Unit	Description
Const_Pos_z0_Rear	[m]	Position offset vector, i.e., the zero position at which the kinematics tables are measured
Const_TrackWidth_Rear	[m]	Track width rear, i.e., distance from left to right wheel centers
Const_WheelBase	[m]	Distance from front to rear axle
Map_Angle_Alpha_Wheel	[rad]	α angle of the wheel, i.e., wheel rotation around the x-axis relative to vehicle
Map_Angle_Beta_Wheel	[rad]	β angle of the wheel, i.e., wheel rotation around the y-axis relative to vehicle
Map_Angle_Gamma_Wheel	[rad]	γ angle of the wheel, i.e., wheel rotation around the z-axis relative to vehicle
Map_Displ_Damper	[m]	Vertical displacement of the damper joint point
Map_Displ_Spring	[m]	Vertical displacement of the spring joint point
Map_Displ_Stabilizer	[m]	Vertical displacement of the stabilizer joint point
Map_Displ_x_Wheel	[m]	x displacement of wheel center relative to vehicle
Map_Displ_y_Wheel	[m]	y displacement of wheel center relative to vehicle

Related topics

References

Body Geometry and Mass (ModelDesk Parameterizing 🕮)
SusKin Rear Sym (ModelDesk Parameterizing 🕮)

Rear Suspension Kinematics Semi-Trailing Arm

Description

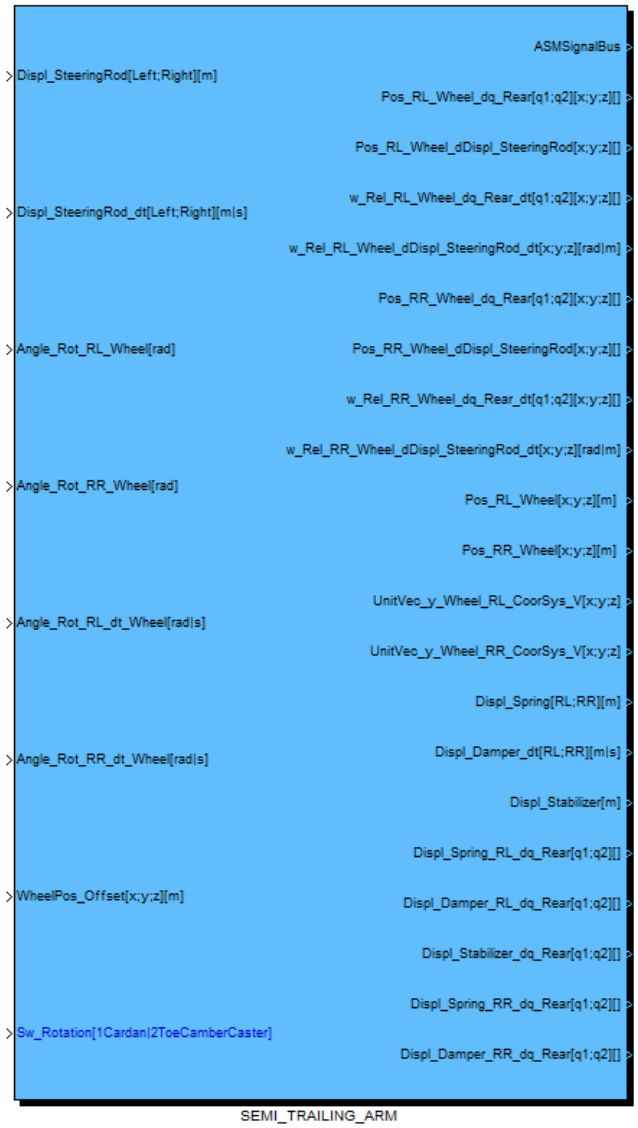
This block describes the wheel suspension kinematics for a Semi-Trailing-Arm suspension. These kinematics depend on the rotation angle of the Semi-Trailing-Arm.

The SEMI_TRAILING_ARM system describes suspension kinematics in geometrical way. There are several linkage points of suspension to chassis and

wheel carrier that have to be configured. Position of wheel midpoint, orientation of wheel, displacements of spring, damper and stabilizer and several partial derivatives are calculated analytically within each simulation step. For a detailed description on calculated derivatives, refer to Rear Suspension Kinematics (Symmetric) on page 209.

The **SEMI_TRAILING_ARM** system can be used alternatively to table-based suspension kinematics by replacing the table-based suspension systems with the **SEMI_TRAILING_ARM** block. For further information on how to replace table-based suspension kinematics system with semi-trailing arm and vice versa, refer to How to Change the Suspension Kinematics Model (ASM Vehicle Dynamics Model Description 🕮).

The table-based compliance systems are superimposed to the analytically calculated semi-trailing arm suspension.



Tip

For visualizing of geometrically described suspension, call `asm_plot_geosusp` at 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` at the MATLAB Command Window or refer to asm_plot_geosusp (ASM User Guide 🕮).

Inports

The following table shows the inports:

Name	Unit	Description
Angle_Rot_RL_dt_Wheel	[rad/s]	Speed of rotation angle about rotation axis, i.e. axis between front connection of control arm to chassis and rear connection of control arm to chassis, of rear left wheel
Angle_Rot_RL_Wheel	[rad]	Rotation angle about rotation axis, i.e. axis between front connection of control arm to chassis and rear connection of control arm to chassis, of rear left wheel
Angle_Rot_RR_dt_Wheel	[rad/s]	Speed of rotation angle about rotation axis, i.e. axis between front connection of control arm to chassis and rear connection of control arm to chassis, of rear right wheel
Angle_Rot_RR_Wheel	[rad]	Rotation angle about rotation axis, i.e. axis between front connection of control arm to chassis and rear connection of control arm to chassis, of rear right wheel
Displ_SteeringRod	[m]	Steering rod displacement
Displ_SteeringRod_dt	[m/s]	Speed of steering rod displacement
Wheel_Pos_Offset	[m]	Additional offset for initial wheel position for left wheel
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (Gamma-Beta-Alpha) ▪ 2: ToeCamberSideview (Toe-Camber-Sideview)

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Collection of ASM signals
Displ_Damper_dt[RL;RR]	[m/s]	Speed of vertical displacement of damper joint points of rear left and rear right wheel
Displ_Damper_RL_dq_Rear[q1;q2]	[]	Derivative of damper displacement of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Damper_RR_dq_Rear[q1;q2]	[]	Derivative of damper displacement of rear right wheel with respect to rotation angle around rotation axis of rear left (q1) and rear right (q2) wheel
Displ_Spring[RL;RR]	[m]	Vertical displacement of spring joint points of rear left and rear right wheel
Displ_Spring_RL_dq_Rear[q1;q2]	[]	Derivative of spring displacement of rear left wheel with respect to rotation angle around rotation axis of rear left (q1) and rear right (q2) wheel
Displ_Spring_RR_dq_Rear[q1;q2]	[]	Derivative of spring displacement of rear right wheel with respect to rotation angle around rotation axis of rear left (q1) and rear right (q2) wheel
Displ_Stabilizer	[m]	Vertical displacement of the stabilizer left joint point relative to the right joint point

Name	Unit	Description
Displ_Stabilizer_dq_Rear[q1;q2]	[]	Derivative of stabilizer displacement of rear left wheel with respect to rotation angle around rotation axis of rear left (q1) and rear right (q2) wheel
Pos_RL_Wheel[x;y;z]	[m]	Position vector of rear left wheel
Pos_RL_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of rear left wheel with respect to steering rod displacement
Pos_RL_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Derivative of wheel position of rear left wheel with respect to rotation angle around rotation axis of rear left (q1) and rear right (q2) wheel
Pos_RR_Wheel[x;y;z]	[m]	Position vector of rear right wheel
Pos_RR_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of rear right wheel with respect to steering rod displacement
Pos_RR_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Derivative of wheel position of rear right wheel with respect to rotation angle around rotation axis of rear left (q1) and rear right (q2) wheel
UnitVec_y_Wheel_RL_CoorSys_V[x;y;z]	[]	Unit vector in y direction of rear left wheel
UnitVec_y_Wheel_RR_CoorSys_V[x;y;z]	[]	Unit vector in y direction of rear right wheel
w_Rel_RL_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel orientation of rear left wheel with respect to steering rod speed
w_Rel_RL_Wheel_dq_Rear_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of rear left wheel with respect to rotation angle around rotation axis of rear left (q1) and rear right (q2) wheel
w_Rel_RR_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel orientation of rear right wheel with respect to steering rod speed
w_Rel_RR_Wheel_dq_Rear_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of rear right wheel with respect to rotation angle around rotation axis of rear left (q1) and rear right (q2) wheel

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Camber_Init_Left	[deg]	Initial camber angle of left wheel
Const_Camber_Init_Right	[deg]	Initial camber angle of right wheel
Const_Damp2Chassis_Left	[m]	Position vector of connection of left damper to chassis
Const_Damp2Chassis_Right	[m]	Position vector of connection of right damper to chassis
Const_Damp2STA_Left	[m]	Position vector of connection of left damper to control arm
Const_Damp2STA_Right	[m]	Position vector of connection of right damper to control arm
Const_Spring2STA_Left	[m]	Position vector of connection of left spring to control arm
Const_Spring2STA_Right	[m]	Position vector of connection of right spring to control arm
Const_Spring2Chassis_Left	[m]	Position vector of connection of left spring to chassis
Const_Spring2Chassis_Right	[m]	Position vector of connection of right spring to chassis

Name	Unit	Description
Const_StabiSelect	[]	Switch to select (1) or clear (0) stabilizer
Const_Stab2STA_Left	[m]	Position vector of connection of left stabilizer to control arm
Const_Stab2STA_Right	[m]	Position vector of connection of right stabilizer to control arm
Const_STA2Chassis_In_Left	[m]	Position vector of connection of left control arm to chassis (inner point)
Const_STA2Chassis_In_Right	[m]	Position vector of connection of right control arm to chassis (inner point)
Const_STA2Chassis_Out_Left	[m]	Position vector of connection of left control arm to chassis (outer point)
Const_STA2Chassis_Out_Right	[m]	Position vector of connection of right control arm to chassis (outer point)
Const_Toe_Init_Left	[deg]	Initial toe-in angle of left wheel
Const_Toe_Init_Right	[deg]	Initial toe-in angle of right wheel
Const_WheelCenter_Left	[m]	Position vector of rear left wheel center
Const_WheelCenter_Right	[m]	Position vector of rear right wheel center

Related topics

References

SusKin Rear SemiTrailing (ModelDesk Parameterizing 🕮)

Rear Suspension Kinematics Rigid Axle

Description

The RIGID_AXLE system describes suspension kinematics in geometrical way. There are several linkage points of suspension to chassis and wheel carrier that have to be configured. Position of wheel midpoint, orientation of wheel, displacements of spring, damper and stabilizer and several partial derivatives are calculated analytically within each simulation step. For a detailed description on calculated derivatives, refer to Rear Suspension Kinematics (Symmetric) on page 209.

The Rigid Axle system can be used as an alternative to the table-based suspension kinematics by replacing the table-based suspension systems with the Rigid Axle block. For further information on how to replace table-based suspension kinematics system with rigid axle and vice versa, refer to How to Change the Suspension Kinematics Model (ASM Vehicle Dynamics Model Description 🕮). For information on using table-based suspension for rigid axles, refer to Rigid Axle with Suspension Kinematics 3DOF (ASM Vehicle Dynamics Addendum 🕮).

The table-based compliance systems are superimposed to the analytically calculated rigid axle suspension.

Tip

For visualizing of geometrically described suspension, call `asm_plot_geosusp` at 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` at the MATLAB Command Window or refer to asm_plot_geosusp (ASM User Guide 🕮).

Inports

The following table shows the inports:

Name	Unit	Description
Angle_Rot_Axle	[rad]	Rotation angle about vehicle centerline
Angle_Rot_dt_Axle	[rad/s]	Speed of rotation angle about vehicle centerline

Name	Unit	Description
Displ_SteeringRod	[m]	Steering rod displacement
Displ_SteeringRod_dt	[m/s]	Speed of steering rod displacement
Displ_z_Axle	[m]	Vertical displacement of midpoint of axle
Displ_z_dt_Axle	[m/s]	Speed of vertical displacement of midpoint of axle
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (Gamma-Beta-Alpha) ▪ 2: ToeCamberSideview (Toe-Camber-Sideview)
Wheel_Pos_Offset	[m]	Additional offset for initial wheel position for left wheel

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Collection of ASM signals
Displ_Damper_dt[RL;RR]	[m/s]	Speed of vertical displacement of damper joint points of rear left and rear right wheel
Displ_Damper_RL_dq_Rear[q1;q2]	[]	Derivative of damper displacement of rear left wheel with respect to rotation angle around vehicle centerline (q1) and vertical displacement of axle (q2)
Displ_Damper_RR_dq_Rear[q1;q2]	[]	Derivative of damper displacement of rear right wheel with respect to rotation angle around vehicle centerline (q1) and vertical displacement of axle (q2)
Displ_Spring[RL;RR]	[m]	Vertical displacement of spring joint points of rear left and rear right wheel
Displ_Spring_RL_dq_Rear[q1;q2]	[]	Derivative of spring displacement of rear left wheel with respect to rotation angle around vehicle centerline (q1) and vertical displacement of axle (q2)
Displ_Spring_RR_dq_Rear[q1;q2]	[]	Derivative of spring displacement of rear right wheel with respect to rotation angle around vehicle centerline (q1) and vertical displacement of axle (q2)
Displ_Stabilizer	[m]	Vertical displacement of the stabilizer left joint point relative to the right joint point
Displ_Stabilizer_dq_Rear[q1;q2]	[]	Derivative of stabilizer displacement of rear left wheel with respect to rotation angle around vehicle centerline (q1) and vertical displacement of axle (q2)
Pos_RL_Wheel[x;y;z]	[m]	Position vector of rear left wheel
Pos_RL_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of rear left wheel with respect to steering rod displacement
Pos_RL_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Derivative of wheel position of rear left wheel with respect to rotation angle around vehicle centerline (q1) and vertical displacement of axle (q2)
Pos_RR_Wheel[x;y;z]	[m]	Position vector of rear right wheel

Name	Unit	Description
Pos_RR_Wheel_dDispl_SteeringRod[x;y;z]	[]	Derivative of wheel position of rear right wheel with respect to steering rod displacement
Pos_RR_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Derivative of wheel position of rear right wheel with respect to rotation angle around vehicle centerline (q1) and vertical displacement of axle (q2)
UnitVec_y_Wheel_RL_CoorSys_V[x;y;z]	[]	Unit vector in y direction of rear left wheel
UnitVec_y_Wheel_RR_CoorSys_V[x;y;z]	[]	Unit vector in y direction of rear right wheel
w_Rel_RL_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel orientation of rear left wheel with respect to steering rod speed
w_Rel_RL_Wheel_dq_Rear_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of rear left wheel with respect to rotation angle around vehicle centerline (q1) and vertical displacement of axle (q2)
w_Rel_RR_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Derivative of wheel orientation of rear right wheel with respect to steering rod speed
w_Rel_RR_Wheel_dq_Rear_dt[q1;q2][x;y;z]	[]	Derivative of wheel orientation of rear right wheel with respect to rotation angle around vehicle centerline (q1) and vertical displacement of axle (q2)

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Camber_Init_Left	[deg]	Initial camber angle of left wheel
Const_Camber_Init_Right	[deg]	Initial camber angle of right wheel
Const_Damper2Chassis_Left	[m]	Position vector of connection of left damper to chassis
Const_Damper2Chassis_Right	[m]	Position vector of connection of right damper to chassis
Const_Damper2Rigid_Left	[m]	Position vector of connection of left damper to rigid axle
Const_Damper2Rigid_Right	[m]	Position vector of connection of right damper to rigid axle
Const_Low2Chassis_Left	[m]	Position vector of connection of left lower link to chassis
Const_Low2Chassis_Right	[m]	Position vector of connection of right lower link to chassis
Const_Low2Rigid_Left	[m]	Position vector of connection of left lower link to rigid axle
Const_Low2Rigid_Right	[m]	Position vector of connection of right lower link to rigid axle
Const_Spring2Chassis_Left	[m]	Position vector of connection of left spring to chassis
Const_Spring2Chassis_Right	[m]	Position vector of connection of right spring to chassis
Const_Spring2Rigid_Left	[m]	Position vector of connection of left spring to rigid axle
Const_Spring2Rigid_Right	[m]	Position vector of connection of right spring to rigid axle
Const_StabiSelect	[]	Switch to select (1) or clear (0) stabilizer
Const_Stabi2Rigid_Left	[m]	Position vector of connection of left stabilizer to rigid axle
Const_Stabi2Rigid_Right	[m]	Position vector of connection of right stabilizer to rigid axle
Const_Toe_Init_Left	[deg]	Initial toe-in angle of left wheel
Const_Toe_Init_Right	[deg]	Initial toe-in angle of right wheel

Name	Unit	Description
Const_Up2Chassis_Left	[m]	Position vector of connection of left upper link to chassis
Const_Up2Chassis_Right	[m]	Position vector of connection of right upper link to chassis
Const_Up2Rigid_Left	[m]	Position vector of connection of left upper link to rigid axle
Const_Up2Rigid_Right	[m]	Position vector of connection of right upper link to rigid axle
Const_WheelCenter_Left	[m]	Position vector of rear left wheel center
Const_WheelCenter_Right	[m]	Position vector of rear right wheel center

Related topics

References

SusKin Rear Rigid (ModelDesk Parameterizing 🕮)

Rear Suspension Kinematics (Asymmetric, 3DOF)

Description

As the same implementation is used here as for the related front suspension, refer to Front Suspension Kinematics (Asymmetric, 3DOF) on page 169 to get detailed information.

Rear Suspension Kinematics (Symmetric, 3DOF)

Description

As the same implementation is used here as for the related front suspension, refer to Front Suspension Kinematics (Symmetric, 3DOF) on page 183 to get detailed information.

Suspension Force Kinematics Rear

Description

This block projects the spring, damper and stabilizer forces in the direction of the axle degrees of freedom. The projection is carried out according to the kinematic relations described in the KINEMATICS block.

The evaluated forces in the Rear Suspension Forces must be transformed in the direction of the relevant degrees of freedom.

For a spring deflections

$z_{Sp} = \left[z_{Sp_RL}\left(q_{1_Rear}, q_{2_Rear}\right) z_{Sp_RR}\left(q_{1_Rear}, q_{2_Rear}\right)\right]^T$ and rear suspension DOFs $\frac{\partial z_{Sp_RL}}{\partial q_{1_Rear}}, \frac{\partial z_{Sp_RL}}{\partial q_{2_Rear}}, \frac{\partial z_{D_RL}}{\partial q_{1_Rear}}, \frac{\partial z_{D_RL}}{\partial q_{2_Rear}} \frac{\partial z_{Stabi}}{\partial q_{1_Rear}}, \frac{\partial z_{Stabi}}{\partial q_{2_Rear}}$ a Jacobian matrix is calculated as follows:

$$J_{Sp} = \frac{\partial z_{Sp}}{\left[\partial q_{Rear}\right]^T}$$

Having the spring forces

$$F_{Sp} = \left[F_{Sp_RL}\left(z_{Sp_RL}\right)\ F_{Sp_RR}\left(z_{Sp_RR}\right)\right]^T$$

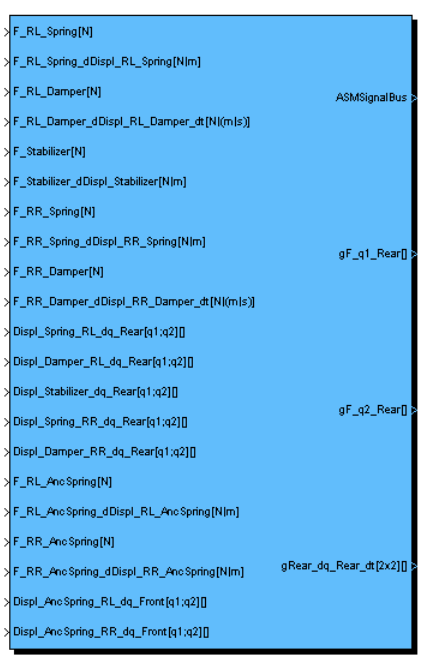
the generalized forces are calculated as follows:

$$gF_{Sp} = J^T{}_{Sp} F_{Sp}$$

The same calculations are carried out for the damper and the stabilizer. Finally, the generalized suspension forces are added.

$$\left[gF_q_{Rear}\right]_{2 \times 1} = J^T{}_{Sp} F_{Sp} + J^T{}_{D} F_D + J^T{}_{Stabi} F_{Stabi}$$

For the stabilization of the equation of motion, the derivatives of the generalized forces with respect to the generalized speeds are required. It can be calculated using the Jacobian matrices and the derivatives of the spring, damper and stabilizer forces with respect to spring displacement, damper velocity, and stabilizer displacement respectively.



Inports

The following table shows the inports:

Name	Unit	Description
Displ_AncSpring_RL_dq_Rear[q1;q2]	[]	Derivative of ancillary spring displacement of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_AncSpring_RR_dq_Rear[q1;q2]	[]	Derivative of ancillary spring displacement of rear right wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Damper_FR_dq_Rear[q1;q2]	[]	Derivative of damper displacement of rear right wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Damper_RL_dq_Rear[q1;q2]	[]	Derivative of damper displacement of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Spring_FR_dq_Rear[q1;q2]	[]	Derivative of spring displacement of rear right wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Spring_RL_dq_Rear[q1;q2]	[]	Derivative of spring displacement of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Displ_Stabilizer_dq_Rear[q1;q2]	[]	Derivative of stabilizer displacement of rear left wheel with respect to vertical wheel displacement of rear left (q1) and rear right (q2) wheel
F_RL_AncSpring	[N]	Force of rear left ancillary spring
F_RL_AncSpring_dDispl_RL_AncSpring	[N/m]	Change of ancillary spring force rear left with respect to vertical displacement of ancillary spring joint point rear left
F_RL_Damper	[N]	Force of rear left damper
F_RL_Damper_dDispl_RL_Damper_dt	[N/(m/s)]	Change of damper force rear left with respect to vertical speed of displacement of damper joint point rear left
F_RL_Spring	[N]	Force of rear left spring
F_RL_Spring_dDispl_RL_Spring	[N/m]	Change of spring force rear left with respect to vertical displacement of spring joint point rear left
F_RR_AncSpring	[N]	Force of rear right ancillary spring
F_RR_AncSpring_dDispl_RR_AncSpring	[N/m]	Change of ancillary spring force rear right with respect to vertical displacement of ancillary spring joint point rear right
F_RR_Damper	[N]	Force of rear right damper
F_RR_Damper_dDispl_RR_Damper_dt	[N/(m/s)]	Change of damper force rear right with respect to vertical speed of displacement of damper joint point rear right
F_RR_Spring	[N]	Force of rear right spring
F_RR_Spring_dDispl_RR_Spring	[N/m]	Change of spring force rear right with respect to vertical displacement of spring joint point rear right
F_Stabilizer	[N]	Force of stabilizer

Name	Unit	Description
F_Stabilizer_dDispl_Stabilizer	[N/m]	Change of stabilizer force with respect to vertical displacement of stabilizer joint point

Outports

The following table shows the outports:

Name	Unit	Description
gF_q1_Rear	[N] or [Nm][1)]	Generalized force in direction of q1_Rear
gF_q2_Rear	[N] or [Nm][1)]	Generalized force in direction of q2_Rear
gRear_dq_Rear_dt[2x2]	[N] or [Nm][1)]	Derivatives of the generalized forces with respect to the generalized speeds

1) The unit of the generalized forces and/or torques depends on the degree of freedom. If q is a displacement, gF is a force and has a unit of [N]. If q is a rotation, gF is a torque and has a unit of [Nm].

Related topics

References

History of the SUSPENSION_FORCE_KINEMATICS_REAR Block..... 450

Rear Suspension Kinematics Ancillary Spring

Description

The same implementation is used as for the related front suspension kinematics ancillary spring. To get detailed information, refer to Front Suspension Kinematics Ancillary Spring on page 199.

Common Suspension Blocks

Where to go from here

Information in this section

Rear Wheel Relative Velocities..... 227
This subsystem calculates the wheel relative translatory and angular velocities.

Relative Position Rear..... 229
This subsystem calculates the wheel relative position and orientation.

Rear Suspension Forces.....230
The forces acting between the wheel and vehicle body, which are generated by spring, damper, and stabilizer, are evaluated in this block.

Rear Suspension Compliance.....233
Due to suspension elasticity, additional elastic displacements are caused by the forces and torques acting on the wheel. Suspension compliance kinematics describes these additional displacements as functions of the forces and torques. There are two suspension compliance blocks available.

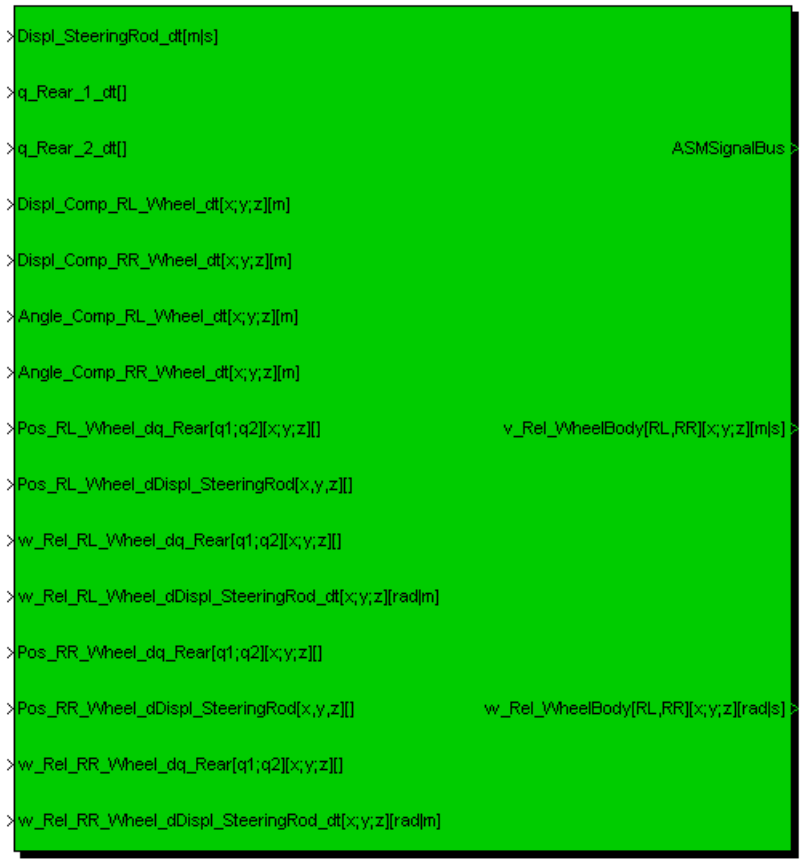
Rear Suspension Forces Ancillary Spring.....236
This block evaluates the forces acting between the wheel and vehicle body, which are generated by the ancillary spring.

Rear Suspension Forces Active.....236
This block evaluates the forces acting between the wheel and vehicle body, which are generated by the spring, damper, and stabilizer.

Rear Wheel Relative Velocities

Description

Due to suspension kinematics and elasticity, the wheels move in and rotate about the three directions (x, y, and z).



This subsystem calculates the wheel relative translatory and angular velocities. The equations for the rear left wheel are as follows:

$$v_{RL} = \frac{\partial v_{RL}}{\partial \dot{q}_{1_Rear}}\dot{q}_{1_Rear} + \frac{\partial v_{RL}}{\partial \dot{q}_{2_Rear}}\dot{q}_{2_Rear} + \frac{\partial v_{RL}}{\partial \dot{q}_{St}}\dot{q}_{St} + v_{RL_Compl}$$

$$\omega_{RL} = \frac{\partial \omega_{RL}}{\partial \dot{q}_{1_Rear}}\dot{q}_{1_Rear} + \frac{\partial \omega_{RL}}{\partial \dot{q}_{2_Rear}}\dot{q}_{2_Rear} + \frac{\partial \omega_{RL}}{\partial \dot{q}_{St}}\dot{q}_{St} + \omega_{RL_Compl}$$

The partial derivatives terms are calculated in the suspension kinematics subsystem and the compliance motion is calculated in the suspension compliance subsystem.

Inports

The following table shows the inports:

Name	Unit	Description
Angle_Comp_RL_Wheel_dt[x;y;z]	[rad/s]	Additional wheel angular velocity due to suspension compliance rear left
Angle_Comp_RR_Wheel_dt[x;y;z]	[rad/s]	Additional wheel angular velocity due to suspension compliance rear right
Displ_Comp_RL_Wheel_dt[x;y;z]	[m]	Additional wheel velocity due to suspension compliance rear left
Displ_Comp_RR_Wheel_dt[x;y;z]	[m]	Additional wheel velocity due to suspension compliance rear right
Displ_SteeringRod_dt	[m/s]	Speed of steering rod displacement
Pos_RL_Wheel_dDispl_SteeringRod[x,y,z]	[]	Change of wheel position rear left with respect to displacement of steering rod
Pos_RL_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Change of wheel position rear left with respect to changes of first and second generalized coordinates
Pos_RR_Wheel_dDispl_SteeringRod[x,y,z]	[]	Change of wheel position rear right with respect to displacement of steering rod
Pos_RR_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Change of wheel position rear right with respect to changes of first and second generalized coordinates
q_Rear_1_dt	[]	Derivative of first generalized coordinate with respect to time
q_Rear_2_dt	[]	Derivative of second generalized coordinate with respect to time
w_Rel_RL_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Change of wheel orientation rear left with respect to displacement of steering rod
w_Rel_RL_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Change of wheel orientation rear left with respect to changes of first and second generalized coordinates
w_Rel_RR_Wheel_dDispl_SteeringRod_dt[x;y;z]	[rad/m]	Change of wheel orientation rear right with respect to displacement of steering rod
w_Rel_RR_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Change of wheel orientation rear right with respect to changes of first and second generalized coordinates

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Velocity of the wheel body relative to vehicle body
v_Rel_WheelBody[RL,RR][x;y;z]	[m/s]	Velocity of wheel body relative to vehicle body, rear
w_Rel_WheelBody[RL,RR][x;y;z]	[rad/s]	Angular velocity of wheel body relative to vehicle body excluding wheel rotational speed, rear

Parameters

With the above inputs and outputs, this subsystem requires no parameters to perform its calculation.

Related topics

References

History of the RELATIVE_VELOCITIES_x Block..... 431

Relative Position Rear

Description

Due to suspension kinematics and elasticity, the wheels move in and rotate about the three directions (x, y and z). In this subsystem, the wheel relative position and orientation are calculated.

Inports

The following table shows the inports:

Name	Unit	Description
Angle_Comp_RL_Wheel[x;y;z]	[rad]	Additional wheel rotation angle due to suspension compliance rear left
Angle_Comp_RR_Wheel[x;y;z]	[rad]	Additional wheel rotation angle due to suspension compliance rear right
Displ_Comp_RL_Wheel[x;y;z]	[m]	Additional wheel displacement due to suspension compliance rear left
Displ_Comp_RR_Wheel[x;y;z]	[m]	Additional wheel displacement due to suspension compliance rear right
Pos_RL_Wheel[x;y;z]	[m]	Position vector of rear left wheel
Pos_RR_Wheel[x;y;z]	[m]	Position vector of rear right wheel

Name	Unit	Description
Sw_Rotation	[1|2]	Switch between two different rotation orders: ▪ 1: Cardan (Gamma-Beta-Alpha) ▪ 2: ToeCamberSideview (Toe-Camber-Sideview)
UnitVec_y_RL_Wheel_CoorSys_V[x;y;z]	[m]	Unit vector in y direction of wheel coordinate system of rear left wheel
UnitVec_y_RR_Wheel_CoorSys_V[x;y;z]	[m]	Unit vector in y direction of wheel coordinate system of rear right wheel

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Collection of ASM signals
Pos_Wheel[RL;RR][x;y;z]	[m]	Position vector of rear wheels including compliance effects
UnitVec_y_Wheel_CoorSys_V[RL;RR] [x;y;z]	[m]	Unit vector in y direction of wheel coordinate system of rear wheels including compliance effects

Related topics

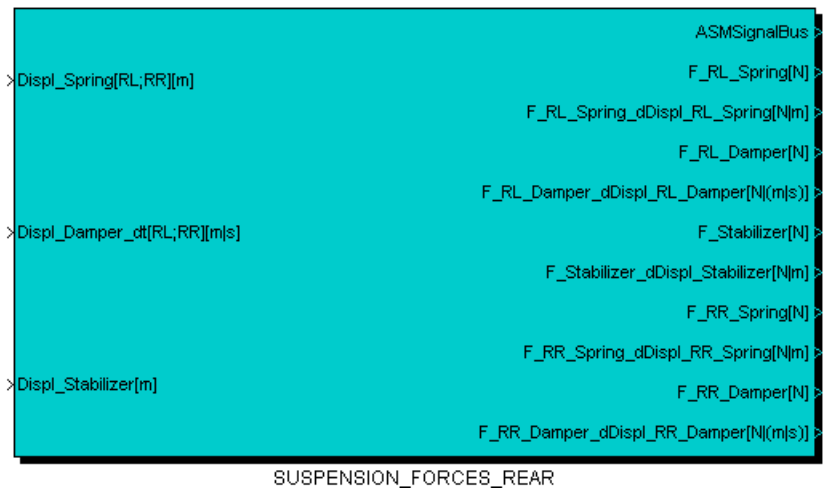
References

History of the RELATIVE_POSITION_x Block 431

Rear Suspension Forces

Description

The forces acting between the wheel and vehicle body, which are generated by spring, damper, and stabilizer, are evaluated in this block.



These forces are generated by the spring, damper, and stabilizer. The forces can be calculated by means of the spring, damper, and stabilizer characteristics, which depend on their displacements. The displacements are calculated from the kinematics subsystem.

$$F_{sp} = F_{sp}(z_{sp}),\ F_d = F_d(\dot{z}_d),\ F_{stab} = F_{stab}(z_{stab})$$

Note

The generalized suspension force in the direction of wheel degrees of freedom is calculated in another block.

Inports

The following table shows the inports:

Name	Unit	Description
Displ_Damper_dt[RL;RR]	[m/s]	Vertical speed of displacement of the damper joint point, rear left and rear right
Displ_Spring[RL;RR]	[m]	Vertical displacement of the spring joint point, rear left and rear right
Displ_Stabilizer	[m]	Vertical displacement of stabilizer left joint point relative to the right joint point

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
F_RL_Damper	[N]	Force of rear left damper
F_RL_Damper_dDispl_RL_Damper	[N/(m/s)]	Change of damper force rear left with respect to vertical speed of displacement of damper joint point rear left
F_RL_Spring	[N]	Force of rear left spring
F_RL_Spring_dDispl_RL_Spring	[N/m]	Change of spring force rear left with respect to vertical displacement of spring joint point rear left
F_RR_Damper	[N]	Force of rear right damper
F_RR_Damper_dDispl_RR_Damper	[N/(m/s)]	Change of damper force rear right with respect to vertical speed of displacement of damper joint point rear right
F_RR_Spring	[N]	Force of rear right spring
F_RR_Spring_dDispl_RR_Spring	[N/m]	Change of spring force rear right with respect to vertical displacement of spring joint point rear right
F_Stabilizer	[N]	Force of stabilizer
F_Stabilizer_dDispl_Stabilizer	[N/m]	Change of stabilizer force with respect to vertical displacement of stabilizer joint point

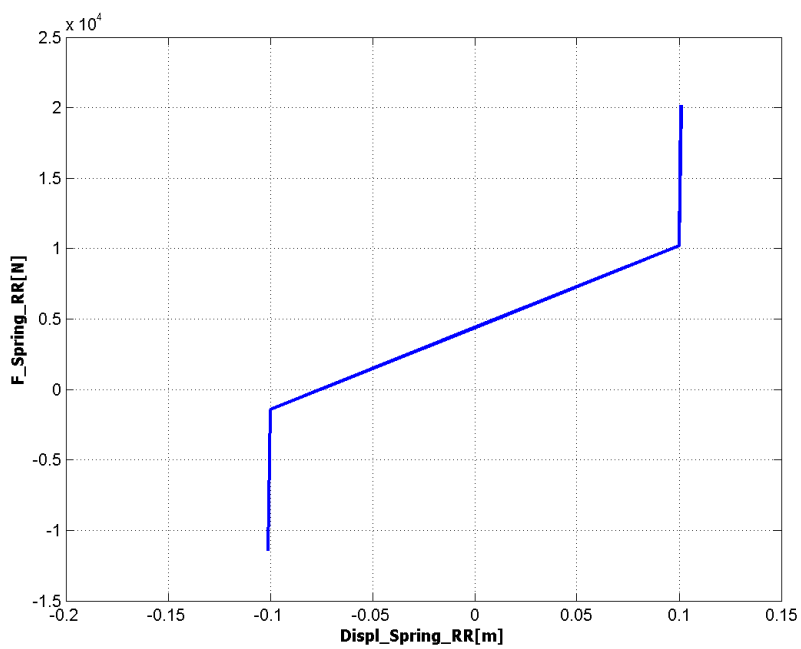
Parameters

The following table shows the parameters:

Name	Unit	Description
Map_Force_LeftDamper	[N]	Left damper force = f(damper speed[m/s])
Map_Force_LeftSpring	[N]	Left spring force = f(Spring displacement[m])
Map_Force_RightDamper	[N]	Right damper force = f(damper speed[m/s])
Map_Force_RightSpring	[N]	Right spring force = f(spring displacement[m])
Map_Force_Stabilizer	[N]	Stabilizer force = f(stabilizer relative displacement of left joint point relative to right joint point[m])

Processing information

The following illustration shows a spring characteristic curve with the dead stops simulated as a very high stiffness. Note that at zero spring deflection, the spring exerts the initial force at the configuration state, which is defined to be the zero position. This state must also be the same as the zero position for the kinematics tables.



The force obtained from the curve is in the spring and not the wheel center, so it must be transferred to the wheel center.

Related topics

References

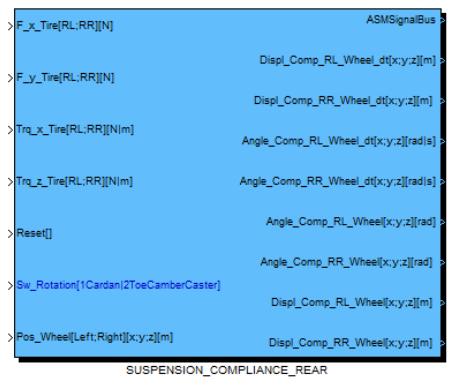
SusForces Rear 2nd (ModelDesk Parameterizing 🕮)

Rear Suspension Compliance

Description

Due to suspension elasticity, additional elastic displacements are caused by the forces and torques acting on the wheel. Suspension compliance kinematics describes these additional displacements as functions of the forces and torques.

There are two suspension compliance blocks available. The SUSPENSION_COMPLIANCE subsystem takes the forces and torques acting on the current wheel center into account, whereas the SUSPENSION_COMPLIANCE_OPP subsystem takes the forces and torques acting on the opposite wheel center into account.



The additional displacements are considered in x and y directions and additional alpha (α) and gamma (γ) angles. These variables are defined to depend on the longitudinal and lateral forces, and the self-aligning torque, i.e.:

$$\Delta x = \sum_i \Delta x_i(Q_i)$$

$$\Delta y = \sum_i \Delta y_i(Q_i)$$

$$\Delta \alpha = \sum_i \Delta \alpha_i(Q_i)'$$

$$\Delta \gamma = \sum_i \Delta \gamma_i(Q_i)$$

where:

$$i = 1, \ldots, 6$$

$$Q_i = F_x, F_y, T_x, T_y, F_{x_opp}, F_{y_opp}, T_{x_opp}, T_{y_opp}$$

The total wheel position and orientation are then the summation of those due to kinematics and elasto-kinematics.

The dynamics of these displacements are approximated by second-order dynamics with a natural frequency ω_n which is defined as a parameter. For example, the dynamic x displacement can be calculated as follows:

$$\Delta x_{dyn} = \frac{{\omega_n}^2}{s^2 + 2\,\zeta\,\omega_n\,s + {\omega_n}^2}\Delta x$$

where:

s is the Laplace operator

ζ is the damping ratio ($\zeta = 0{,}707$, set internally)

Note

The sample time of the model must be considered when defining ω_n, such that $(\omega_n/2\pi) \ll (1/ModelSampleTime)$.

Inports

The following table shows the inports:

Name	Unit	Description
F_x_Tire[RL;RR]	[N]	Longitudinal tire force, rear left and rear right
F_y_Tire[RL;RR]	[N]	Lateral tire force, rear left and rear right
Reset	[]	Resets all integrators in the block to their initial conditions
Trq_x_Tire[RL;RR]	[Nm]	Tire torque about the longitudinal axis, rear left and rear right
Trq_z_Tire[RL;RR]	[Nm]	Tire torque about the vertical axis, rear left and rear right

Outports

The following table shows the outports:

Name	Unit	Description
Angle_Comp_RL_Wheel[x,y,z]	[rad]	Wheel orientation change due to suspension compliance, rear left
Angle_Comp_RL_Wheel_dt[x,y,z]	[rad/s]	Relative wheel angular velocity due to suspension compliance, rear left
Angle_Comp_RR_Wheel[x,y,z]	[rad]	Wheel orientation change due to suspension compliance, rear right
Angle_Comp_RR_Wheel_dt[x,y,z]	[rad/s]	Relative wheel angular velocity due to suspension compliance, rear right
Displ_Comp_RL_Wheel[x,y,z]	[m]	Wheel position change due to suspension compliance, rear left
Displ_Comp_RL_Wheel_dt[x,y,z]	[m/s]	Relative wheel speed due to suspension compliance, rear left
Displ_Comp_RR_Wheel[x,y,z]	[m]	Wheel position change due to suspension compliance, rear right
Displ_Comp_RR_Wheel_dt[x,y,z]	[m/s]	Relative wheel speed due to suspension compliance, rear right

Parameters

The following table shows the parameters for the SUSPENSION_COMPLIANCE subsystem. The parameters for the SUSPENSION_COMPLIANCE_OPP subsystem are structurally identical, but refer to the opposite wheel each. Both subsystems must be parameterized for the left wheel only. The right wheel compliance effects are symmetrically mirrored from the left ones below the mask.

Example

- The `Map_Displ_y_Wheel_F_y` parameter describes the lateral displacement of the left wheel center as a function of the lateral force (F_y) acting on the left wheel center. This parameter is needed for independent and dependent suspension types.
- The `Map_Displ_y_Wheel_F_y_opp` parameter describes the lateral displacement of the left wheel center as a function of the lateral force (F_y_opp) acting on the right wheel center. This parameter is needed for dependent suspension types only.

Name	Unit	Description
Const_NaturalFrequency_Angle_Alpha	[1/s]	Natural frequency for the dynamics of the α angle
Const_NaturalFrequency_Angle_Gamma	[1/s]	Natural frequency for the dynamics of the γ angle
Const_NaturalFrequency_Displ_x	[1/s]	Natural frequency for the dynamics of the displacement in x direction
Const_NaturalFrequency_Displ_y	[1/s]	Natural frequency for the dynamics of the displacement in y direction
Map_Angle_Alpha_Wheel_F_x	[deg]	α angle as a function of tire longitudinal force
Map_Angle_Alpha_Wheel_F_y	[deg]	α angle as a function of tire lateral force
Map_Angle_Alpha_Wheel_Trq_x	[deg]	α angle as a function of tire torque about longitudinal axis
Map_Angle_Alpha_Wheel_Trq_z	[deg]	α angle as a function of tire torque about the vertical axis
Map_Angle_Gamma_Wheel_F_x	[deg]	γ angle as a function of tire longitudinal force
Map_Angle_Gamma_Wheel_F_y	[deg]	γ angle as a function of tire lateral force
Map_Angle_Gamma_Wheel_Trq_x	[deg]	γ angle as a function of tire torque about longitudinal axis
Map_Angle_Gamma_Wheel_Trq_z	[deg]	γ angle as a function of tire torque about the vertical axis
Map_Displ_x_Wheel_F_x	[mm]	x displacement of wheel center as a function of tire longitudinal force
Map_Displ_x_Wheel_F_y	[mm]	x displacement of wheel center as a function of tire lateral force
Map_Displ_x_Wheel_Trq_x	[mm]	x displacement of wheel center as a function of tire torque about the longitudinal axis
Map_Displ_x_Wheel_Trq_z	[mm]	x displacement of wheel center as a function of tire torque about the vertical axis
Map_Displ_y_Wheel_F_x	[mm]	y displacement of wheel center as a function of tire longitudinal force
Map_Displ_y_Wheel_F_y	[mm]	y displacement of wheel center as a function of tire lateral force
Map_Displ_y_Wheel_Trq_x	[mm]	y displacement of wheel center as a function of tire torque about the longitudinal axis
Map_Displ_y_Wheel_Trq_z	[mm]	y displacement of wheel center as a function of tire torque about the vertical axis

Related topics

References

SusComp Rear (ModelDesk Parameterizing 🕮)
SusComp Rear 2nd (ModelDesk Parameterizing 🕮)
SusComp Rear 3rd (ModelDesk Parameterizing 🕮)
SusCompOpp Rear (ModelDesk Parameterizing 🕮)
SusCompOpp Rear 2nd (ModelDesk Parameterizing 🕮)
SusCompOpp Rear 3rd (ModelDesk Parameterizing 🕮)

Rear Suspension Forces Ancillary Spring

Description

The same implementation is used as for the related front suspension forces ancillary spring. To get detailed information, refer to Front Suspension Forces Ancillary Spring on page 143.

Rear Suspension Forces Active

Description

The same implementation is used as for the related front suspension forces active. To get detailed information, refer to Front Suspension Forces Active on page 141.

Steering

Where to go from here

Information in this section

Steering.....237
The STEERING subsystem transfers the steering wheel angle defined by the driver to a movement of the steering rod.

Steering Tie Rod.....243
The STEERING_TIE_ROD block can be used to simulate the forces acting at the left and right steering tie rods.

Steering 3DOF Variable Ratio.....244
In this steering model, three degrees of freedom are considered.

Variable Steering Ratio.....257
A variable steering ratio can be simulated with the STEERING_VARIABLE_RATIO block.

Steering

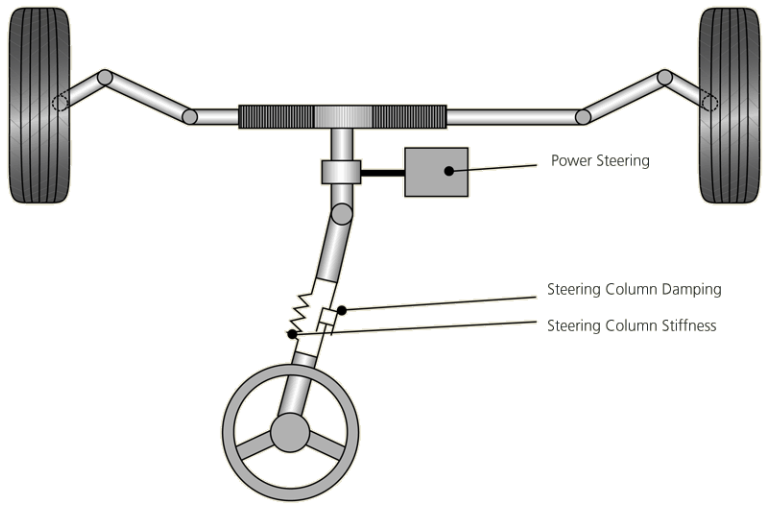
Description

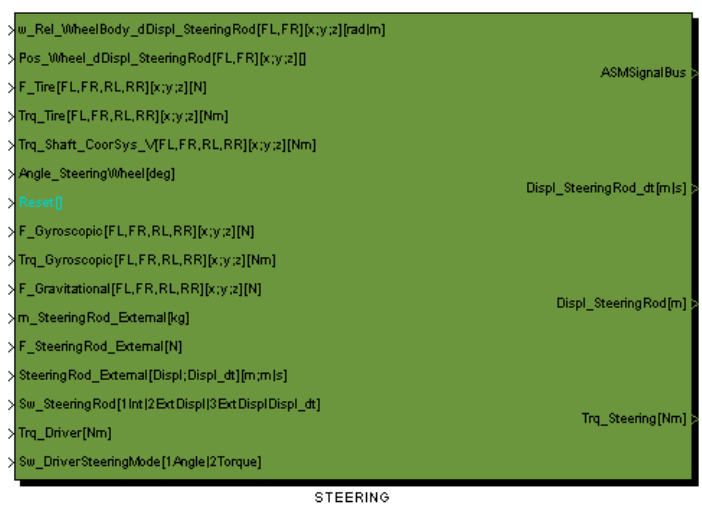
The STEERING subsystem transfers the steering wheel angle defined by the driver to a movement of the steering rod. The steering rod movement is passed on to the suspension kinematics so that the wheel position and orientation can be calculated.

The steering model is composed of a steering column, steering transmission, power steering, and steering rod. The generalized degree of freedom is the displacement of the steering rod q_{St}.

Note

For the sign convention of the ASM steering system, refer to Sign Conventions (ASM Vehicle Dynamics Addendum 🕮)





Inports

The following table shows the inports:

Name	Unit	Description
Angle_SteeringWheel	[deg]	Steering wheel angle, defined by the driver
F_Gravitational[FL,FR,RL,RR][x;y;z]	[N]	Gravitational forces of the wheel
F_Gyroscopic[FL,FR,RL,RR][x;y;z]	[N]	Gyroscopic forces of the wheel
F_SteeringRod_External	[N]	Additional external force on steering rod
F_Tire[FL,FR,RL,RR][x;y;z]	[N]	Tire force vector in vehicle reference system
m_SteeringRod_External	[kg]	Additional external steering rod mass
Pos_Wheel_dDispl_SteeringRod [FL,FR][x;y;z]	[]	Change of wheel position with respect to the displacement of steering rod
Reset	[]	Resets all integrators in the block to their initial conditions.

Name	Unit	Description
SteeringRod_External[Displ;Displ_dt]	[m;m/s]	Vector including external calculated steering rod displacement and steering rod speed
Sw_DriverSteeringMode	[1|2]	Switch to change between driver angle and driver torque as input to the steering system ▪ 1: Angle ▪ 2: Torque
Sw_SteeringRod	[1|2|3]	Switch to change steering rod displacement and speed from internal calculations to external ones: ▪ 1: Internal calculation of Displ_SteeringRod and Displ_SteeringRod_dt ▪ 2: External calculation of Displ_SteeringRod, internal calculation of Displ_SteeringRod_dt ▪ 3: External calculation of Displ_SteeringRod and Displ_SteeringRod_dt
Trq_Driver	[Nm]	External torque from driver (driver steering switch mode has to be set to 2 for using the signal)
Trq_Gyroscopic[FL,FR,RL,RR][x;y;z]	[Nm]	Gyroscopic torques of the wheel
Trq_Shaft_CoorSys_V[FL,FR,RL,RR][x;y;z]	[Nm]	Driving torque on the wheel in vehicle reference coordinate system
Trq_Tire[FL,FR,RL,RR][x;y;z]	[Nm]	Tire torque vector in vehicle reference system
w_Rel_WheelBody_dDispl_SteeringRod [FL,FR][x;y;z]	[rad/m]	Change of wheel orientation with respect to the displacement of steering rod

Outports

The following table shows the outports:

Name	Unit	Description
Displ_SteeringRod	[m]	Displacement of steering rod
Displ_SteeringRod_dt	[m/s]	Speed of steering rod displacement
Trq_Steering	[Nm]	Torque in the steering column

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Angel_Max_SteeringWheel	[deg]	Maximal steering wheel angle
Const_Coeff_Transient_Friction	[N/(m/s)]	Transient friction coefficient
Const_Damping_SteeringColumn	[Nm/(rad/s)]	Damping, steering column
Const_Force_Friction_max	[N]	Maximum friction force in the steering
Const_i_Steering	[deg/m]	Ratio angle steering wheel to displacement steering rod
Const_m_Steering_Rod	[kg]	Mass of steering rod
Const_NaturalFrequency_SteeringCompliance	[1-/s]	Natural frequency for the steering compliance dynamics

Name	Unit	Description
Const_Stiffness_SteeringColumn	[Nm/rad]	Torsion stiffness, steering column
Const_Time_PowerSteering	[s]	Time constant, power steering force dynamic
Const_Time_SteeringWheel	[s]	Time constant, steering wheel dynamic
Map_SteeringCompliance	[m]	Additional steering rod displacement [m] = f(force at steering rod [N]
Map_Torque_PowerSteering	[Nm]	Power steering servo-torque = f(torque in steering column)

Steering force balance

Neglecting the coupling between the steering degree of freedom and the other vehicle degrees of freedom, the equation of motion for the steering rod movement can be written as follows:

$$M_{St}\,\ddot{q}_{St} = Q_{St_FL} + Q_{St_FR} + Q_{St_Gear} + Q_{St_Fric}$$

where:

Q_{St_FL} is the generalized force due to front left tire forces and torques and depends on the suspension kinematics

Q_{St_FR} is the generalized force due to front right tire forces and torques and depends on the suspension kinematics

Q_{St_Gear} is the generalized force at the steering gear from the steering column including power steering

Q_{St_Fric} is the generalized force due to friction in the steering rod

M_{St} is the generalized mass in the direction of q_{St} which depends on the wheel inertias and masses

Friction in steering mechanism

The friction in the steering gear resists the movement of the steering rod. The friction force is composed of two components. Static force F_{St_FricSt} balances the other steering forces. The transition from a moving state to the block state is performed by a dynamic term, which depends on the speed of the steering rod.

$$F_{St_{FricSt}} = \begin{cases} F_{St_Fric,max}, & if \quad F_{St_{FricSt}} > F_{St_Fric,max} \\ F_{St_FricSt}, & if \quad |F_{St_FricSt}| \le F_{St_Fric,max} \\ -F_{St_Fric,max}, & if \quad F_{St_FricSt} < -F_{St_Fric,max} \end{cases}$$

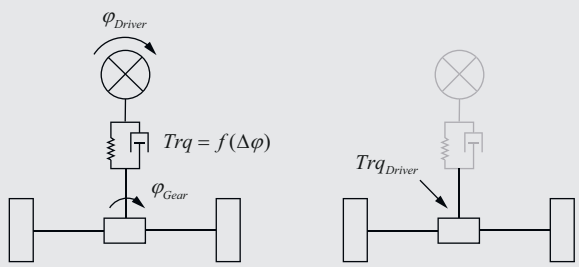
$$Q_{St_Fric} = -F_{St_{FricSt}} - D\,\dot{q}_{St}$$

$$Q_{St_Fric} = \begin{cases} F_{St_Fric,max}, & if \quad Q_{St_{Fric}} > F_{St_Fric,max} \\ Q_{St_Fric}, & if \quad |Q_{St_Fric}| \le F_{St_Fric,max} \\ -F_{St_Fric,max}, & if \quad Q_{St_Fric} < -F_{St_Fric,max} \end{cases}$$

Power steering

Note

There are two different inputs to the steering system (Sw_DriverSteeringMode): either driver angle or driver torque.
If driver angle is used as input, a steering torque is calculated at the steering column from the relative angle coming from the driver and the steering gear that is calculated from the steering rod dynamics.
If driver torque is used as input, the steering column is not taken into account but the driver torque acts directly on the steering rod.



In a vehicle equipped with power steering, the steering torque from the driver is increased by additional torque from a hydraulic or electrical mechanism. The additional torque depends on the torque exerted in the steering column and in some systems on vehicle speed.

$$T_{St_Col} = k_{St}\left(\delta_{St_Wheel} - \delta_{St_Gear}\right) + \left(\dot{\delta}_{St_Wheel} - \dot{\delta}_{St_Gear}\right)$$

$$T_{St_Pow} = f\left(T_{St_Col}\right)$$

where:

k_{St}	is the stiffness of the steering column
d_{St}	is the damping coefficient of the steering column
δ_{St_Wheel}	is the steering wheel angle
δ_{St_Gear}	is the angle at the input shaft of the steering gearbox
T_{St_Col}	is the torque in the steering column
T_{St_Pow}	is the additive power steering torque

The model describes function f by a look-up table. The dynamics of the power steering mechanism are emulated by a first-order delay with the time constant T_{St_Pow}, i.e.

$$\tau_{St_Pow}\,\dot{T}_{St_Pow} = f\left(T_{St_Col}\right) - T'_{St_Pow}$$

where

T'_{St_Pow}	is the dynamic power steering torque

The torques are then added and transferred to the steering rod side with the steering gear ratio.

External power steering

The block also supports external power steering models. For example, a more detailed model of an electric power steering can be connected. For this use case

the torque of the steering column (Trq_Steering[Nm]) is provided as an output. Using this torque, the external steering model can calculate the additional steering force which is introduced on the steering rod. This force has to be connected to the F_SteeringRod_External[N] input. Further additional inertias can be taken into account. They have to be reduced on the steering rod and assigned to the m_SteeringRod_External[kg] input.

If the external power steering is used, the Map_Torque_PowerSteering has to be initialized with zeros, otherwise it will also support the driver.

The position can also be calculated externally. It is therefore possible to integrate the position of an electric power steering system on a hardware integrator. To use this feature, the steering rod velocity (Disp_SteeringRod_dt[m|s]) has to be sent to the hardware integrator and the result has to be assigned to the Displ_SteeringRod_External[m] input. The Sw_Displ_SteeringRod[1Internal|2External] switch has to be set to external.

Recirculating ball steering

A recirculating ball steering (also known as ball joint steering) can be simulated with the standard ASM steering system. For details on simulating recirculating ball steering, refer to Recirculating Ball (ASM Vehicle Dynamics Addendum 🕮).

Subsystems

- Variable Steering Ratio on page 257
- Steering Tie Rod on page 243
- Steering 3DOF Variable Ratio on page 244

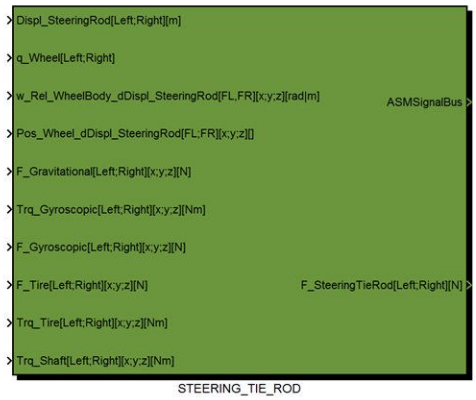
Related topics

References

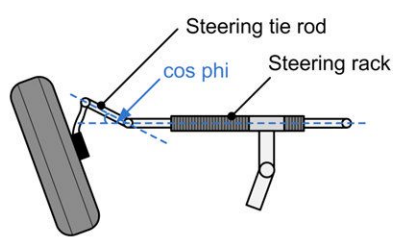
Steering (ModelDesk Parameterizing 🕮)

Steering Tie Rod

Description



The STEERING_TIE_ROD block can be used to simulate the forces acting at the left and right steering tie rods. For this, a 2-D look-up table has to be parameterized, which describes the geometric relation between the steering tie rod and the steering rack as function of the steering rod displacement and the vertical wheel displacement.



Inports

The following table shows the inports:

Name	Unit	Description
Displ_SteeringRod	[m]	Displacement of steering rod at left and right steering tie rod connection points
F_Gravitational	[N]	Gravitational force vector of left and right wheel
F_Gyroscopic	[N]	Gyroscopic force vector of left and right wheel
F_Tire	[N]	Tire force vector of left and right wheel in vehicle reference system
Pos_Wheel_dDispl_SteeringRod	[]	Change of wheel position with respect to the displacement of steering rod
q_Wheel	[]	Generalized coordinates of left and right wheel
Trq_Gyroscopic	[Nm]	Gyroscopic torque vector of left and right wheel
Trq_Shaft	[Nm]	Driving torque vector of left and right wheel in vehicle reference coordinate system
Trq_Tire	[Nm]	Tire torque vector of left and right wheel in vehicle reference system

Name	Unit	Description
w_Rel_WheelBody_dDispl_SteeringRod	[]	Change of left and right wheel orientation with respect to the displacement of steering rod

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
F_SteeringTieRod	[N]	Force at the left and right steering tie rod

Parameters

The following table shows the parameters:

Name	Unit	Description
Map_CosPhi_SteeringTieRod	[]	Map of geometric relation between the steering tie rod and the steering rack as function of the steering rod displacement and the vertical wheel displacement.

Related topics

References

Steering Tie Rod (ModelDesk Parameterizing 🕮)

Steering 3DOF Variable Ratio

Description

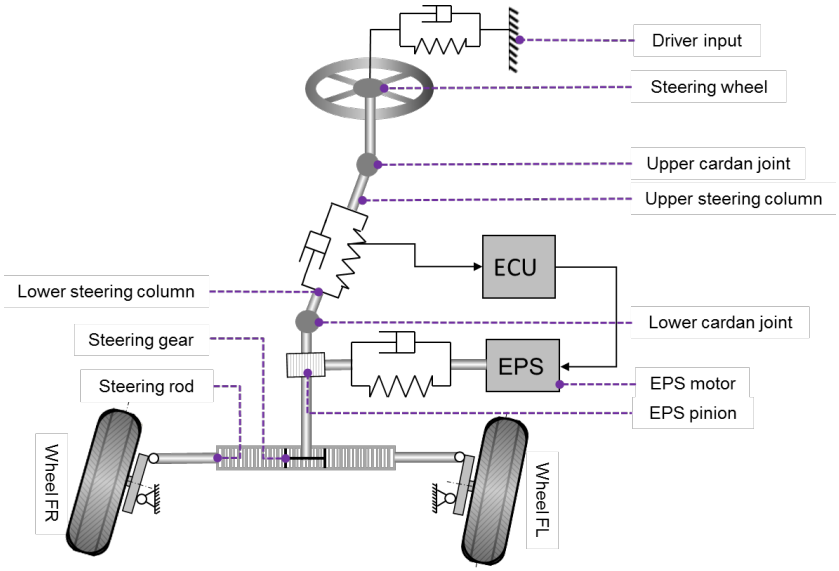
The steering subsystem transfers the desired steering wheel angle defined by the driver to a corresponding linear movement of the steering rod. This is then provided as input to suspension kinematics, so that the wheel position and orientation can be calculated.

The steering model comprises steering wheel, steering column, steering transmission (with variable transmission ratio), power steering motor, power steering transmission, steering rod, friction elements at steering rod and steering wheel, and two cardan joints.

Three degrees of freedom are considered in the model:

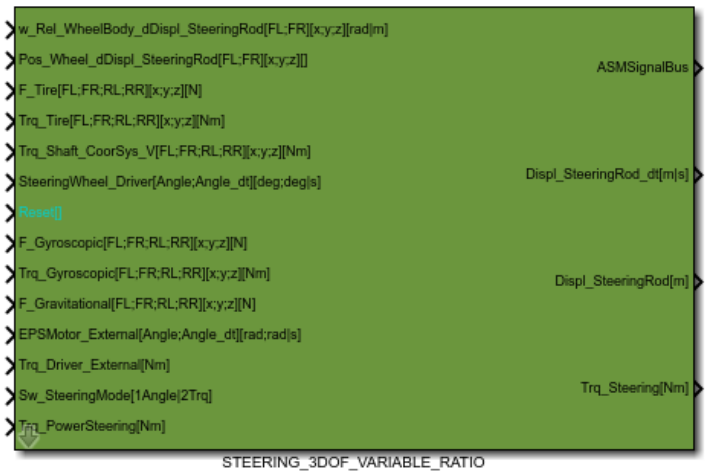
- Angular displacement of steering wheel with upper steering column
- Angular displacement of lower steering column with steering rod
- Angular displacement of EPS motor

The different components of the steering model are shown in the following illustration:



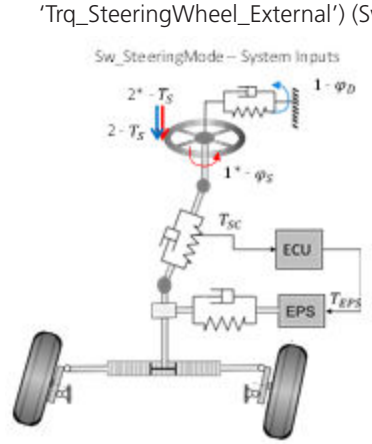
Note

For the sign convention of the ASM steering system, refer to Sign Conventions (ASM Vehicle Dynamics Addendum 🕮).



Inports

The following table shows the inports:

Name	Unit	Description
SteeringWheel_Driver[Angle;Angle_dt]	[rad] [rad/s]	Desired steering wheel angular displacement and velocity calculated by driver model (or) Steering wheel angular displacement and velocity stimulus defined by measurement data
EPSMotor_External[Angle;Angle_dt]	[rad] [rad/s]	EPS motor angular displacement and velocity in case of external EPS motor simulation
F_Gravitational[FL,FR,RL,RR][x;y;z]	[N]	Gravitational forces of the wheel
F_Gyroscopic[FL,FR,RL,RR][x;y;z]	[N]	Gyroscopic forces of the wheel
F_Tire[FL,FR,RL,RR][x;y;z]	[N]	Tire force vector in vehicle reference system
Pos_Wheel_dDispl_SteeringRod [FL,FR] [x;y;z]	[]	Change of wheel position with respect to displacement of steering rod
Reset	[]	Resets all integrators in the block to their initial conditions.
Sw_SteeringMode	[1Angle|2Trq]	Switch to change steering wheel control between input angle and torque. In torque mode, the torque is always applied on the steering wheel. However, in angle mode the angle is applied either at driver's hand (with force elements representing the driver's hand) or directly at the steering wheel depending on the parameter (Sw_DriverHand) 1. Angle φ_S applied at driver's hand (input to be provided as a vector through inport – 'SteeringWheel_Driver') (Sw_DriverHand = 1) 2. Torque T_S applied at steering wheel (input to be provided through inport – "Trq_SteeringWheel_External") (Sw_DriverHand = 1) 1. Stimulus Angle φ_S applied at steering wheel (input to be provided as a vector through inport – "SteeringWheel_Driver") (Sw_DriverHand = 0) 2. Torque T_S applied at steering wheel in the same way as mode 2 (input to be provided through inport – 'Trq_SteeringWheel_External') (Sw_DriverHand = 0) 

Name	Unit	Description
Trq_Gyroscopic[FL,FR,RL,RR][x;y;z]	[Nm]	Gyroscopic torques of the wheel
Trq_Driver_External	[Nm]	Torque input from driver (for torque based steering control)
Trq_PowerSteering	[Nm]	Torque at power steering motor (internal EPS simulation) (or) Torque at EPS pinion (external EPS simulation)
Trq_Shaft_CoorSys_V[FL,FR,RL,RR][x;y;z]	[Nm]	Driving torque on the wheel in vehicle reference coordinate system
Trq_Tire[FL,FR,RL,RR][x;y;z]	[Nm]	External torque on steering wheel in case of torque-based steering wheel control
w_Rel_WheelBody_dDispl_SteeringRod [FL,FR][x;y;z]	[rad/m]	Change of wheel orientation with respect to the displacement of steering rod

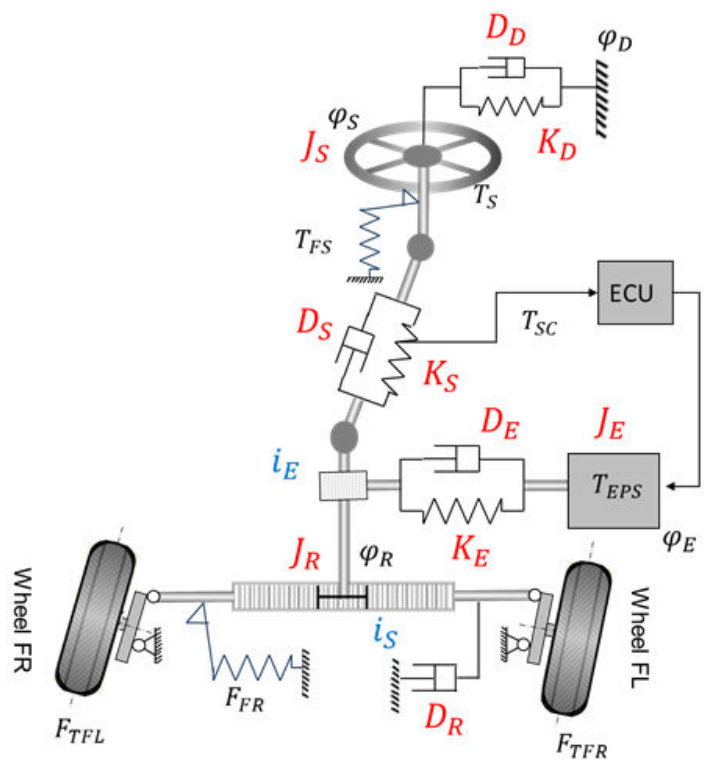
Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Displ_SteeringRod	[m]	Displacement of steering rod
Displ_SteeringRod_dt	[m/s]	Speed of steering rod displacement
Trq_Steering	[Nm]	Torque in the steering column

Model components

The following illustration displays the key components of the 3DOF steering model:



Inertial masses:

- Steering wheel + upper steering column (J_S)
- Lower steering column + steering rod (J_R)
- EPS motor (J_E)

System states:

- Steering wheel angular displacement (φ_S) and velocity ($\dot{\varphi}_S$)
- Lower steering column angular displacement (φ_R) and velocity ($\dot{\varphi}_R$)
- EPS motor angular displacement (φ_E) and velocity ($\dot{\varphi}_E$)

Friction elements

- Friction element at steering wheel (*TFS*)
- Friction element at steering rod (*FFR*)

Force elements

- Steering column stiffness (K_S)
- Steering column damping (D_S)
- Lower steering column damping (D_R)
- Driver hand stiffness (K_D)
- Driver hand damping (D_D)

- EPS system stiffness (K_E)
- EPS damping (D_E)

System input:

- Driver hand angle (φ_D) / Steering torque (T_S)

Equations of motion

The following equations of motion describe the 3DOF steering system (for EPS at lower steering column configuration):

$$J_S \cdot \ddot{\varphi}_S = K_D \cdot (\varphi_D - \varphi_S) + D_D \cdot (\dot{\varphi}_D - \dot{\varphi}_S) - K_S \cdot (\varphi_S - \varphi_R) - D_S \cdot (\dot{\varphi}_S - \dot{\varphi}_R) - T_{FS}$$

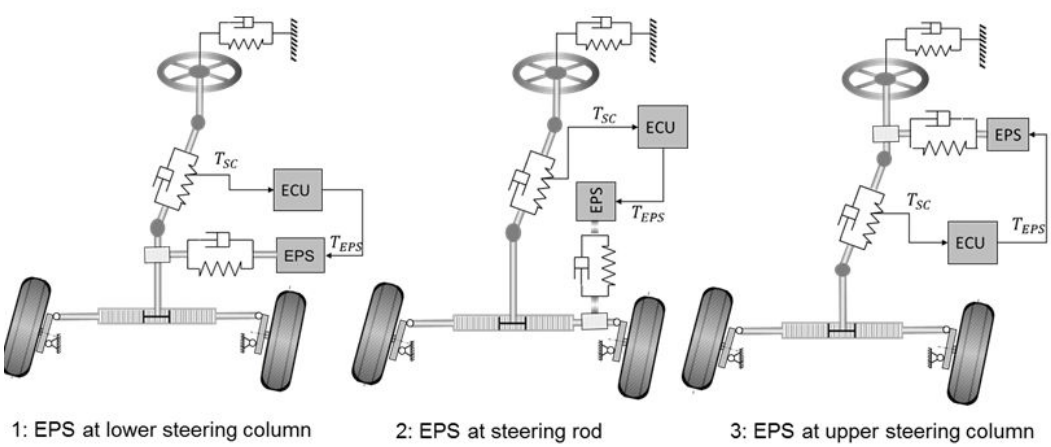
$$J_R \cdot \ddot{\varphi}_R = K_S \cdot (\varphi_S - \varphi_R) + D_S \cdot (\dot{\varphi}_S - \dot{\varphi}_R) + i_E \cdot (K_E \cdot (\varphi_E - i_E \cdot \varphi_R) + D_E \cdot (\dot{\varphi}_E - i_E \cdot \dot{\varphi}_R)) + \frac{1}{i_S(\varphi_G)} \cdot (F_{TFL} + F_{TFR} - F_{FR})$$

$$J_E \cdot \ddot{\varphi}_E = T_E - K_E \cdot (\varphi_E - i_E \cdot \varphi_R) - D_E \cdot (\dot{\varphi}_E - i_E \cdot \dot{\varphi}_R)$$

The steering ratio $i_S(\varphi_G)$ is a function of the angular displacement of the steering gear $\varphi_G \cdot (\varphi_G = f(\varphi_R))$.

System configuration

With the 3DOF steering model, it is possible to choose between the following three steering system configurations. The choice is based on the location of the EPS system of the steering system to be simulated.



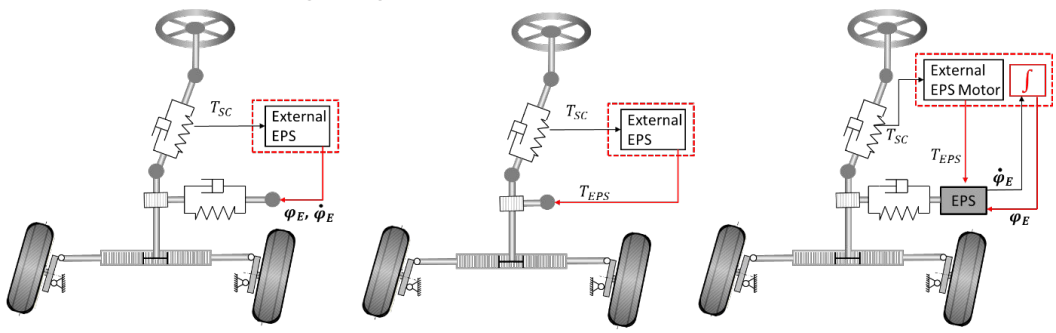
Note

It is important to consider the location of steering torque sensor T_{SC} when choosing the configuration.

Note

The EPS motor represented in the model simulates the dynamics of the mechanical components of the motor (including motor inertia, motor shaft stiffness and damping, EPS gearbox transmission ratio). The electrical behavior of the motor should be simulated outside the model. The output torque from the simulation can be provided as input to the EPS motor inertia via the **Trq_PowerSteering[Nm]** inport. For simulation of the electrical behavior of an EPS motor, refer to the ASM Electric Components Library.

In addition to the 3DOF configurations mentioned above, 2DOF configurations can also be simulated with this model to incorporate external EPS motor simulation. Based on the type of output from external EPS simulation, one of the following configurations can be chosen.



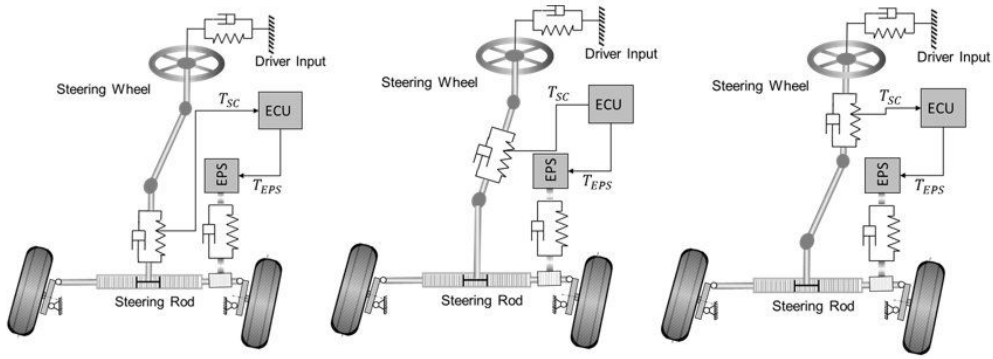
1. EPS States 2. EPS Torque 3. EPS Torque and Angle

When using configuration `1`, the EPS motor angular displacement and velocity input has to be provided to the model through the `EPSMotor_Ext[Angle;Angle_dt]` inport. When using configuration `2`, the EPS system torque acting on the EPS pinion has to be provided to the model through the `Trq_PowerSteering` inport.

Configuration 3 is used when an FPGA-based EPS electric motor simulation has to be integrated to the steering model. In configuration `3`, the EPS system torque acting on the EPS pinion has to be provided to the model through the **Trq_PowerSteering** inport and the EPS motor velocity calculated by the model has to be provided to FPGA APU simulation. Again, the EPS motor angle calculated by the FPGA APU should be fed back to the model through the first vector signal in the **EPSMotor_Ext[Angle;Angle_dt]** inport (the second vector value can be provided with a zero).

Based on the configuration, the equations of motion and the associated mass, the stiffness and damping matrices defining the steering system are automatically updated.

The above 2DOF configurations are more commonly used in steering test bench applications. Also the location of Steering column force elements with respect to cardan joints can be configured using *SW_LocationColumnStiffness*. The following figure shows the three different configurations which can be chosen.



1. Steering column force elements below cardan joints

2. Steering column force elements between cardan joints

3. Steering column force elements above cardan joints

Note

It is important to consider the location of steering torque sensor (T_{SC}) when choosing the correct configuration.

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Angle_Max_SteeringWheel	[deg]	Maximum steering wheel angle
Const_Damping_DriverHand	[Nm/(rad/s)]	Driver hand torsional damping
Const_Damping_EPS	[Nm/(rad/s)]	EPS system torsional damping
Const_Damping_LowerSteeringColumn	[Nm/(rad/s)]	Lower steering column torsional damping
Const_Damping_SteeringColumn	[Nm/(rad/s)]	Steering column torsional damping
Const_i_EPSSteeringColumn	[deg//deg]	Ratio angle EPS pinion to angle steering column (To be parameterized when SW_EPSLocation = 1 or 3)
Const_i_EPSSteeringRod	[deg/m]	Ratio angle EPS pinion to displacement steering rod. (To be parameterized when SW_EPSLocation = 2)
Const_Inertia_EPS	[kg m^2]	Inertia of power steering motor
Const_Inertia_LowerSteeringColumn	[kg m^2]	Equivalent inertia of lower steering column and steering rod
Const_Inertia_SteeringWheel	[kg m^2]	Equivalent inertia of steering wheel and upper steering column
Const_MaxTorque_Driver	[Nm]	Maximum driver torque
Const_Stiffness_DriverHand	[Nm/rad]	Torsional stiffness driver hand
Const_Stiffness_EPS	[Nm/rad]	Torsional stiffness EPS system
Const_Stiffness_SteeringColumn	[Nm/rad]	Torsional stiffness steering column
Const_Time_SteeringWheel	[s]	Time constant of steering wheel dynamics

Name	Unit	Description
Map_i_Steering	[deg/m]	Steering ratio [deg/m] = f(Angle_SteeringGear [deg]) Ratio of Angle_SteeringGear[deg] to Disp_SteeringRod[m] When having steering rod displacement as a function of steering gear angular displacement, use M-function `asm_steering_conversion` for calculating steering ratio.
Map_TrqSpring_SteeringColumn	[Nm]	Torque Spring Steering Column [Nm] = f(Angle_delta_SteeringColumn [rad]) Use this parameter when you have a variable steering column stiffness with respect to steering column twist angle. ▪ If this map has to be used for the simulation, set the SW_SteeringColumn_Stiffness parameter to 1. ▪ If the SW_SteeringColumn_Stiffness is set to 0, the parameter Const_Stiffness_SteeringColumn is used for the simulation.
SW_DriverHand	[1|0]	Switch for driver's hand: ▪ 0: Off ▪ 1: On
SW_EPSLocation	[1|2|3]	Switch for EPS system location: ▪ 1: Lower steering column ▪ 2: Steering rod ▪ 3: Upper steering column
SW_EPSSimulation	[0|1]	Switch for EPS system simulation: ▪ 0: Internal ▪ 1: External
SW_ExternalEPSOutput	[1|2]	Switch for external EPS simulation output: ▪ 1: States ▪ 2: Torque ▪ 3:Angle only
SW_SteeringColumn_Stiffness	[0|1]	Switch for steering column stiffness ▪ 0: Constant ▪ 1: Variable

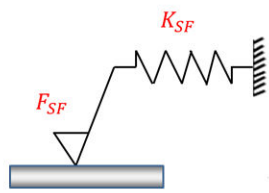
Friction elements

The friction elements used in the steering model include:

- Spring friction element
- Columb friction element
- Exponential spring friction element
- Maxwell friction element

These friction elements were used in the steering rod. Spring and exponential spring friction elements were implemented in the steering wheel. The different friction elements can be activated with a switch based on measurement results. The implementation and characteristics of different friction elements are explained in the following.

Spring friction element The spring friction element comprises a spring with linear stiffness and a slipping element which limits the maximal friction force offered by the friction element. The physical and mathematical representations of the spring friction element are shown in the following illustration.



$$F_{friction} = \begin{Bmatrix} K_{SF} \cdot (x) & if\ X_{Llim} < x < X_{Ulim} \\ F_{SF} & if\ x > X_{Ulim} \\ -F_{SF} & if\ x > X_{Llim} \end{Bmatrix}$$

where:

x is the component displacement (from point of zero velocity)

K_{SF} is the linear spring stiffness

F_{SF} is the limiting friction force

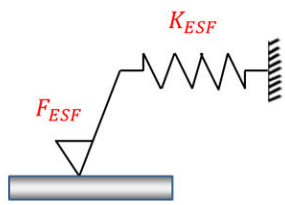
X_{Ulim} is F_{SF}/K_{SF}

X_{Llim} is $-(F_{SF})/K_{SF}$

The following table shows the parameters corresponding to spring friction element at steering rod and steering wheel.

Name	Unit	Description
Const_LimitingForce_SF_SteeringRod	[N]	Spring friction element at steering rod maximum friction force
Const_LimitingTorque_SF_SteeringWheel	[Nm]	Spring friction element at steering wheel maximum friction torque
Const_Stiffness_SF_SteeringRod	[Nm]	Spring friction element at steering rod stiffness
Const_Stiffness_SF_SteeringWheel	[Nm/rad]	Spring friction element at steering wheel stiffness
SW_SF_SteeringRod	[0|1]	Spring friction element at steering rod switch ▪ 0: Off ▪ 1: On
SW_SF_SteeringWheel	[0|1]	Spring friction element at steering wheel switch ▪ 0: Off ▪ 1: On

Exponential spring friction element The exponential spring friction element comprises a spring with exponential stiffness and a slipping element which limits the maximal friction force offered by the friction element. The physical and mathematical representations of the exponential spring friction element are shown in the following illustration.



$$F_{friction} = \begin{cases} F_{ESF} \cdot \left(1 - e^{-K_{ESF} \cdot (x)}\right) & if\ x > 0 \\ -F_{ESF} \cdot \left(1 - e^{-K_{ESF} \cdot (x)}\right) & if\ x < 0 \end{cases}$$

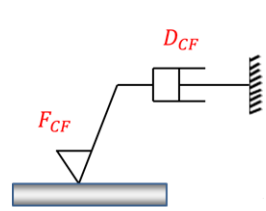
where:

x	is the component displacement (from point of zero velocity)
K_{ESF}	is the exponential spring stiffness
F_{ESF}	is the limiting friction force

The following table shows the parameters corresponding to the exponential spring friction element at steering rod and steering wheel.

Name	Unit	Description
Const_LimitingForce_ESF_SteeringRod	[N]	Exponential spring friction element at steering rod maximum friction force
Const_LimitingTorque_ESF_SteeringWheel	[Nm]	Exponential spring friction element at steering wheel maximum friction torque
Const_Stiffness_ESF_SteeringRod	[Nm]	Exponential spring friction element at steering rod stiffness
Const_Stiffness_ESF_SteeringWheel	[Nm/rad]	Exponential spring friction element at steering wheel stiffness
SW_ESF_SteeringRod	[0|1]	Exponential spring friction element at steering rod switch ▪ 0: Off ▪ 1: On
SW_ESF_SteeringWheel	[0|1]	Exponential spring friction element at steering wheel switch ▪ 0: Off ▪ 1: On

Columb friction element This friction element was carried over from the 1DOF steering model. The columb friction element comprises a static friction component and a dynamic component. The static component produces a limiting friction force. Any external force acting on the component must overcome this limit to impart motion to the component. The dynamic component comprises a damper and a slipping component. The friction force produced by the dynamic component is linearly proportional to the velocity of the body and the slipping limit for the slipping component is the same as for the static friction limit.



$$F_{St_Friction} = \begin{Bmatrix} F_{CF} & if\ F_r > F_{CF} \\ F_r & if\ |F_r| \leq F_{CF} \\ -F_{CF} & if\ F_r < -F_{CF} \end{Bmatrix}$$

$$Q_{st_Friction} = -F_{St_Friction} - D_{CF} \cdot \dot{x}$$

where:

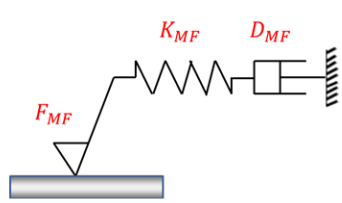
$\dot{x}$ is the component velocity

$$F_{DynamicFriction} = \begin{Bmatrix} F_{CF} & if\ Q_{st_Friction} > F_{CF} \\ F_{CF} & if\ |Q_{st_Friction}| \leq F_{CF} \\ -F_{CF} & if\ Q_{st_{Friction}} > -F_{CF} \end{Bmatrix}$$

The following table shows the parameters corresponding to the columb friction element at steering rod:

Name	Unit	Description
Const_Damping_CF_SteeringRod	[N/(m/s)]	Columb friction element at steering rod damping
Const_LimitingForce_CF_SteeringRod	[N]	Columb friction element at steering rod maximum friction force
SW_CF_SteeringRod	[0|1]	Columb friction element at steering rod switch ▪ 0: Off ▪ 1: On

Maxwell friction element The maxwell friction element comprises a spring with exponential stiffness in series with a damper and a slipping element which limits the friction force offered by the friction element. The physical and mathematical representations of the maxwell friction element are shown below.



$$F_{Friction} = \begin{Bmatrix} F_{MF} \cdot \left[1 - e^{K_{MF}}\left(0_x - \int \frac{F_{Friction}}{D_{MF}}\right)\right] & if\ 0_x > 0 \\ -F_{MF} \cdot \left[1 - e^{K_{MF}}\left(0_x - \int \frac{F_{Friction}}{D_{MF}}\right)\right] & if\ 0_x < 0 \end{Bmatrix}$$

where:

0_x is the component displacement (from origin)

K_{MF} is the exponential spring stiffness

D_{MF} is the maxwell damping coefficient

F_{MF} is the limiting friction force

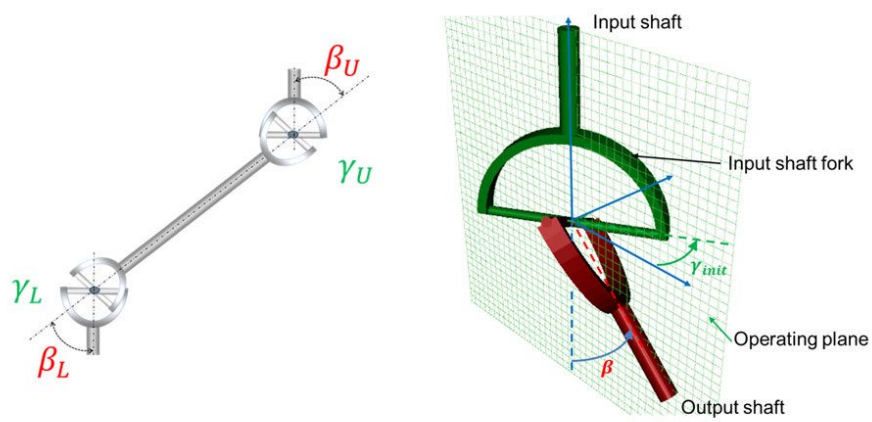
The following table shows the parameters corresponding to the maxwell friction element at steering rod.

Name	Unit	Description
Const_Damping_MF_SteeringRod	[N/(m/s)]	Maxwell friction element at steering rod damping
Const_LimitingForce_MF_SteeringRod	[N]	Maxwell friction element at steering rod maximum friction force
Const_Stiffness_MF_SteeringRod	[N/m]	Maxwell friction element at steering rod stiffness
SW_MF_SteeringRod	[0|1]	Maxwell friction element at steering rod switch ▪ 0: Off ▪ 1: On

Cardan joints

Two cardan joints are added to the steering system, to simulate the effect of cardan joints on the steering system states.

The key parameters which define the cardan joints are explained in the following figure.



γ_{init} – *Initial angle – Cardan Joint (Angle between operating plane and Input shaft fork of cardan joint)*

Operating Plane – Plane comprising of input and output shaft axis

β – *Cardan Angle – Acute angle measured between input and output shaft (measured in operating plane)*

β_U – *Cardan angle upper joint*

γ_U – *Initial angle upper cardan joint*

β_L – *Cardan angle lower joint*

γ_L – *Initial angle lower cardan joint*

Note

It is important to consider the direction in which the offset angles are measured. Here, angles measured in anticlockwise direction are positive.

Parameters The following table shows the parameters:

Name	Unit	Description
SW_LocationColumnStiffness	[1|2|3]	Switch for steering column stiffness location ▪ 1: Below cardan joints

Name	Unit	Description
		▪ 2: Between cardan joints ▪ 3: Above cardan joints
Const_InitialAngle_UpperCardanJoint	[deg]	Initial angular offset of upper cardan joint (angle between operating plane and input shaft fork)
Const_InitialAngle_LowerCardanJoint	[deg]	Initial angular offset of lower cardan joint (constant velocity drives - 90deg)
Const_CardanAngle_UpperJoint	[deg]	Cardan angle of upper cardan joint
Const_CardanAngle_LowerJoint	[deg]	Cardan angle of lower cardan joint

Related topics

References

Steering 3DOF Variable (ModelDesk Parameterizing 🕮)

Variable Steering Ratio

Description

A variable steering ratio can be simulated with the STEERING_VARIABLE_RATIO block. You must parameterize two 1-D look-up tables, one containing the steering ratio and one containing the steering column angle, as functions of the displacement of the steering rod.

If only the steering torque at the steering column is considered, the generalized steering force at the steering rod is calculated by:

$$Q_{St} = i_{St}(q_{St}) \cdot Trq_{St_Col}$$

where

$$Trq_{St_Col} = k_{St} \cdot \left(\delta_{St_{Wheel}} - \delta_{St_{Gear}}\right) + d_{St} \cdot \left(\dot{\delta}_{St_{Wheel}} - \dot{\delta}_{St_{Gear}}\right) = k_{St} \cdot \left(\delta_{St_{Wheel}} - \delta_{St_{Gear}}(q_{St})\right) + d_{St} \cdot \left(\dot{\delta}_{St_{Wheel}} - i_{St}(q_{St}) \cdot \dot{u}_{St_Gear}\right)$$

Both the steering ratio $i_{St}(q_{St})$ and the angle at the input shaft of the steering gearbox $\delta_{St_{Gear}}(q_{St})$ are functions of the steering rod displacement q_{St}.

If the parameterization of the steering ratio is available only as a function of the steering wheel angle instead of the steering rod displacement, you can use the MATLAB function `asm_steering_conversion` to transform the parameter table.

Inports and outports

The interfaces for inports and outports are the same as for the STEERING subsystem with constant steering ratio. Refer to Steering on page 237.

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Angle_Max_SteeringWheel	[deg]	Maximal steering wheel angle
Const_Coeff_Transient_Friction	[N/(m/s)]	Transient friction coefficient
Const_Damping_SteeringColumn	[Nm/(rad/s)]	Damping, steering column
Const_Force_Friction_max	[N]	Maximum friction force in the steering
Const_m_SteeringRod	[kg]	Mass of steering rod
Const_NaturalFrequency_SteeringCompliance	[1/s]	Natural frequency for the steering compliance dynamics
Const_Stiffness_SteeringColumn	[Nm/rad]	Torsion stiffness, steering column
Const_Time_PowerSteering	[s]	Time constant, power steering force dynamic
Const_Time_SteeringWheel	[s]	Time constant, steering wheel dynamic
Map_i_Steering	[deg/m]	Steering ratio [deg/m] = f(Displacement steering rod [m])
Map_phi_SteeringColumn	[deg]	Angle steering column [deg] = f(Displacement steering rod [m])
Map_SteeringCompliance	[m]	Additional steering rod displacement [m] = f(force at steering rod [N]
Map_Torque_PowerSteering	[Nm]	Torque power steering[Nm] = f(Torque_SteeringColumn [Nm])
StepSize	[s]	Simulation step size

Related topics

References

Steering Variable (ModelDesk Parameterizing 🕮)

Tire

Where to go from here

Information in this section

Additional CP Calculations.....260
The block can be used to calculate additional geometric values of the tire contact point.

Contact Point Calculation.....261
The characteristics of the tire-road contact point can be evaluated according to the wheel position and orientation in the earth coordinate system.

Tire Forces and Torques Vector.....262
The forces and torques calculated in the tire model are oriented to the contact point coordinate system and must therefore be transformed to the vehicle coordinate system to be used in solving the vehicle equation of motion.

Tire Mode.....264
The Magic Formula 6.1 implementation uses the ISO sign convention.

Tire Model.....265
ASM Vehicle Dynamics offers multiple tire models.

Magic Formula Tire Model.....266
The model uses the Magic Formula function to approximate the tire characteristics and the interaction between the longitudinal and lateral forces.

Tire Model Magic Formula 6.1.....268
The model uses the Magic Formula mathematical function to approximate the tire characteristics and the interaction between the longitudinal and lateral forces.

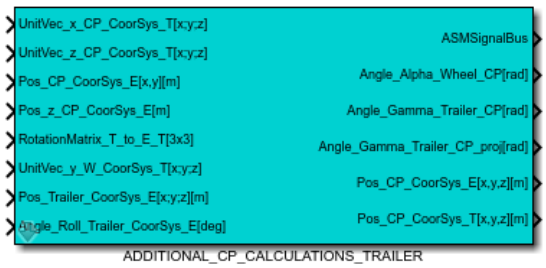
Tire Model TMeasy.....271
The model contains a simplified formula that approximates the tire characteristics and the interaction between the longitudinal and lateral forces.

Tire Velocity Sampling Points.....273
The block is used to parameterize tire parameters depending on the longitudinal velocity.

Additional CP Calculations

Description

The ADDITIONAL_CP_CALCULATIONS_TRAILER block can be used to calculate additional geometric values of the tire contact point. This can be used for detailed tire forces investigations.



Inports

The following table shows the inports:

Name	Unit	Description
Angle_Roll_Vehicle_CoorSys_E	[deg]	Roll angle of the vehicle in earth coordinate system
Pos_CP_CoorSys_E	[m]	xy-position vector of tire contact point in earth coordinate system
Pos_Vehicle_CoorSys_E	[m]	Position vector of vehicle reference point in earth coordinate system
Pos_z_CP_CoorSys_E	[m]	z-position of tire contact point in earth coordinate system
RotationMatrix_V_to_E_T	[]	Transposed rotation matrix from vehicle to earth coordinate system
UnitVec_x_CP_CoorSys_V	[]	Unit vector in x-axis direction of the contact point coordinate system
UnitVec_y_W_CoorSys_V	[]	Unit vector in y-axis direction of the wheel coordinate system
UnitVec_z_CP_CoorSys_V	[]	Unit vector in z-axis direction of the contact point coordinate system

Outports

The following table shows the outports:

Name	Unit	Description
Angle_Alpha_Wheel_CP	[rad]	Alpha-angle around the x-axis in the tire contact point
Angle_Gamma_Vehicle_CP	[rad]	Gamma-angle about vehicle z-axis
Angle_Gamma_Vehicle_CP_proj	[rad]	Gamma-angle about vehicle z-axis, projected into road plane
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Pos_CP_CoorSys_E	[m]	Position vector of tire contact point in earth coordinate system
Pos_CP_CoorSys_V	[m]	Position vector of tire contact point in vehicle coordinate system

Parameters

The subsystem requires no parameters to perform its calculation.

Contact Point Calculation

Description

The characteristics of the tire-road contact point can be evaluated according to the wheel position and orientation in the earth coordinate system, index E. The speeds of the contact point in the contact point coordinate system, index CP, are also calculated in this subsystem.

The estimated position of the contact point is sent to the road model, which calculates the contact point height, the road friction coefficient and the tire parameters set at the contact point. A unit vector of the normal track is also calculated in the road model.

From this data, and from the speed of the wheel center in the coordinate system, index V, the speeds of the contact point in x and y directions are calculated, as well as the tire current radius and the tire bore speed. This data is then given to the tire model to calculate the tire forces and torques.

For detailed information, refer to Contact Point Calculation (ASM Vehicle Dynamics Addendum 🕮).

Inports

The following table shows the inports:

Name	Unit	Description
Pos_Wheels_CoorSys_E[x,y,z]	[m]	Position of wheel center in earth coordinate system
Pos_z_CP_CoorSys_E	[m]	Road hight at the contact point
RotationMatrix_V_E_T[3x3]	[]	Rotation matrix from earth to vehicle coordinate system

Name	Unit	Description
UnitVec_y_W_CoorSys_V[x,y,z]	[]	Unit vector in y-axis direction of the wheel coordinate system
UnitVec_z_CP_CoorSys_E[x,y,z]	[]	Unit vector perpendicular to contact point, described in earth coordinate system
v_Abs_WheelBody[x y z]	[m/s]	Absolute velocity of the wheel
w_Abs_Wheel[x,y,z]	[rad/s]	Absolute angular velocity of the wheel

Outports

The following table shows the outports:

Name	Unit	Description
Pos_CP_CoorSys_E[x,y]	[m]	Estimated x- and y-position of the contact point
r_Tire	[m]	Tire radius (scalar value)
r_Tire_CoorSys_V[x,y,z]	[m]	Tire radius vector in vehicle coordinate system
Sin_Angle_Alpha_Tire	[]	Sin camber angle between tire and road
UnitVec_x_CP_CoorSys_V[x,y,z]	[]	Unit vector in x-axis direction of the contact point coordinate system
UnitVec_y_CP_CoorSys_V[x,y,z]	[]	Unit vector in y-axis direction of the contact point coordinate system
UnitVec_z_CP_CoorSys_V[x,y,z]	[]	Unit vector in z-axis direction of the contact point coordinate system
v_x_Wheel_CoorSys_CP	[m/s]	Tire speed in tire longitudinal direction
v_y_Wheel_CoorSys_CP	[m/s]	Tire speed in tire lateral direction
w_z_Wheel_CoorSys_CP	[rad/s]	Angular speed in direction of the contact point perpendicular

Parameters

With the above inputs and outputs, this subsystem requires no parameters to perform its calculation.

Related topics

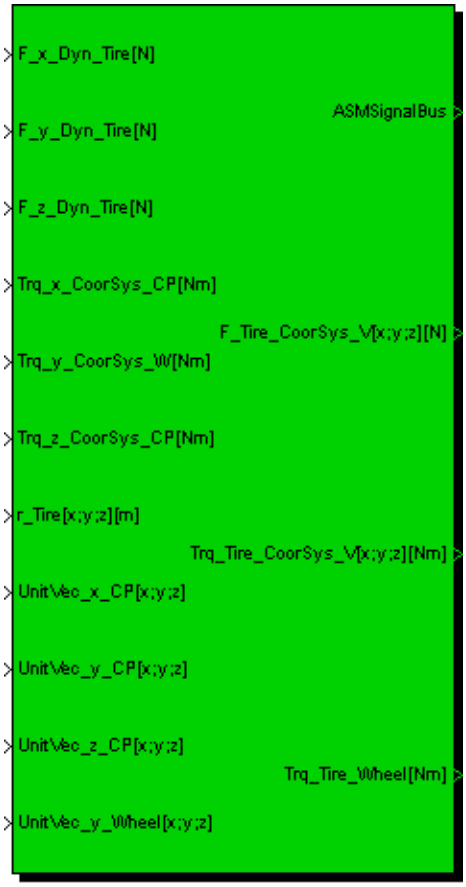
References

Contact Point Calculation (ASM Vehicle Dynamics Addendum 🕮)

Tire Forces and Torques Vector

Description

The forces and torques calculated in the tire model are oriented to the contact point coordinate system, index CP, which is not normally on the vehicle coordinate system due to wheel and vehicle orientation. The tire forces and torques vectors must therefore be transformed to the vehicle coordinate system to be used in solving the vehicle equation of motion.



TIRE_FORCES_AND_TORQUES_VECTORS_FR

The effective components of the tire forces and torques in the direction of wheel rotation are the rolling resistance torque and the longitudinal (circumferential) force multiplied by the wheel radius. These components are calculated in this block and sent to the WHEEL_SPEED block.

Inports

The following table shows the inports:

Name	Unit	Description
F_x_Dyn_Tire_CoorSys_CP	[N]	Longitudinal dynamic force
F_y_Dyn_Tire_CoorSys_CP	[N]	Lateral dynamic force
F_z_Dyn_Tire_CoorSys_CP	[N]	Vertical dynamic force
r_Tire[x,y,z]	[m]	Tire radius vector
Trq_x_CoorSys_CP	[Nm]	Torque around the x-axis of the contact point
Trq_y_CoorSys_W	[Nm]	Torque around the y-axis of the wheel
Trq_z_CoorSys_CP	[Nm]	Torque around the z-axis of the contact point
UnitVec_x_CP[x,y,z]	[]	Unit vector in x-axis direction of the contact point coordinate system
UnitVec_y_CP[x,y,z]	[]	Unit vector in y-axis direction of the contact point coordinate system

Name	Unit	Description
UnitVec_y_Wheel[x,y,z]	[]	Unit vector in y-axis direction of the wheel coordinate system
UnitVec_z_CP[x,y,z]	[]	Unit vector in z-axis direction of the contact point coordinate system

Outports

The following table shows the outports:

Name	Unit	Description
F_Tire_CoorSys_V[x;y;z]	[N]	Tire force vector in vehicle coordinate system
Trq_Tire_CoorSys_V[x;y;z]	[Nm]	Tire torque vector in vehicle coordinate system
Trq_Tire_Wheel	[Nm]	Torque from tire to the wheel, i.e., effect of circumferential force and rolling resistance torque

Parameters

With the above inputs and outputs, this subsystem requires no parameters to perform its calculation.

Tire Mode

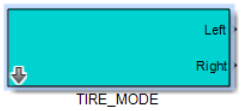
Introduction

The Magic Formula 6.1 implementation uses the ISO sign convention. In the ISO sign convention, the x-axis is the vehicle positive direction, the y-axis is to the left, and z-axis is the vertical axis moving upward

In most cases, the parameter of Magic Formula tire shows asymmetric behavior. In this subsystem, the following tire mode calculations can be chosen:

- Tire mode 1: normal calculation
- Tire mode 2: mirror calculation

For example: If the "LEFT" tire parameter is mounted on the "RIGHT" side of the vehicle, then mirroring the right tire is applied on the tire calculation and the total vehicle behaves symmetrically.



Outports

The following table shows the outports:

Name	Unit	Description
Left	[]	Tire mode for the left side. ▪ 1: Normal ▪ 2: Mirror

Name	Unit	Description
Right	[]	Tire mode for the right side. ▪ 1: Normal ▪ 2: Mirror

Parameters

The following table shows the parameters:

Name	Unit	Description
ModeLeft	[]	Tire mode for the left side. ▪ 1: Normal ▪ 2: Mirror
ModeRight	[]	Tire mode for the right side. ▪ 1: Normal ▪ 2: Mirror

Related topics

Basics

Magic Formula Tire Model (Earlier than Version 6.1) (ASM Vehicle Dynamics Addendum 🕮)

References

Tire Model Magic Formula 6.1 268

Tire Model

Model overview

Several types of mathematical models were developed over the last few decades, each of them for a specific purpose and with a different level of accuracy and complexity. Real-time applications use empirical tire models that describe measured tire characteristics by means of a table or mathematical formula and certain interpolation schemes, which reduces the computation time.

ASM Vehicle Dynamics offers the following tire models:

- Tire Model TMeasy on page 271
- Magic Formula Tire Model on page 266
- Tire Model Magic Formula 6.1 on page 268

The principle of approximating measured tire forces by an interpolation curve described by various parameters is the same for both models, but the formulations of the interpolation curves are different.

TMEasy has the advantage that it can be easily parameterized via some supporting points on the friction force curves. The parameters of Magic Formula tire model are less intuitive but they can be provided by the tire manufacturers, as this model is a well-known standard in tire forces and torques simulation.

ASM Vehicle Dynamics lets you switch between 4 sets of tire parameters to simulate different surface conditions which affect not only the friction coefficient but also the forces and torques curve shapes. The friction coefficient can also be changed.

Tip

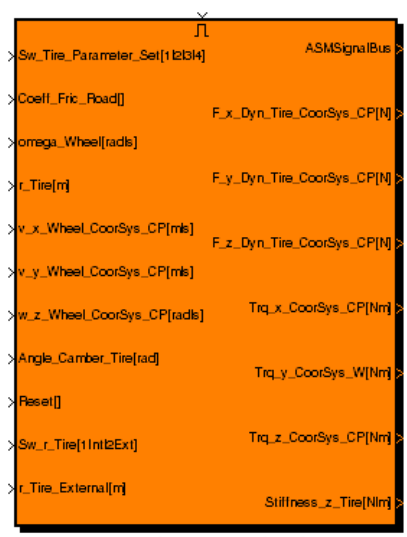
The friction coefficient is defined in the road model and then passed to the tire model. The friction coefficient can also be defined for the tire regardless of the value saved in the road model by setting the value externally.

Magic Formula Tire Model

Description

A well-known and widely used model is the Magic Formula by Prof. Pacejka, Delft University of Technology, Delft, Netherlands. It is a semi-empirical tire model that describes longitudinal and lateral tire force characteristics and the self-aligning torque as functions of longitudinal and/or lateral slip.

The model uses the mathematical function Magic Formula to approximate the tire characteristics and the interaction between the longitudinal and lateral forces.



The Magic Formula tire model requires more parameters than the TMeasy tire model, and interpolation using the Magic Formula requires more computation time. The parameters are often provided by the tire manufacturer. As an

advantage over TMEasy, the model includes the effect of the wheel load and there is no need to define two sets of parameters.

The Magic Formula tire model includes the following features:

- Dynamic tire radius used to calculate the slip quantities
- Combined slip force calculation
- Self-aligning torque
- Scalable friction coefficient
- Longitudinal and lateral tire dynamics
- Wheel load influence
- Rolling resistance torque and tipping torque

For detailed information on the model, refer to Magic Formula Tire Model (Earlier than Version 6.1) (ASM Vehicle Dynamics Addendum 🕮).

Inports

The following table shows the inports:

Name	Unit	Description
Angle_Camber_Tire	[rad]	Camber angle between tire and road
Coef_Fric_Road	[]	Friction coefficient of the road at the contact point. This value is evaluated in the road model. It can also be set externally.
omega_Wheel	[rad/s]	Rotational speed of the wheel around its axis of rotation
Reset	[]	Resets all integrators in the block to their initial conditions.
r_Tire	[m]	Tire radius (scalar value)
r_Tire_External	[m]	External tire radius
Sw_r_Tire	[1|2]	Switch between internal tire radius and external tire radius: ▪ 1: Internal tire radius ▪ 2: External tire radius
Sw_Tire_Parameter_Set	[1|2|3|4]	Switch between four sets of tire parameters. This value is evaluated in the road model. It can also be set externally.
Trq_Shaft_CoorSys_V [FL,FR,RL,RR][x;y;z]	[Nm]	Driving torque on the wheel in vehicle coordinate system
v_x_Wheel_CoorSys_CP	[m/s]	Wheel speed in longitudinal direction
v_y_Wheel_CoorSys_CP	[m/s]	Wheel speed in lateral direction
w_z_Wheel_CoorSys_CP	[rad/s]	Angular speed in the perpendicular direction of the contact point

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
F_x_Dyn_Tire_CoorSys_CP	[N]	Longitudinal dynamic force
F_y_Dyn_Tire_CoorSys_CP	[N]	Lateral dynamic force

Name	Unit	Description
F_z_Dyn_Tire_CoorSys_CP	[N]	Vertical dynamic force
Stiffness_z_Tire	[N/m]	Vertical tire stiffness
Trq_x_CoorSys_CP	[Nm]	Overturning coupling torque
Trq_y_CoorSys_W	[Nm]	Rolling resistance torque
Trq_z_CoorSys_CP	[Nm]	Self-aligning torque

Parameters

The following table shows the parameters:

Name	Unit	Description
Para_Set_1	[]	Parameter set number 1
Para_Set_2	[]	Parameter set number 2
Para_Set_3	[]	Parameter set number 3
Para_Set_4	[]	Parameter set number 4

Related topics

Basics

Magic Formula Tire Model (Earlier than Version 6.1) (ASM Vehicle Dynamics Addendum 🕮)

References

Tire <xyz> Parameter Set <n> (Magic Formula) (ModelDesk Parameterizing 🕮)

Tire Model Magic Formula 6.1

Description

A well-known and widely used model is the Magic Formula by Prof. Pacejka, Delft University of Technology, Delft, Netherlands. It is a semi-empirical tire model that describes longitudinal and lateral tire force characteristics and the self-aligning torque as functions of longitudinal and/or lateral slip.

The model uses the mathematical function Magic Formula to approximate the tire characteristics and the interaction between the longitudinal and lateral forces.

The Magic Formula tire model includes the following features:

- Dynamic tire radius used to calculate the slip quantities
- Combined slip force calculation
- Self-aligning torque
- Scalable friction coefficient
- Wheel load influence
- Calculation of rolling resistance torque and tipping torque
- Wheel inclination angle (camber angle) influence
- Tire inflation pressure effect

For detailed information on the model, refer to Magic Formula Tire Model 6.1 (ASM Vehicle Dynamics Addendum 🕮).

Inports

The following table shows the inports:

Name	Unit	Description
Coef_Fric_Road	[]	Friction coefficient of the road at the contact point. This value is evaluated in the road model. It can also be set externally.
Mode	[1|2]	Switch between mode calculation: ▪ 1: normal tire calculation ▪ 2: mirror tire calculation
omega_Wheel	[rad/s]	Rotational speed of the wheel around its axis of rotation
p	[Pa]	Tire inflation pressure
r_Tire	[m]	Tire radius (scalar value)
r_Tire_External	[m]	External tire radius
ResetOrDisable	[]	Reset all integrators in the block to their initial conditions or disable the calculation.
Sin_Angle_Alpha_Tire	[]	Sin camber angle between tire and road

Name	Unit	Description
Sw_r_Tire	[1|2]	Switch between internal tire radius and external tire radius: ▪ 1: Internal tire radius ▪ 2: External tire radius
Sw_Tire_Parameter_Set	[1|2|3|4]	Switch between four sets of tire parameters. This value is evaluated in the road model. It can also be set externally.
v_x_Wheel_CoorSys_CP	[m/s]	Wheel speed in longitudinal direction
v_y_Wheel_CoorSys_CP	[m/s]	Wheel speed in lateral direction
w_z_Wheel_CoorSys_CP	[rad/s]	Angular speed in the perpendicular direction of the contact point

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus containing signals of ASM components
ForceTorque_cz	[]	Array with the following components: ▪ 1: Force Fx [N] ▪ 2: Force Fy [N] ▪ 3: Force Fz [N] ▪ 4: Torque Mx [Nm] ▪ 5: Torque My [Nm] ▪ 6: Torque Mz [Nm] ▪ 7: Vertical stiffness cz [N/m]

Parameters

The following table shows the parameters:

Name	Unit	Description
Base	[]	Base parameters
Para_Set_1	[]	Geometry and scaling parameters set number 1
Para_Set_2	[]	Geometry and scaling parameters set number 2
Para_Set_3	[]	Geometry and scaling parameters set number 3
Para_Set_4	[]	Geometry and scaling parameters set number 4

Related topics

Basics

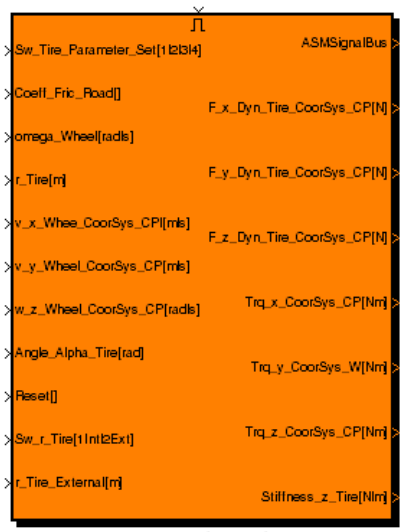
Introduction (ASM Vehicle Dynamics Addendum 🕮)

Tire Model TMeasy

Description

The model was developed by Prof. G. Rill, College of Regensburg, Germany. It is a semi-empirical tire model, which describes longitudinal, lateral tire force characteristics and the self-aligning torque as functions of longitudinal and lateral slip.

The model contains a simplified formula that approximates the tire characteristics and the interaction between the longitudinal and lateral forces. Due to the simplified inter-extrapolation formulas used in the model, it requires less computation time than the Magic Formula tire model.



TIRE_MODEL_TMEASY_TRAILER_FL

Calculating the tire forces and torques requires parameters that can be easily obtained from empirical tire force curves. These parameter sets have to be specified at the nominal and the double nominal wheel load to cover a wider range of operation. A description of the tire's dynamic behavior is also available by means of a first-order dynamic.

The TMeasy tire model includes the following features:

- Dynamic tire radius used to calculate the slip quantities
- Combined slip force calculation
- Self-aligning torque and bore torque
- Scalable friction coefficient
- Longitudinal and lateral tire dynamics
- Wheel load influence
- Rolling resistance torque and tipping torque

For detailed information on the model, refer to EasyToUse Tire Model (ASM Vehicle Dynamics Addendum 🕮).

Inports

The following table shows the inports:

Name	Unit	Description
Angle_Camber_Tire	[rad]	Camber angle between tire and road
Coef_Fric_Road	[]	Friction coefficient of the road at the contact point. This value is evaluated in the road model, it can also be set externally.
omega_Wheel	[rad/s]	Rotational speed of the wheel about its axis of rotation
Reset	[]	Resets all integrators in the block to their initial conditions
r_Tire	[m]	Tire radius (scalar value)
r_Tire_External	[m]	External tire radius
Sw_r_Tire	[1|2]	Switch between internal tire radius and external tire radius. ▪ 1: Internal tire radius ▪ 2: External tire radius
Sw_Tire_Parameter_Set	[1|2|3|4]	Switch between four sets of tire parameters. This value is evaluated in the road model, it can also be set externally.
v_x_Wheel_CoorSys_CP	[m/s]	Wheel speed in longitudinal direction
v_y_Wheel_CoorSys_CP	[m/s]	Wheel speed in lateral direction
w_z_Wheel_CoorSys_CP	[rad/s]	Angular speed in direction of the contact point perpendicular

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
F_x_Dyn_Tire_CoorSys_CP	[N]	Longitudinal dynamic force
F_y_Dyn_Tire_CoorSys_CP	[N]	Lateral dynamic force
F_z_Dyn_Tire_CoorSys_CP	[N]	Vertical dynamic force
Stiffness_z_Tire	[N/m]	Vertical tire stiffness
Trq_x_CoorSys_CP	[Nm]	Tipping torque
Trq_y_CoorSys_W	[Nm]	Rolling resistance torque
Trq_z_CoorSys_CP	[Nm]	Self-aligning torque plus bore torque

Parameters

The following table shows the parameters:

Name	Unit	Description
Para_Set_1	[]	Parameter set number 1
Para_Set_2	[]	Parameter set number 2
Para_Set_3	[]	Parameter set number 3
Para_Set_4	[]	Parameter set number 4

Related topics

References

Tire <xyz> Parameter Set <n> (TMEasy) (ModelDesk Parameterizing 🕮)

Tire Velocity Sampling Points

Description

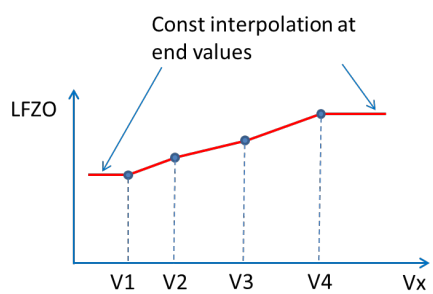
The TIRE_VEL_SAMPLING_POINTS block is used to parameterize tire parameters depending on the longitudinal velocity.

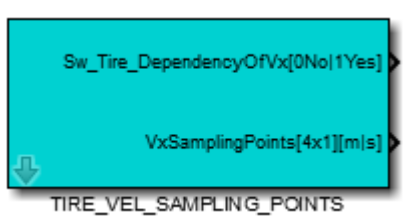
Four tire velocity sampling points can be defined. They should be strictly increasing. If the SW_Tire_DependencyOfVelocity flag is on (= 1), the four tire parameter sets are interpreted as tire parameters for four different tire longitudinal velocities. If the SW_Tire_DependencyOfVelocity is off (= 0), the four tire parameter sets are interpreted as *dry*, *damp*, *wet*, and *icy* road.

Note

At this point, the block is only supported by Magic Formula tire models and only the scaling parameters of the Magic Formula are interpolated between different velocities.

The following figure shows an example of the interpolation of the scaling parameter `LFZO`, depending on the tire longitudinal velocity (`LFZO=f(V_x)`):





Outports

The following table shows the outports:

Name	Unit	Description
Sw_Tire_DependencyOfVx	[1|2]	Flag to switch on/off the tire dependency of velocity: ▪ 1: No ▪ 2: Yes
VxSamplingPoints[4x1]	[m/s]	Four velocity sampling points. The values must be strictly increasing.

Parameters

The following table shows the parameters:

Name	Unit	Description
SW_Tire_DependencyOfVelocity	[]	Switch to activate/deactivate the tire dependency of velocity. At this point, only Magic Formula tires can have this feature.
VelocitySamplingPoints	[km/h]	Four velocity sampling points. The values must be strictly monotonically increasing.

Related topics

References

Tire Vel Sampling Points (ModelDesk Parameterizing 🕮)

Vehicle Movement

Where to go from here

Information in this section

Coordinate Transformation.....275
The transformation matrix between the vehicle reference coordinate system and the earth coordinate system is calculated in this block.

Mass Forces and Torques.....279
To calculate the generalized forces and torques, it is necessary to calculate not only the external forces and torques, i.e., tire and aerodynamics, but also the mass forces and torques.

Sensor Motion.....281
This block calculates the simulation variables, for example, acceleration and velocity, at a certain point in the vehicle that you define.

Vehicle and Wheel Velocities.....283
In this subsystem, the velocities of the vehicle center of gravity and the wheels are calculated according to the generalized degrees of freedom, position of CoG, and the wheels' positions and relative velocities.

Vehicle Mass and Additional Loads.....284
ASM Vehicle Dynamics lets you define six additional loads, each at a different position in the vehicle.

Vehicle Motion.....287
In the VEHICLE_MOTION_CAR block, the MBS technique is used to write the equation of motion.

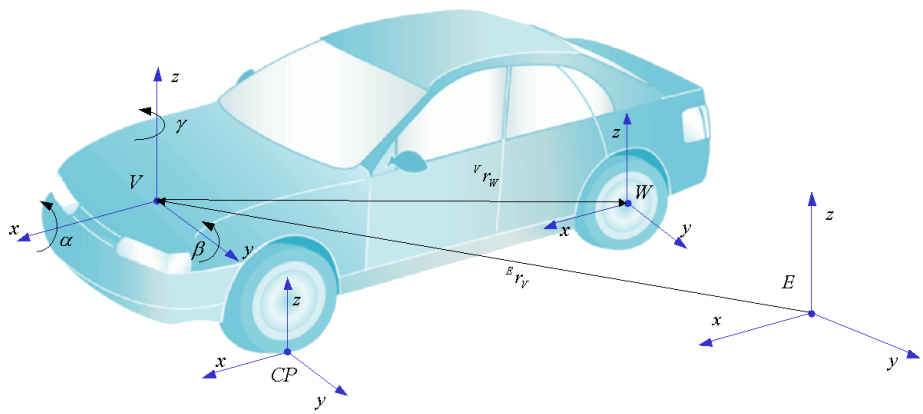
Vehicle Movement Info Car.....291
The block collects the signals from the blocks inside the Vehicle Movement subsystems and calculates typical vehicle data.

Coordinate Transformation

Description

To describe the kinematics of vehicle dynamics adequately, a system of several coordinate systems, all rotating clockwise, is defined as a basis for modeling. The following illustration shows the systems used in the ASM Vehicle Dynamics Model.

In the following explanations, the superscript of a variable indicates the coordinate system in which the variable is described. The subscript indicates the relevant body of the variable. For example, ${}^{E}r_{V}$ is the position vector of the vehicle (subscript $_V$) described in the earth coordinate system (superscript ^{E}).



As shown in the illustration, the following coordinates are used:

- Earth coordinate system, index E

  represents the fixed reference system.
- Vehicle reference coordinate system, index V

  is fixed to the vehicle body. Its origin is at zero position – configuration position – at mid-point between the front wheel centers. The x-axis is in the longitudinal direction of the vehicle and points forwards, the y-axis points to the vehicle's left, and the z-axis direction follows the right-hand rule and points upwards.
- Wheel coordinate system, index W

  is at the wheel center. Its orientation is determined by the wheel orientation, which depends on the suspension kinematics.
- Contact point coordinate system, index CP

  has its origin in the contact point. The x-y plane is parallel to the road local plane. For details on calculating the contact point coordinate system, refer to Contact Point Calculation (ASM Vehicle Dynamics Addendum 🕮).

All variables are described in the vehicle coordinate system. To describe these variables in the earth coordinate system, index E, coordinate transformation must be performed.

Vehicle degrees of freedom

The position and orientation of the vehicle reference coordinate system is a function of the following vehicle degrees of freedom:

$$\begin{bmatrix}\dot{q}_1\\ \dot{q}_2\\ \dot{q}_3\end{bmatrix} = {}^{V}\begin{bmatrix}V_x\\ V_y\\ V_z\end{bmatrix}_V$$

The translatory vehicle velocities of the origin of coordinate system V in x, y, and z directions described in vehicle reference coordinate system V

$$\begin{bmatrix}\dot{q}_4\\ \dot{q}_5\\ \dot{q}_6\end{bmatrix} = {}^{V}\begin{bmatrix}\omega_x\\ \omega_y\\ \omega_z\end{bmatrix}_V$$

The angular vehicle velocities about the x, y, and z axes of the vehicle reference coordinate system V

Note

The other vehicle degrees of freedom are not used to determine the position and orientation of the vehicle coordinate system against the earth coordinate system.

The orientation angles roll, pitch, and yaw are calculated from the degrees of freedom listed above

$$\begin{bmatrix}\dot{\alpha}\\ \dot{\beta}\\ \dot{\gamma}\end{bmatrix} = \begin{bmatrix}1 & 0 & -\sin(\beta)\\ 0 & \cos(\alpha) & \sin(\alpha)\cos(\beta)\\ 0 & -\sin(\alpha) & \cos(\alpha)\cos(\beta)\end{bmatrix}^{-1}{}^{V}\begin{bmatrix}\omega_x\\ \omega_y\\ \omega_z\end{bmatrix}_V$$

where:

α is the roll angle

β is the pitch angle

γ is the yaw angle

The vehicle orientation angles can then be calculated by integration.

The overall rotation matrix can now be calculated by multiplying the subsequence rotation matrices as follows:

$${}^{E}T_V = T_z(\gamma)T_y(\beta)T_x(\alpha) = \begin{bmatrix}\cos(\gamma) & -\sin(\gamma) & 0\\ \sin(\gamma) & \cos(\gamma) & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\cos(\beta) & 0 & \sin(\beta)\\ 0 & 1 & 0\\ -\sin(\beta) & 0 & \cos(\beta)\end{bmatrix}\begin{bmatrix}1 & 0 & 0\\ 0 & \cos(\alpha) & -\sin(\alpha)\\ 0 & \sin(\alpha) & \cos(\alpha)\end{bmatrix}$$

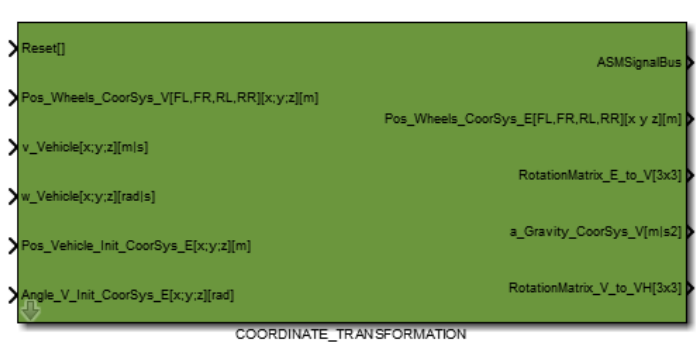
Using the rotation matrix, the vehicle velocity in the earth coordinate system is calculated as follows:

$${}^{E}V_V = {}^{E}T_V{}^{V}V_V$$

The vehicle position can then be calculated by integration:

$${}^{E}r_V = {}^{E}\begin{bmatrix}x\\ y\\ z\end{bmatrix}_V = {}^{E}\begin{bmatrix}x_0\\ y_0\\ z_0\end{bmatrix}_V + \int {}^{E}\begin{bmatrix}v_x\\ v_y\\ v_z\end{bmatrix}_V dt$$

The above equations provide the position and orientation of the vehicle reference coordinate system against the earth coordinate system. The following illustration shows the Simulink implementation of this block.



The transformation matrix between the vehicle reference coordinate system and the earth coordinate system ${}^{E}T_{V}$ is calculated in this block. This matrix is then used to calculate the position of the vehicle and the wheels in the earth coordinate system. The inverse of the transformation matrix is used to calculate the acceleration gravity in the vehicle reference coordinate system.

The position vector of each wheel ${}^{E}r_{Wi}$ is calculated as follows:

$${}^{E}r_{Wi} = {}^{E}r_{V} + {}^{E}T_{V}\,{}^{V}r_{Wi}$$

The acceleration of gravity ${}^{V}g$ in the vehicle reference coordinate system is calculated as follows:

$${}^{V}g = {}^{E}T^{-1}{}_{V}\,{}^{E}g$$

Inports

The following table shows the inports:

Name	Unit	Description
Angle_V_Init_CoorSys_E[x;y;z]	[rad]	Initial vehicle orientation angles
Pos_Vehicle_Init_CoorSys_E[x;y;z]	[m]	Initial vehicle position
Pos_Wheels_CoorSys_V[FL,FR,RL,RR][x;y;z]	[m]	Wheel relative position in vehicle coordinate system
Reset	[]	Resets all integrators in the block to their initial conditions
v_Vehicle[x,y,z]	[m/s]	Velocity of vehicle reference system
w_Vehicle[x,y,z]	[rad/s]	Angular velocity of vehicle reference system

Outports

The following table shows the outports:

Name	Unit	Description
a_Gravity_CoorSys_V	[m/s^2]	Acceleration of gravity represented in the vehicle reference coordinate system
Pos_Vehicle_CoorSys_E[x y z]	[m]	Position of vehicle reference coordinate system in the earth coordinate system
Pos_Wheels_CoorSys_E[FL,FR,RL,RR][x;y;z]	[m]	Wheel position in the earth coordinate system
RotationMatrix_V_to_E_Trans[3x3]	[]	Rotation matrix from earth to vehicle reference coordinate system
RotationMatrix_V_to_VH[3x3]	[]	Rotation matrix from vehicle horizontal coordinate system (xy-plane of global coordinate system) to vehicle reference coordinate system.

Parameters

With the above inputs and outputs, this subsystem requires no parameters to perform its calculation.

Mass Forces and Torques

Description

To calculate the generalized forces and torques, it is necessary to calculate not only the external forces and torques, i.e., tire and aerodynamics, but also the mass forces and torques. The mass forces and torques depend on residual acceleration, gravity acceleration, and angular velocity.

Residual Acceleration and Mass Force

In the acceleration expression, there are some terms which are not the time derivative of the velocity components. These terms are collected in the term residual acceleration. For instance, for any given wheel, the expression for the acceleration a_{Wi} can be written as follows

$$a_{Wi} = \dot{V}_V + (\dot{\omega}_V \times r_{Wi}) + [\omega_V \times (V_{Wi} + \dot{r}_{Wi})]$$

where:

V_V	is the vehicle velocity
ω_V	is the vehicle angular velocity
V_{Wi}	is the absolute wheel velocity
r_{Wi}	is the wheel position vector

Note

This expression for acceleration is simplified, which is possible because the MBS algorithm is for a passenger car and is not a general-purpose MBS simulation.

The residual acceleration a_{RWi} is the third term in the previous equation

$$a_{RWi} = \omega_V \times (V_{Wi} + \dot{r}_{Wi})$$

The relevant mass forces due to the residual acceleration F_{RWi} can then be calculated as follows:

$$F_{RWi} = -m_{Wi} a_{RWi}$$

The total mass force for a wheel is the mass force due to residual acceleration and the gravity force due to earth acceleration, i.e.,

$$F_{mWi} = F_{RWi} + m_{Wi}g$$

Mass Torque

The expression for mass torque includes not only the residual angular acceleration, but also the effect of the angular velocities, i.e.:

$$T_{mWi} = -[J_{Wi}\alpha_{Wi} + \omega_{Wi} \times J_{Wi}\omega_{Wi} + \omega_{Wi} \times \Theta\,\Omega\, e_{Wi}]$$

where:

α_{Wi} is the wheel residual acceleration
ω_{Wi} is the absolute wheel angular velocity
W is the wheel rotation speed about its rotation axis
e_{Wi} is the wheel position vector
Q is the inertia about the wheel rotation axis
J_{Wi} is the inertia tensor of the wheel

The last term in the previous equation is the gyroscopic torque. Experience shows that the last term is the dominant term over the first two terms, i.e.:

$$T_{mWi} \cong T_{GyWi} = -[\omega_{Wi} \times \Theta\,\Omega\, e_{Wi}]$$

For the overall vehicle, the second term is the dominant term due to the large vehicle inertia, i.e.:

$$T_{mV} \cong -[\omega_V \times J_V\, \omega_V]$$

Inports

The following table shows the inports:

Name	Unit	Description
a_Gravity[x,y,z]	[m/s^2]	Acceleration of gravity represented in the vehicle reference system
Inertia_Tensor_TotalVehicle[3x3]	[kgm^2]	Inertia tensor of the total vehicle including the additional loads
m_TotalVehicle	[kg]	Mass of the total vehicle including the additional loads
omega_Wheel[FL,FR,RL,RR]	[rad/s]	Rotational speed of the wheel about its axis of rotation
v_Abs_Vehicle_CoG[x,y,z]	[m/s]	Velocity of the vehicle CoG
v_Abs_WheelBody[FL,FR,RL,RR][x,y,z]	[m/s]	Absolute velocity of wheel center
v_Rel_WheelBody[FL,FR,RL,RR][x,y,z]	[m/s]	Relative velocity of wheel center
w_Abs_WheelBody[FL,FR,RL,RR][x,y,z]	[rad/s]	Absolute angular velocities of the wheel centers
w_Vehicle[x,y,z]	[rad/s]	Angular velocity of the vehicle CoG
w_Wheel_CoorSys_V[FL,FR,RL,RR][x;y;z]	[rad]	Wheel rotation speed in vehicle reference coordinate system

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
F_Gravitational[FL,FR,RL,RR][x,y,z]	[N]	Gravitational forces of the wheel
F_Gyroscopic[FL,FR,RL,RR][x,y,z]	[N]	Gyroscopic forces of the wheel
Trq_Gyroscopic[FL,FR,RL,RR][x,y,z]	[Nm]	Gyroscopic torques of the wheel

Parameters

With the above inputs and outputs, this subsystem requires only the parameters defined at the **Common Vehicle Parameter** subsystem.

Related topics

References

History of the MASS_FORCES_AND_TORQUES Block..... 429

Sensor Motion

Description

The sensors in the vehicle can be mounted anywhere in the vehicle body and do not have to be in the CoG. This block calculates the simulation variables, for example, acceleration and velocity, at a certain point in the vehicle that you define.

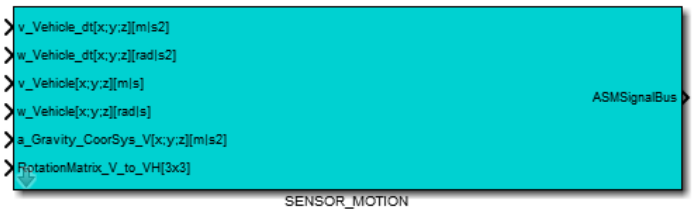
There are two lateral acceleration signals available at the **SENSOR_MOTION** block. Both signals are part of the ASMSignalBus:

- One lateral acceleration signal does not take the acceleration of gravity into account (**a_y_Sensor_CoorSys_V signal**).
- The other lateral acceleration signals include the effect of the acceleration of gravity (**a_y_Total_Sensor_CoorSys_V signal**).

Note

If you connect an ABS or ESP system (i.e. an ECU) to the sensor's signals, for example, for hardware-in-the-loop (HIL) simulation, use the correct lateral acceleration signal from the sensor block.

- If the sensor compensates the acceleration of gravity itself, connect it to the **a_y_Sensor_CoorSys_V** signal.
- If the sensor does not compensate the acceleration of gravity, connect it to the **a_y_Total_Sensor_CoorSys_V** signal.



Note

SENSOR_MOTION calculates sensor signals with respect to vehicle coordinate system as well as horizontal sensor signals.
The following signal name suffix determines the type of coordinate system:

- *xxx_CoorSys_V*: Coordinate system is related to the vehicle coordinate system
- *xxx_CoorSys_VH*: Coordinate system is related to the horizontal sensor signal (xy-plane of global coordinate system)

Inports

The following table shows the inports:

Name	Unit	Description
a_Gravity_CoorSys_V[x;y;z]	[m/s^2]	Acceleration of gravity
RotationMatrix_V_to_VH[3x3]	[]	Rotation matrix from vehicle to vehicle horizontal coordinate system.
v_Vehicle[x,y,z]	[m/s]	Velocity of vehicle reference system
v_Vehicle_dt[x,y,z]	[m/s^2]	Time derivative of the vehicle reference system velocity
w_Vehicle[x,y,z]	[rad/s]	Angular velocity of vehicle reference system
w_Vehicle_dt[x,y,z]	[rad/s^2]	Time derivative of the vehicle reference system angular velocity

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_PosVec_Sensor1	[m]	Position of sensor no. 1
Const_PosVec_Sensor2	[m]	Position of sensor no. 2
Const_PosVec_Sensor3	[m]	Position of sensor no. 3
Const_PosVec_Sensor4	[m]	Position of sensor no. 4

Related topics

References

Sensor Position (ModelDesk Parameterizing 🕮)

Vehicle and Wheel Velocities

Description

In this subsystem, the velocities of the vehicle center of gravity (CoG) and the wheels are calculated according to the generalized degrees of freedom, position of CoG, and the wheels' positions and relative velocities.

For the vehicle CoG, the translatory and angular velocities can be calculated as follows:

$$^{V}v_{CoG} = {}^{V}v_{V} + {}^{V}\omega_{V} \times {}^{V}r_{CoG}$$

$$^{V}\omega_{CoG} = {}^{V}\omega_{V}$$

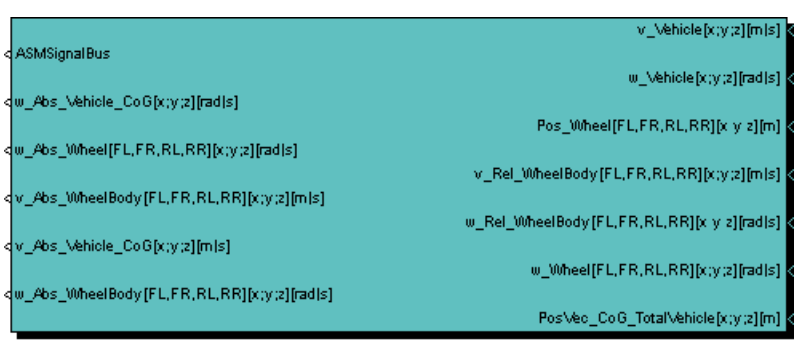
And for the wheel center:

$$^{V}v_{Wi} = {}^{V}v_{V} + {}^{V}\omega_{V} \times {}^{V}r_{Wi} + {}^{V}\dot{r}_{Wi}$$

$$^{V}v_{Wi} = {}^{V}\omega_{V} + {}^{V}\omega_{Wi_rel}$$

where:

i	stands for FL, FR, RL, or RR
$^{V}r_{Wi}$	is the wheel position, represented in the vehicle system
$^{V}\dot{r}_{Wi}$	is the wheel velocity relative to the vehicle
$^{V}\omega_{Wi_rel}$	is the wheel angular velocity relative to the vehicle



Inports

The following table shows the inports:

Name	Unit	Description
Pos_CoG_TotalVehicle[x,y,z]	[m]	Position vector to the vehicle CoG
Pos_Wheel[FL,FR,RL,RR][x;y;z]	[m]	Wheel relative position in vehicle coordinate system
UnitVec_y_Wheels_CoorSys_V[FL,FR,RL,RR][x;y;z]	[]	Unit vector in y direction of the wheel coordinate system
v_Vehicle[x,y,z]	[m/s]	Velocity of vehicle reference system
v_Rel_WheelBody[FL,FR,RL,RR][x,y,z]	[m/s]	Velocity of wheel body relative to vehicle body
w_Rel_WheelBody[FL,FR,RL,RR][x y z]	[rad/s]	Angular velocity of wheel body relative to vehicle body, i.e., excluding wheel rotational speed
w_Vehicle[x,y,z]	[rad/s]	Angular velocity of vehicle reference system
w_Wheel_CoorSys_V[FL,FR,RL,RR][x;y;z]	[rad]	Wheel rotation speed in vehicle reference coordinate system

Outports

The following table shows the outports:

Name	Unit	Description
w_Abs_Body[x,y,z]	[rad/s]	Angular velocity of the vehicle body
v_Abs_Vehicel_CoG[x,y,z]	[m/s]	Absolute velocity of the vehicle CoG
w_Abs_Wheel[FL,FR,RL,RR][x,y,z]	[rad/s]	Absolute angular velocity of the wheel in vehicle coordinate system
v_Abs_WheelBody[FL,FR,RL,RR][x,y,z]	[m/s]	Absolute velocity of the wheel in vehicle coordinate system
w_Abs_WheelBody[FL,FR,RL,RR][x,y,z]	[rad/s]	Absolute angular velocity of the wheel body, excluding wheel rotation speed

Parameters

With the above inputs and outputs, this subsystem requires no parameters to perform its calculation.

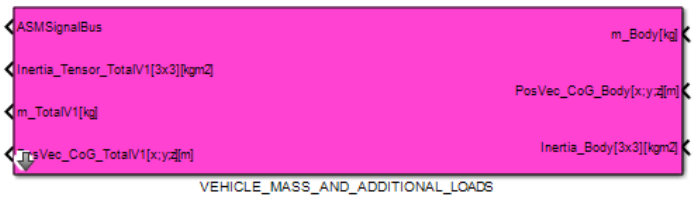
Vehicle Mass and Additional Loads

Description

ASM Vehicle Dynamics lets you define six additional loads, each at a different position in the vehicle. These additional loads affect the overall vehicle mass, inertia, and position of the center of gravity (CoG).

Note

Inertia tensor of the vehicle and additional loads are defined with respect to their center of gravity, respectively.



To calculate the total entity, the following equations are used:

$$m_{Total} = m_{V1} + m_b$$

$$r_{cog,T} = \frac{m_{V1} \cdot r_{cog,v1} + m_b \cdot r_{cog,b}}{m_{Total}}$$

$$J_{Total} = J_{V1} - m_{V1} \cdot \overline{\left(r_{cog,T} - r_{cog,v1}\right)} \cdot \overline{\left(r_{cog,T} - r_{cog,v1}\right)} + J_b - m_b \cdot \overline{\left(r_{cog,T} - r_{cog,b}\right)} \cdot \overline{\left(r_{cog,T} - r_{cog,b}\right)}$$

where:

m_{Total}	is the total mass of the vehicle (including wheels)
m_{V1}	is the the vehicle mass (including wheels) without the external/additional (moving) body (e.g. cabin or additional motor mass)
m_b	is the external/additional (moving) body (e.g. cabin or additional motor) mass
$r_{cog,T}$	is the center of gravity of the vehicle (including wheels)
$r_{cog,v1}$	is the center of gravity without the external/additional (moving) body (e.g. cabin or additional motor mass)
$r_{cog,b}$	is the external/additional (moving) body (e.g. cabin or additional motor) center of gravity
J_{Total}	is the inertia of the vehicle (including wheels)
J_{V1}	is the inertia without the external/additional (moving) body (e.g. cabin or additional motor mass)
m_{V1}	is the vehicle mass (including wheels) without the external/additional (moving) body (e.g. cabin or additional motor mass)
J_b	is the external/additional (moving) body (e.g. cabin or additional motor) inertia
m_b	is the external/additional (moving) body (e.g. cabin or additional motor) mass

Inports

The following table shows the inports:

Name	Unit	Description
Inertia_Body[3x3]	[kg m^2]	Inertia of additional external body with regard to the corresponding center of gravity
M_Body	[kg]	Additional external body mass which is calculated separately (e.g.: cabin mass)
PosVec_CoG_Body[x;y;z]	[m]	Position vector center of gravity of additional external body

Outports

The following table shows the outports:

Name	Unit	Description
Inertia_Tensor_TotalV1[3x3]	[k gm^2]	Inertia tensor of vehicle without additional external body but including the additional loads
m_TotalV1	[kg]	Mass of the vehicle without additional external body but including the additional loads
PosVec_CoG_TotalVehicle[x,y,z]	[m]	Position of CoG of the vehicle without additional external body but including the additional loads

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Inertia_Tensor_AddLoad1	[kgm^2]	Inertia tensor of additional body no. 1 with respect to center of gravity of additional load no. 1
Const_Inertia_Tensor_Vehicle	[kgm^2]	Inertia tensor of vehicle body and wheels with respect to the vehicle center of gravity
Const_m_AddLoad1	[kg]	Mass of additional body no. 1
Const_m_Vehicle	[kg]	Mass of vehicle body and wheels
Const_PosVec_CoG_AddLoad1	[m]	Position of CoG of additional body no. 1 in vehicle reference system
Const_PosVec_CoG_Vehicle	[m]	Position of vehicle CoG in vehicle reference system

Note

Only the parameters for the first additional load are described here. They are the same for the other five additional loads.

The wheel-related parameters, like wheel mass and inertias are defined in the **Common Vehicle Parameters** subsystem (see Common Vehicle Parameters on page 124).

Related topics

References

Body Geometry and Mass (ModelDesk Parameterizing 🕮)

Vehicle Motion

Description

As described in Vehicle Motion on page 287, the MBS technique is used to write the equation of motion, which can be written as follows:

$$[M][\ddot{q}] = [Q]$$

where:

$[M]$ is the [10 × 10] generalized mass matrix
$[\ddot{q}]$ is the [10 × 1] generalized degrees of freedom vector
$[Q]$ is the [10 × 1] generalized forces and torques vector

Vehicle generalized degrees of freedom

The generalized degrees of freedom are listed in detail below.

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = {}^{V}\begin{bmatrix} V_x \\ V_y \\ V_z \end{bmatrix}_V$$

The translatory vehicle velocities of the origin of coordinate system V in x, y, and z directions described in vehicle reference coordinate system V

$$\begin{bmatrix} \dot{q}_4 \\ \dot{q}_5 \\ \dot{q}_6 \end{bmatrix} = {}^{V}\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}_V$$

The angular vehicle velocities about the x, y, and z-axes of vehicle reference coordinate system V

$$\begin{bmatrix} \dot{q}_{FL} \\ \dot{q}_{FR} \\ \dot{q}_{RL} \\ \dot{q}_{RR} \end{bmatrix} = {}^{V}\begin{bmatrix} \dot{Z}_{FL} \\ \dot{Z}_{FR} \\ \dot{Z}_{RL} \\ \dot{Z}_{RR} \end{bmatrix}$$

The vertical speed of the wheel represented in vehicle reference coordinate system V

Each translatory or angular degree of freedom has a corresponding generalized force or torque. The mass matrix is a function of the wheel position and kinematic suspension relations, i.e., it is not constant and is calculated at every simulation step. The generalized forces and torques $[Q]$ are calculated from the tire, aerodynamics, mass forces, and torques and suspension forces in the direction of the relevant degrees of freedom.

Inports

The following table shows the inports:

Name	Unit	Description
F_Aero_CoorSys_V[x,y,z]	[N]	Aerodynamic forces represented in vehicle reference coordinate system
F_External_CoorSys_V[x;y;z]	[N]	External forces acting on the vehicle reference coordinate system
F_Gravitational[V,FL,FR,RL,RR][x,y,z]	[N]	Gravitational forces of the vehicle and wheels
F_Gyroscopic[V,FL,FR,RL,RR][x,y,z]	[N]	Gyroscopic forces of the vehicle and wheels
F_Tire[FL,FR,RL,RR][x y z]	[N]	Tire forces vectors represented in vehicle reference coordinate system
gF_Susp[q1_Front;q2_Front;q1_Rear;q2_Rear]	[]	Generalized forces and/or torques in the direction of the generalized coordinates (i.e. degrees of freedom).
gFront_dq_Front_dt[2x2]	[]	Derivatives of the generalized forces with respect to the generalized speeds for front suspension

Name	Unit	Description
gRear_dq_Rear_dt[2x2]	[]	Derivatives of the generalized forces with respect to the generalized speeds for rear suspension
Inertia_Tensor_TotalVehicle[3x3]	[kgm^2]	Total inertia tensor of the vehicle including body and wheels
m_TotalVehicle	[kg]	Total mass of the vehicle (body and wheels)
Pos_CoG_TotalVehicle[x,y,z]	[m]	Position vector to the vehicle CoG
Pos_FL_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of front left wheel position with respect to the vertical wheel displacement of front left (q1) and front right (q2) wheel
Pos_FR_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of front right wheel position with respect to the vertical wheel displacement of front left (q1) and front right (q2) wheel
Pos_RL_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Derivative of rear left wheel position with respect to the vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Pos_RR_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Derivative of rear right wheel position with respect to the vertical wheel displacement of rear left (q1) and rear right (q2) wheel
Pos_Wheel[FL,FR,RL,RR][x y z]	[m]	Wheel relative position in vehicle reference coordinate system
q_dt_Init[10x1]	[]	Initial vehicle velocity (1-3: translational, 4-6: rotational) and relative wheel velocity with respect to the vehicle body (7-10)
q_Wheel_Init[4x1]	[]	Initial relative wheel position with respect to the vehicle body
Reset	[]	Resets all integrators in the block to their initial conditions
Stabilization_F_External[3x6]	[]	Numerical stabilization matrix for external torques, Jacobian-Matrix dForces/dGeneralizedCoordinates multiplied with the simulation step size
Stabilization_Trq_External[3x6]	[]	Numerical stabilization matrix for external torques, Jacobian-Matrix dTorques/dGeneralizedCoordinates multiplied with the simulation step size
Stiffness_z_Tire[FL,FR,RL,RR]	[N/m]	Vertical tire stiffness
Sw_test benchMode	[]	Value to switch between vehicle and test bench mode: 1: Vehicle 2: Test bench mode
Trq_Aero_CoorSys_V[x,y,z]	[Nm]	Aerodynamic torques represented in vehicle reference coordinate system
Trq_External_CoorSys_V[x;y;z]	[Nm]	External torques acting on the vehicle reference coordinate system
Trq_Gyroscopic[V,FL,FR,RL,RR][x,y,z]	[Nm]	Gyroscopic torques of the vehicle and wheels
Trq_Shaft_CoorSys_V[FL,FR,RL,RR][x;y;z]	[Nm]	Driving torque on the wheel in vehicle reference coordinate system

Name	Unit	Description
Trq_Tire[FL,FR,RL,RR][x y z]	[Nm]	Tire torques vectors represented in vehicle reference coordinate system
w_Rel_FL_Wheel_dq_Front_dt[q1;q2][x;y;z]	[]	Derivative of front left wheel angular velocity with respect to the vertical wheel speeds of front left (q1) and front right (q2) wheel
w_Rel_FR_Wheel_dq_Front[q1;q2][x;y;z]	[]	Derivative of front right wheel angular velocity with respect to the vertical wheel speeds of front left (q1) and front right (q2) wheel
w_Rel_RL_Wheel_dq_Rear_dt[q1;q2][x;y;z]	[]	Derivative of rear left wheel angular velocity with respect to the vertical wheel speeds of rear left (q1) and rear right (q2) wheel
w_Rel_RR_Wheel_dq_Rear[q1;q2][x;y;z]	[]	Derivative of rear right wheel angular velocity with respect to the vertical wheel speeds of rear left (q1) and rear right (q2) wheel

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
q_Wheel [q1_Front;q2_Front;q1_Rear;q2_Rear]	[]	The wheel generalized coordinates
q_Wheel_dt [q1_Front;q2_Front;q1_Rear;q2_Rear]	[]	The wheel generalized speeds
v_Vehicle[x,y,z]	[m/s]	Velocity of vehicle reference system
v_Vehicle_dt[x,y,z]	[m/s^2]	Time derivative of the vehicle reference system velocity
w_Vehicle[x,y,z]	[rad/s]	Angular velocity of vehicle reference system
w_Vehicle_dt[x,y,z]	[rad/s^2]	Time derivative of the vehicle reference system angular velocity

Parameters

With the above inputs and outputs, this subsystem requires only the parameters defined at the Common Vehicle Parameters subsystem to perform its calculation.

Related topics

References

History of the VEHICLE_MOTION_CAR Block 461

Vehicle Movement Info Car

Description

The block collects the signals from the blocks inside the **Vehicle Movement** subsystems and calculates typical vehicle data. The signals are then routed to the ASMSignalBus.

Inports

The following table shows the inports:

Name	Unit	Description
ASMBUS_VehicleDynamics_CoordinateTransformation	[]	Contains the signals from the coordinate transformation.
ASMBUS_VehicleDynamics_MassForcesAndTorques	[]	Contains the signals from the mass forces and torques.
ASMBUS_VehicleDynamics_SensorMotion	[]	Contains the signals from the sensor motion.
ASMBUS_VehicleDynamics_Steering	[]	Contains the signals from the steering.
ASMBUS_VehicleDynamics_Suspension	[]	Contains the signals from the suspension.
ASMBUS_VehicleDynamics_VehicleMass	[]	Contains the signals from the vehicle mass.

Name	Unit	Description
ASMBUS_VehicleDynamics_VehicleMotion	[]	Contains the signals from the vehicle motion.
CoGAndWheelVelocities	[]	Centre of gravitiy and wheel velocities.
Reset	[]	Reset of states.

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).

Related topics

Basics

ASMSignalBus (ASM User Guide 🕮)

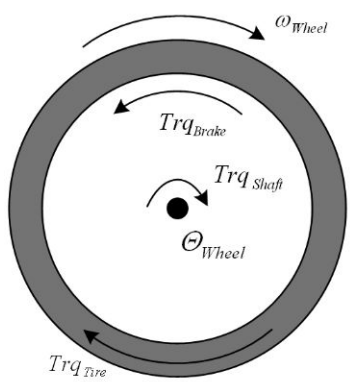
Wheel Speed

Wheel Speed

Description

Only the rotational degree of freedom of the wheel movement is considered. The torque balance around the wheel axis of rotation is calculated to determine the wheel rotation speed.

There are mainly three components acting on the wheel, as shown in the illustration below.



These components are used to calculate the torque balance about the wheel rotational axis. The wheel speed is calculated from the following equation:

$$\Theta_{Wheel} \cdot \dot{\omega}_{Wheel} = Trq_{Shaft} + Trq_{Tire} - Trq_{Brake}$$

where:

Θ_{Wheel}	is the inertia about the wheel rotational axis
ω_{Wheel}	is the wheel speed
Trq_{Shaft}	is the driving torque, transferred through the drivetrain
Trq_{Tire}	is the effective tire torque in the direction of wheel rotation
Trq_{Brake}	is the effective braking torque

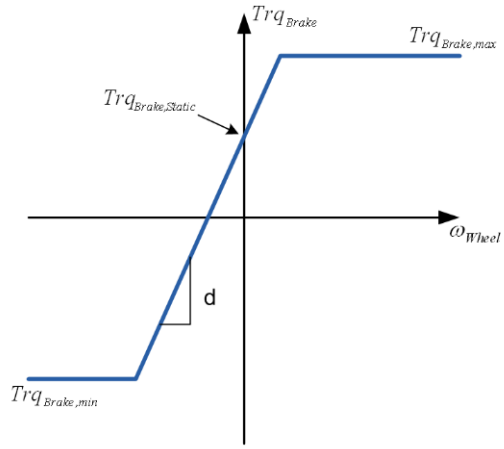
The effective braking torque is modeled as a friction torque with sticking capability. The friction torque consists of two components, the static and the dynamic torque. The static torque is used to balance the shaft and tire torques acting on the wheel. The dynamic torque is added to allow the transition from the wheel rotation to standstill. The static torque and the sum of static and dynamic torques are limited by the maximum braking torque, which is calculated in the brake model.

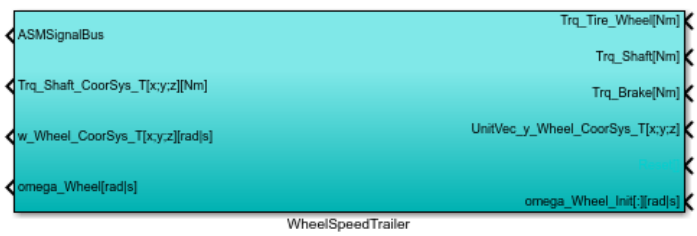
The effective braking torque is calculated as follows:

$$Trq_{BrakeStatic} = Trq_{Shaft} + Trq_{Tire} \leq Trq_{Brake,max}$$

$$Trq_{Brake} = Trq_{BrakeStatic} + d \cdot \omega_{Wheel} \leq Trq_{Brake,max}$$

The following illustration shows the braking torque as a function of wheel speed.





Inports

The following table shows the inports:

Name	Unit	Description
omega_Wheel_Init [FL;FR;RL;RR]	[rad/s]	Initial wheel rotational speed
Reset	[]	Resets all integrators in the block to their initial conditions.
Trq_Brake[FL,FR,RL,RR]	[Nm]	Brake torque, calculated in the Brake subsystem
Trq_Shaft[FL,FR,RL,RR]	[Nm]	Driving torque exerted on the driving shafts of the drivetrain
Trq_Tire[FL,FR,RL,RR]	[Nm]	Torque from tire to the wheel, i.e., effect of circumferential force and rolling resistance torque
UnitVec_y_Wheels_CoorSys_V [FL,FR,RL,RR][x;y;z]	[]	Unit vector in y direction of the wheel coordinate system

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
omega_Wheel[FL,FR,RL,RR]	[rad/s]	Rotational speed of the wheel about its axis of rotation

Name	Unit	Description
Trq_Shaft_CoorSys_V [FL,FR,RL,RR] [x;y;z]	[Nm]	Driving torque on the wheel in vehicle reference coordinate system
w_Wheel_CoorSys_V [FL,FR,RL,RR] [x;y;z]	[rad]	Wheel rotation speed in vehicle reference coordinate system

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_Coef_DynamicBrakeTrq	[Nm/(rad/s)]	Change of braking torque with wheel speed (transient friction coefficient d) (defined in the COMMON_VEHICLE_PARAMETERS subsystem)
Const_Inertia_y	[kgm^2]	Inertia about the rotation axis of the wheel (defined in the COMMON_VEHICLE_PARAMETERS subsystem)

Related topics

References

History of the WHEEL_SPEED Block..... 463

Switches

Vehicle Dynamics Variant Switches

Description

This block contains switches for the vehicle dynamics.

Parameters

The following table shows the parameters:

Name	Unit	Description
SW_Tire_Model	[1|2]	Switch for the tire model: ▪ 1: TMeasy ▪ 2: Magic Formula

Demos

Where to go from here

Information in this section

Vehicle Dynamics.....298

ASM Modules.....301
To describe the modules of the model.

Subsystems.....336
To describe the subsystems inside the ASM modules.

Vehicle Dynamics

Where to go from here

Information in this section

Basics on the Vehicle Dynamics Demo Model.....298
Gives you basic information about the Vehicle Dynamics Demo Model

Overview of the VehicleDynamics Demo.....299
Gives you an overwiew of the Vehicle Dynamics Demo Model and the ASM modules used.

Basics on the Vehicle Dynamics Demo Model

Introduction

This topic gives you basic information about the Vehicle Dynamics Demo Model.

Basics

The ASM Vehicle Dynamics Model simulates the dynamics of a passenger car. The vehicle is composed of engine, drivetrain, rigid vehicle body, and four wheels. The model represents the vehicle's longitudinal, lateral, and vertical dynamics and simulates in detail an elastic drivetrain, suspension kinematics and compliance, tire-road friction forces and moments, steering and brakes. The combustion process in the engine is not modeled.

As the focus is on vehicle dynamics, a look-up table is used to calculate the engine torque, i.e., the combustion process is not modeled.

The Drivetrain model consists of a crankshaft, a starter, a manual transmission with a clutch, an automatic transmission with a lockup clutch and a torque converter, differentials, and drive shafts. The Drivetrain model can be switched to simulate front-, rear-, or four-wheel drive. It is also possible to model part-time four-wheel drives with the lockup clutch of the central differential.

The Vehicle Dynamics block contains components models which simulate the dynamics and/or kinematics of tire, suspension, steering, wheel spin, brake, and aerodynamics, and also the dynamic coupling between these components to simulate the motion of the whole vehicle.

The environment is divided into driver, maneuver, and road models.

The two blocks on the left and right of the model blocks manage the signal mapping from and to the I/O. They also manage the signal flow to and from the soft ECU for Simulink simulation.

The engine, the drivetrain, and the vehicle dynamic blocks are connected via shaft speeds and shaft torque signals. This makes it easy to extend the ASM Vehicle Dynamics Model with other models (for example, a user drivetrain

model). The ASM Vehicle Dynamics Model can be used in combination with real controllers in a hardware-in-the-loop environment (HIL mode), or for simulation of an ASM Vehicle Dynamics in combination with software controller algorithms (PC mode).

Overview of the VehicleDynamics Demo

Introduction

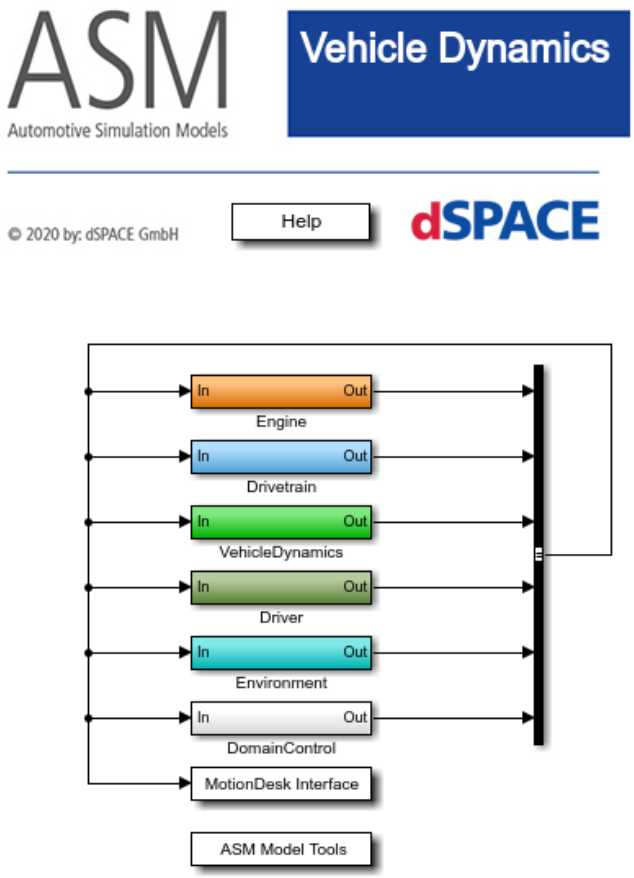
This topic gives you an overwiew of the Vehicle Dynamics Demo Model and the ASM modules used.

Opening the model

Refer to How to Start with an ASM Demo Model via MATLAB (ASM Vehicle Dynamics Model Description 🕮).

Top layer

The following illustration shows the top layer of the model:



ASM modules

The model consists of the following ASM modules:

- Electrical System Basic Module on page 309
- Engine Basic Module on page 310
- Drivetrain Module on page 306
- Vehicle Dynamics Module on page 327
- Driver Module on page 301
- Environment Module on page 313
- Start System Module on page 323
- Torque Manager Basic Module on page 325
- Driver Assistance Simple Module on page 305

Related topics

HowTos

How to Start with an ASM Demo Model via MATLAB (ASM Vehicle Dynamics Model Description 🕮)

ASM Modules

Where to go from here

Information in this section

Driver Module.....301
To control the pedals, gear, and steering wheel in such a way that the vehicle follows given references and stays on the road.

Driver Assistance Simple Module.....305
To provide default values for Adaptive Cruise Control (ACC) signals.

Drivetrain Module.....306
To simulate different drivetrain components, such as clutch, transmission systems, differentials, and drive shafts.

Electrical System Basic Module.....309
To implement constant values of the voltage, state of charge and temperature of a low and a high voltage battery.

Engine Basic Module.....310
To model an engine as a simple look-up table, where the engine torque is calculated as a function of engine speed and the driver accelerator pedal position.

Environment Module.....313
To model maneuver and road models.

Start System Module.....323
To implement a start system for switching the engine on and off.

Torque Manager Basic Module.....325
To serve as a mockup for a hybrid manager. It is used if the vehicle only has a conventional internal combustion engine.

Vehicle Dynamics Module.....327
To implement a multibody system of a passenger car including tire, aerodynamics and suspension calculation.

Driver Module

Description

The driver model is essential for controlled maneuvers. It controls the pedals, gear, and steering wheel in such a way that the vehicle follows given references and stays on the road. The driver plans the cruising speed for a given road and can navigate safely through the curves by controlling the pedals in such a way

that given dynamics limits are not exceeded. It is also possible to incorporate the model with different stimulus signals, such as stimulating the gear and clutch while following reference velocity.

The following tables show the external inports and outports of the ASM module.

Subsystems

The module contains the following subsystems:

- Lateral Controller Subsystem on page 336
- Longitudinal Controller Subsystem on page 338
- Reference Velocity Subsystem on page 341

Inports Plant

The following table shows the ports of the Interface_External_In:

Name	Unit	Description
a_x_Max	[m/s^2]	Maximum longitudinal acceleration for reference velocity calculation
a_x_Min	[m/s^2]	Minimum longitudinal acceleration for reference velocity calculation
a_x_Vehicle_Ref_TrafficDriver	[m/s^2]	Reference acceleration of the ASM vehicle by the traffic driver feature
a_y_Max	[m/s^2]	Maximum lateral acceleration (for velocity calculation in curves)
Angle_SteeringWheel_Maneuver	[deg]	Steering wheel angle stimulus from maneuver scheduler
Angle_Yaw_Veh_Road	[deg]	Yaw angle of the reference line of the road at the current position of the ASM vehicle
Angle_Yaw_Vehicle_CoorSys_E	[deg]	Vehicle yaw angle
CorneringStiffness_Tire_Front	[N/rad]	Cornering stiffness of front tires
CorneringStiffness_Tire_Rear	[N/rad]	Cornering stiffness of rear tires
Curv_Preview_Road	[1/m]	Road curvature at previewed road positions
Curv_Veh_Road	[1/m]	Curvature of the road at the vehicle position
Dist_Border_Left	[m]	Distance from the ASM vehicle reference coordinate system to the left road border
Dist_Border_Right	[m]	Distance from the ASM vehicle reference coordinate system to the right road border
Enable_v_Init	[0|1]	Flag that indicates if the ASM vehicle simulation has a velocity different from zero
F_Driving_Res	[N]	Sum of driving resistances acting on the vehicle
Gear_Maneuver	[]	Gear stimulus from maneuver
Gear_Max	[0|1_n]	Gear setpoint defined by maneuver input ▪ 0: Off ▪ 1_n: Gear setpoint
i_Steering_Mean	[]	Mean value of steering gear ratio
Inertia_z_Vehicle_Total	[kg m^2]	Vehicle inertia around z axis

Name	Unit	Description
m_Total_Vehicle	[kg]	Total vehicle mass
Mode_AcceleratorBrake	[1|2]	Controls whether the accelerator and brake pedal positions are stimulus signals or controlled by the driver: ▪ 1: Stimulus ▪ 2: Driver
Mode_AdaptVelocityToRoad	[1|2]	Controls whether the reference velocity calculation on roads is active: ▪ 1: Off ▪ 2: On
Mode_CurvatureCalculation	[1|2]	Specifies the curvature calculation mode: ▪ 1: Road ▪ 2: Driver
Mode_ForceToRoad	[1|2]	Controls the limitation to valid lane indices: ▪ 1: Off ▪ 2: On
Mode_GearClutch	[1|2|3|4]	Switch for mode signal of the gear and clutch pedal source: ▪ 1: Stimulus ▪ 2: Driver ▪ 3: Open clutch ▪ 4: Reference gear
Mode_LatCtrl	[1|2]	Lateral control type: ▪ 1: Pos ▪ 2: Yaw
Mode_SelectorLever	[1|2]	Switch for mode signal of the selector lever source: ▪ 1: Stimulus ▪ 2: Driver
Mode_Steer	[1|2|3]	Controls whether the steering wheel angle is controlled by a stimulus signal or by driver: ▪ 1: Stimulus ▪ 2: Driver ▪ 3: Fix
Mode_TrafficDriver	[1|2]	Controls the Traffic Driver mode of the ASM vehicle: ▪ 1: Off ▪ 2: On
Mode_Transmission	[1|2]	Transmission mode switch ▪ 1: Manual ▪ 2: Automatic
n_Engine	[rpm]	Speed of combustion engine
n_Engine_Idle_Set	[rpm]	Setpoint for idle speed of combustion engine
omega_Out_Trm	[rad/s]	Transmission output speed
Pitch_Preview_Road	[deg]	Pitch angle of the road at the positions of the preview points
Pos_AccPedal_Maneuver	[%]	Accelerator pedal position stimulus from maneuver scheduler
Pos_BrakePedal_Maneuver	[%]	Brake pedal position stimulus from maneuver scheduler

Name	Unit	Description
Pos_ClutchPedal_Maneuver	[%]	Clutch pedal position stimulus from maneuver scheduler
Pos_x_CoG_Vehicle_Total	[m]	x-element of position vector of vehicle centre of gravity
Pos_x_Preview_DriverCurv	[m]	x-positions of the preview points in the earth coordinate system taken for the curvature for the reference velocity calculation
Pos_x_Preview_Road	[m]	Vector of previewed road positions in x direction (earth coordinates)
Pos_x_Ref_Vehicle_Road	[m]	x-position of the reference line of the road at the current position of the vehicle
Pos_x_Vehicle_CoorSys_E	[m]	X vehicle position
Pos_y_Preview_DriverCurv	[m]	y-positions of the preview points in the earth coordinate system taken for curvature for the reference velocity calculation
Pos_y_Preview_Road	[m]	Vector of previewed road positions in y direction (earth coordinates)
Pos_y_Ref_Vehicle_Road	[m]	y-position of the reference line of the road at the current position of the vehicle
Pos_y_Vehicle_CoorSys_E	[m]	Y vehicle position
Pos_z_Preview_Road	[m]	Vector of the previewed road positions in the z-direction (earth coordinates)
Reset_States	[0|1]	Reset states
Roll_Preview_Road	[deg]	Roll angle of the road at the positions of the preview points
Route_Direction_Init	[]	Direction on the route for the all fellow vehicles
s_Total_Vehicle_Road	[m]	The total distance along the route
s_Vehicle_Init	[m]	Initial distance s along the road
SelectorLever_Maneuver	[-3|-2|-1|0|1]	Selector lever position from maneuver: ▪ -3: TipShift ▪ -2: Park ▪ -1: Reverse ▪ 0: Neutral ▪ 1: Drive
Speed_Limit_Preview_Road	[km/h]	Speed limit as defined in the scenery section of the road element at the position of the ASM vehicle
State_Engine	[0|1|2|3|4]	Engine state ▪ 0: Engine off ▪ 1: Ignition on ▪ 2: Ignition on and starter activated ▪ 3: Engine is running ▪ 4: Ignition is switched off, shutdown active
State_StartStop	[1|2|3|4]	Start-stop state: ▪ 1: Off ▪ 2: Not ready ▪ 3: Ready to stop the engine ▪ 4: Engine actively stopped by the system
Sw_LatDriver	[1|2]	Controls whether lateral_control1 or lateral_control2 is active:

Name	Unit	Description
		▪ 1: Lateral_control 1 active ▪ 2: Lateral_control 2 active
v_Vehicle_Preview_Ref_Maneuver	[m/s]	Reference signal for preview speed calculation
v_Vehicle_Ref_Maneuver	[m/s]	Vehicle reference velocity provided by maneuver
v_x_Vehicle_CoG	[km/h]	Longitudinal vehicle velocity
v_y_Vehicle_CoG	[km/h]	Lateral vehicle velocity at centre of gravity
Width_Deadzone_LatCtrl	[m]	Width of the dead zone for the lateral deviation from theroad reference line
YawRate_Vehicle	[deg/s]	Vehicle yaw rate

Outports Plant

The following table shows the ports of the Interface_External_Out:

Name	Unit	Description
Angle_SteeringWheel	[deg]	Steering wheel angle
Gear	[]	Gear
Pos_AccPedal	[%]	Position of the accelerator pedal
Pos_BrakePedal	[%]	Brake pedal position
Pos_ClutchPedal	[%]	Clutch pedal position
s_Preview	[m]	Vector with preview distances for the lateral control and vehicle reference velocity calculation.
SelectorLever	[-3|-2|-1|0|1]	Selector lever position: ▪ -3: TipShift ▪ -2: Park ▪ -1: Reverse ▪ 0: Neutral ▪ 1: Drive
t_Preview_vRef_Driver	[s]	Preview time for the driver model

Driver Assistance Simple Module

Description

This module provides default values for adaptive cruise control (ACC) signals.

The following tables show the external inports and outports of the ASM module.

Outports Control

The following table shows the ports of the Interface_External_Out:

Name	Unit	Description
AEB	[0|1]	Autonomous emergency brake flag from soft ecu ACC
Enable_Trq_Decrease_ACC	[0|1]	Enable torque decrease ACC
Enable_Trq_Increase_ACC	[0|1]	Enable torque increase ACC
FCW	[0|1]	Forward collision warning flag from soft ecu ACC
Trq_DecreaseRequest_ACC	[Nm]	Torque decrease request ACC
Trq_IncreaseRequest_ACC	[Nm]	Torque increase request ACC

Drivetrain Module

Description

The drivetrain transfers the torque from the engine to the wheels. The drivetrain model simulates different drivetrain components, such as clutch, transmission systems, differentials, and drive shafts. The used modeling approach includes the stiffness of drive shafts and takes the elasticity and the inertias of the drivetrain components into account. The model can be switched to simulate different drivetrain topologies, such as front-, rear- and all-wheel drive.

The following tables show the external inports and outports of the ASM module.

Subsystems

The module contains the following subsystems:

- Final Drive Assembly Subsystem on page 342
- Transmission Subsystem on page 348
  - Automatic Transmission Subsystem on page 349
  - Manual Transmission Subsystem on page 352

Inports Control

The following table shows the ports of the Interface_External_In:

Name	Unit	Description
Enable_v_Init	[0|1]	Flag that indicates if the ASM vehicle simulation has a velocity different from zero
Gear_Maneuver	[]	Gear stimulus from maneuver
Gear_Max	[0Off|1_n]	Gear set point defined by maneuver input

Name	Unit	Description
Mode_GearClutch	[1|2|3|4]	Switch for mode signal of gear and clutch pedal source: ▪ 1: Stimulus ▪ 2: Driver ▪ 3: Open clutch ▪ 4: Reference gear
Pos_AccPedal	[%]	Position of the accelerator pedal
Reset_States	[0|1]	Reset states
SelectorLever	[-3|-2|-1|0|1]	Selector lever position: ▪ -3: TipShift ▪ -2: Park ▪ -1: Reverse ▪ 0: Neutral ▪ 1: Drive
State_Engine	[0|1|2|3|4]	Engine state: ▪ 0: Engine off ▪ 1: Ignition on ▪ 2: Ignition on and starter activated ▪ 3: Engine is running ▪ 4: Ignition is switched off, shutdown active
State_StartStop	[1|2|3|4]	Start-stop state: ▪ 1: Off ▪ 2: Not ready ▪ 3: Ready to stop the engine ▪ 4: Engine actively stopped by the system
Sw_TipShift	[-1|0|1]	TipShift request signal: ▪ -1: Downshift ▪ 0: Off ▪ 1: Upshift

The following table shows the interface to connect an external ECU (real ECU):

Name	Unit	Description
Ctrl_CentralDiff	[0_1]	Lockup control signal of central differential
Ctrl_Clutch_4WD	[0_1]	Lockup control signal of clutch 4WD
Ctrl_FrontDiff	[0_1]	Lockup control signal of front differential
Ctrl_Lockup_Clutch	[0_1]	Position of the clutch
Ctrl_Parking_Pawl	[0|1]	Parking pawl control signal
Ctrl_RearDiff	[0_1]	Lockup control signal of rear differential
Factor_Trq_Engine_Red	[%]	Engine torque reduction factor
Gear	[]	Gear

Outports Control

The following table shows the ports of the Interface_External_Out:

Name	Unit	Description
Ctrl_Parking_Pawl	[0|1]	Parking pawl control signal
Factor_Trq_Engine_Red	[%]	Engine torque reduction factor

Inports Plant

The following table shows the ports of the Interface_External_In:

Name	Unit	Description
CrankAngle_Cyl	[aTDC]	Crank angle per cylinder, positive value is after TDC
Enable_v_Init	[0|1]	Flag that indicates if the ASM vehicle simulation has a velocity different from zero
Gear	[]	Gear
n_Engine_Set	[rpm]	Setpoint for speed of combustion engine
omega_Wheel	[FL;FR;RL;RR][rad/s]	Wheel speed
Pos_ClutchPedal	[%]	Clutch pedal position
Reset_States	[0|1]	Reset states
Sw_Starter	[0|1]	Starter activate switch
Sw_Testbench	[0|1]	Switch to activate the test bench: ▪ 0: Off ▪ 1: On
Trq_MeanEff_Engine	[Nm]	Mean effective engine torque

Outports Plant

The following table shows the ports of the Interface_External_Out:

Name	Unit	Description
Gear	[]	Gear
i_FinalDrive	[]	Total transmission ratio of the final driveline (between transmission output and wheel speed)
i_unsync_Trm	[]	Unsynchronized transmission ratio
Mode_Drivetrain	[1|2|3|4|5]	Drivetrain mode
Mode_Transmission	[1|2]	Transmission mode switch: ▪ 1: Manual ▪ 2: Automatic
n_Engine	[rpm]	Speed of the combustion engine
omega_Out_Trm	[rad/s]	Transmission output speed

Name	Unit	Description
Sw_CrankShaft_Reset	[0|1]	Crankshaft reset switch: ▪ 0: Disabled ▪ 1: Enabled
Trq_BSG	[Nm]	Torque of the belt-driven starter generator (BSG)
Trq_Out_Front_Diff	[Nm]	Output torque of the front differential
Trq_Shaft	[FL;FR;RL;RR][Nm]	Torque of the output shaft

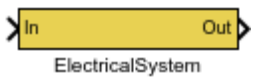
Animation

The following table shows the ports of the Animation:

Name	Unit	Description
Gear	[]	Gear
n_Engine	[rpm]	Speed of combustion engine

Electrical System Basic Module

Description



The ASM Module Electrical System Basic provide constant values of the voltage, state of charge and temperature of a low and a high voltage battery. The DC link voltage represents the voltage of the high voltage battery. It can be exchanged with a more complex ASM module.

The following tables show the external in- and outports of the ASM module.

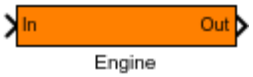
Outports Plant

The following table shows the ports of the Interface_External_Out:

Name	Unit	Description
SOC_Bat_HV	[%]	State of charge high voltage battery
SOC_Bat_LV	[%]	State of charge low voltage battery
T_Bat_HV	[°C]	Temperature high voltage battery
T_Bat_LV	[°C]	Temperature low voltage battery
V_Bat_HV	[V]	Voltage of high voltage battery
V_Bat_LV	[V]	Voltage of low voltage battery
V_DCLink	[V]	DC Link voltage

Engine Basic Module

Description



The engine is modelled as a simple look-up table, where the engine torque is calculated as a function of engine speed and the driver accelerator pedal position.

The following tables show the external inports and outports of the ASM module.

Subsystems

The module contains the following subsystems:

- Engine Basic Subsystem on page 355
- Soft ECU Engine Subsystem on page 355

Inports Control

The following table shows the ports of the Interface_External_In:

Name	Unit	Description
Enable_n_Engine_Idle_Request	[0|1]	Flag to activate the engine idle speed request: ▪ 0: Off ▪ 1: On
Enable_Trq_Decrease_Fast_ESP	[0|1]	Flag to enable or disable the request for ESP fast torque decreasing
Enable_Trq_Increase_Fast_ESP	[0|1]	Flag to enable or disable the request for ESP fast torque increasing
Enable_Trq_Request_Extern1	[0|1]	Flag to enable or disable the first external torque request
Enable_Trq_Request_Extern2	[0|1]	Flag to enable or disable the second external torque request
Factor_Trq_Engine_Red	[%]	Engine torque reduction factor
Mode_StartButtonState	[1|2]	Controls whether the signal for the activation of the start button is calculated by the soft ecu or provided by a stimulus signal. ▪ 1: Stimulus ▪ 2: Soft ECU
n_Engine	[rpm]	Speed of the combustion engine
n_Engine_Idle_Request	[rpm]	Engine idle speed request
Pos_AccPedal_Driver	[%]	Position of the accelerator pedal from the longitudinal controller

Name	Unit	Description
Reset_States	[0|1]	Reset states
State_StartButton	[-1|0|1|2]	State of the start button: ▪ -1: Power off ▪ 0: Acceleration ▪ 1: Power on ▪ 2: Starter on
Sw_CrankShaft_Reset	[0|1]	Crankshaft reset switch: ▪ 0: Disabled ▪ 1: Enabled
Sw_Ignition	[0|1]	Ignition signal - terminal 15: ▪ 0: Off ▪ 1: On
Sw_StarterReq	[0|1]	Starter request signal (usually sent to engine ECU): ▪ 0: Off ▪ 1: On
Trq_Decrease_Request_Fast_ESP	[Nm]	ESP fast torque decreasing request
Trq_Increase_Request_Fast_ESP	[Nm]	ESP fast torque increasing request
Trq_Request_Extern1	[Nm]	First external torque request
Trq_Request_Extern2	[Nm]	Second external torque request
Trq_Request_ICE	[Nm]	Torque request ICE
Vehicle_Topology	[-2|-1|0|2]	Vehicle topology: ▪ -2: EM only ▪ -1: ICE only ▪ 0: P0 hybrid ▪ 2: P2 hybrid

Outports Control

The following table shows the ports of the Interface_External_Out:

Name	Unit	Description
CrankAngle_Cyl	[aTDC]	Crank angle per cylinder, positive value is after TDC
Enable_Trq_Shift_Request	[0|1]	Flag to enable the ICE torque intervention during gear shift
EngOP_Num	[]	Engine operation point
n_Engine_Idle_Set	[rpm]	Setpoint for the idle speed of combustion engine
n_Engine_Meas	[rpm]	Measured speed of the combustion engine
p_Ambient	[Pa]	Ambient pressure
Pos_AccPedal_Meas	[%]	Measured accelerator pedal position

Name	Unit	Description
State_ICEngine	[0|1|2|3|4]	ICE state: ▪ 0: Off or error ▪ 1: Ignition on ▪ 2: Starter on ▪ 3: Running ▪ 4: Shutdown
State_StartStop	[1|2|3|4]	Start-stop state: ▪ 1: Off, ▪ 2: Not ready ▪ 3: Ready to stop the engine ▪ 4: Engine actively stopped by the system
Sw_Replace_Env	[0|1]	Switch for replacing the ambient conditions ▪ 0: Off - source is parameters ▪ 1: On - source is measurements
Sw_Starter	[0|1]	Starter activate switch: ▪ 0: Off ▪ 1: On
T_Ambient	[°C]	Ambient temperature
Trq_MeanEff_Engine_Meas	[Nm]	Measured mean effective engine torque
Trq_Shift_Request	[Nm]	Torque intervention during gear shift

Inports Plant

The following table shows the ports of the Interface_External_In:

Name	Unit	Description
n_Engine	[rpm]	Speed of combustion engine
Reset_States	[0|1]	Reset states
v_x_Vehicle	[km/h]	Vehicle velocity in forward direction

Outports Plant

The following table shows the ports of the Interface_External_Out:

Name	Unit	Description
T_Coolant_ICE	[°C]	Temperature of the engine coolant
Trq_Fric_Engine	[Nm]	Friction torque of the engine
Trq_MeanEff_Engine	[Nm]	Mean effective engine torque

Environment Module

Description

This ASM module is used for ASM Vehicle Dynamics. The ASM Module Environment for ASM Vehicle Dynamics provides the environment for the vehicle model. It contains maneuver and road models.

The maneuver provides driving instructions for accelerating, braking, and steering the vehicle model. The road model provides the connection to the road on which the vehicle model travels.

The following tables show the external inports and outports of the ASM module.

Submodules

The module contains the following submodules:

- Maneuver Submodule on page 358
- Road Submodule on page 361

Inports

The following table shows the ports of the Interface_External_In:

Name	Unit	Description
a_y_Vehicle_CoG	[m/s^2]	Lateral acceleration of vehicle with COG in consideration
Active_CP_VehDyn_Trailer	[]	Vector of all active contact points
Angle_SteeringWheel	[deg]	Steering wheel angle
Angle_SteeringWheel_SteeringSystem	[deg]	Steering wheel angle
Angle_Vehicle_CoorSys_E	[x;y;z][deg]	Vehicle orientation (roll, pitch, yaw)
Coef_RollingRes	[FL;FR;RL;RR][]	Rolling resistance coefficient
Coef_RollingRes_Mean	[]	Coefficient for rolling resistance
CorneringStiffness	[FL;FR;RL;RR] [N]	Cornering stiffness of the tires
Gear	[]	Gear
Mode_Drivetrain	[1|2|3|4|5]	Drivetrain mode: ▪ -5: FF ▪ -4: FF-4WD ▪ -3: RWD ▪ -2: FWD ▪ -1: AWD

Name	Unit	Description
Mode_Transmission	[1|2]	Transmission mode switch: ▪ 1: Manual ▪ 2: Automatic
Pos_AccPedal	[%]	Position of the accelerator pedal
Pos_BrakePedal	[%]	Brake pedal position
Pos_ClutchPedal	[%]	Clutch pedal position
Pos_CP_CoorSys_E	[:][x;y][m]	Position of the contact points in the earth coordinates
Pos_CP_FL_CoorSys_E	[x;y][m]	Position of the front left contact points in the earth coordinates
Pos_CP_FL_Trailer_CoorSys_E	[x;y][m]	Position of trailer front left contact points in earth coordinates
Pos_CP_FL2_Trailer_CoorSys_E	[x;y][m]	Position of trailer front (inner) left contact points in earth coordinates
Pos_CP_FR_CoorSys_E	[x;y][m]	Position of the front right contact points in the earth coordinates
Pos_CP_FR_Trailer_CoorSys_E	[x;y][m]	Position of trailer front right contact points in earth coordinates
Pos_CP_FR2_Trailer_CoorSys_E	[x;y][m]	Position of trailer front (inner) right contact points in earth coordinates
Pos_CP_R2L_CoorSys_E	[x;y][m]	Position of the second rear left contact points in the earth coordinates
Pos_CP_R2L_Trailer_CoorSys_E	[x;y][m]	Position of trailer second rear left contact points in earth coordinates
Pos_CP_R2L2_CoorSys_E	[x;y][m]	Position of second rear (inner) left contact points in earth coordinates
Pos_CP_R2L2_Trailer_CoorSys_E	[x;y][m]	Position of trailer second rear (inner) left

Name	Unit	Description
		contact points in earth coordinates
Pos_CP_R2R_CoorSys_E	[x;y][m]	Position of second rear right contact points in earth coordinates
Pos_CP_R2R_Trailer_CoorSys_E	[x;y][m]	Position of trailer second rear right contact points in earth coordinates
Pos_CP_R2R2_CoorSys_E	[x;y][m]	Position of second rear (inner) right contact points in earth coordinates
Pos_CP_R2R2_Trailer_CoorSys_E	[x;y][m]	Position of trailer second rear (inner) right contact points in earth coordinates
Pos_CP_RL_CoorSys_E	[x;y][m]	Position of the rear left contact points in the earth coordinates
Pos_CP_RL_Trailer_CoorSys_E	[x;y][m]	Position of trailer rear left contact points in earth coordinates
Pos_CP_RL2_CoorSys_E	[x;y][m]	Position of the rear (inner) left contact points in the earth coordinates
Pos_CP_RL2_Trailer_CoorSys_E	[x;y][m]	Position of trailer rear (inner) left contact points in earth coordinates
Pos_CP_RR_CoorSys_E	[x;y][m]	Position of the rear right contact points in the earth coordinates
Pos_CP_RR_Trailer_CoorSys_E	[x;y][m]	Position of trailer rear right contact points in earth coordinates
Pos_CP_RR2_CoorSys_E	[x;y][m]	Position of the rear (inner) right contact points in the earth coordinates
Pos_CP_RR2_Trailer_CoorSys_E	[x;y][m]	Position of trailer rear (inner) right contact points in earth coordinates

Name	Unit	Description
r_Dyn_Tire	[FL;FR;RL;RR] [m]	Dynamic tire radius
r_Dyn_Tire_Mean	[m]	Mean dynamic tire radius
RollRate_Vehicle_CoG	[deg/s]	Vehicle roll rate
RotationMatrix_V_to_E	[1x9]	Rotation matrix to transfrom values from the vehicle to the earth coordinate system
s_Preview	[m]	Vector with preview distances for lateral control and vehicle reference velocity calculation.
s_Vehicle	[m]	Driven distance of the vehicle
SelectorLever	[-3|-2|-1|0|1]	Selector lever position: ▪ -3: TipShift ▪ -2: Park ▪ -1: Reverse ▪ 0: Neutral ▪ 1: Drive
Sw_MasterBrakeCylinder_Mode	[1|2|3]	Master brake cylinder mode: ▪ 1: Modeled linearly ▪ 2: Modeled physically ▪ 3: Brake booster
Sw_Tire_Model	[1|2]	Tire model switch: ▪ 1: TMeasy ▪ 2: Magic Formula
t_Preview_vRef_Driver	[s]	Preview time for the driver model
Trq_SteeringWheel_SteeringSystem	[Nm]	Torque in the steering column
v_x_Vehicle_CoG	[km/h]	Longitudinal vehicle velocity
v_y_Vehicle_CoG	[km/h]	Lateral vehicle velocity at the centre of gravity
YawRate_Vehicle_CoG	[deg/s]	Vehicle yaw rate

Outports

The following table shows the ports of the Interface_External_Out:

Name	Unit	Description
Angle_SteeringWheel_Maneuver	[deg]	Steering wheel angle stimulus from the maneuver scheduler
Angle_Vehicle_Init_CoorSys_E	[x;y;z] [rad]	Initial yaw angle of the ASM vehicle in the earth coordinate system
Angle_Yaw_Veh_Road	[deg]	Yaw angle of the reference line of the road at the current position of the ASM vehicle
a_x_Max	[m/s^2]	Maximum longitudinal acceleration for the reference velocity calculation
a_x_Min	[m/s^2]	Minimum longitudinal acceleration for the reference velocity calculation
a_x_Vehicle_Ref_TrafficDriver	[m/s^2]	Reference acceleration of the ASM vehicle by the traffic driver feature
a_y_Max	[m/s^2]	Maximum lateral acceleration (for velocity calculation in curves)
Const_Steering_Mode_Manual	[1|2]	Controls whether the manual steering mode expects the steering wheel angle or the steering wheel torque: ▪ 1: Steering wheel angle ▪ 2: Steering wheel torque
Curv_Preview_Road	[1/m]	Road curvature at previewed road positions
Curv_Vehicle_Road	[1/m]	Curvature of the road reference line at the current position of the ASM vehicle
d_Vehicle_Road	[m]	Lateral distance of the vehicle regarding the preferred lane
Dist_Border_Left	[m]	Distance from the ASM vehicle reference coordinate system to the left road border
Dist_Border_Right	[m]	Distance from the ASM vehicle reference coordinate system to the right road border
ECU_Commands	[]	ECU commands derived from the customized maneuver user signals

Name	Unit	Description
Enable_v_Init	[0|1]	Flag that indicates if the ASM vehicle simulation has a velocity or then zero
EngOPNum	[]	Engine operating point number
Fric_Coeff_CP	[:][]	Friction coefficients at the contact point positions
Gear_Maneuver	[]	Gear stimulus from the maneuver
Gear_Max	[0Off|1_n]	Gear set point defined by maneuver input
Keep_VehiclePosition	[0|1]	Flag to disable all integrators in the model in order to keep the vehicle at its current position
Mode_AcceleratorBrake	[1|2]	Controls whether accelerator and brake pedal positions are stimulus signals or controlled by the driver: ▪ 1: Stimulus signals ▪ 2: Controlled by driver
Mode_AdaptVelocityToRoad	[1|2]	Controls whether the reference velocity calculation on roads is active: ▪ 1: Off ▪ 2: On
Mode_CurvatureCalculation	[1|2]	Specifies the curvature calculation mode: ▪ 1: Road ▪ 2: Driver
Mode_ForceToRoad	[1|2]	Controls the limitation to valid lane indices: ▪ 1: Off ▪ 2: On
Mode_GearClutch	[1|2|3|4]	Switch for the mode signal the of gear and clutch pedal source: ▪ 1: Stimulus ▪ 2: Driver ▪ 3: Open clutch ▪ 4: Reference gear
Mode_LatCtrl	[1|2]	Lateral control type: ▪ 1: Pos ▪ 2: Yaw
Mode_RelativeLane	[0|1]	Controls the interpretation of the lane index in the road calculation: ▪ 0: Center lane

Name	Unit	Description
		▪ 1: Preferred lane
Mode_Road	[1|2|3|4]	Controls whether vehicle travels on a flat area, on a straight or circular road or on the road that is created with the ModelDesk Road Generator: ▪ 1: No road ▪ 2: Straight road ▪ 3: Circular road ▪ 4: Road from Road Generator
Mode_SelectorLever	[1|2]	Switch for the mode signal of the selector lever source: ▪ 1: Stimulus ▪ 2: Driver
Mode_StartButtonState	[1|2]	Controls whether the signal for the activation of the start button is calculated by the soft ECU or provided by a stimulus signal: ▪ 1: Stimulus ▪ 2: Soft ECU
Mode_Steer	[1|2|3]	Controls whether the steering wheel angle is controlled by the stimulus signal or by the driver: ▪ 1: Stimulus ▪ 2: Driver ▪ 3: Fix
Mode_TrafficDriver	[1|2]	Controls the Traffic Driver mode of the ASM vehicle: ▪ 1: Off ▪ 2: On
n_Engine_Set	[rpm]	Setpoint for the speed of combustion engine
p_Ambient	[Pa]	Ambient pressure
Pitch_Preview_Road	[deg]	Pitch angle of the road at the positions of the preview points
Pos_AccPedal_Maneuver	[%]	Accelerator pedal position stimulus from the maneuver scheduler
Pos_BrakePedal_Maneuver	[%]	Brake pedal position stimulus from the maneuver scheduler
Pos_ClutchPedal_Maneuver	[%]	Clutch pedal position stimulus from the maneuver scheduler

Name	Unit	Description
Pos_Vehicle_Init_CoorSys_E	[x;y;z][m]	Initial position of the ASM vehicle in the earth coordinate system
Pos_x_Preview_Road	[m]	Vector of the previewed road positions in x-direction (earth coordinates
Pos_x_Ref_Vehicle_Road	[m]	x-position of the reference line of the road at the current position of the vehicle
Pos_y_Preview_Road	[m]	Vector of previewed road positions in y-direction (earth coordinates)
Pos_y_Ref_Vehicle_Road	[m]	y-position of the reference line of the road at the current position of the vehicle
Pos_z_CP	[:][m]	Z-position of the contact points
Pos_z_Preview_Road	[m]	Vector of the previewed road positions in the z-direction (earth coordinates)
q_dt_Vehicle_Init	[6x1]	Vector signal with the initial velocity and angular velocity of the ASM vehicle
Reset_VehicleStates	[0|1]	Reset signal from the maneuver scheduler
Roll_Preview_Road	[deg]	Roll angle of the road at the positions of the preview points
Route_Direction_Init	[]	Direction on the route for the all fellow vehicles
SelectorLever_Maneuver	[-3|-2|-1|0|1]	Selector lever position from the maneuver: ▪ -3: TipShift ▪ -2: Park ▪ -1: Reverse ▪ 0: Neutral ▪ 1: Drive
Slope_Veh_Road	[%]	Slope of the road at the current distance s of the vehicle
Speed_Limit_Preview_Road	[km/h]	Speed limit as defined in the scenery section of the road element at the position of the ASM vehicle
s_Total_Vehicle_Road	[m]	The total distance along the route
s_Vehicle_Init	[m]	Initial distance s along the road

Name	Unit	Description
s_Vehicle_Road	[m]	The distance s along the route at the position of the vehicle. Reset to zero after each turn on a closed route
State_StartButton	[-1P|0|1|2]	State of the start button: ▪ -1: Power off ▪ 0: Acceleration ▪ 1: Power on ▪ 2: Starter on
Sw_LatDriver	[1|2]	Controls whether lateral_control1 or lateral_control2 is active: ▪ 1: Lateral_control 1 active ▪ 2: Lateral_control 2 active
Sw_ParkingBrake	[0|1]	Controls the parking brake: ▪ 0: Off ▪ 1: On
Sw_StartButton	[0|1]	Start button actuation input: ▪ 0: Off ▪ 1: On
Sw_Steering_Mode	[1|2]	Controls whether the steering angle is determined by the maneuver or manually: ▪ 1: Manual ▪ 2: Maneuver Scheduler
Sw_SteeringMode	[1|2]	Controls the steering angle calculation method (torque or angle): ▪ 1: Angle ▪ 2: Torque
Sw_Testbench	[0|1]	Switch to activate the test bench: ▪ 0: Off ▪ 1: On
Sw_Tire_Parameter_Set_CP	[:][1|2|3|4]	Controls the used tire parameter set
Sw_TipShift_Maneuver	[-1|0|1]	TipShift requests the stimulus signal from the maneuver: ▪ -1: Down shift ▪ 0: Off ▪ 1: Up shift
T_Ambient	[°C]	Ambient temperature
Trq_Driver_External	[Nm]	Reference torque for the steering angle calculation

Name	Unit	Description
UnitVec_z_CP	[:][x;y;z]	Unit vector in z-direction at the contact point positions
x_Preview_DriverCurv	[m]	Vector of the previewed road positions in the x-direction (earth coordinates) for curvature calculation
v_Vehicle_Preview_Ref_Maneuver	[m/s]	Reference signal for the preview speed calculation
v_Vehicle_Ref_Maneuver	[m/s]	Vehicle reference velocity provided by the maneuver
v_Wind_Coor_Sys_E	[x;y;z][m|s]	Velocity of the wind in the earth coordinate system
Width_Deadzone_LatCtrl	[m]	Width of the dead zone for lateral deviation from road reference line
y_Preview_DriverCurv	[m]	Vector of the previewed road positions in the y-direction (earth coordinates) for the curvature calculation

Animation

The following table shows the ports of the Animation:

Name	Unit	Description
ASM_Vehicle_TurnLights	[]	Signal bus containing the turn light signals
Distance_MeasPnt_Marker_Left	[m]	Lateral distance from the lane sensor position to the left lane marking
Distance_MeasPnt_Marker_Right	[m]	Lateral distance from the lane sensor position to the right lane marking
ManeuverState	[]	Maneuver state signal
ManeuverTime	[s]	Time since start of maneuver
MarkerType_Left	[]	Type of the left lane marking. The signal is -1 if there is no marker
MarkerType_Right	[]	Type of the right lane marking. The signal is -1 if there is no marker
Network_Item_Type_Veh	[]	Type of network item at the position of the ASM vehicle
Objects_DataStreams	[]	Data stream to control states in the model
Pos_Left	[x;y;z] [m]	Position of the left lane marking in earth coordinate system at each

Name	Unit	Description
		enabled lane sensor evaluation distance
Pos_Right	[x;y;z] [m]	Position of the right lane marking in earth coordinate system at each enabled lane sensor evaluation distance
s_Veh_Road	[m]	The distance s along the route at the position of the vehicle. Reset to zero after each turn on a closed route
Sw_BrakeLight	[0|1]	Switch for the brake light: ▪ 0: Off ▪ 1: On
Sw_ManualControl_Lights	[1|2]	Switch that determines if the lights are controlled by the MANEUVER_SCHEDULER block or the user interface: ▪ 1: Maneuver ▪ 2: Manual
Sw_TurnLights	[0Off| 1Left| 2Right| 3Both]	Switch for the turn light: ▪ 0: Off ▪ 1: Left ▪ 2: Right ▪ 3: Both
Velocity_MeasPnt_Marker_Left	[m/s]	Velocity from each measurement point towards the left marker in d-direction
Velocity_MeasPnt_Marker_Right	[m/s]	Velocity from each measurement point towards the right marker in the d-direction
Yaw_Angle_Left	[deg]	Yaw angle of the left lane marking in earth coordinate system
Yaw_Angle_Right	[deg]	Yaw angle of the right lane marking in the earth coordinate system

Start System Module

Description

The StartSystem module contains models responsible for switching the engine on and off. It is mainly an ECU which responds to the user actuation request and sends the corresponding instructions to the engine ECU to start or turn off the engine.

The following tables show the external inports and outports of the ASM module.

Inports Control

The following table shows the ports of the Interface_External_In:

Name	Unit	Description
Mode_StartButtonState	[1|2]	Controls whether the signal for the activation of the start button is calculated by the soft ECU or provided by a stimulus signal ▪ 1: By Stimulus signal ▪ 1: By Soft ECU
Mode_Transmission	[1|2]	Transmission mode switch ▪ 1: Manual ▪ 2: Automatic
Pos_BrakePedal	[%]	Brake pedal position
Pos_ClutchPedal	[%]	Clutch pedal position
Reset_States	[0|1]	Reset states
SelectorLever	[-3|-2|-1|0|1]	Selector lever position ▪ -3: TipShift ▪ -2: Park ▪ -1: Reverse ▪ 0: Neutral ▪ 1: Drive
State_Engine	[0|1|2|3|4]	Engine state ▪ 0: Engine off ▪ 1: Ignition on ▪ 2: Ignition on and starter activated ▪ 3: Engine is running ▪ 4: Ignition is switched off, shutdown active
State_StartButton	[-1|0|1|2]	State of the start button ▪ -1: Power off ▪ 0: Accessory on ▪ 1: Power on ▪ 2: Starter on
Sw_CrankShaft_Reset	[0|1]	Crankshaft reset switch ▪ 0: Disabled ▪ 1: Enabled
Sw_StartButton	[0|1]	Start button actuation input ▪ 0: Off ▪ 1: On

The following table shows the interface to connect an external ECU (real ECU):

Name	Unit	Description
Sw_Ignition	[0|1]	Ignition signal - terminal 15
Sw_StarterReq	[0|1]	Starter request signal (usually sent to engine ECU): ▪ 0: Off ▪ 1: On

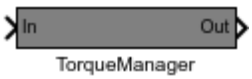
Outports Control

The following table shows the ports of the Interface_External_Out:

Name	Unit	Description
Sw_Ignition	[0|1]	Ignition signal - terminal 15 ▪ 0: Off ▪ 1: On
Sw_StarterReq	[0|1]	Starter request signal (usually sent to engine ECU) ▪ 0: Off ▪ 1: On

Torque Manager Basic Module

Description



The hybrid manager basic model serves as a mock-up for a hybrid manager. It is used if the vehicle only has only a conventional internal combustion engine. The signals are remapped so that the engine is set to internal combustion engine only.

It can be exchanged with a more advanced ASM module.

The following tables show the external inports and outports of the ASM module.

Inports Control

The following table shows the ports of the Interface_External_In:

Name	Unit	Description
Pos_BrakePedal	[%]	Brake pedal position
State_Engine	[0|1|2|3|4]	Engine state ▪ 0: Engine off ▪ 1: Ignition on ▪ 2: Ignition on and starter activated ▪ 3: Engine is running ▪ 4: Ignition is switched off, shutdown active

Name	Unit	Description
Sw_Ignition	[0|1]	Ignition signal - terminal 15: ▪ 0: Off ▪ 1: On
Sw_Starter	[0|1]	Starter activate switch: ▪ 0: Off ▪ 1: On
Sw_StarterReq	[0|1]	Starter request signal (usually sent to the engine ECU): ▪ 0: Off ▪ 1: On

Outports Control

The following table shows the ports of the Interface_External_Out:

Name	Unit	Description
HybridMode_ICE	[0|1|2|3]	Hybrid mode of the combustion engine: ▪ 0: ICE off ▪ 1: ICE on, idle ▪ 2: ICE on, power ▪ 3: Error
Ignition_ICE	[0|1]	Ignition ICE: ▪ 0: Off ▪ 1: On
Mode_BSG	[-1|0|1|2|3|4|5]	Mode of the belt-driven starter generator (BSG): ▪ -1: Not available ▪ 0: Stop combustion engine ▪ 1: Start combustion engine ▪ 2: Generator ▪ 3: Recuperation ▪ 4: Boost ▪ 5: Idle
Pos_BrakePedal_Desired	[%]	Desired brake pedal position
StarterReq_ICE	[0|1]	Starter request ICE: ▪ 0: Off ▪ 1: On
Starter_Drivetrain	[0|1]	Engine starter switch: ▪ 0: Off ▪ 1: On
State_DCLink	[0|1]	State DC link: ▪ 0: Off ▪ 1: On
State_Engine	[0|1|2|3|4]	Engine state: ▪ 0: Engine off ▪ 1: Ignition on ▪ 2: Ignition on and starter activated

Name	Unit	Description
		▪ 3: Engine is running ▪ 4: Ignition is switched off, shutdown active
Sw_DriveMode	[0|1|2]	Drive mode: ▪ 0: Hybrid ▪ 1: ICE only ▪ 2: EM only
Torque_Request_Brake	[Nm]	Torque request brake
Trq_Request_EM	[Nm]	Torque request EM
Trq_Request_ICE	[Nm]	Torque request ICE
Vehicle_Topology	[-2|-1|0|2]	Vehicle topology: ▪ -2: EM only ▪ -1: ICE only ▪ 0: P0 hybrid ▪ 2: P2 hybrid

Vehicle Dynamics Module

Description

This module contains the implementation of a multibody system for a passenger car including tire, aerodynamics, and suspension calculation.

The following tables show the external inports and outports of the ASM module.

Subsystems

The module contains the following subsystems:

- Brake Subsystem on page 363
- Suspension Subsystem on page 364
  - Front Suspension Subsystem on page 365
  - Rear Suspension Subsystem on page 367
- Vehicle Movement Subsystem on page 369

Inports control

The following table shows the ports of the Interface_External_In:

Name	Unit	Description
ECU_Commands	[]	ECU commands derived from the customized maneuver user signals
Enable_Trq_Brake_Extern	[0|1]	Enable using external torque brake
Pos_AccPedal	[%]	Position of the accelerator pedal

Name	Unit	Description
Pos_BrakePedal	[%]	Brake pedal position
Reset_VehicleStates	[0|1]	Reset signal from the maneuver scheduler
Sw_Ignition	[0|1]	Ignition signal - terminal 15 ▪ 0: Off ▪ 1: On
Trq_Brake_Request_Extern	[Nm]	External torque brake request
Trq_Eff_Engine	[Nm]	Effective engine torque
Torque_EM_Front	[Nm]	Front EM torque
Torque_Request_Brake	[Nm]	Torque request brake
Vehicle_Topology	[-2|-1|0|2]	Vehicle topology: ▪ -2: EM only ▪ -1: ICE only ▪ 0: P0 hybrid ▪ 2: P2 hybrid

The following table shows the interface to connect an external ECU (real ECU):

Name	Unit	Description
Ctrl_ChangeOverValve_1	[0_1]	Change over valve control signal of the first braking circuit
Ctrl_ChangeOverValve_2	[0_1]	Change over valve control signal of the second braking circuit
Ctrl_HydPump	[0_1]	Control signal of the hydraulic pump
Ctrl_InletValve_FL	[0_1]	Control signal of the front left inlet valve
Ctrl_InletValve_FR	[0_1]	Control signal of the front right inlet valve
Ctrl_InletValve_RL	[0_1]	Control signal of the rear left inlet valve
Ctrl_InletValve_RR	[0_1]	Control signal of the rear right inlet valve
Ctrl_OutletValve_FL	[0_1]	Control signal of the front left outlet valve
Ctrl_OutletValve_FR	[0_1]	Control signal of the front right outlet valve
Ctrl_OutletValve_RL	[0_1]	Control signal of the rear left outlet valve
Ctrl_OutletValve_RR	[0_1]	Control signal of the rear right outlet valve
Ctrl_PreChargeValve_1	[0_1]	Precharge valve control signal of first braking circuit
Ctrl_PreChargeValve_2	[0_1]	Precharge valve control signal of second braking circuit

Name	Unit	Description
Enable_TrqDecrease_Fast_ESP	[0|1]	Enable torque decrease fast ESP
Enable_TrqDecrease_Slow_ESP	[0|1]	Enable slow torque decrease request for ESP
Enable_TrqIncrease_ESP	[0|1]	Enable ESP torque increase
ESP_Intervention	[0|1]	ESP intervention
Trq_DecreaseRequest_Fast_ESP	[Nm]	Torque decrease request fast ESP
Trq_DecreaseRequest_Slow_ESP	[Nm]	ESP slow torque decrease request
Trq_IncreaseRequest_ESP	[Nm]	ESP torque increase request
Trq_PowerSeering	[Nm]	Power-steering torque

Outports Control

The following table shows the ports of the Interface_External_Out:

Name	Unit	Description
ECU_ESP_Mode	[0|1|2|3]	Switch ESP mode ▪ 0: None ▪ 1: ABS ▪ 2: ESP ▪ 3: ASR
Enable_p_Brake_Desired	[0|1]	Enable desired brake pressure
Enable_TrqDecrease_Fast_ESP	[0|1]	Enable torque decrease fast ESP
Enable_TrqDecrease_Slow_ESP	[0|1]	Enable slow torque decrease request for ESP
Enable_TrqIncrease_ESP	[0|1]	Enable ESP torque increase
p_Brake_Desired	[bar]	Desired brake pressure
Trq_DecreaseRequest_Fast_ESP	[Nm]	Torque decrease request fast ESP
Trq_DecreaseRequest_Slow_ESP	[Nm]	ESP slow torque decrease request
Trq_IncreaseRequest_ESP	[Nm]	ESP torque increase request

Inports Plant

The following table shows the ports of the Interface_External_In:

Name	Unit	Description
Angle_SteeringWheel	[deg]	Steering wheel angle
Angle_Vehicle_Init_CoorSys_E	[x;y;z][rad]	Initial yaw angle of the ASM vehicle in the earth coordinate system
Const_Steering_Mode_Manual	[1|2]	Controls whether the manual steering mode expects the steering wheel

Name	Unit	Description
		angle or the steering wheel torque ▪ 1: Steering wheel angle ▪ 2: Steering wheel torque
Pos_BrakePedal	[%]	Brake pedal position
Fric_Coeff_CP	[FL;FR;RL;RR][]	Friction coefficients at the contact point positions
Keep_VehiclePosition	[0|1]	Flag to disable all integrators in the model in order to keep the vehicle at its current position
M_Articulated_T	[1x36]	Articulated mass matrix
Pos_AccPedal	[%]	Position of the accelerator pedal
p_Ambient	[Pa]	Ambient pressure
Pos_Vehicle_Init_CoorSys_E	[x;y;z][m]	Initial position of the ASM vehicle in earth coordinate system
Pos_z_CP	[FL;FR;RL;RR][m]	z-position of the contact points
Q_Articulated_T	[6x1]	External articulated wrench
q_dt_Vehicle_Init	[6x1]	Vector signal with the initial velocity and angular velocity of the ASM vehicle
Reset_VehicleStates	[0|1]	Reset signal from maneuver scheduler
Sw_ParkingBrake	[0|1]	Controls the parking brake ▪ 0: Off ▪ 1: On
Sw_SteeringMode	[1|2]	Controls the steering angle calculation method: ▪ 1: Angle ▪ 2: Torque
Sw_Steering_Mode	[1|2]	Controls whether the steering angle is determined by the maneuver or manually ▪ 1: Manually ▪ 2: Maneuver schedule
Sw_Tire_Parameter_Set_CP	[FL;FR;RL;RR][1|2|3|4]	Controls the used tire parameter set.
Trq_Driver_External	[Nm]	Reference torque for the steering angle calculation

Name	Unit	Description
Trq_Hitch_CoorSys_V	[x;y;z][Nm]	Hitch torque w.r.t vehicle coordinate system
Trq_Shaft	[FL,FR,RL,RR] [Nm]	Torque of the output shaft
T_Ambient	[°C]	Ambient temperature
UnitVec_z_CP	[FL;FR;RL;RR] [x;y;z]	Unit vector in the z-direction at the contact point positions
v_Wind_Coor_Sys_E	[x;y;z][m|s]	Velocity of the wind in the earth coordinate system

The following table shows the interface to connect an external plant (real plant):

Name	Unit	Description
Angle_dt_EPSMotor	[rad/s]	EPS motor angular velocity
Angle_EPSMotor	[rad]	EPS motor rotation angle
Displ_Steering	[m]	Steering rod displacement
Displ_Steering_dt	[m/s]	Steering rod velocity

Outports Plant

The following table shows the ports of the Interface_External_Out:

Name	Unit	Description
a_y_Vehicle_CoG	[m/s^2]	Lateral acceleration of vehicle with COG in consideration
Active_CP_VehDyn	[]	Vector of active vehicle dynamics contact points
Angle_SteeringWheel	[deg]	Steering wheel angle
Angle_Vehicle_CoorSys_E	[x;y;z][deg]	Vehicle orientation (roll, pitch, yaw)
Angle_Yaw_Vehicle_CoorSys_E	[deg]	Vehicle yaw angle
Coef_RollingRes	[FL;FR;RL;RR][]	Rolling resistance coefficient
Coef_RollingRes_Mean	[]	Coefficient for rolling resistance
CorneringStiffness	[FL;FR;RL;RR][N]	Cornering stiffness of tires
CorneringStiffness_Tire_Front	[N/rad]	Cornering stiffness of the front tires
CorneringStiffness_Tire_Rear	[N/rad]	Cornering stiffness of rear tires
Cw_x	[]	Longitudinal aerodynamics coefficients

Name	Unit	Description
F_Driving_Res	[N]	Sum of the driving resistances acting on the vehicle
F_Vehicle_Gravitational	[x;y;z][N]	Gravitational force
F_x_Aero_CoorSys_V	[N]	Longitudinal aerodynamic force
i_Steering_Mean	[]	Mean value of steering gear ratio
Inertia_z_Vehicle_Total	[kgm^2]	Vehicle inertia around the z-axis
m_TotalVehicle	[kg]	Total vehicle mass
omega_Mean_Wheel	[rad/s]	Mean wheel speed
omega_Wheel	[FL;FR;RL;RR] [rad/s]	Wheel speed
Pos_CP_FL_CoorSys_E	[x;y][m]	Position of the front left contact points in the earth coordinates
Pos_CP_FR_CoorSys_E	[x;y][m]	Position of the front right contact points in the earth coordinates
Pos_CP_R2L_CoorSys_E	[x;y][m]	Position of the second rear left contact points in the earth coordinates
Pos_CP_R2L2_CoorSys_E	[x;y][m]	Position of the second rear (inner) left contact points in the earth coordinates
Pos_CP_R2R_CoorSys_E	[x;y][m]	Position of the second rear right contact points in the earth coordinates
Pos_CP_R2R2_CoorSys_E	[x;y][m]	Position of the second rear (inner) right contact points in the earth coordinates
Pos_CP_RL_CoorSys_E	[x;y][m]	Position of the rear left contact points in the earth coordinates
Pos_CP_RL2_CoorSys_E	[x;y][m]	Position of the rear (inner) left contact points in the earth coordinates
Pos_CP_RR_CoorSys_E	[x;y][m]	Position of the rear right contact points in the earth coordinates
Pos_CP_RR2_CoorSys_E	[x;y][m]	Position of the rear (inner) right contact points in the earth coordinates

Name	Unit	Description
Pos_FL_Wheel	[x;y;z][m]	Front left wheel position
Pos_FR_Wheel	[x;y;z][m]	Front right wheel position
Pos_Vehicle_CoorSys_E	[x;y;z][m]	Vector containing the ASM vehicle position values in the earth coordinate system [x,y,z]
Pos_x_CoG_Vehicle_Total	[m]	x-element of position vector of vehicle centre of gravity
Pos_x_Vehicle_CoorSys_E	[m]	x vehicle position
Pos_y_Vehicle_CoorSys_E	[m]	y vehicle position
r_Dyn_Tire	[FL;FR;RL;RR][m]	Dynamic tire radius
r_Dyn_Tire_Mean	[m]	Mean dynamic tire radius
r_Tire_extern	[m]	External tire radius
RollRate_Vehicle_CoG	[deg/s]	Vehicle roll rate
RotationMatrix_V_to_E	[1x9]	Rotation matrix to transfrom the values from the vehicle to the earth coordinate system
RotationMatrix_V_to_E_T	[1x9]	Rotation matrix from the earth to vehicle coordinate system
s_Vehicle	[m]	Driven distance of the vehicle
Sw_MasterBrakeCylinder_Mode	[1|2|3]	Master brake cylinder mode ▪ 1: Modeled linearly ▪ 2: Modeled physically ▪ 3: Brake booster
Sw_r_Tire	[1|2]	Switch for the source of the tire radius: ▪ 1: Intern ▪ 2: Extern
Sw_Rotation	[1|2]	Flag between the Bryant angle and Toe-Camber-SideView interpretation ▪ 1: Cardan angles ▪ 2: Toe-Camber-Caster angles
Sw_Tire_Model	[1|2]	Switch tire model ▪ 1: TMeasy ▪ 2 Magic formula
Trq_Max_Brake	[Nm]	Maximum brake torque
Trq_SteeringWheel	[Nm]	Steering wheel torque at steering column

Name	Unit	Description
v_Vehicle	[x;y;z][km/h]	Vehicle velocity
v_Vehicle	[km/h]	Vehicle velocity
v_Vehicle_dt	[x;y;z][m/s^2]	Derivative of vehicle velocity
v_x_Vehicle_CoG	[km/h]	Longitudinal vehicle velocity
v_y_Vehicle_CoG	[km/h]	Lateral vehicle velocity at centre of gravity
w_Vehicle	[x;y;z][deg/s]	Vehicle angular velocity
w_Vehicle_dt	[x;y;z][rad/s^2]	Derivative of vehicle angular velocity
YawRate_Vehicle_CoG	[deg/s]	Vehicle yaw rate

Animation

The following table shows the ports of the Animation:

Name	Unit	Description
Angle_FL_Wheel	[Toe;Camber][deg]	Animation signal
Angle_FR_Wheel	[Toe;Camber][deg]	Animation signal
Angle_RL_Wheel	[Toe;Camber][deg]	Animation signal
Angle_Rot_FL_Wheel	[deg]	Animation signal
Angle_Rot_FR_Wheel	[deg]	Animation signal
Angle_Rot_RL_Wheel	[deg]	Animation signal
Angle_Rot_RR_Wheel	[deg]	Animation signal
Angle_RR_Wheel	[Toe;Camber][deg]	Animation signal
Angle_SteeringWheel	[deg]	Steering wheel angle
Angle_Vehicle_CoorSys_E	[x;y;z][deg]	Vehicle orientation (roll, pitch, yaw)
Angle_Yaw_Vehicle_CoorSys_E	[deg]	Vehicle yaw angle
a_y_Vehicle_CoG	[m/s^2]	Lateral acceleration of vehicle with COG in consideration
F_FL_Tire_CoorSys_V	[x;y;z][N]	Animation signal
F_FR_Tire_CoorSys_V	[x;y;z][N]	Animation signal
F_RL_Tire_CoorSys_V	[x;y;z][N]	Animation signal
F_RR_Tire_CoorSys_V	[x;y;z][N]	Animation signal
p_Brake	[FL;FR;RL;RR][bar]	Animation signal
Pos_FL_Wheel	[x;y;z][m]	Front left wheel position
Pos_FR_Wheel	[x;y;z][m]	Front right wheel position
Pos_RR_Wheel	[x;y;z][m]	Animation signal
Pos_Vehicle_CoorSys_E	[x;y;z][m]	Vector containing the ASM vehicle position

Name	Unit	Description
		values in the earth coordinate system [x,y,z]
r_Dyn_FL_Tire	[m]	Animation signal
r_Dyn_FR_Tire	[m]	Animation signal
r_Dyn_RL_Tire	[m]	Animation signal
r_Dyn_RR_Tire	[m]	Animation signal
s_Vehicle	[m]	Driven distance of the vehicle
v_x_Vehicle_CoG	[km/h]	Longitudinal vehicle velocity
YawRate_Vehicle_CoG	[deg/s]	Vehicle yaw rate

Subsystems

Introduction

The following subsystems are contained inside the ASM modules.

Where to go from here

Information in this section

Subsystems of the Driver Module.....336
Subsystems of the Drivetrain Module.....341
Subsystems of the Engine Basic Module.....355
Subsystems of the Environment Module.....358
Subsystems of the Vehicle Dynamics Module.....362

Subsystems of the Driver Module

Where to go from here

Information in this section

Lateral Controller Subsystem.....336
The LateralController keeps the vehicle on the road by controlling the steering wheel.

Longitudinal Controller Subsystem.....338
The LongitudinalController subsystem includes the main longitudinal driver models which are used to control the accelerator pedal, brake pedal, selector lever, and clutch pedal as well as gear.

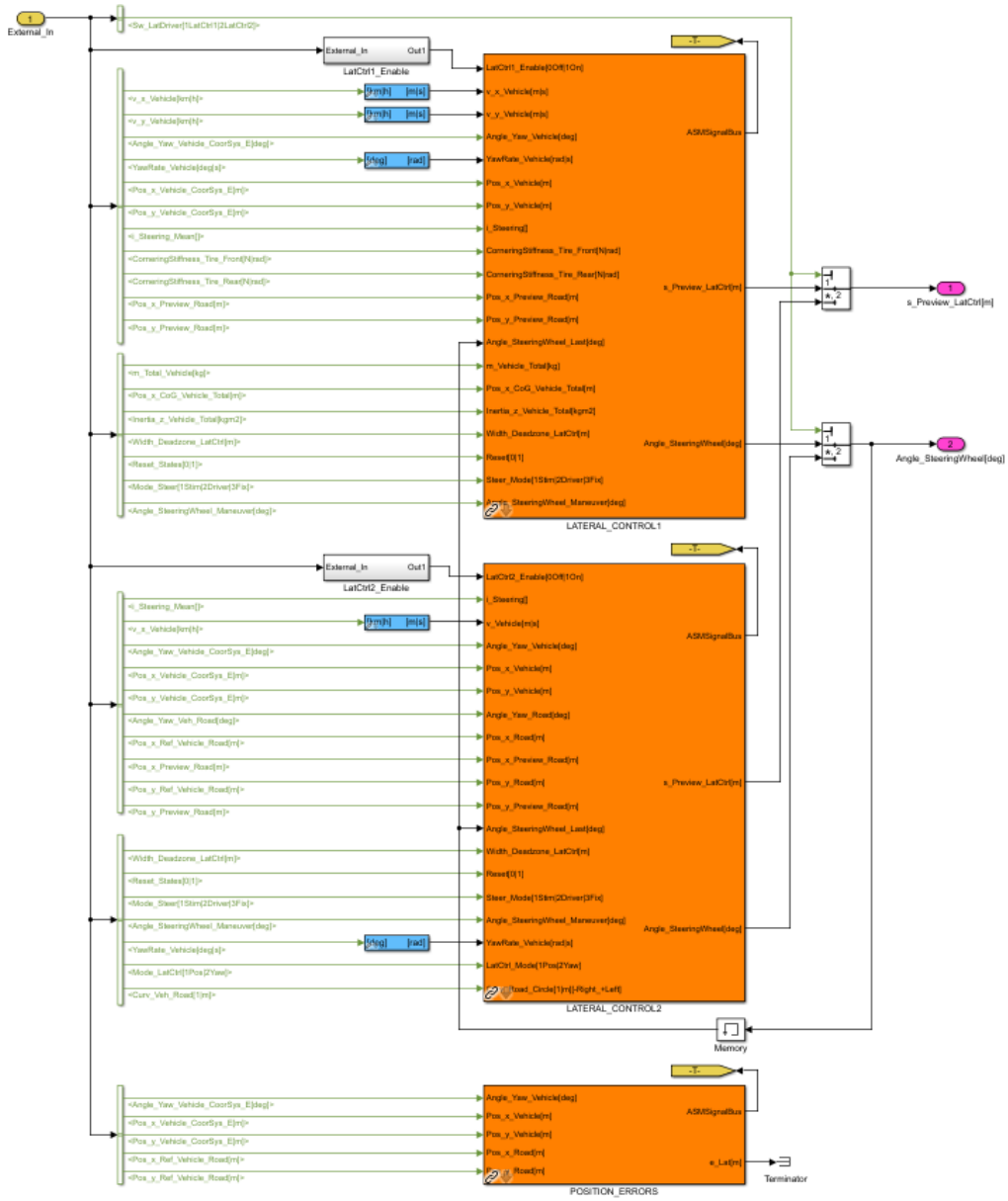
Reference Velocity Subsystem.....341
The ReferenceVelocity subsystem provides the reference velocity for the longitudinal controller.

Lateral Controller Subsystem

Overview

The LateralController keeps the vehicle on the road by controlling the steering wheel.

There are a variety of driving situations a driver model has to handle. To provide optimal control results in every driving situation, two lateral controllers are implemented. One lateral controller offers optimal results for driving on roads, the other lateral controller is optimized for stationary driving situations such as driving in a circle. The maneuver selects the best lateral controller for each driving situation. Both lateral controllers use preview information on the road for the control law.



The Lateral_Control1 subsystem is recommended for road following maneuvers.

The **Lateral_Control2** subsystem is recommended for stationary driving situations such as driving in a circle at a given lateral acceleration.

The **Position_Errors** subsystem provides data on the deviation between vehicle position and current road position.

Subsystems

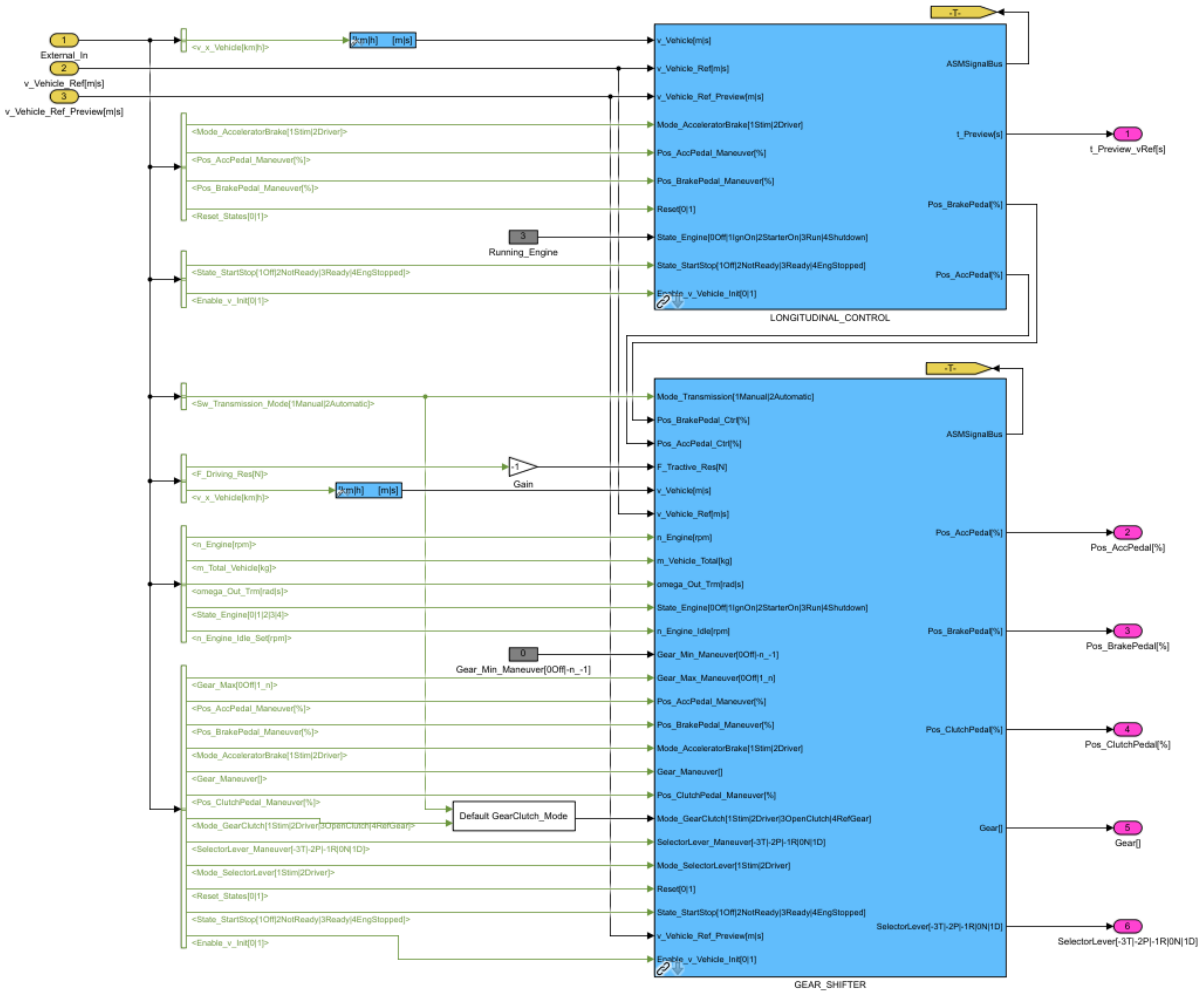
- Lateral Controller 1 (ASM Environment Reference 🕮)
- Lateral Controller 2 (ASM Environment Reference 🕮)
- Position Errors (ASM Environment Reference 🕮)

Longitudinal Controller Subsystem

Overview

The **LongitudinalController** subsystem includes the main longitudinal driver models which are used to control the accelerator pedal, brake pedal, selector lever, and clutch pedal as well as gear.

The longitudinal driver models can handle several drivetrain and vehicle variants, for example four-wheel drive and front-wheel drive. Therefore, the block interface allows easy adaptation to a new variant. The following figure shows the **LongitudinalController** subsystem:



LONGITUDINAL CONTROL block The controller controls the accelerator pedal and brake pedal according to a reference velocity. This velocity is provided by the reference velocity subsystem. Refer to Reference Velocity Subsystem on page 341.

GEAR_SHIFTER block The gear shifter controls the gears and clutch in manual transmission. The block also contains the vehicle move-off functionality.

Simulation scenarios

The longitudinal driver models are configurable and can be set to perform several simulation scenarios. These are shown in the following table:

Scenario	Source		Driver task	
	Reference	Stimulus	Manual transmission	Automatic transmission
1	Velocity	–	Driver controls AccPedal, BrakePedal, Gear and ClutchPedal	Driver controls AccPedal, BrakePedal and SelectorLever
2	Velocity Gear	–	Driver controls AccPedal, BrakePedal and ClutchPedal	Driver controls AccPedal, BrakePedal and SelectorLever

Scenario	Source		Driver task	
	Reference	Stimulus	Manual transmission	Automatic transmission
3	Velocity	ClutchPedal Gear	Driver controls AccPedal and BrakePedal	Driver controls AccPedal, BrakePedal and SelectorLever
4	–	AccPedal BrakePedal	Driver controls ClutchPedal and Gear	Driver controls SelectorLever

These simulation scenarios are illustrated in more details in the following:

Scenario 1: The driver follows the reference velocity by controlling the accelerator, brake, and clutch pedals. Additionally, the gear or the selector lever is controlled depending on the transmission type.

Scenario 2: The driver follows the reference velocity and the gear by controlling the accelerator, brake, and clutch pedals. In this case, the reference gear is used to trigger the clutch pedal control or the selector lever depending on the transmission type.

Scenario 3: The driver follows the reference velocity. The gearshift and the clutch pedal are predefined. In this case, the driver controls the accelerator and brake pedal. If automatic transmission is active, the predefined gear is used to trigger the selector lever control.

Scenario 4: The driver controls the clutch pedal and gear as well as the selector lever according to the accelerator and brake pedal, which come from stimulus.

The longitudinal driver models offer a wide variety of features, which you can configure via the model parameters. The following table shows the important features.

Feature	Description
Engine stall detection	All controls are disabled, when the engine has been stalled
Gear skipping and hold during down shift	During emergency or hard brake, the driver holds the current gear or skips gears during fast down shift
Gear skipping during up shift	The driver skips gears during acceleration
Startup gear	The driver selects the most suitable gear during startup
Gear passing through neutral during gearshift	The driver simulates passing the gear through neutral during gearshift
Clutch hold during gearshift	The driver holds the clutch completely open during gearshift
Following small velocities	The driver controls a slipping clutch to follow small velocities
Considering the tractive resistance	The driver takes the tractive resistance into consideration during the startup and gearshift

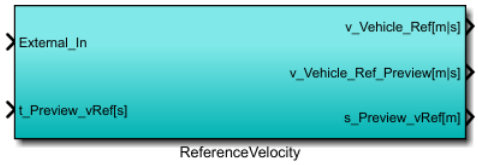
Subsystems

- Longitudinal Control (ASM Environment Reference 🕮)
- Gear Shifter (ASM Environment Reference 🕮)

Reference Velocity Subsystem

Overview

The **ReferenceVelocity** subsystem provides the reference velocity for the longitudinal controller. The defined speed course is adapted according to the horizontal (curves) as well as vertical (uphill, downhill) road conditions. The reference velocity provided by the maneuver is automatically adapted to the road according to preview information and user-defined driver characteristics.



For a detailed description of the reference velocity calculation, refer to Road Reference Velocity (ASM Environment Reference 🕮).

Subsystems

- Road Reference Velocity (ASM Environment Reference 🕮)
- Speed Profiler (ASM Environment Reference 🕮)
- Curvature (ASM Environment Reference 🕮)

Subsystems of the Drivetrain Module

Where to go from here

Information in this section

Final Drive Assembly Subsystem.....342
The final drive assembly is the part of the drivetrain which transmits the power from the transmission to the wheels. Depending on the drivetrain configuration, it can consist of differentials, shafts, gearboxes and clutches.

Transmission Subsystem.....348
The transmission system converts the engine speed and engine torque with different gear ratios to the rest of the drivetrain. The transmission system contains a manual and an automatic transmission.

Automatic Transmission Subsystem.....349
The automatic transmission converts the engine speed and engine torque with different gear ratios to the output side of the transmission.

Manual Transmission Subsystem.....352
The manual transmission converts the engine speed and engine torque with different gear ratios to the output side of the transmission.

Final Drive Assembly Subsystem

Description

The final drive assembly is the part of the drivetrain which transmits the power from the transmission to the wheels. Depending on the drivetrain configuration, it can consist of differentials, shafts, gearboxes and clutches.

The library comes with standard final drive assembly demos for flexible and rigid drivetrain simulation.

These standard demos support several drivetrain configurations:

- All wheel drive (AWD)
- Front wheel drive (FWD)
- Rear wheel drive (RWD)
- Front wheel front engine (FF)
- Front wheel front engine with four wheel drive extension (FF-4WD)

You can switch between the different drivetrain configurations by setting the drivetrain mode switch. To switch between flexible and rigid drivetrain, you have to additionally exchange several Simulink subsystems.

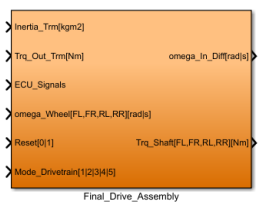
After you updated your model with the correct subsystem (flexible or rigid drivetrain), the illustrated drivetrain configurations can also be simulated by selecting the drivetrain variant in ModelDesk. Refer to How to Configure a Parameter Set (ModelDesk Parameterizing 🕮).

For a detailed description of how to switch between different drivetrain types, refer to How to Change the Drivetrain Configurations (ASM Vehicle Dynamics Model Description 🕮).

Stabilization blocks are also prepared for the flexible drivetrain simulation with standard demos. However, to build a non-standard drivetrain configuration, for example, the stabilization blocks have to be adapted accordingly. For information on stabilizing the drivetrain solver, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).

For the rigid drivetrain simulation, no stabilization is needed. This means, in addition to the demos, further drivetrain configurations can be built easily. Therefore, this type of simulation offers more flexibility when different drivetrain configurations have to be simulated.

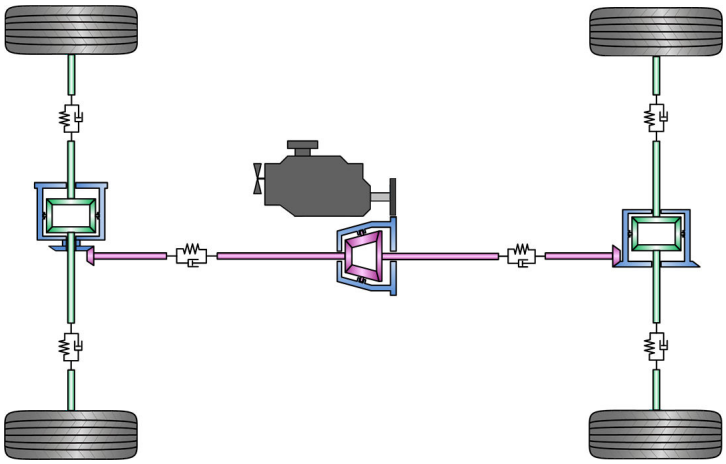
The following figure shows a **Final_Drive_Assembly** subsystem:



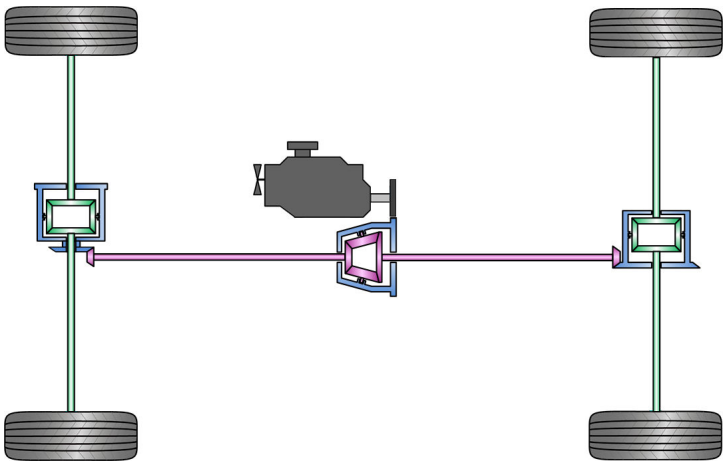
All wheel drive (AWD)

The power is distributed to the front and rear axles via a central differential. Both front and rear axle are permanently driven.

AWD (flexible drivetrain)



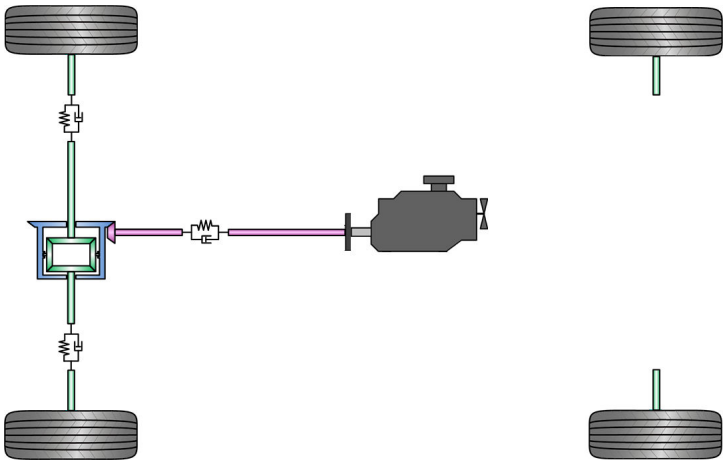
AWD (rigid drivetrain)



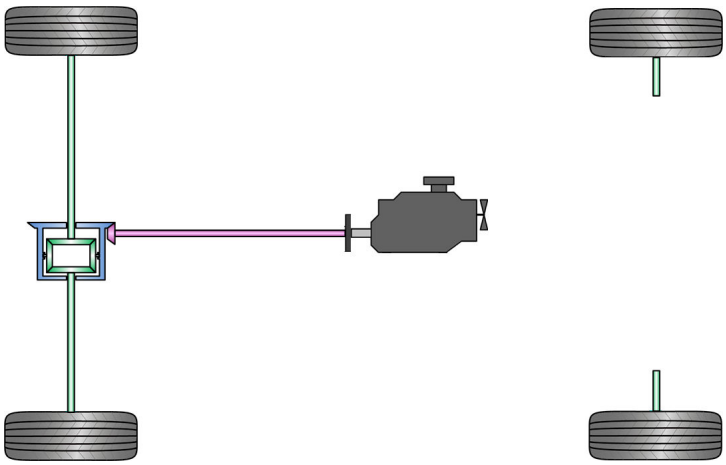
Front wheel drive (FWD)

The whole power is transmitted to the front axle, whereas the rear axle is dragged.

FWD (flexible drivetrain)



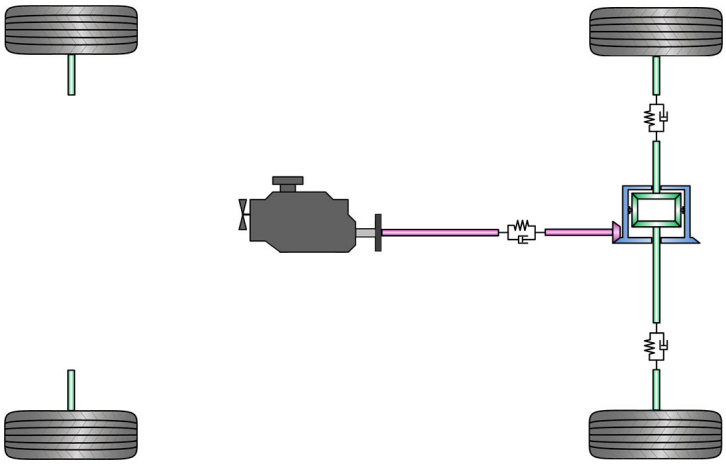
FWD (rigid drivetrain)



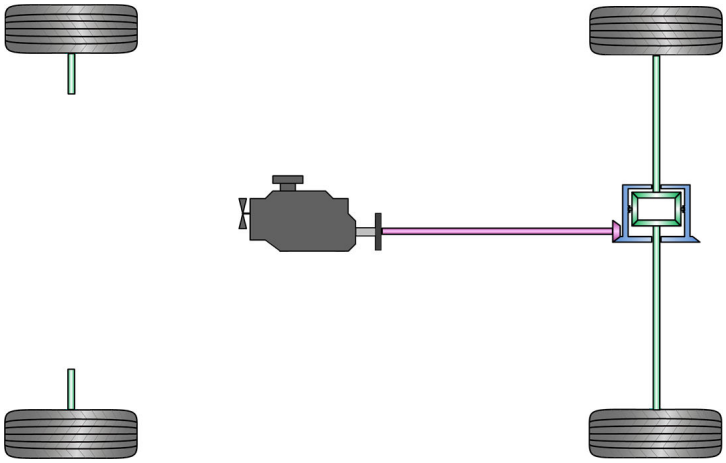
Rear wheel drive (RWD)

The whole power is transmitted to the rear axle, whereas the front axle is pushed.

RWD (flexible drivetrain)



RWD (rigid drivetrain)



Front wheel front engine drive (FF)

The power source is attached to the front axle, and thereby the front axle is directly driven. The rear axle is dragged.

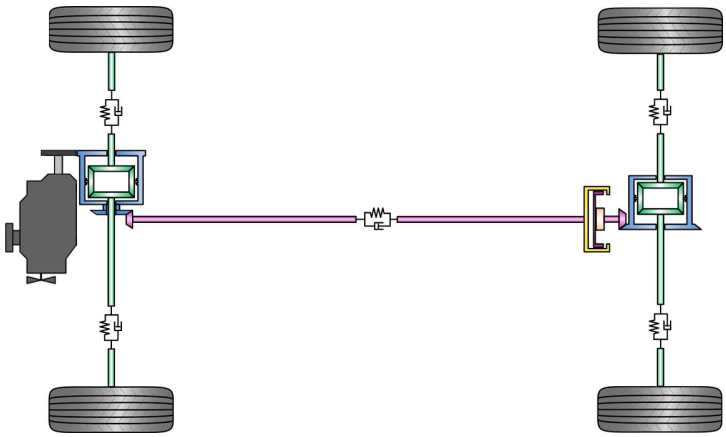
FF (Flexible Drivetrain)

FF (rigid drivetrain)

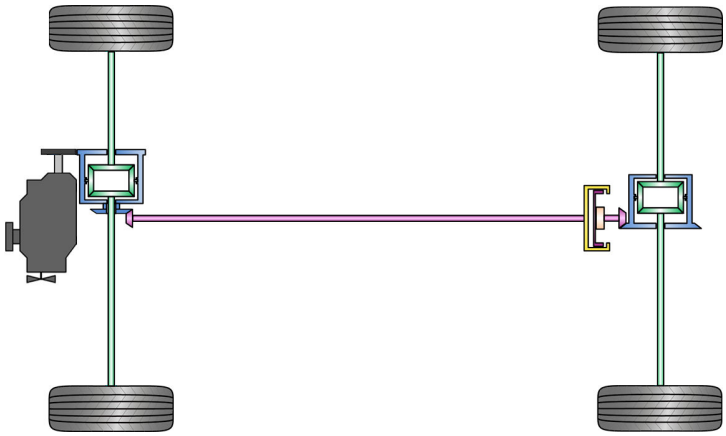
Front wheel front engine drive with optional four wheel drive extension (FF-4WD)

The power source is attached to the front axle, and thereby the front axle is directly driven. The rear axle can be optionally driven via a controllable clutch.

FF-4WD (Flexible Drivetrain)



FF-4WD (rigid drivetrain)



Inports

The following table shows the inports:

Name	Unit	Description
ECU_Signals	[]	Control signals from the transmission control unit
Inertia_Trm	[kg/m^2]	Transmission inertia reduced to the output shaft
Mode_Drivetrain	[1|2|3|4|5]	Drivetrain mode: ▪ 1: AWD ▪ 2: FWD ▪ 3: RWD ▪ 4: FF 4WD ▪ 5: FF
omega_Wheel[FL,FR,RL,RR]	[rad/s]	Rotational speed of the wheel about its axis of rotation
Reset	[0|1]	Reset of states
Trq_Out_Trm	[Nm]	Transmission output torque

Outports

The following table shows the outports:

Name	Unit	Description
omega_In_Diff	[rad/s]	Differential input speed
Trq_Shaft[FL,FR,RL,RR]	[Nm]	Driving torque acting on the driving shafts

Related topics

HowTos

How to Change the Drivetrain Configurations (ASM Vehicle Dynamics Model Description 🕮)
How to Configure a Parameter Set (ModelDesk Parameterizing 🕮)

References

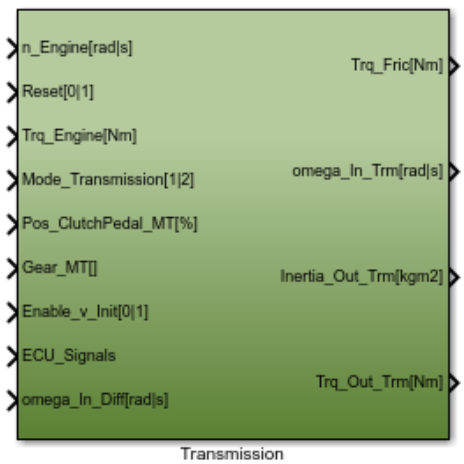
Integration Method (ASM Vehicle Dynamics Addendum 🕮)

Transmission Subsystem

Overview

The transmission system converts the engine speed and engine torque with different gear ratios to the rest of the drivetrain. The transmission system contains a manual and an automatic transmission. The manual transmission contains a clutch model and a gearbox model. The automatic transmission contains a torque converter, lockup clutch, and gearbox model. The models are implemented as enabled subsystems, so only one of them can be active at a time. This makes it possible to switch between the transmission systems efficiently.

The library comes with standard transmission demos for flexible and rigid drivetrain simulation.



Note

To obtain meaningful simulation results, it is not recommended to switch between the transmission models while the vehicle model is in operating mode. Stop the vehicle first.

Inports

The following table shows the inports:

Name	Unit	Description
ECU_Signals	[]	Controls signals from the transmission control unit
Enable_v_Vehicle_Init	[0|1]	Drivetrain initialization flag for scenarios with non-zero vehicle initial velocity.
Gear_MT	[]	Gear of the manual transmission
n_Engine	[rad/s]	Engine speed
omega_in_Diff	[rad/s]	Differential input speed
Pos_ClutchPedal_MT	[%]	Position of clutch pedal: 0% means fully closed and 100% means fully open
Reset	[0|1]	Reset of states
Mode_Transmission	[1|2]	Transmission mode: ▪ 1: Manual transmission ▪ 2: Automatic transmission
Trq_Engine	[Nm]	Engine torque

Outports

The following table shows the outports:

Name	Unit	Description
Inertia_Out_Trm	[kg m^2]	Gearbox inertia reduced to the transmission output shaft
omega_In_Trm	[rad/s]	Transmission output speed
Trq_Fric	[Nm]	Clutch friction torque in manual transmission. Converter and lockup clutch torque in automatic transmission
Trq_Out_Trm	[Nm]	Transmission output torque

Subsystems

- Automatic Transmission Subsystem on page 349
- Manual Transmission Subsystem on page 352

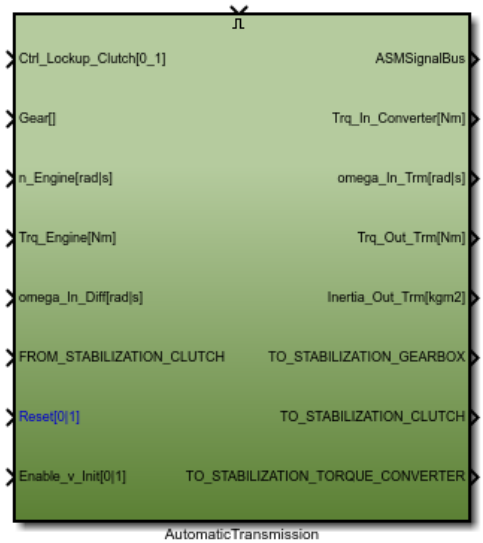
Automatic Transmission Subsystem

Description

The automatic transmission converts the engine speed and engine torque with different gear ratios to the output side of the transmission. The gear is set by the transmission control unit, which can be a real ECU or a soft ECU. For more information about the transmission soft ECU, refer to Transmission on page 95. The automatic transmission contains a torque converter, a lockup clutch, and a gearbox model.

Transmission system for flexible drivetrain

The following illustration shows the automatic transmission subsystem for the flexible drivetrain:



Inports The following table shows the inports:

Name	Unit	Description
Ctrl_Lockup_Clutch	[%]	Lockup clutch control signal: 0[%] means fully closed, 100[%] means fully open
Enable_v_Vehicle_Init	[0|1]	Drivetrain initialization flag for scenarios with non-zero vehicle initial velocity.
FROM_STABILIZATION_CLUTCH	[]	Interface from drivetrain stabilization to **ASM LOCKUP_CLUTCH** block. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
Gear	[]	Gear
n_Engine	[rad/s]	Engine speed
omega_In_Diff	[rad/s]	Differential input speed
Reset	[0|1]	Reset of states
Trq_Engine	[Nm]	Engine torque

Outports The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Inertia_Out_Trm	[kgm^2]	Inertia reduced to the transmission output shaft
TO_STABILIZATION_CLUTCH	[]	Interface from **ASM CLUTCH** block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).

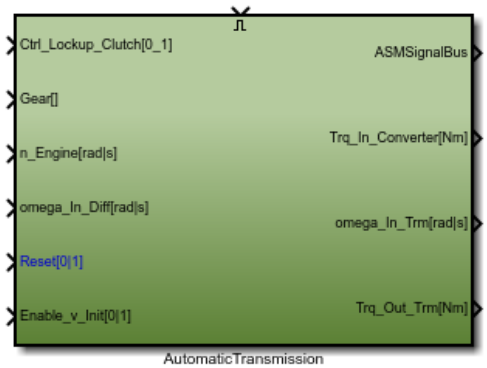
Name	Unit	Description
TO_STABILIZATION_GEARBOX	[]	Interface from ASM GEARBOX block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
TO_STABILIZATION_TORQUE_CONVERTER	[]	Interface from ASM TORQUE_CONVERTER block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
Trq_In_Converter	[Nm]	Converter and lockup clutch torque
Trq_Out_Trm	[Nm]	Transmission output torque

Blocks

- Torque Converter on page 31
- Lockup Clutch on page 29
- Gearbox Automatic Transmission on page 53

Transmission system for rigid drivetrain

The following illustration shows the automatic transmission subsystem for the rigid drivetrain:



Inports The following table shows the inports:

Name	Unit	Description
Ctrl_Lockup_Clutch	0_1	Lockup clutch control signal. *0* means fully closed, *1* means fully open.
Enable_v_Vehicle_Init	[0|1]	Drivetrain initialization flag for scenarios with non-zero vehicle initial velocity
Gear	[]	Reqeusted gear from the transmission control unit
n_Engine	[rad/s]	Engine speed
omega_In_Diff	[rad/s]	Differential input speed
Reset	[]	Reset of states

Outports The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).

Name	Unit	Description
omega_In_Trm	[rad/s]	Transmission output speed
Trq_In_Converter	[Nm]	Converter and lockup clutch torque
Trq_Out_Trm	[Nm]	Transmission output torque

Blocks

- Torque Converter Rigid on page 34
- Lockup Clutch Rigid on page 30
- Gearbox Automatic Transmission Rigid on page 55

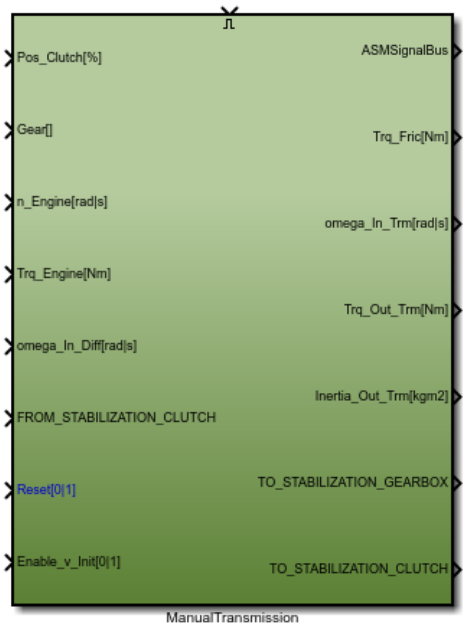
Manual Transmission Subsystem

Description

The manual transmission converts the engine speed and engine torque with different gear ratios to the output side of the transmission. The manual transmission contains a clutch model and a gearbox model.

Manual transmission for flexible drivetrain

The figure below shows the ManualTransmission subsystem for the flexible drivetrain.



Inports The following table shows the inports:

Name	Unit	Description
Enable_v_Vehicle_Init	[0|1]	Drivetrain initialization flag for scenarios with non-zero vehicle initial velocity
FROM_STABILIZATION_CLUTCH	[]	Interface from drivetrain stabilization to **ASM CLUTCH** block. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
Gear	[]	Gear requested by the driver
n_Engine	[rad/s]	Engine speed
omega_in_Diff	[rad/s]	Differential input speed
Pos_Clutch	[%]	Position of clutch pedal in percent: 0% position means clutch fully closed and 100% means fully open
Reset	[0|1]	Reset of states
Trq_Engine	[Nm]	Engine torque

Outports The following table shows the outports:

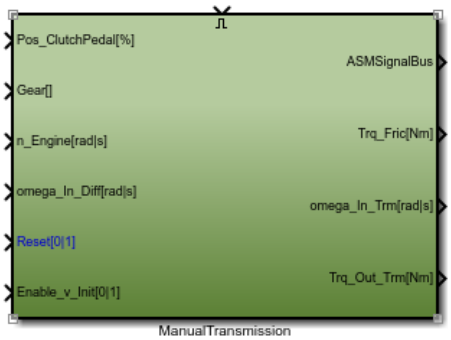
Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Inertia_Out_Trm	[kg m^2]	Gearbox inertia reduced to the transmission output shaft
omega_In_Trm	[rad/s]	Transmission output speed
TO_STABILIZATION_CLUTCH	[]	Interface from **ASM CLUTCH** block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
TO_STABILIZATION_GEARBOX	[]	Interface from **ASM GEARBOX** block to drivetrain stabilization. For more information on the stabilization, refer to Integration Method (ASM Vehicle Dynamics Addendum 🕮).
Trq_fric	[Nm]	Friction torque
Trq_Out_Trm	[Nm]	Transmission output torque

Blocks

- Clutch on page 18
- Gearbox Manual Transmission on page 56

Manual transmission for rigid drivetrain

The figure below shows the ManualTransmission subsystem for the rigid drivetrain:



Inports The following table shows the inports:

Name	Unit	Description
Enable_v_Vehicle_Init	[0|1]	Drivetrain initialization flag for scenarios with non-zero vehicle initial velocity
Gear	[]	Gear requested by the driver
n_Engine	[rad/s]	Engine speed
omega_In_Diff	[rad/s]	Differential input speed
Pos_ClutchPedal	[%]	Position of clutch pedal in percent: 0[%] means fully closed and 100[%] means fully open
Reset	[0|1]	Reset of states

Outports The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
omega_In_Trm	[rad/s]	Transmission output speed
Trq_Fric	[Nm]	Friction torque
Trq_Out_Trm	[Nm]	Transmission output torque

Blocks

- Clutch Rigid on page 22
- Gearbox Manual Transmission Rigid on page 58

Subsystems of the Engine Basic Module

Where to go from here

Information in this section

Engine Basic Subsystem.....355
The engine is the power source in the drivetrain, and the driver controls the engine by the accelerator pedal.

Soft ECU Engine Subsystem.....355
The soft ECU engine gets torque requests from different ECUs and systems, for example, an ESP ECU, a CC ECU or a hybrid management system.

Engine Basic Subsystem

Overview

The engine is the power source in the drivetrain, and the driver controls the engine by the accelerator pedal. The output of the engine is the torque resulting from both the combustion and the resulting engine speed. The engine model does not model the combustion process. In the engine block, the engine torque is calculated as a function of engine speed and accelerator pedal position

Subsystems

- Engine on page 73
- Fuel Consumption on page 75

Soft ECU Engine Subsystem

Description

The soft ECU engine gets torque requests from different ECUs and systems, for example, an ESP ECU, a CC ECU or a hybrid management system. These requests depend on the dynamic states of the vehicle, which require the engine torque to be decreased or increased. The engine ECU can control engine torque in a slow way (for example, by setting the accelerator pedal position) or in a fast way (for example, by controlling the injection and ignition).

Also an idle speed control is included in the soft ECU.

Inports

The following table shows the inports.

Name	Unit	Description
Enable_n_Engine_Idle_Request	[0|1]	Switch to activate the engine idle speed request: ▪ 0: Off ▪ 1: On
Enable_Trq_Decrease_Fast_ESP	[0|1]	Signal from ESP ECU to enable or disable the fast torque decreasing request
Enable_Trq_Increase_Fast_ESP	[0|1]	Signal from ESP ECU to enable or disable the fast torque increasing request
Enable_Trq_Request_Extern	[0|1]	Signal to enable or disable the external torque request
Factor_Trq_Engine_Red	[%]	Requested engine torque reduction factor from the transmission control unit
Mode_StartButtonState	[1|2]	Mode of start button state: ▪ 1: Stimulus ▪ 2: Soft ECU
n_Engine	[rpm]	Engine speed
n_Engine_Idle_Request	[rpm]	Engine idle speed request
Pos_AccPedal_Driver	[%]	Driver accelerator pedal position
Reset	[]	Reset of states
State_StartButton	[-1|0|1|2]	Start button state in case of stimulus: ▪ -1: Power off

Name	Unit	Description
		▪ 0: Power on ▪ 1:Accessory on ▪ 2: Starter on
Sw_Crankshaft_Reset	[0|1]	Switch to enable/disable crankshaft reset: ▪ 0: Disable ▪ 1: Enable
Sw_Ignition	[0|1]	Ignition signal (terminal 15) ▪ 0: Off ▪ 1: On
Sw_StarterReg	[0|1]	Starter request signal (usually sent to engine ECU) ▪ 0: Off ▪ 1: On
Trq_Decrease_Request_Fast_ESP	[Nm]	The torque request from ESP ECU, fast decreasing
Trq_Driver_Des	[Nm]	Driver desired torque
Trq_Driver_Ind_Des	[Nm]	Desired indicated torque reqeusted by the driver
Trq_Engine_Ind_Max	[Nm]	Maximum available indicated engine torque
Trq_Increase_Request_Fast_ESP	[Nm]	The torque request from ESP ECU, fast increasing
Trq_Request_Extern	[Nm]	The external torque request

Outports

The following table shows the outports.

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Pos_AccPedal_Driver_Des	[%]	Driver desired accelerator pedal position
Pos_AccPedal_Ctrl	[%]	Accelerator pedal position influenced by the idle speed controller and the torque intervention
State_Engine	[0|1|2|3|4]	Engine state ▪ 0: Engine off ▪ 1: Ignition on ▪ 2: Ignition on and starter activated ▪ 3: Engine running ▪ 4: Ignition switched off, shutdown
Sw_Engine_Dynmaics_Fast	[0|1]	Flag, routed to engine basic, to increase the engine dynamic ▪ 0: Standard ▪ 1: Fast
Sw_Starter	[0|1]	Switch for starter activation (output from ECU) ▪ 0: Off ▪ 1: On

Blocks

- Idle Speed Control Engine Basic
- ESP Fast Torque Set
- Torque Intervention Basic
- Shift Torque Set
- Engine Operation Basic

Subsystems of the Environment Module

Where to go from here

Information in this section

Maneuver Submodule.....358
The **Maneuver** submodule controls the vehicle, road, and driver models to perform maneuvers.

Road Submodule.....361
The **Road** submodule provides information about the road the vehicle moves on.

Maneuver Submodule

Overview

Maneuvers are predefined sequences of driving instructions for simulating a variety of driving situations with the vehicle model. Typical maneuvers are braking on a road with different friction on two lanes (μ-split maneuver), performing a double-lane change, step steering, and the Fishhook maneuver. A maneuver also defines how to drive on road networks on predefined routes by specifying where to do lane changes, when to stop at junctions, etc. The submodule controls the vehicle, road, and driver models to perform maneuvers.

A maneuver is made up of sequences that comprise one ore more segments. Each segment has a segment end condition. When this condition is met, the next segment is executed. After the last segment of a sequence is simulated, either

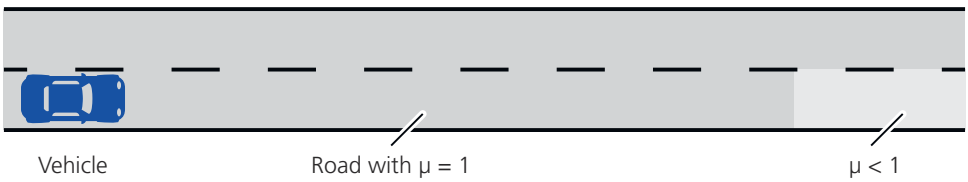
the start condition of the next sequence is evaluated or the maneuver terminates if all sequences are finished.

The core of the submodule is the MANEUVER_SCHEDULER block. Refer to Maneuver Scheduler (ASM Environment Reference 🕮).

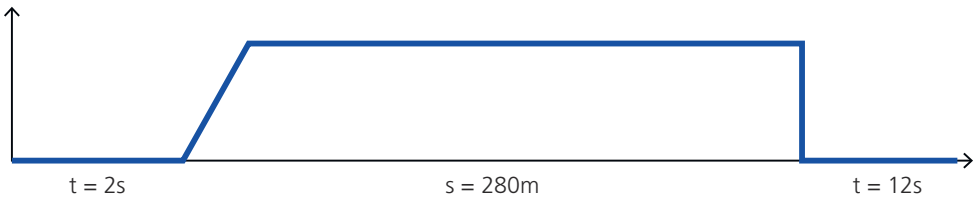
Example

The μ-split maneuver is described as an aid to understanding maneuvers in general.

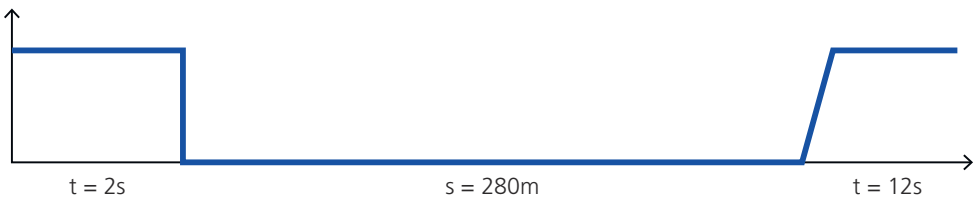
Maneuver segment	**0**	**1**	**2**
Description	Init and standstill	Accelerate to a certain velocity, drive until low mue spot	Brake



Reference velocity



Brake pedal



A route on a straight road with a low μ value on one lane after 280 m is assumed for this maneuver. The aim is to accelerate the vehicle to a certain velocity, drive the vehicle until the μ spot is reached, and initialize braking. This task is split into sequential steps that the driver and vehicle models have to perform.

- The maneuver starts with an initialization phase. The initial vehicle position is set to 0 m. After two seconds, the initialization phase ends.
- In the first segment, the vehicle drives to the low μ area. A segment with a length of 280 m is specified. The steering wheel is controlled by the driver model, which keeps the vehicle on the road. A reference velocity profile is defined so that a final velocity is reached with reasonable acceleration. The pedals and gears are controlled by the driver model.
- At the start of the low μ area, the driver model initiates the braking process. It holds the last steering wheel angle and opens the clutch.

Maneuver types

There are two maneuver types.

- In *stimulus* maneuvers, the vehicle model is controlled by stimulus signals for accelerator pedal, brake pedal, clutch pedal, gear, and steering wheel.
- In *controlled* maneuvers, the driver model controls the vehicle model. Accelerator, brake and clutch pedal positions, and gear are controlled by the driver model's longitudinal controller. The steering wheel is controlled by the lateral controller.

Combinations of both maneuver types are possible in the maneuver segments.

Mode signals

The maneuver uses mode signals to control the vehicle, driver, and road models. These enable model components which are used in the current maneuver segment.

The accelerator and brake pedal have *stimulus* and *driver* modes. In *stimulus* mode, the reference signals for both pedals in the vehicle model are provided by the maneuver. In *driver* mode, the driver controls accelerator and brake pedal.

The clutch pedal and gear have *stimulus*, *driver*, and *open clutch* modes. In *stimulus* mode, the reference signals for the vehicle model are provided by the maneuver. In *driver* mode, the driver controls clutch and gear. In *open clutch* mode, the clutch is opened by the maneuver, the current gear is held.

The steering wheel has *stimulus*, *driver*, and *fixed steering* modes. In *stimulus* mode the maneuver provides a stimulus signal for the steering wheel. In *driver* mode the driver controls the steering wheel. In *fixed steering* mode, the current steering wheel angle is held.

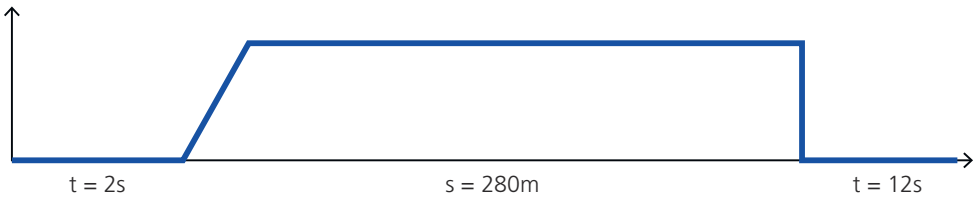
The road model has *no road*, *straight*, *circle*, and *road* modes. In *no road* mode, the vehicle can perform stimulus maneuvers on a flat area. The *straight* and *circle* modes are for straight and circular driving on a flat area. In *road* mode, the road information is provided by the road model.

Example

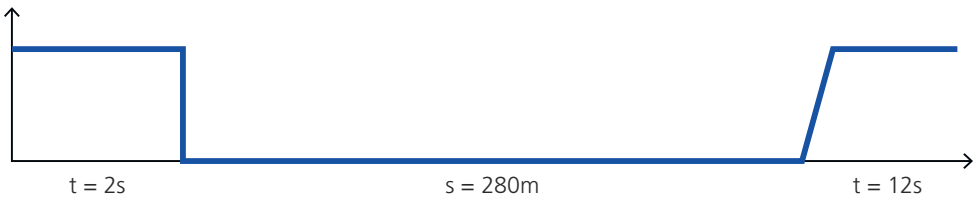
The following illustration shows the use of mode signals in the μ-split maneuver.

Maneuver segment	0	1	2
Description	Init and standstill	Accelerate unit a certain veocity, drive until low mue spot	Brake
Steering mode	Stimulus	Driver	Fix steering
AccBr mode	Stimulus	Driver	Stimulus
Gear clutch mode	Stimulus	Driver	Open Clutch

Reference velocity



Brake pedal



Maneuver definition

Maneuvers are defined with the **Scenario Editor** in ModelDesk. Refer to Basics of the Scenario Editor (ModelDesk Scenario Creation 🕮). The maneuver definition that is used in the model is stored in a vector. This vector contains data on the steps that the model has to carry out sequentially as well as the table reference data for the pedals, gears, steering wheel, and vehicle velocity.

Related topics

Basics

Basics of the Scenario Editor (ModelDesk Scenario Creation 🕮)

Road Submodule

Overview

The **Road** submodule provides information about the road the vehicle moves on. This information is calculated in the road model and forwarded to both the vehicle and driver models. Different road modes are selected by the maneuver according to the maneuver definition. Roads can be built up using the **Road Generator** in ModelDesk.

Subsystems

- Ambient (ASM Environment Reference 🕮)
- Basic Road (ASM Environment Reference 🕮)
- GPS Position (ASM Environment Reference 🕮)
- Lane Network (ASM Environment Reference 🕮)
- Lane Sensor Dynamic Parameters (ASM Environment Reference 🕮)
- Lane Sensor Dynamic Output (ASM Environment Reference 🕮)
- Road (ASM Environment Reference 🕮)

Related topics

Basics

Introduction to the Road Generator (ModelDesk Road Creation 🕮)

Subsystems of the Vehicle Dynamics Module

Where to go from here

Information in this section

Brake Subsystem.....363
The **Brake** subsystem simulates the transformation from brake pedal to brake torque at each wheel.

Suspension Subsystem.....364
The **Suspension** block calculates the suspension kinematics and the suspension forces in the spring, shock absorber, and the stabilizer.

Front Suspension Subsystem.....365
The **SuspensionFront** subsystem calculates the front suspension kinematics, compliance kinematics, and the suspension forces in the spring, damper, and the stabilizer.

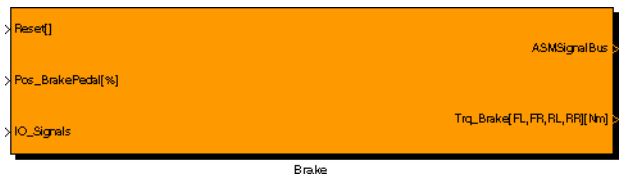
Rear Suspension Subsystem.....367
The **SuspensionRear** subsystem calculates the rear suspension kinematics, compliance kinematics, and the suspension forces in the spring, shock absorber, and the stabilizer.

Vehicle Movement Subsystem 369
In the **VehicleMovement** block, the motion of the vehicle and the vertical wheel movements are calculated as functions of tires, aerodynamics, suspension, and mass forces and torques.

Brake Subsystem

Description

This block simulates the transformation from brake pedal to brake torque at each wheel.



Inports

The following table shows the inports:

Name	Unit	Description
IO_Signals	[]	Signals from ECU to control the brake hydraulic circuit if present
Pos_BrakePedal	[%]	Brake pedal position, defined by the driver
Reset	[]	Resets all integrators in the block to their initial conditions.

Outports

The following table shows the outports:

Name	Unit	Description
Trq_Brake[FL,FR,RL,RR]	[Nm]	Brake torques on the wheels

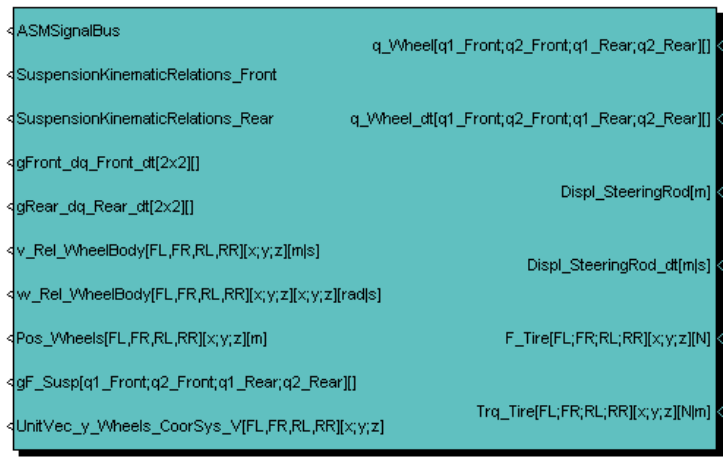
Blocks

- **Brake Hydraulics**, refer to ASM Brake Hydraulics Reference 🕮.
- Brake Booster on page 112
- Manifold Pressure on page 121
- Brake Disc on page 117
- Drum Brake on page 119
- Drum Brake 2-D on page 120

Suspension Subsystem

Description

The **Suspension** block calculates the suspension kinematics and the suspension forces in the spring, shock absorber, and the stabilizer.



Suspension kinematics means the relative position of the wheel center, the orientation of the wheel and the spring-damper-stabilizer movements of two generalized coordinates (for example, vertical wheel movements of left (q1) and right (q2) sides) and steering rod displacement for both front and rear suspension. The movements of the spring, damper, and stabilizer are calculated to determine the forces that they exert.

The compliance kinematics describe the limited movements of the wheel due to suspension elasticity as a function of the tire forces and torques.

The kinematics are configured via either look-up tables or geometrical linkage points. The compliances are configured via look-up tables. Kinematics and compliance look-up tables are generated by K&C Tester (Kinematics and Compliances Tester) or by a suspension analysis program like ADAMS/Car Suspension Kit. Geometrical linkage points are contained in suspension design data.

The suspension is described in greater detail below. For details on the coordinate systems and the sign conventions, refer to Coordinate Systems (ASM Vehicle Dynamics Addendum 🕮) and Sign Conventions (ASM Vehicle Dynamics Addendum 🕮).

Inports

The following table shows the inports:

Name	Unit	Description
Displ_SteeringRod	[m]	Steering rod displacement
Displ_SteeringRod_dt	[m/s]	Speed of steering rod displacement
F_Tire[FL;FR;RL;RR][x;y;z]	[N]	Tire forces at wheel center, used for the calculation of suspension compliance displacements

Name	Unit	Description
q_Wheel[q1_Front;q2_Front;q1_Rear;q2_Rear]	[]	Generalized coordinates for front and rear suspension
q_Wheel_dt[q1_Front;q2_Front;q1_Rear;q2_Rear]	[]	Derivative with respect to time of generalized coordinates for front and rear suspension
Trq_Tire[FL;RF;RL;RR][x;y;z]	[N/m]	Tire torques at wheel center, used for the calculation of suspension compliance displacement

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
gF_Front_Susp_dq_Front_dt[2x2]	[]	Derivative of generalized forces with respect to generalized speed of front suspension
gF_Rear_Susp_dq_Rear_dt[2x2]	[]	Derivative of generalized forces with respect to generalized speed of rear suspension
gF_Susp[q1_Front;q2_Front;q1_Rear;q2_Rear]	[]	Generalized forces of front and rear suspension
Pos_Wheels[FL,FR,RL,RR][x;y;z]	[m]	Wheel relative position in vehicle coordinate system
SuspensionKinematicRelations_Front	[]	Collection of suspension kinematics derivative signals of front suspension
SuspensionKinematicRelations_Rear	[]	Collection of suspension kinematics derivative signals of rear suspension
UnitVec_y_Wheels_CoorSys_V[FL,FR,RL,RR][x;y;z]	[]	Unit vector in y direction of wheel coordinate system
v_Rel_WheelBody[FL,FR,RL,RR][x;y;z]	[m/s]	Velocity of wheel body relative to vehicle body
w_Rel_WheelBody[FL,FR,RL,RR][x;y;z]	[rad/s]	Angular velocity of wheel body relative to vehicle body, i.e. excluding wheel rotational speed

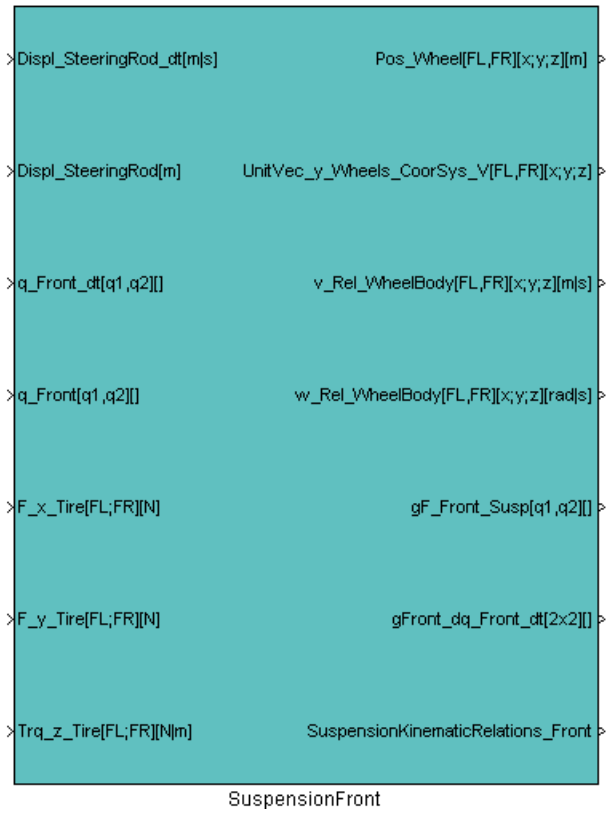
Blocks

- Rotation Switch on page 130
- Front Suspension Subsystem on page 365
- Rear Suspension Subsystem on page 367
- Subframe on page 131
- Displacement Sensor on page 129

Front Suspension Subsystem

Description

The SuspensionFront subsystem calculates the front suspension kinematics, compliance kinematics, and the suspension forces in the spring, damper, and the stabilizer.



Suspension kinematics is calculated by symmetric or asymmetric look-up table or by geometrical description (MacPherson Strut). Each suspension kinematics system can be replaced in the model with the relevant system at the ASM Vehicle Dynamics Library. For further information on how to replace the suspension kinematics system, refer to How to Change the Suspension Kinematics Model (ASM Vehicle Dynamics Model Description 🕮). The suspension compliances subsystem calculates the elastic displacements and orientations as functions of tire forces and torques.

Blocks

There are five variants to calculate the suspension kinematics of the front suspension: by symmetric or asymmetric look-up tables (as a function of two or three variables) or by geometrical description (MacPherson strut). Each suspension kinematics system calculates wheel midpoint, orientation of wheel and displacements of spring, damper and stabilizer.

- Front Suspension Kinematics (Symmetric) on page 174
- Front Suspension Kinematics (Symmetric, 3DOF) on page 183
- Suspension Kinematics Rigid Axle (Symmetric) on page 194
- Suspension Kinematics Rigid Truck Axle on page 189
- Front Suspension Kinematics (Asymmetric) on page 163
- Front Suspension Kinematics (Asymmetric, 3DOF) on page 169
- Front Suspension Kinematics MacPherson Strut on page 179
- Front Suspension Kinematics Ancillary Spring on page 199

- Front Suspension Compliance on page 145
- Suspension Compliance Rigid Axle (Symmetric) on page 148
- Suspension Compliance 2-D (Symmetric) on page 155
- Stabilizer Compliance (Symmetric) on page 160
- Front Suspension Forces on page 138
- Front Suspension Forces Active on page 141
- Front Suspension Forces Ancillary Spring on page 143
- Suspension Force Kinematics Front on page 152
- Relative Position Front on page 137
- Front Wheels Relative Velocities on page 135

Related topics

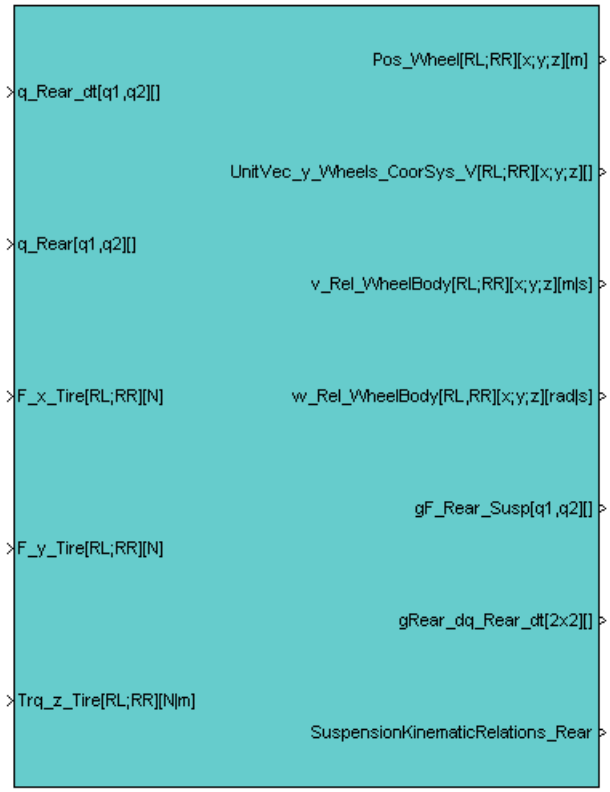
HowTos

How to Change the Suspension Kinematics Model (ASM Vehicle Dynamics Model Description 🕮)

Rear Suspension Subsystem

Description

The **SuspensionRear** subsystem calculates the rear suspension kinematics, compliance kinematics, and the suspension forces in the spring, shock absorber, and the stabilizer.



SuspensionRear

Suspension kinematics is calculated by symmetric or asymmetric look-up table or by geometrical description (semi-trailing arm or rigid axle). Each suspension kinematics system can be replaced at the model with the relevant system at the ASM Vehicle Dynamics Library. For further information on how to replace suspension kinematics system, refer to How to Change the Suspension Kinematics Model (ASM Vehicle Dynamics Model Description 🕮). The suspension compliances subsystem calculates the elastic displacements and orientations as functions of tire forces and torques.

Blocks

There are six variants to calculate the suspension kinematics of the front suspension: by symmetric or asymmetric look-up tables (as functions of two or three variables) or by geometrical description (semi-trailing arm or rigid axle). Each suspension kinematics system calculates wheel midpoint, orientation of wheel and displacements of spring, damper and stabilizer.

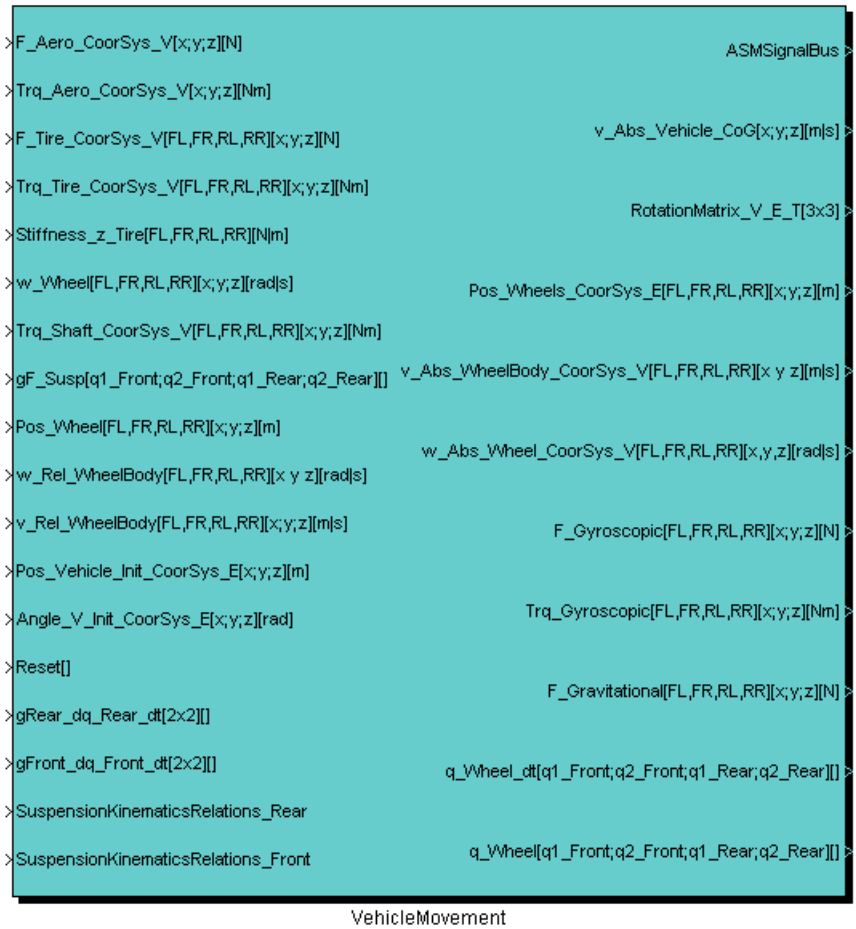
- Rear Suspension Kinematics (Symmetric) on page 209
- Rear Suspension Kinematics (Symmetric, 3DOF) on page 222
- Rear Suspension Kinematics (Asymmetric) on page 203
- Rear Suspension Kinematics (Asymmetric, 3DOF) on page 222
- Rear Suspension Kinematics Semi-Trailing Arm on page 213
- Rear Suspension Kinematics Rigid Axle on page 218
- Rear Suspension Kinematics Ancillary Spring on page 225
- Rear Suspension Compliance on page 233

- Rear Suspension Forces on page 230
- Rear Suspension Forces Active on page 236
- Rear Suspension Forces Ancillary Spring on page 236
- Suspension Force Kinematics Rear on page 222
- Relative Position Rear on page 229
- Rear Wheel Relative Velocities on page 227

Vehicle Movement Subsystem

Overview

In the **VehicleMovement** block, the motion of the vehicle and the vertical wheel movements are calculated as functions of tires, aerodynamics, suspension, and mass forces and torques. ASM Vehicle Dynamics uses a multibody system (MBS) technique to write the equation of motion. In this subsystem the equation of motion is solved for the vehicle degrees of freedom. These degrees of freedom are used to calculate the vehicle and wheel velocities and also to calculate the vehicle and wheel positions in the earth reference system. For details of the coordinate systems, refer to Coordinate Systems (ASM Vehicle Dynamics Addendum 🕮).



Inports

The following table shows the inports:

Name	Unit	Description
Angle_Vehicle_Init_CoorSys_E[x;y;z]	[rad]	Initial vehicle orientation angles
F_Aero_CoorSys_V[x,y,z]	[N]	Aerodynamic forces represented in vehicle reference coordinate system
F_Susp[FL,FR,RL,RR]	[N]	Forces from suspension system, i.e., spring, damper and stabilizer
F_Tire_CoorSys_V[FL,FR,RL,RR][x;y;z]	[N]	Tire forces vectors represented in vehicle reference coordinate system
gFront_dq_Front_dt[2x2]	[]	Derivatives of the generalized forces with respect to the generalized speeds for front suspension
gRear_dq_Rear_dt[2x2]	[]	Derivatives of the generalized forces with respect to the generalized speeds for rear suspension
Pos_Vehicle_Init_CoorSys_E[x;y;z]	[m]	Initial vehicle position
Pos_Wheel[FL,FR,RL,RR][x;y;z]	[m]	Wheel relative position in vehicle coordinate system

Name	Unit	Description
Reset	[]	Resets all integrators in the block to their initial conditions
Stiffness_z_Tire[FL,FR,RL,RR]	[N/m]	Vertical tire stiffness
SuspensionKinematicRelations_Front	[]	Collection of suspension kinematics derivative signals of front suspension
SuspensionKinematicRelations_Rear	[]	Collection of suspension kinematics derivative signals of rear suspension
Trq_Aero_CoorSys_V[x,y,z]	[Nm]	Aerodynamic torques represented in vehicle reference coordinate system
Trq_Shaft_CoorSys_V[FL,FR,RL,RR][x;y;z]	[Nm]	Driving torque on the wheel in vehicle reference coordinate system
Trq_Tire_CoorSys_V[FL,FR,RL,RR][x;y;z]	[Nm]	Tire forces vectors represented in vehicle reference coordinate system
v_Rel_WheelBody[FL,FR,RL,RR][x,y,z]	[m/s]	Velocity of the wheel body relative to vehicle body
w_Rel_WheelBody[FL,FR,RL,RR][x y z]	[rad/s]	Angular velocity of the wheel body relative to vehicle body, i.e., excluding wheel rotational speed
w_Wheel_CoorSys_V[FL,FR,RL,RR][x;y;z]	[rad]	Wheel rotation speed in vehicle reference coordinate system

Outports

The following table shows the outports:

Name	Unit	Description
F_Gravitational[FL,FR,RL,RR][x,y,z]	[N]	Gravitational forces of the wheel
F_Gyroscopic[FL,FR,RL,RR][x,y,z]	[N]	Gyroscopic forces of the wheel
Pos_Wheels_CoorSys_E[FL,FR,RL,RR][x;y;z]	[m]	Wheel position in the earth coordinate system
q_Wheel[q1_Front;q2_Front;q1_Rear;q2_Rear]	[]	The wheel generalized coordinates
q_Wheel_dt[q1_Front;q2_Front;q1_Rear;q2_Rear]	[]	The wheel generalized speeds
RotationMatrix_V_E_T[3x3]	[]	Rotation matrix from earth to vehicle coordinate system
Trq_Gyroscopic[FL,FR,RL,RR][x,y,z]	[Nm]	Gyroscopic torques of the wheel
v_Abs_Vehicle_CoG[x,y,z]	[m/s]	Absolute velocity of the vehicle center of gravity
v_Abs_WheelBody_CoorSys_V[FL,FR,RL,RR][x y z]	[m/s]	Absolute velocity of the wheel in vehicle coordinate system
w_Abs_Wheel_CoorSys_V[FL,FR,RL,RR][x,y,z]	[rad/s]	Absolute angular velocity of the wheel in vehicle coordinate system

Blocks

- Common Vehicle Parameters on page 124
- Coordinate Transformation on page 275
- Vehicle Motion on page 287
- Vehicle and Wheel Velocities on page 283
- Vehicle Mass and Additional Loads on page 284
- Sensor Motion on page 281
- Mass Forces and Torques on page 279

Blocks from Former Versions

Introduction

The following topics provide information on blocks that were used in former library versions.

Where to go from here

Information in this section

Engine Basic (Version 2.0 or Earlier)..... 374
This is a former version of the engine basic model.

Engine Basic (Version 3.0)..... 374
This is a former version of the ENGINE_BASIC block.

ESP Interventions Fast (Version 3.0)..... 376
This is a former version of the ESP_TORQUE_INTERVENTION_FAST block.

ESP Interventions Slow (Version 3.0)..... 377
This is a former version of the ESP_TORQUE_INTERVENTION_SLOW block.

Soft ECU Transmission (Version 8.0 or Earlier)..... 378
This is a former version of the SOFT_ECU_TRANSMISSION block.

Soft ECU Transmission (Version 9.0 and 10.0)..... 382
This is a former version of the SOFT_ECU_TRANSMISSION block.

Test Bench (Version 1.0)..... 389
This is a former version of the TEST_BENCH block.

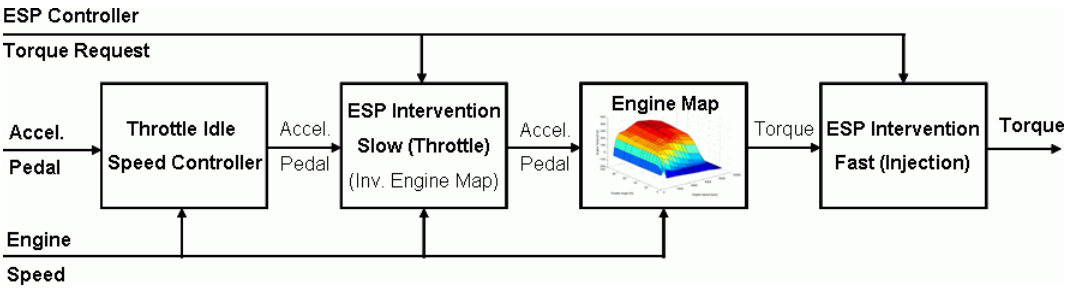
Throttle (Version 2.0)..... 390
This is a former version of the THROTTLE block.

Engine Basic (Version 2.0 or Earlier)

Description

Connecting the model with an electronic stability program (ESP) results in torque requests from the ECU. These torque requests are implemented in this block.

The following illustration shows a schematic diagram of the main component in the engine subsystem. As shown in the illustration, the throttle block works as an idle speed controller. The modified accelerator pedal position is influenced by the ESP slow torque request and the resulting accelerator pedal position. It is then passed to the engine block to calculate the engine torque. The resulting torque is in turn influenced by the ESP fast torque requests.



For detailed descriptions of the subsystems, refer to the following sections.

Subsystems

- Throttle (Version 2.0) on page 390
- Engine Basic (Version 3.0) on page 374
- ESP Interventions Slow (Version 3.0) on page 377
- ESP Interventions Fast (Version 3.0) on page 376

Related topics

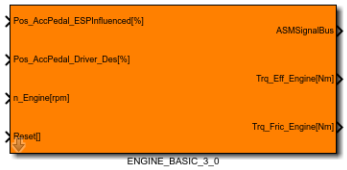
References

Engine.....73

Engine Basic (Version 3.0)

Description

As the focus of ASM Vehicle Dynamics is on vehicle dynamics, the engine modeling details, i.e., the combustion process, are not included in this engine model. Instead, there is a 2-D look-up table with engine speed and accelerator pedal position as inputs and the engine torque as an output. Another 1-D look-up table is used to get the engine friction torque as a function of engine speed. The engine dynamics is approached by a first-order delay element.



Inports

The following table shows the inports:

Name	Unit	Description
n_Engine	[rpm]	Engine speed
Pos_AccPedal_Driver_Des	[%]	Driver desired accelerator pedal position without ESP intervention
Pos_AccPedal_ESPInfluenced	[%]	Accelerator pedal position including the ESP torque influences
Reset	[]	Resets all integrators in the block to their initial conditions

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_Eff_Engine	[Nm]	Mean effective engine torque
Trq_Fric_Engine	[Nm]	Engine friction torque

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_TimeConst	[s]	Engine torque time constant
Map_FrictionTorque	[Nm]	Engine friction torque = f(Engine Speed[rpm])
Map_Torque	[Nm]	Engine torque = f(Engine Speed[rpm], Pos_AccPedal[%])

Related topics

References

Engine Basic (Version 3.0) (ModelDesk Parameterizing 🕮)

ESP Interventions Fast (Version 3.0)

Description

When the Vehicle Dynamics model is connected to an ECU for vehicle dynamics control, the ECU sends torque requests to the engine ECU. These requests depend on the dynamic states of the vehicle, which require the engine torque to be decreased or increased. The engine ECU can control engine torque in two ways: by setting the accelerator pedal position or by controlling the injection. Controlling the injection is implemented in this subsystem.

Injection control is not possible in the engine model because it does not simulate the functionality of the engine in detail. Therefore, if the Enable flag is set, the desired torque is applied directly instead of engine torque from the Engine Basic block. The transition between the requested and the calculated engine torque is carried out via a first-order delay.

Inports

The following table shows the inports:

Name	Unit	Description
Enable_TrqDecrease_Fast	[]	Signal from ECU to enable or disable the fast torque decreasing request
Enable_TrqIncrease	[]	Signal from ECU to enable or disable the torque increasing request
Reset	[]	Resets all integrators in the block to their initial conditions
Trq_DecreaseRequest_Fast	[Nm]	The torque request from ECU, decreasing
Trq_Eff_Engine	[Nm]	Engine mean effective torque
Trq_Fric_Engine	[Nm]	Engine friction torque
Trq_IncreaseRequest	[Nm]	The torque request from ECU, increasing

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_ESPInfluenced	[Nm]	Engine mean effective torque influenced by the ESP requests

Parameters

With the above inputs and outputs, this subsystem requires no parameters to perform its calculation.

Related topics

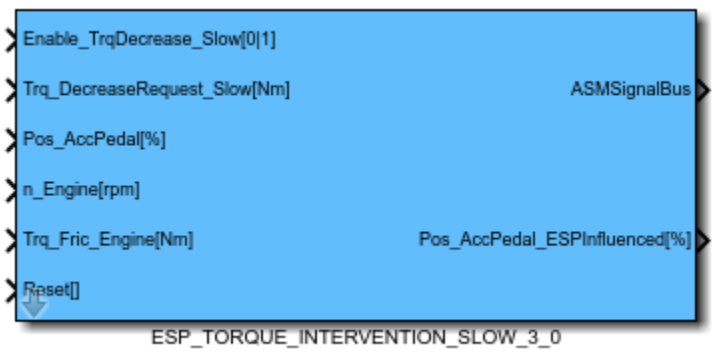
References

Engine Basic (Version 3.0)..... 374

ESP Interventions Slow (Version 3.0)

Description

When the Vehicle Dynamics model is connected to an ECU for vehicle dynamics control, the ECU sends torque requests to the engine ECU. These requests depend on the dynamic states of the vehicle, which require the engine torque to be decreased or increased. The engine ECU can control engine torque in two ways: by setting the accelerator pedal position or by controlling the injection. Controlling the accelerator pedal is implemented in this subsystem.



To get a certain torque from the engine, the accelerator pedal position must be calculated at the current engine speed.

This value is passed on instead of the value defined by the driver only if the Enable_TrqDecrease_Slow input is enabled. The accelerator pedal position is calculated as a function of the engine torque and engine speed by inverting the look-up table for engine torque. This inversion can be done by the parameterization tool, and you do not have to define it yourself.

Inports

The following table shows the inports:

Name	Unit	Description
Enable_TrqDecrease_Slow	[]	Signal from ECU to enable or disable the slow torque reduction request
n_Engine	[rpm]	Engine speed

Name	Unit	Description
Pos_AccPedal	[%]	Accelerator pedal position influenced by the idle speed control
Reset	[]	Resets all integrators in the block to their initial conditions
Trq_DecreaseRequest_Slow	[Nm]	The torque request from ECU
Trq_Fric_Engine	[Nm]	Engine friction torque

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Pos_AccPedal_ESPInfluenced	[%]	Accelerator pedal position influenced by the ESP request

Parameters

The following table shows the parameters:

Name	Unit	Description
Map_Trq_Engine_Inv	[%]	Inverse Engine Map Pos_AccPedal = f(Engine Speed [rpm], Engine torque [Nm])

Related topics

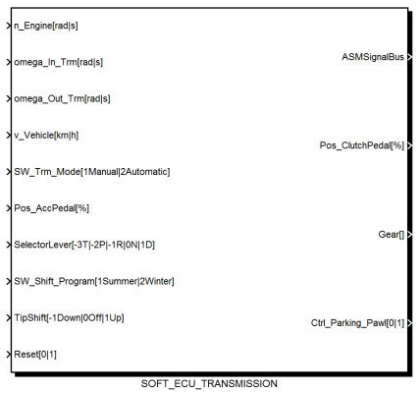
References

Engine Basic (Version 3.0) (ModelDesk Parameterizing 🕮)

Soft ECU Transmission (Version 8.0 or Earlier)

Description

The SOFT_ECU_TRANSMISSION subsystem controls the automatic gearbox. It engages the gears and controls the lockup clutch according to lever position.

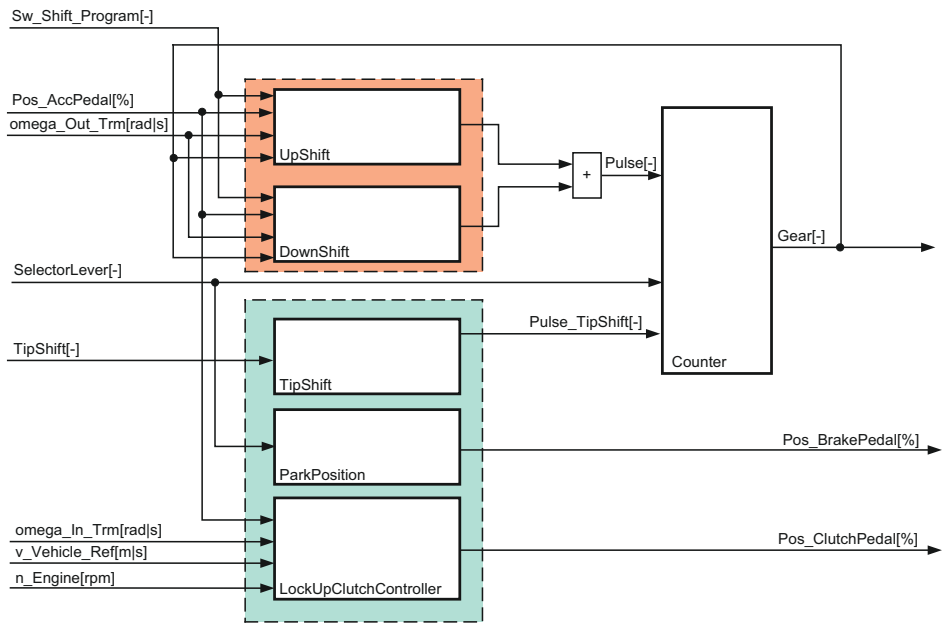


The automatic gearbox lever has five positions:

Position	Description
P - Park	Disengages the clutch and brakes the vehicle for parking position
R - Reverse	Engages the reverse gear
N - Neutral	Engages the neutral gear
D - Drive	Engages the first gear and shifts up or down depending on the transmission output speed and on the accelerator pedal position
T - TipShift	Manual shifting with automatic transmission

In the drive position, the soft ECU engages the first gear and shifts automatically according to the transmission output speed and the accelerator pedal position. The lockup clutch is also controlled according to vehicle speed and to the accelerator pedal position. This makes it possible to drive the vehicle without the user shifting gear.

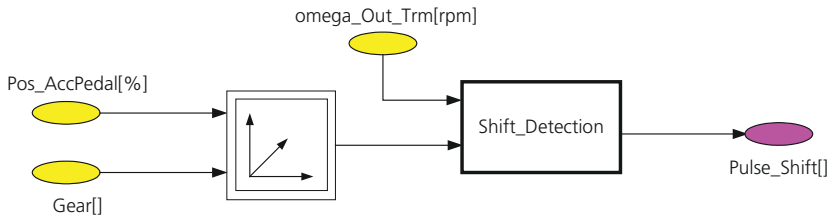
The following illustration gives a basic overview of the SOFT_ECU_TRANSMISSION subsystem:



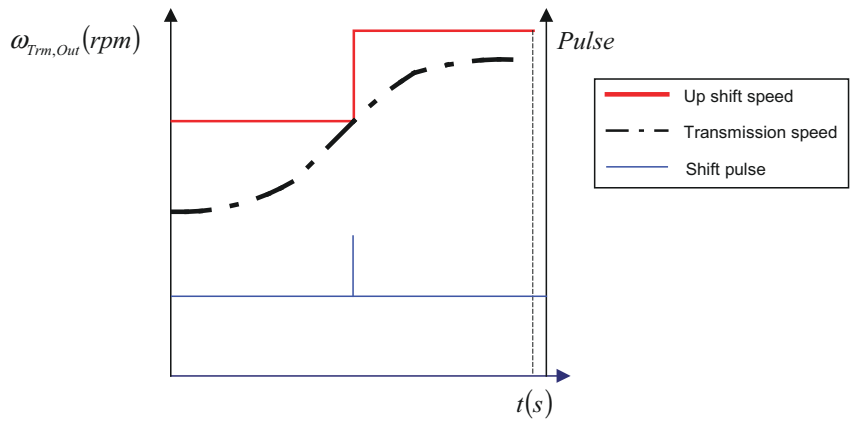
The model is divided into three sections. The light red section contains all the functions which shift a gear up or down. The cyan subsystem contains functions for particular driving situations. Finally, the counter section is the gearshifter itself, which generates the gear for the transmission and the pedal positions. The mode switch section enables the correct signals for the current driving situation. The following three paragraphs describe the different parts of the SOFT_ECU_TRANSMISSION subsystem.

Shifting Section (light red) Generally speaking, the shifting functionality compares the current transmission output speed with an up shift or down shift speed. If the transmission output speed crosses the up shift or down shift speed for the current driving situation, a pulse is generated. Inside the gearshifter, this

pulse is used to engage the clutch, to decrease the accelerator pedal, and to increase or decrease the current gear position.



The following illustration shows a typical up shift functionality:



If the transmission output speed crosses the up shift signal, a shift pulse is generated. This forces the transmission to shift up a gear.

Particular Driving Situations (cyan) The models for particular driving situations are used to handle additional functionalities inside the SOFT_ECU_TRANSMISSION subsystem.

- The `TipShift` function is used to handle manual gearshifting with the automatic transmission.
- The `ParkPosition` function is used to engage the brakes if the selector lever is in parking position.
- The `LockUpClutch Controller` function is used to control the lockup clutch of the torque converter according to the current driving situation.

Counter The counter engages the gear according to the current selector lever position or to the manually selected gear if the `TipShift` function is used.

Inports

The following table shows the inports:

Name	Unit	Description
n_Engine	[rad/s]	Engine speed
omega_In_Trm	[rad/s]	Transmission input speed
omega_Out_Trm	[rad/s]	Transmission output speed
Pos_AccPedal	[%]	Accelerator pedal position
Reset	[0|1]	Reset of all integrators and internal states

Name	Unit	Description
Selector_Lever	[-3T|-2P|-1R|0N|1D]	Position of the gear lever for automatic transmission
SW_Shift_Program	[1|2]	Switch for different shift programs of the soft ECU transmission ▪ 1: Summer ▪ 2: Winter
SW_Trm_Mode	[1|2]	Switch for the transmission mode. Activates the SOFT_ECU_TRANSMISSION subsystem if automatic transmission is used. ▪ 1: Manual ▪ 2: Automatic
TipShift	[-1|0|1]	Input for TipShift function of the soft ECU transmission
v_Vehicle	[km/h]	Vehicle speed

Outports

The following table shows the outports:

Name	Unit	Description
Ctrl_Parking_Pawl	[0|1]	Control signal of the parking pawl of automatic transmission
Gear	[]	Gear selected for automatic transmission
Pos_Clutch_Pedal	[%]	Position of the lockup clutch

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_I_LockupClutch	[]	I gain for lockup clutch controller
Const_P_LockupClutch	[]	P -gain for lockup clutch controller
Map_ClutchActive	[]	Table for lockup clutch state = f(Vehice Speed)
Map_LockupClutch_Set	[]	Table for setpoint of lockup clutch slip = f(Vehicle Speed, Pos_AccPedal)
Map_omega_Down shift	[rpm]	Table with transmission speed for gear down shift, [rpm] = f(Pos_AccPed, Gear)
Map_omega_Up shift	[rpm]	Table with transmission speed for gear up shift, [rpm] = f(Pos_AccPed, Gear)
StepSize	[s]	Step size of model

Related topics

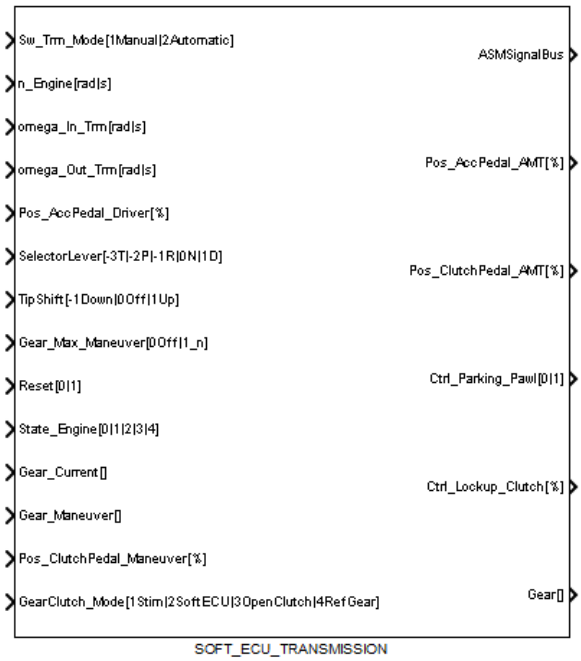
References

Soft ECU AT Basic V7 (ModelDesk Parameterizing 🕮)
Soft ECU AT V8 (ModelDesk Parameterizing 🕮)

Soft ECU Transmission (Version 9.0 and 10.0)

Description

The block controls the gearshift process of an automatic transmission. The shift process can be initiated by using the selector lever or the TipShift. It also offers a control strategy for the lockup clutch of the torque converter. For an automated manual transmission (AMT), the clutch is also controlled while the gear is being shifted.



The driver can request gear engagement by controlling the selector lever, which has the following positions:

Position	Description
P	Park: Activates the parking pawl
R	Reverse: Engages the reverse gear
N	Neutral: Engages the neutral gear
D	Drive: Engages the first gear and afterwards shifts up or down automatically according to the driving situation
T	Tip shift: Activates the tip shift (TipTronic) function. The gears can then be shifted according to the driver's demand

The gearshift process and the clutch control, both lockup and AMT clutch, can also be influenced by stimuli and external requests. The gear can be influenced by the Gear_Maneuver inport. The clutch is influenced by the Pos_ClutchPedal_Maneuver inport.

Therefore, the block performs different simulation scenarios according to the GearClutch_Mode inport.

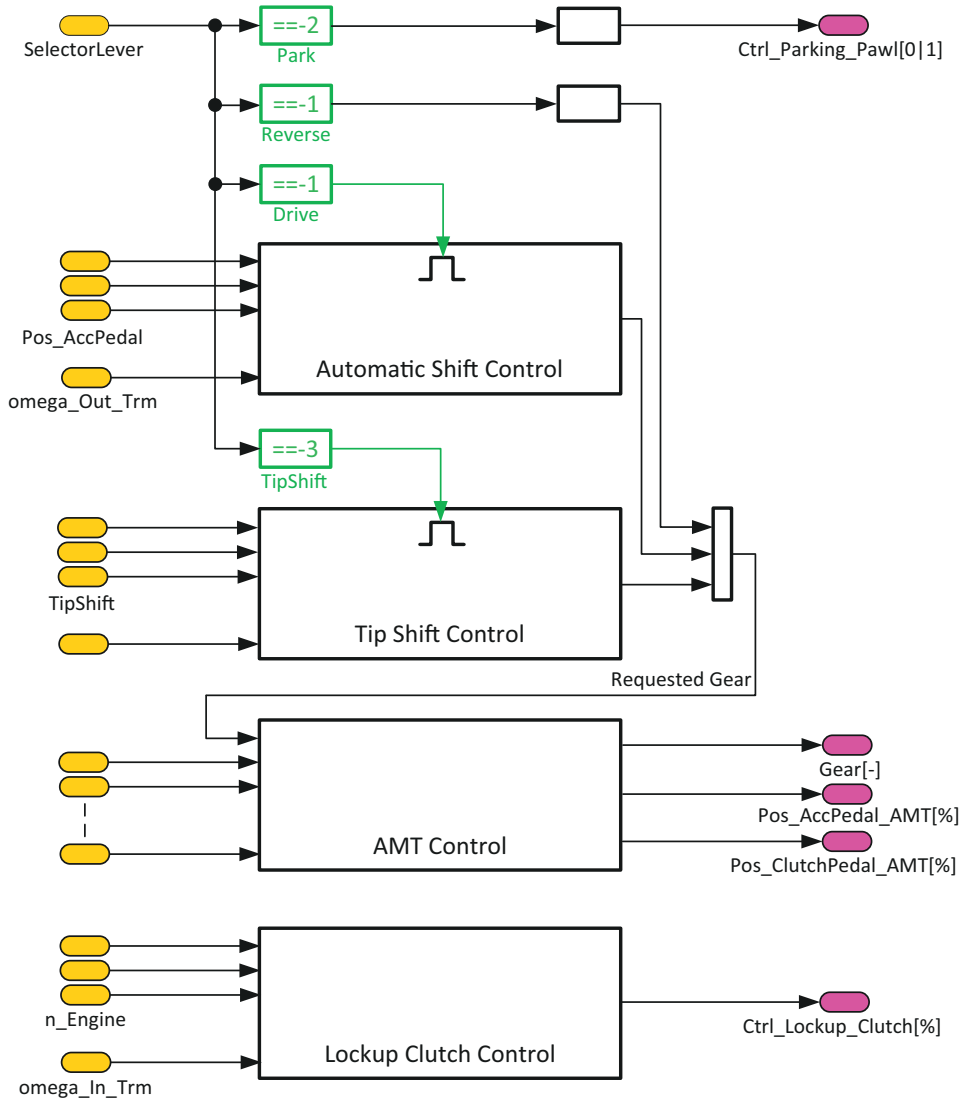
The following table illustrates the possible simulation scenarios performed by the block:

GearClutch_Mode	**Gear**	**Lockup Clutch (Sw_AMT = 0) or AMT Clutch (Sw_AMT = 1)**
1 Stim: Stimulus mode	Gear_Maneuver	Pos_ClutchPedal_Maneuver
2 SoftECU: Automatic control mode	Controlled by the soft ECU	Controlled by the soft ECU
3 OpenClutch: Open clutch mode	Controlled by the soft ECU	Pos_ClutchPedal_Maneuver (if simulated with Maneuver Scheduler, the clutch is opened gradually)
4 RefGear: Reference gear mode	Controlled according to Gear_Maneuver, which is used as a set value	Controlled by the soft ECU

Note

Both stimulus and external request features are deactivated in the standard demos. By default, the demos are configured to mode 2.

The Soft ECU Transmission model is divided into several subsystems according to specific functions, see the following figure:



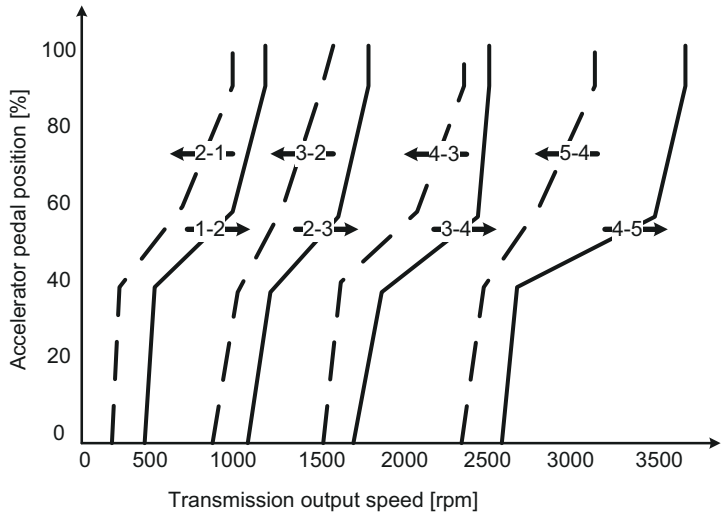
The following sections describe the control strategies of the Soft ECU Transmission.

Automatic shift control

The subsystem is active if the driver shifts the selector lever to the D position. The gear is then shifted up and down automatically according to the accelerator pedal position and the transmission output speed.

Two shift patterns for up shift and down shift are parameterized inside the block. Each shift pattern calculates the shift thresholds of the transmission output speed for several possible gears. These thresholds are then compared with the actual transmission output speed to request an up shift or down shift.

The following illustration displays these shift patterns:



To illustrate the shift strategy, up shift 1-2 and down shift 2-1 are clarified in the following:

As the driver shifts the selector lever to the D position, the first gear is requested and engaged at the same time. Meanwhile, the transmission output speed is being increased to some point between curves 2-1 and 1-2. If the driver demands more acceleration by pressing the accelerator pedal, the transmission output speed will be increased accordingly and curve 1-2 will be crossed. After curve1-2 is crossed, the second gear is requested.

Down shift 2-1 can be understood in a similar way. As the driver decelerates, the transmission output speed is decreased to some point, where curve 2-1 is crossed and the first gear is requested.

To prevent the phenomenon of gear hunting, i.e., frequent up shifting and down shifting, a minimum time between two successive shifts is considered (**Const_t_Min_Shift[s]** parameter). In this way and after a successful gearshift, another shift request within the mentioned time is disabled. If the user sets a sufficient time, gear skipping can be reached. However, this strongly depends on the shift patterns and the driving situation.

The gear request can be reset at any time by using the reset of states. Moreover, this is automatically done if the engine was stalled.

Tip shift control

The subsystem is active if the driver shifts the selector lever to the T position. Afterwards, the driver can take the control to shift the gear up and down.

If the driver shifts the selector lever to the T position, the tip shift signal will be observed.

This signal has three values:

- -1: Down shift
- 0: Off
- 1: Up shift

Each time the driver tips the tip shift up or down, a gear up shift or down shift is requested, respectively.

However, two conditions have to be fulfilled:

- The lowest possible gear is -1.
- A gearshift request can be considered only if the previously requested gear was successfully engaged.

Automated manual transmission control

The functionality of an automated manual transmission (AMT) control is implemented in this subsystem. It can be activated using the Sw_AMT[0|1] parameter.

The block controls the clutch pedal position while shifting gear. This is essential, for example, in parallel hybrid topologies, where the clutch between the electric machine and the transmission needs to be controlled. It is ensured that the gear is requested only if the clutch pedal is completely disengaged.

Additionally, the accelerator pedal position is scaled while shifting gear by a user-defined function.

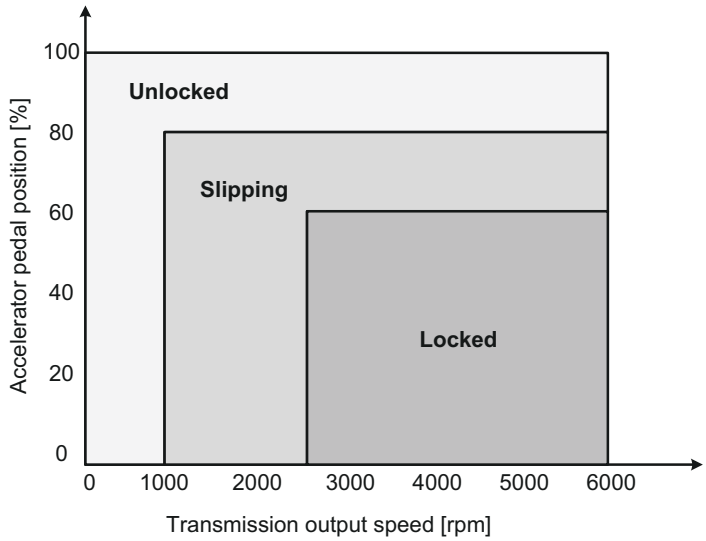
Note

At the moment, the model does not have a startup controller to ensure a soft vehicle launch. Therefore, it might be not suitable to use it for vehicles equipped with AMT and without a startup element. as for example a torque converter.

Lockup clutch control

The block controls the lockup clutch in automatic transmissions. The main function of the lockup clutch is to mechanically connect or disconnect the engine from the rest of the transmission units to increase efficiency and avoid oscillations in the drivetrain.

The lockup states are simplified as a function of the accelerator pedal position and the transmission output speed. See the following illustration:



The lockup clutch is unlocked at the lowest transmission output speed and the highest accelerator pedal position to enable vehicle startup and isolate the engine oscillations, respectively.

The lockup clutch is locked at relatively high transmission speeds to increase the efficiency of the drivetrain by avoiding losses in the torque converter.

Between the lock and unlock states, slipping is allowed. Lockup clutch slipping is controlled by using a PI controller with the speed ratio as a feedback signal. The actual speed ratio is calculated as the ratio of the transmission input speed to the engine speed and can be used as an indicator of slipping in the torque converter.

Inports

The following table shows the inports:

Name	Unit	Description
Gear_Current	[]	Current gear
Gear_Maneuver	[]	Stimulus gear from maneuver
Gear_Max_Maneuver	[0Off|1_n]	Highest allowed gear for the maneuver
GearClutch_Mode	[1|2|3|4]	Gear and clutch mode: ▪ 1: Stimulus mode ▪ 2: SoftECU mode ▪ 3: Open clutch mode ▪ 4: Reference gear mode
n_Engine	[rad/s]	Engine speed
omega_In_Trm	[rad/s]	Transmission input speed
omega_Out_Trm	[rad/s]	Transmission output speed
Pos_AccPedal_Driver	[%]	Accelerator pedal position from driver
Pos_ClutchPedal_Maneuver	[%]	Stimulus clutch pedal position

Name	Unit	Description
Reset	[0|1]	Reset of states
SelectorLever	[-3|-2|-1|0|1]	Selector lever position for automatic transmission ▪ -3: TipShift ▪ -2: Park ▪ -1: Reverse ▪ 0: Neutral ▪ 1: Drive
State_Engine	[0|1|2|3|4]	Engine state switch: ▪ 0: Engine off/Error ▪ 1: Ignition on ▪ 2: Starter on ▪ 3: Engine running ▪ 4: Engine shutdown
Sw_Trm_Mode	[1|2]	Transmission mode switch: ▪ 1: Manual transmission ▪ 2: Automatic transmission
TipShift	[-1|0|1]	TipShift request signal for TipTronic function: ▪ -1: Down shift ▪ 0: Off ▪ 1: Up shift

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Ctrl_Lockup_Clutch	[%]	Lockup clutch control signal
Ctrl_Parking_Pawl	[0|1]	Parking pawl control signal
Gear	[]	Requested gear
Pos_AccPedal_AMT	[%]	Scaled accelerator pedal position for automated manual transmission
Pos_ClutchPedal_AMT	[%]	Controlled clutch pedal position for automated manual transmission

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_I_LockupClutch	[1/s]	I gain of lockup clutch controller
Const_Max_Gear_Gearbox	[]	Maximum gearbox gear
Const_P_LockupClutch	[]	P gain of lockup clutch controller
Const_t_Min_Shift	[s]	Minimum time required between two successive shifts
Map_Factor_AccPedal	[0_1]	Accelerator pedal scaling factor during clutch actuation in AMT
Map_omega_Down shift	[rpm]	Transmission output speed for gear down shift schedule

Name	Unit	Description
Map_omega_Up shift	[rpm]	Transmission output speed for gear up shift schedule
Map_Pos_Diseng_ClutchPedal	[%]	Clutch pedal disengagement position while shifting gear in AMT
Map_Pos_Eng_ClutchPedal	[%]	Clutch pedal engagement position while shifting gear in AMT
Map_State_LockupClutch	[0_1]	Lockup clutch state
StepSize	[s]	Simulation step size
Sw_AMT	[0|1]	Switch to activate AMT control

Related topics

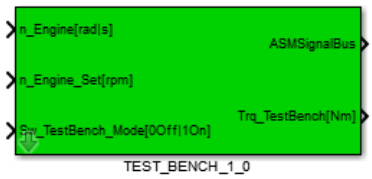
References

Soft ECU Transmission V10 (ModelDesk Parameterizing 🕮)

Test Bench (Version 1.0)

Description

The test bench model allows you to set a fixed engine speed controlled by the test bench controller. This is comparable to an engine test bench and can be used to examine the engine at a fixed engine speed.



The test bench model is implemented as a PI controller.

Inports

The following table shows the inports:

Name	Unit	Description
n_Engine	[rad/s]	Engine speed
n_Engine_Set	[rpm]	Engine speed setpoint
Sw_test bench_Mode	[0|1]	Switch for test bench activation ▪ 0: Off ▪ 1: On

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Trq_test bench	[Nm]	Test bench torque

Parameters

The following table shows the block parameters:

Name	Unit	Description
Const_I_Gain	[]	Test bench controller I-Gain
Const_P_Gain	[]	Test bench controller P-Gain
Const_Trq_max	[Nm]	Maximum dynamometer torque

Related topics

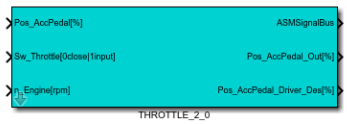
References

Test Bench VD V1 (ModelDesk Parameterizing 🕮)

Throttle (Version 2.0)

Description

The throttle controls the airflow to the engine to control combustion. As these details are not modeled in the engine, the throttle is also not modeled. The throttle subsystem contains only the idle speed control.



The idle speed controller aims to keep the engine speed at a preset value (Const_n_EngineIdle_Set parameter).

Inports

The following table shows the inports:

Name	Unit	Description
n_Engine	[rpm]	Engine speed
Pos_AccPedal	[%]	Position of gas pedal, defined by the driver
Sw_Throttle	[0|1]	Switch for activating throttle angle calculation or completely closing the throttle ▪ 0: Close

Name	Unit	Description
		▪ 1: Input

Outports

The following table shows the outports:

Name	Unit	Description
ASMSignalBus	[]	Signal bus that contains signals of ASM components. Refer to ASMSignalBus (ASM User Guide 🕮).
Pos_AccPedal_Driver_Des	[%]	Accelerator pedal position without intervention of the idle speed controller
Pos_AccPedal_Out	[%]	Idle speed controller influenced accelerator pedal position

Parameters

The following table shows the parameters:

Name	Unit	Description
Const_n_EngineIdle_Set	[rpm]	Set or reference value for the idle speed controller

Related topics

References

Engine Basic (Version 3.0) (ModelDesk Parameterizing 🕮)

New Features/Migration History of the ASM Vehicle Dynamics Blockset

Introduction

The following topics provide an overview of the changes to the ASM products in the previous Releases.

For an overview of the new features and migration of the current Release, refer to Automotive Simulation Models (ASM) (New Features and Migration 🕮).

Where to go from here

Information in this section

General Changes to the ASM Vehicle Dynamics Blockset.....402
Provides an overview of the new features and migration of the ASM blockset in the previous Releases.

History of the ADDITIONAL_CP_CALCULATIONS Block.....408
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the AERODYNAMICS Block.....408
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the BASIC_ROADS Block.....409
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the BRAKE_BOOSTER Block.....409
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the BRAKE_DISC Block.....410
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the BRAKE_HYDRAULICS_BASIC Block.....410
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the BRAKING_CIRCUIT_BASIC Block.....410
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the CENTRAL_DIFFERENTIAL Block.....410
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the CENTRAL_DIFFERENTIAL_RIGID Block.....412
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the CLUTCH Block.....412
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the CLUTCH_ENGAGEMENT_CONTROL Block.....413
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the CLUTCH_RIGID Block.....413
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the CLUTCH_4WD Block.....413
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the CLUTCH_4WD_RIGID Block.....414
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the COMMON_VEHICLE_PARAMETERS Block.....414
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the CONTACT_POINT_CALCULATION Block.....415
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the CONTACT_POINT_CALCULATION_x Block.....415
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the COORDINATE_TRANSFORMATION Block.....415
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the CRANK_SHAFT Block.....416
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the CRANK_SHAFT_RIGID Block.....417
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the DISPLACEMENT_SENSOR Block.....417
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the DRIVETRAIN_VARIANT_SWITCHES Block.....418
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the DRUM_BRAKE Block.....418
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the DRUM_BRAKE_2D Block.....418
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the ENGINE Block.....418
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the ENGINE_BASIC Block.....419
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the ENGINE_OPERATION_BASIC Block.....419
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the ESP_TORQUE_INTERVENTION_FAST Block.....420
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the ESP_TORQUE_INTERVENTION_SLOW Block.....420
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the FRONT_DIFFERENTIAL Block.....421
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the FRONT_DIFFERENTIAL_RIGID Block.....422
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the FUEL_CONSUMPTION Block.....422
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the GEARBOX_AT Block.....422
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the GEARBOX_AT_RIGID Block.....423
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the GEARBOX_MT Block.....424
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the GEARBOX_MT_RIGID Block.....425
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the IDLE_SPEED_CONTROL_ENGINE_BASIC Block.....426
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the LATERAL_CONTROL1 Block.....426
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the LATERAL_CONTROL2 Block.....427
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the LOCKUP_CLUTCH Block.....427
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the LOCKUP_CLUTCH_CONTROL Block.....428
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the LOCKUP_CLUTCH_RIGID Block.....428
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the MANIFOLD_PRESSURE Block.....428
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the MASS_FORCES_AND_TORQUES Block.....429
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the MC_PHERSON_STRUT Block.....429
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the REAR_DIFFERENTIAL Block.....429
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the REAR_DIFFERENTIAL_RIGID Block.....430
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the RELATIVE_POSITION_x Block.....431
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the RELATIVE_VELOCITIES_x Block.....431
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the RIGID_AXLE Block.....431
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SEMI_TRAILING_ARM Block.....432
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SENSOR_MOTION Block.....432
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SHAFT_RIGID Block.....433
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SHAFT_TGCL4WD Block.....433
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SHAFT_x Block.....434
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SHIFT_STRATEGY Block.....434
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SOFT_ECU_POWERSTEERING Block.....434
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SOFT_ECU_TRANSMISSION Block.....435
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SOFT_ECU_TRANSMISSION_10_0 Block.....437
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the STABI_COMP_FRONT Block.....437
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the STABILISATION Block.....437
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the STABILISATION_FF_FF4WD Block.....437
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the STABILISATION_TO_TRANSMISSION Block.....438
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the STARTER Block.....438
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the START_BUTTON Block.....439
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the STEERING Block.....439
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSCOMP_2D_REAR Block.....440
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the STEERING_TIE_ROD Block.....441
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the STEERING_VARIABLE_RATIO Block.....441
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the STEERING_3DOF_VARIABLE_RATIO Block.....442
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUBFRAME Block.....443
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSCOMP_RIGID_SYM_x Block.....443
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSCOMP_2D_FRONT Block.....444
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSFORCES_ANC_SPRING Block.....444
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSKIN_ANC_SPRING Block.....444
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSKIN_x_ASYM_3DOF Block.....445
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSKIN_FRONT_ASYM_3DOF Block.....445
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSKIN_FRONT_SYM_3DOF Block.....445
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSKIN_REAR_ASYM_3DOF Block.....446
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSKIN_REAR_SYM_3DOF Block.....446
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSKIN_RIGID_SYM Block.....446
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSKIN_RIGID_TRUCK_x Block.....447
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSKIN_x_SYM_3DOF Block.....447
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_COMPLIANCE Block.....447
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_COMPLIANCE_FRONT Block.....448
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_COMPLIANCE_OPP_x Block.....448
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_COMPLIANCE_REAR Block.....448
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_COMPLIANCE_x Block.....449
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_FORCE_KINEMATICS Block.....449
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_FORCE_KINEMATICS_FRONT Block.....450
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_FORCE_KINEMATICS_REAR Block.....450
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_FORCES_ACTIVE Block.....450
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_FORCES_FRONT Block.....451
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_FORCES_REAR Block.....451
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_KINEMATICS_x_ASYMMETRIC Block.....451
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_KINEMATICS_x_SYMMETRIC Block.....451
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_KINEMATICS_FRONT_ASYMMETRIC Block.....451
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_KINEMATICS_FRONT_SYMMETRIC Block.....452
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_KINEMATICS_REAR_ASYMMETRIC Block.....452
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the SUSPENSION_KINEMATICS_REAR_SYMMETRIC Block.....452
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TEST_BENCH Block.....453
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the THROTTLE Block.....453
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TIRE_MF Block....453
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TIRE_MODEL_MAGIC_FORMULA Block....454
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TIRE_MODEL_MAGIC_FORMULA_DUMMY_x Block....454
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TIRE_MODEL_MAGIC_FORMULA_x Block....454
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TIRE_MODEL_TMEASY Block....455
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TORQUE_INTERVENTION_CONTROL Block....456
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TIRE_MODEL_TMEASY_DUMMY_x Block....456
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TIRE_MODEL_TMEASY_x Block....456
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TIRE_VEL_SAMPLING_POINT Block....457
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TORQUE_CONVERTER Block....458
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TORQUE_INTERVENTION_FAST_ENGINE_BASIC Block....458
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TORQUE_INTERVENTION_SLOW_ENGINE_BASIC Block....459
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TRANSFER_GEARBOX Block....459
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the TRANSMISSION_TO_STABILISATION Block.....460
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the VEHICLE_DYNAMICS_VARIANT_SWITCHES Block.....460
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the VEHICLE_MASS_AND_ADDITIONAL_LOADS Block.....460
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the VEHICLE_MOTION_1_0 Block.....461
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the VEHICLE_MOTION_CAR Block.....461
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the VEHICLE_MOVEMENT_INFO_1_0 Block.....462
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the VEHICLE_MOVEMENT_INFO_CAR Block.....462
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the WHEEL_SPEED Block.....463
Provides an overview of all the new features and migration of the ASM block in the previous Releases.

History of the Vehicle Dynamics Demo Model.....464
Provides an overview of the new features and migration of the ASM demo model in the previous Releases.

General Changes to the ASM Vehicle Dynamics Blockset

Release 2020-B

Scenarios with initial vehicle speed Drivetrain and driver components are now automatically initialized for scenarios with an initial vehicle speed. This includes setting the correct gear or selector lever and initializing the different drivetrain speeds accordingly to ensure a smooth scenario startup.

Adams2ASM Converter The Adams2ASM converter was expanded to include new functionalities. In addition to the parameterization for the suspension kinematics and compliance, it is now possible to import parameters of the entire vehicle into ASM. These include the mass, center of gravity, inertia tensor of the entire vehicle as well as the engine map and the configuration of the drivetrain.

Release 2020-A

Adams2ASM Converter In addition to the steering rod displacement, the steering wheel angle can now be selected as an input variable in the Adams simulation environment (for independent suspensions).

Release 2019-B

Adams2ASM Converter With the Adams2ASM Converter you can convert maps for the kinematics and compliance of the wheel suspensions from Adams Car™. The Adams2ASM Converter provides Adams command files to automate the simulation in Adams Car as much as possible. The parameterization of the maps can be done via the user interface in Adams Car. Both dependent and independent suspensions are supported.

To visualize the results generated with Adams Car and to convert into the correct model structure for the ASM model, the Adams2ASM Converter provides a user interface in MATLAB. Furthermore, the results can be written directly to ModelDesk.

Release 2019-A

The *caster* angle was renamed to *sideview* angle.

This applies to the following blocks:

- MC_PHERSON_STRUT
- SUSKIN_FRONT_ASYM_3DOF
- SUSKIN_FRONT_SYM_3DOF
- SUSKIN_RIGID_SYM_FRONT
- SUSKIN_RIGID_TRUCK_FRONT
- SUSPENSION_KINEMATICS_FRONT_ASYMMETRIC
- SUSPENSION_KINEMATICS_FRONT_SYMMETRIC
- RELATIVE_POSITION_FRONT
- SUSCOMP_2D_FRONT
- SUSCOMP_RIGID_SYM_FRONT
- SUSPENSION_COMPLIANCE_FRONT
- SUSPENSION_COMPLIANCE_OPP_FRONT
- ROTATION_SWITCH
- RIGID_AXLE
- SEMI_TRAILING_ARM
- SUSKIN_REAR_ASYM_3DOF
- SUSKIN_REAR_SYM_3DOF
- SUSKIN_RIGID_SYM_REAR
- SUSKIN_RIGID_TRUCK_REAR
- SUSPENSION_KINEMATICS_REAR_ASYMMETRIC
- SUSPENSION_KINEMATICS_REAR_SYMMETRIC
- RELATIVE_POSITION_REAR
- SUSCOMP_2D_REAR
- SUSCOMP_RIGID_SYM_REAR

- SUSPENSION_COMPLIANCE_OPP_REAR
- SUSPENSION_COMPLIANCE_REAR

Release 2018-B

A new inport was added to the suspension blocks: Trq_y_Tire

This applies to the following blocks:

- SUSCOMP_2D_FRONT
- SUSCOMP_2D_REAR
- SUSCOMP_RIGID_SYM_FRONT
- SUSCOMP_RIGID_SYM_REAR
- SUSPENSION_COMPLIANCE_FRONT
- SUSPENSION_COMPLIANCE_REAR
- SUSPENSION_COMPLIANCE_OPP_FRONT
- SUSPENSION_COMPLIANCE_OPP_REAR

Release 2017-B

TIRE Magic Formula A new implementation of Magic Formula according to Magic Formula 6.1 is added to the blockset. For more information on the model, refer to Magic Formula Tire Model 6.1 (ASM Vehicle Dynamics Addendum 🕮).

TIRE MODE The blockset contains the new TIRE_MODE block that is used to mirror the calculation of Magic Formula.

Release 2017-A

Final drive assembly subsystem All the delivered final drive assembly subsystems were merged into one subsystem. Therefore, to switch between different drivetrain configurations, only the relevant option has to be set in ModelDesk and no Simulink subsystems need to be exchanged anymore.

Import of magic formula "*.tir" file to ModelDesk To import the magic formula (MF) `*.tir` file to ModelDesk XML, a script is available: `asm_vd_tire_import_mf.m`.

New drivetrain stabilization Due to the new structure of the drivetrain stabilization, some blocks cannot be automatically migrated. Therefore, during migration, the links to the blocks are changed to the former versions.

The following blocks are affected:

- STABILIZATION_4_0
- STABILIZATION_FF_FF4WD_2_0
- STABILIZATION_TO_TRANSMISSION_2_0
- TRANSMISSION_TO_STABILIZATION_3_0

DIFFERENTIAL blocks New inports and outports were added: FROM_STABILIZATION[] and TO_STABILIZATION[]. These ports represent the connection with the drivetrain stabilization. During migration, these ports are connected with the corresponding Goto and From blocks.

GEARBOX blocks The Stabilisation_Out outport was renamed TO_STABILIZATION[] for unification purposes.

SHAFT blocks New inports and outports were added: FROM_STABILIZATION[] and TO_STABILIZATION[]. During migration, these ports are connected with the corresponding Goto and From blocks.

Release 2016-B

Look-up table migration The discontinued Simulink blocks Lookup and Lookup2D in the ASM library blocks were updated to the new standard Simulink Look-up table (n-D) block. Refer to Changes to all ASM Products (ASM User Guide 🕮).

The look-up tables were updated in the following blocks within this library:

- AERODYNAMICS
- BRAKE_BOOSTER
- CENTRAL_DIFFERENTIAL
- CLUTCH
- CLUTCH_4WD
- ENGINE
- ENGINE_BASIC_3_0
- ESP_TORQUE_INTERVENTION_SLOW_3_0
- FRONT_DIFFERENTIAL
- GEARBOX_AT
- GEARBOX_MT
- MANIFOLD_PRESSURE
- REAR_DIFFERENTIAL
- SOFT_ECU_TRANSMISSION_8_0
- STEERING
- STEERING_VARIABLE_RATIO
- STEERING_3DOF_VARIABLE_RATIO
- SUBFRAME
- SUSCOMP_RIGID_SYM_FRONT
- SUSCOMP_RIGID_SYM_REAR
- SUSPENSION_COMPLIANCE_FRONT
- SUSPENSION_COMPLIANCE_OPP_FRONT
- SUSPENSION_COMPLIANCE_OPP_REAR
- SUSPENSION_COMPLIANCE_REAR
- TIRE_MODEL_MAGIC_FORMULA_***
- TIRE_MODEL_TMEASY_***
- TORQUE_CONVERTER
- TORQUE_CONVERTER_RIGID
- TORQUE_INTERVENTION_SLOW_ENGINE_BASIC
- WHEEL_SPEED

License check of ASM Utils blocks The ASM_UTILS license was discontinued. The ASM Utils blocks now check the license of the ASM blockset in which they are used.

The Utils blocks in the following blocks within this library were updated:

- CLUTCH
- DRUM_BRAKE_FRONT
- DRUM_BRAKE_REAR
- RIGID_AXLE
- SOFT_ECU_TRANSMISSION
- STEERING_VARIABLE_RATIO
- STEERING_3DOF_VARIABLE_RATIO
- SUSKIN_FRONT_ASYM_3DOF
- SUSKIN_FRONT_ASYM_ANC_SPRING
- SUSKIN_FRONT_ASYM_ANC_SPRING_3DOF
- SUSKIN_FRONT_SYM_3DOF
- SUSKIN_FRONT_SYM_ANC_SPRING
- SUSKIN_FRONT_SYM_ANC_SPRING_3DOF
- SUSKIN_RIGID_SYM_FRONT
- SUSKIN_REAR_ASYM_3DOF
- SUSKIN_REAR_ASYM_ANC_SPRING
- SUSKIN_REAR_ASYM_ANC_SPRING_3DOF
- SUSKIN_REAR_SYM_3DOF
- SUSKIN_REAR_SYM_ANC_SPRING
- SUSKIN_REAR_SYM_ANC_SPRING_3DOF
- SUSKIN_RIGID_SYM_REAR
- SUSFORCES_FRONT_ANC_SPRING
- SUSFORCES_REAR_ANC_SPRING
- SUSPENSION_FORCES_ACTIVE_FRONT
- SUSPENSION_FORCES_ACTIVE_REAR
- SUSPENSION_FORCES_FRONT, SUSPENSION_FORCES_FRONT_1_0
- SUSPENSION_FORCES_REAR, SUSPENSION_FORCES_REAR_1_0
- SUSPENSION_KINEMATICS_FRONT_ASYMMETRIC
- SUSPENSION_KINEMATICS_FRONT_SYMMETRIC
- SUSPENSION_KINEMATICS_FRONT_1_0
- SUSPENSION_KINEMATICS_REAR_ASYMMETRIC
- SUSPENSION_KINEMATICS_REAR_SYMMETRIC
- SUSPENSION_KINEMATICS_REAR_1_0

Release 2016-A

Scope handling During migration, all disabled scopes within the model need to be restored to ensure that the model works properly, including the new scope handling procedure. You might have to disable these scopes again after migration. You can easily disable scopes via the **Scope Handling GUI** button provided by ASM. This makes direct use of the commenting feature in Simulink.

Release 2015-A

The drivetrain blocks in the library were combined into subsystems to provide a better overview and easy access. The blocks are divided according to their functionalities.

Release 2014-B

The Drivetrain model was expanded with a new modeling approach. Besides the original flexible drivetrain, i.e., with elastic shafts, a rigid drivetrain can now be simulated. The main differences between the two approaches are described in the following:

Flexible Drivetrain Simulates the behavior of a drivetrain with stiff drive shafts and takes the elasticity and the inertias of the drivetrain components into account.

By including the stiff drive shafts, the drivetrain dynamics and oscillations can be simulated. However, the relatively high stiffness and small inertias of the different components can result in a stiff system of equations. This raises the difficulty of the numerical integration and might lead to an unstable system. Therefore, in this case, the complete drivetrain model is stabilized.

Rigid Drivetrain Simulates the behavior of a drivetrain without stiff drive shafts. No elasticity or inertias of the drivetrain components are taken into account.

Using this model approach, you can simulate the drivetrain components with simpler models and fewer parameters. It is also possible to extend the standard drivetrain model for other configurations with relatively low effort.

For the rigid drivetrain, several new blocks were introduced:

- CENTRAL_DIFFERENTIAL_RIGID
- REAR_DIFFERENTIAL_RIGID
- FRONT_DIFFERENTIAL_RIGID
- CLUTCH_RIGID
- CLUTCH_4WD_RIGID
- TORQUE_CONVERTER_RIGID
- GEARBOX_MT_RIGID
- GEARBOX_AT_RIGID
- TRANSFER_GEARBOX_RIGID
- CRANK_SHAFT_RIGID
- LOCKUP_CLUTCH_RIGID

Different final drive assembly configurations are supported and offered as standard demos for both flexible and rigid drivetrain. You can switch between these configurations by exchanging specific subsystems and adapting the related settings and switches.

Release 2014-A

Before migrating a ModelDesk project that was created with RLS2013-B, make sure that the first ASM patch is installed. If it is not, the warning **`Renaming parameter failed`** appears in the ModelDesk log file with reference to the Const_phi_Inj_PostLimit parameter. This does not affect simulation or model

handling and can be fixed by importing the SoftECU parameter file from the RLS2014-A demo model into the corresponding parameter pool location `…\Parameterization\MOD_XXX\Pool\Parameter\ASM\SoftECU\ Soft ECU Diesel\Soft ECU Diesel.xml`.

Release 2013-B

Several blocks in the **Environment**, **Drivetrain** and **Soft ECU** subsystems now have a reset inport that is triggered by the maneuver reset.

Release 7.3

The suspension compliance systems were extended with additional displacements and rotational angles as functions of longitudinal torques acting on the wheels' centers.

The steering systems were extended with compliance effects, i.e., additional steering rod displacement as a function of general forces and torques acting on the steering rod.

Release 6.2

Now this blockset is available as operator version. This is a model variant specifically designed for offline simulation with Simulink®.

The model offers the same functionality, simulation quality and parameterization options as the standard simulation package. The operator version is compatible with the standard model (developer version) and can be parameterized in ModelDesk, the parameterization software.

The fundamental difference with this model is the way the library components are implemented: They are encapsulated in separate systems to ensure good performance during Simulink simulation. The systems are accessible in the model so that their input/output behavior can be studied.

History of the ADDITIONAL_CP_CALCULATIONS Block

Release 2014-A

The **ADDITIONAL_CP_CALCULATIONS** block can be used to calculate additional geometric values of the tire contact point, which might be necessary for detailed tire forces investigations.

History of the AERODYNAMICS Block

Release 2014-B

The calculation of the wind force now takes the vertical wind velocity (v_z) into account. Thus, the amount of the wind vector during pitch is considered correctly.

Release 6.2

The aerodynamics forces and torques are now functions of the aerodynamics coefficients (which are functions of the angle of incidence), the longitudinal shadow area of the vehicle, and the characteristic length of the vehicle (mostly the vehicle wheelbase). Thus, several new parameters were added to the block's mask. The main difference in the new aerodynamics implementation compared to the former one is the way the aerodynamics coefficients are included. These are now functions of the angle of incidence. These functions can be gathered from wind tunnel tests. In addition the pressure point was replaced by the aerodynamics reference point. If you want to change back to the former aerodynamics implementation, you can find the former block in the FormerVersions subsystem in the aerodynamics library. Just drag & drop the former block to your Simulink model.

As the new implementation of the vehicle's aerodynamics changed a lot and the parameters cannot be migrated automatically, the link to the aerodynamics library is changed to the former implementation (FormerVersion/AERODYNAMICS_1_0 subsystem) during the migration of older ASM models. Thus, the simulation behavior is not changed. If you want to use the new aerodynamics calculation, just drag & drop the AERODYNAMICS block from the ASM_VehicleDynamics_lib Simulink library to your model and adapt the new parameters (mainly the aerodynamics coefficients as functions of the angle of incidence) to your needs.

History of the BASIC_ROADS Block

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

History of the BRAKE_BOOSTER Block

Release 6.5

The new model simulates a vacuum brake booster that supplements the brake pedal force with a force generated by a pneumatic booster system.

History of the BRAKE_DISC Block

Release 2020-B	New signals have been added to the ASMSignalBus.
Release 6.5	Some new signals were added to the **ASMSignalBus**. These signals are needed for the brake booster support of the driver in the ASM **Environment** Blockset.
Release 6.4	A new inport for the parking brake function was added. The additional inputs are connected to default values. They were chosen so that the change does not affect the simulation result. No additional migration steps are required.
Release 6.2	A new **p_Ambient** input port for ambient pressure was added. This solves issues due to mixed use of absolute and relative pressures in the brake system.

History of the BRAKE_HYDRAULICS_BASIC Block

Release 6.1	This block now has an inport for resetting its integrators.

History of the BRAKING_CIRCUIT_BASIC Block

Release 6.1	This block now has an inport for resetting its integrators.

History of the CENTRAL_DIFFERENTIAL Block

Release 2016-B	New inports were added for an external initialization of the speed integrators.
Release 7.3	To support the multi-instance feature, this drivetrain block requires a MaskInitialization function to ensure unique Goto/From connection for data exchange with stabilization. This is added during migration.

Release 7.1

New input ports for an external torque introduced on the cage (Trq_Ext_Cage[Nm]) and an additional inertia of parts connected to the cage (Inertia_Ext_Cage[kgm2]) were added. The main reduction ratio and the torque distribution were added to the ASMSignalBus.

The bus to the stabilization now contains the inertia for each separate part and for all ratios.

Release 7.0

The new omega_Cage[rad|s] outport was added. This makes it easier to use this block also in other drivetrain structures.

Release 6.5

The CENTRAL_DIFFERENTIAL block now supports a parking pawl for automatic transmissions. The Ctrl_Parking_Pawl[0|1] input was introduced for this. The parking pawl function was inserted in this block and not in the transmission block because the transmission output speed is calculated by the central differential.

The new Ctrl_Parking_Pawl[0|1] input is connected to a constant block. To introduce the new functionality, is has to be connected to the Ctrl_Parking_Pawl[0|1] output of the SOFT_ECU_TRANSMISSION block.

Release 6.2

The friction model was improved with additional saturation of the static torque.

The differential lock was redesigned. You can select the parts to be connected by the lock clutch with the new Sw_DifferentialLock_Mode parameter. It is now possible to simulate a part-time all-wheel drive with appropriate parameters.

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The Mux block was replaced by the BusCreator block for creating bus signals.

The unit of the damping parameters was corrected from [Nm/rad/s] to [Nm/(rad/s)].

The change of friction torque according to relative velocity in the lock clutch was put in the block's mask as Const_Coef_DynamicClutchTrq. The default value is equal to the old model value. The inner gear ratio parameter was put in the block's mask as Const_i_InnerGear. The default value is set to match the old implementation. The default value of Sw_DifferentialLock_Mode is set so that the two output shafts are connected. This matches the old implementation.

History of the CENTRAL_DIFFERENTIAL_RIGID Block

Release 2017-A

The SW_DrivetrainMode[1|2|3] inport was renamed Sw_DiffMode[1FrontRear|2Front|3Rear]. This does not cause any functional change in the block itself.

History of the CLUTCH Block

Release 2017-A

The Stabilisation_Out outport was renamed TO_STABILIZATION[] for unification purposes.

The omega_dt_Clutch[rad|s] inport was renamed FROM_STABILIZATION[] for unification purposes.

Release 2016-B

A new inport was added for an external initialization of the speed integrator.

Release 2015-A

To simulate the kiss point, a new parameter (Map_Trq_Clutch_Red) was introduced. The parameter describes the clutch nominal torque capacity as a function of the clutch actuation signal: i.e., clutch pedal position or lockup clutch control.

A new parameter (Map_Trq_Clutch_Red) is added to the block. The parameter is initialized with a linear relationship, so the former behavior of the block remains unchanged.

Release 6.2

The friction model was improved with additional saturation of the static torque.

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The unit of the damping parameters was corrected from [Nm/rad/s] to [Nm/(rad/s)].

The Map_Stiffenss parameter was renamed to Map_Stiffness.

History of the CLUTCH_ENGAGEMENT_CONTROL Block

Release 2020-B	The block has a new inport: **Enable_v_Vehicle_Init**. To keep the model behavior unchanged, it is connected to constant block during migration. The new inport is used to enable drivetrain initialization for scenarios with an initial vehicle speed.
Release 2018-A	A problem with the block not passing reverse gears was resolved.

History of the CLUTCH_RIGID Block

Release 2015-A	To simulate the kiss point, a new parameter (**Map_Trq_Clutch_Red**) was introduced. The parameter describes the clutch nominal torque capacity as a function of the clutch actuation signal: i.e., clutch pedal position or lockup clutch control. A new parameter (**Map_Trq_Clutch_Red**) is added to the block. The parameter is initialized with a linear relationship, so the former behavior of the block remains unchanged.

History of the CLUTCH_4WD Block

Release 2017-A	New inports and outports were added: **FROM_STABILIZATION[]** and **TO_STABILIZATION[]**. During migration, these ports are connected with the corresponding Goto and From blocks.
Release 2016-B	The block now gets the main reduction ratio of the rear differential via an inport instead of the GoToFrom blocks. A new inport was added for an external initialization of the speed integrator.
Release 2015-A	To simulate the kiss point, a new parameter (**Map_Trq_Clutch_Red**) was introduced. The parameter describes the clutch nominal torque capacity as a function of the clutch actuation signal: i.e., clutch pedal position or lockup clutch control.

A new parameter (**Map_Trq_Clutch_Red**) is added to the block. The parameter is initialized with a linear relationship, so the former behavior of the block remains unchanged.

Release 2014-A

During migration, the inports **Trq_RL_Shaft[Nm]** and **Trq_RR_Shaft[Nm]** are scaled to keep the old simulation behavior unchanged. An annotation is added to recommend that you change to the new implementation.

Release 7.3

To support the multi-instance feature, this drivetrain block requires a MaskInitialization function to ensure unique Goto/From connection for data exchange with stabilization. This is added during migration.

Release 7.1

New drivetrain variants for Front Wheel Front Engine Vehicle with Optional 4WD Extension.

In addition to the existing drivetrain variants All Wheel Drive, Front Wheel Drive and Rear Wheel Drive, the ASM **Vehicle Dynamics** Model now also supports front wheel front engine and front wheel front engine with 4WD extension drivetrain. To use the new variant, you can replace the subsystem of the existing drivetrain variant by a prepared demo subsystem.

This block was added to support the new variants.

History of the CLUTCH_4WD_RIGID Block

Release 2015-A

To simulate the kiss point, a new parameter (**Map_Trq_Clutch_Red**) was introduced. The parameter describes the clutch nominal torque capacity as a function of the clutch actuation signal: i.e., clutch pedal position or lockup clutch control.

A new parameter (**Map_Trq_Clutch_Red**) is added to the block. The parameter is initialized with a linear relationship, so the former behavior of the block remains unchanged.

History of the COMMON_VEHICLE_PARAMETERS Block

Release 7.3

Parameters for the dynamics brake torques were added and are used at the **WHEEL_SPEED** block. All signals are available at the new outport bus and no longer routed via Goto blocks from below the masked subsystem. The Goto blocks are automatically added after the subsystem during the migration process.

History of the CONTACT_POINT_CALCULATION Block

Release 6.4	The Pos_CP_CoorSys_E output signal is now a 1D-array Simulink signal type instead of a row/column matrix.
Release 6.3	The CONTACT_POINT_CALCULATION block now contains a reset signal inport. The memory block inside this block can be reset to the initial value to avoid NaN values. A new inport for the reset signal was added.

History of the CONTACT_POINT_CALCULATION_x Block

Release 2013-B	The contact point calculation considers the tire camber angle. The outport Angle_Alpha_Tire[rad] was renamed to Sin_Angle_Alpha_Tire[]. The bus signal label was changed from Angle_Alpha_xxx_Tire[deg] to Sin_Angle_Alpha_xxx_Tire[] and the trigonometry calculation asin was removed.
Release 2013-A	The Simulink subsystem was restructured.
Release 7.4	You can reset the tire radius calculation.

History of the COORDINATE_TRANSFORMATION Block

Release 2015-A	There is a new outport for the horizontal sensor signal: RotationMatrix_V_to_VH
Release 6.4	The a_Gravity_CoorSys_V output signal is now a 1D-array Simulink signal type instead of a row/column matrix. The mapping calculation from angular velocity to the first time derivative of the generalized coordinates was improved.

Release 6.2	The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask. **Note** The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

History of the CRANK_SHAFT Block

Release 2017-A	New inports and outports were added: FROM_STABILIZATION[] and TO_STABILIZATION[]. During migration, these ports are connected with the corresponding Goto and From blocks.
Release 2016-B	A new inport was added for an external initialization of the speed integrator.
Release 2015-A	The crankshaft inertia is added to the ASMSignalBus block.
Release 2014-B	The CRANK_SHAFT block has the new Sw_Trq_Mass_Mode parameter. This parameter was provided via signal routing and inport. The deletion of the corresponding inport is compensated during migration.
Release 2013-B	The lower saturation limit of the integrator was changed to `"-inf"`. This makes the block compatible with virtual vehicle systems.
Release 7.3	To support the multi-instance feature, this drivetrain block requires a MaskInitialization function to ensure unique Goto/From connection for data exchange with stabilization. This is added during migration.
Release 6.5	The CRANK_SHAFT block is now consistent with the CRANK_SHAFT block of the ASM Drivetrain Basic Blockset. The modulated mass torque map was introduced as a new parameter. The effects of accelerating and decelerating the pistons can now be considered together with the crank angle and the engine speed. The modulated mass torque can be activated and deactivated by the new SW_Trq_Mass_Mode[1On|2Off] inport. During migration, this switch is set to off.

The new **Trq_test bench[Nm]** inport was added to include additional engine test bench torque on the crankshaft. During migration, this port is connected to a constant block with a value of 0. The new **Inertia_External[kgm2]** inport was added to consider additional inertias (for example, of the starter). During migration, this port is connected to a constant block with a value of 0.

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (**SIM.StepSize** parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The **Mux** block was replaced by the **BusCreator** block for creating bus signals.

History of the CRANK_SHAFT_RIGID Block

Release 2017-A

The CopyFcn callback was removed from the block. This does not cause any functional change in the block itself.

Release 2016-B

A new inport was added for an external initialization of the speed integrator.

Release 2015-A

The crankshaft inertia is added to the ASMSignalBus block.

History of the DISPLACEMENT_SENSOR Block

Release 2014-A

The **DISPLACEMENT_SENSOR** block can be used to simulate wheel displacement sensors. The sensor characteristics are defined via a 1-D look-up table, as a function of the vertical wheel displacement.

History of the DRIVETRAIN_VARIANT_SWITCHES Block

Release 2016-B	The comments of the drivetrain mode for the FF4WD variants were extended from [1FourWheel|2FrontWheel|3RearWheel] to [1AWD|2FWD|3RWD|4FF-4WD|5FF].
Release 2014-B	The Sw_Trq_Mass_Mode parameter was deleted. This parameter was provided to the CRANK_SHAFT block via signal routing and inport. The CRANK_SHAFT block now includes this parameter instead of getting it as input.

History of the DRUM_BRAKE Block

Release 6.6	A map-based implementation is used to define the brake torques as functions of the brake pressures. Thus, nonlinear characteristics of drum brakes can be simulated. Note that the drum brake blocks are not supported by ModelDesk 2.2 (as included in Release 6.6).

History of the DRUM_BRAKE_2D Block

Release 2018-B	A new drum brake block modeled as 2-D look-up table was added.

History of the ENGINE Block

Release 2020-B	There is a new outport for engine-indicated torque: Trq_Ind_Engine. The outport is terminated during migration.
Release 2016-B	The size of the engine torque parameter was expanded from [30x30] to [31x30]. The old parameter is extrapolated and initialized in `asmmigratepost`. The engine torque look-up table Map_EffectiveEngineTorque was renamed to Map_Trq_MeanEff_Engine[Nm].
Release 2016-A	Some port and parameter names were revised. There are no functional changes.

History of the ENGINE_BASIC Block

Release 2015-A

The engine full load, i.e., the engine effective torque at full accelerator pedal position, is calculated and added to the ASMSignalBus block.

Release 7.2

To simulate the engine dynamic, a separate time constant for torque increase and decrease can be set.

The block has a new input to simulate the engine torque with a fast dynamic, for example, the ECU torque request.

As the new implementation of the engine basic model has changed a lot and parameter migration cannot be done automatically, the link to the engine basic library is changed to the former implementation (**FormerVersion/ENGINE_BASIC_3_0** subsystem) during migration of older ASM models. Thus, the simulation behavior is not changed. If you want to use the new engine basic implementation, drag the **ENGINE_BASIC** block from the **ASM_VehicleDynamics_lib** Simulink library to your model and adapt the inports, outports and the new parameters to your needs. Note that the interface to the ESP torque intervention and throttle idle speed model of the new engine basic model has changed.

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (**SIM.StepSize** parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The **AngleThrottle[deg]** variable was replaced by **Pos_AccPedal[%]**. The related input port and parameter maps were changed. The new variable is more suitable for the simulation of diesel and gasoline engines.

This block now has an inport for resetting its integrators.

History of the ENGINE_OPERATION_BASIC Block

Release 2020-B

The block has new inports.

To keep the model behavior unchanged, the new inports are connected to constant blocks during migration.

Release 2015-A

This block is new. It detects the engine operation state from the engine speed and the ignition as well as the starter request switch. The block activates the starter and gives information about the engine operation state.

History of the ESP_TORQUE_INTERVENTION_FAST Block

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

> **Note**
>
> The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

This block now has an inport for resetting its integrators.

History of the ESP_TORQUE_INTERVENTION_SLOW Block

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

> **Note**
>
> The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The AngleThrottle[deg] variable was replaced by Pos_AccPedal[%]. The related input and output ports, labels and parameter maps were changed. During the migration process AngleThrottle_ESPInfluenced[deg] is replaced by Pos_AccPedal_ESPInfluenced [%] in all bus selector blocks. The Map_Throttle parameter was replaced with Map_Trq_Engine_Inv.

This block now has an inport for resetting its integrators.

History of the FRONT_DIFFERENTIAL Block

Release 7.3

To support the multi-instance feature, this drivetrain block requires a MaskInitialization function to ensure unique Goto/From connection for data exchange with stabilization. This is added during migration.

Release 7.1

New input ports for an external torque introduced on the cage (Trq_Ext_Cage[Nm]), an additional inertia of parts connected to the cage (Inertia_Ext_Cage[kgm2]) and the parking pawl (Ctrl_Parking_Pawl[0|1]) were added. The main reduction ratio and the torque distribution were added to the ASMSignalBus.

The bus to the stabilization now contains the inertia for each separate part and for all ratios.

Release 7.0

The new omega_Cage[rad|s] outport and Inertia_Ext_InputShaft[kgm2] inport were added. This makes it easier to use this block in other drivetrain structures.

Release 6.2

The friction model was improved with additional saturation of the static torque.

The differential lock was redesigned. You can select the parts to be connected by the lock clutch with the new Sw_DifferentialLock_Mode parameter. It is now possible to simulate a part-time all-wheel drive with appropriate parameters.

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The Mux block was replaced by the BusCreator block for creating bus signals.

The unit of the damping parameters was corrected from [Nm/rad/s] to [Nm/(rad/s)].

The change of friction torque according to relative velocity in the lock clutch was put in the block's mask as Const_Coef_DynamicClutchTrq. The default value is equal to the old model value. The inner gear ratio parameter was put in the block's mask as Const_i_InnerGear. The default value is set to match the old implementation. The default value of Sw_DifferentialLock_Mode is set so that the two output shafts are connected. This matches the old implementation.

History of the FRONT_DIFFERENTIAL_RIGID Block

Release 2017-A	The SW_DrivetrainMode[1|2|3] inport was renamed Sw_Axle_Driven[0Undriven|1Driven]. During migration, this change is compensated.

History of the FUEL_CONSUMPTION Block

Release 2016-A	Some port and parameter names were revised. There are no functional changes.
Release 2015-B	The FUEL_CONSUMPTION block was added to the ENGINE subsystem to simulate the fuel consumption and the carbon dioxide emissions. The calculations are approximated, since the focus of the demo is the vehicle dynamics. The fuel consumption is calculated on the basis of a brake-specific fuel consumption map, which is a measurement of the fuel efficiency. The block is implemented in the new demo only and will not be automatically added during the migration. To use the block, drag it from the library and connect the related ports.

History of the GEARBOX_AT Block

Release 2020-B	The block has a new inport: Enable_v_Vehicle_Init. To keep the model behavior unchanged, it is connected to constant block during migration. The new inport is used to enable drivetrain initialization for scenarios with an initial vehicle speed.
Release 2019-A	The issue of gear synchronization failure during shifting through neutral was solved.
Release 2015-A	The damping coefficient of the gearbox input shaft is added as a mask parameter: Const_Damping_InputGear
Release 6.5	The look-up method of all tables was changed to interpolation/extrapolation.

Release 6.2

The inertia parameterization was adapted to a more general implementation. The transmission inertia reduced to the output shaft is now parameterized with a look-up table. The inertia of the input shaft is now only needed for simulating neutral gear. The transmission efficiency parameter was replaced by a forward and a reverse efficiency look-up table which are functions of the gear. The transient behavior between neutral and first gear was improved.

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The unit of the damping parameters was corrected from [Nm/rad/s] to [Nm/(rad/s)].

The new inertia map (Map_Inertia_Out) is initialized with default values. To achieve the original simulation result it has to be changed to: `Map_Inertia_Out(Gear) = Const_Inertia_OutputGear + Map_Gear_Ratio(Gear)2 * Const_Inertia_OutputGear`. The new default parameterization already fulfills this equation with the old default parameterization.

The new transmission efficiencies (`Map_Forward_Efficiency`,`Map_Reverse_Efficiency`) are initialized with default values. To achieve the original simulation result they have to be changed to:

```
Map_Forward_Efficiency(Gear) = Const_Transmission_Efficiency
```

```
Map_Reverse_Efficiency(Gear) = Const_Transmission_Efficiency
```

Related topics

References

Gearbox Automatic Transmission..... 53

History of the GEARBOX_AT_RIGID Block

Release 2020-B

The block has a new inport: Enable_v_Vehicle_Init. To keep the model behavior unchanged, it is connected to constant block during migration.

The new inport is used to enable drivetrain initialization for scenarios with an initial vehicle speed.

Release 2019-A	The issue of gear synchronization failure during shifting through neutral was solved.
Release 2015-A	The damping coefficient of the gearbox input shaft is added as a mask parameter: Const_Damping_InputGear
Related topics	References Gearbox Automatic Transmission Rigid..... 55

History of the GEARBOX_MT Block

Release 2020-B	The block has a new inport: Enable_v_Vehicle_Init. To keep the model behavior unchanged, it is connected to constant block during migration. The new inport is used to enable drivetrain initialization for scenarios with an initial vehicle speed.
Release 2019-A	The issue of gear synchronization failure during shifting through neutral was solved. The shifting through neutral feature can be activated in the GEAR_SHIFTER block.
Release 2015-A	The damping coefficient of the gearbox input shaft is added as a mask parameter: Const_Damping_InputGear
Release 6.5	The look-up method of all tables was changed to interpolation/extrapolation.
Release 6.2	The inertia parameterization was adapted to a more general implementation. The transmission inertia reduced to the output shaft is now parameterized with a look-up table. The inertia of the input shaft is now only needed for simulating neutral gear. The transmission efficiency parameter was replaced by a forward and a reverse efficiency look-up table which are functions of the gear. The transient behavior between neutral and first gear was improved.

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The unit of the damping parameters was corrected from [Nm/rad/s] to [Nm/(rad/s)].

Related topics

References

Gearbox Manual Transmission 56

History of the GEARBOX_MT_RIGID Block

Release 2020-B

The block has a new inport: Enable_v_Vehicle_Init. To keep the model behavior unchanged, it is connected to constant block during migration.

The new inport is used to enable drivetrain initialization for scenarios with an initial vehicle speed.

Release 2019-A

The issue of gear synchronization failure during shifting through neutral was solved. The shifting through neutral feature can be activated in the GEAR_SHIFTER block.

Release 2015-A

The damping coefficient of the gearbox input shaft is added as a mask parameter: Const_Damping_InputGear

Related topics

References

Gearbox Manual Transmission Rigid 58

History of the IDLE_SPEED_CONTROL_ENGINE_BASIC Block

Release 2020-B

The block has a new outport for set point of idle speed: n_Engine_Idle_Set. The outport is terminated during migration.

Release 2016-A

A new functionality to offer an external engine idle speed request was introduced.

The block has new inports for the external idle speed request. These inports are connected to dummy values during migration. Moreover, the idle speed controller parameters are promoted to mask parameters, which are added and initialized during migration.

Release 2013-B

A reset inport was added.

Release 7.2

The soft ECU engine gets torque requests from different ECUs and systems, for example, an ESP ECU, an ACC ECU or a hybrid management system. These requests depend on the dynamic states of the vehicle, which require the engine torque to decrease or increase. The engine ECU can control engine torque slowly, for example, by setting the accelerator pedal position, or fast, for example, by controlling the injection and ignition. An idle speed control is also included in the soft ECU.

The model consists of the following library blocks:

- TORQUE_INTERVENTION_FAST_ENGINE_BASIC
- TORQUE_INTERVENTION_SLOW_ENGINE_BASIC
- IDLE_SPEED_CONTROL_ENGINE_BASIC

History of the LATERAL_CONTROL1 Block

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

History of the LATERAL_CONTROL2 Block

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

History of the LOCKUP_CLUTCH Block

Release 2017-A

The Stabilisation_Out outport was renamed TO_STABILIZATION[] for unification purposes.

The omega_dt_Clutch[rad|s] inport was renamed FROM_STABILIZATION[] for unification purposes.

Release 2016-B

The inertia parameter of the driven plate is transfered to a mask parameter.

A new inport was added for an external initialization of the speed integrator.

Release 2015-A

To simulate the kiss point, a new parameter (Map_Trq_Clutch_Red) was introduced. The parameter describes the clutch nominal torque capacity as a function of the clutch actuation signal: i.e., clutch pedal position or lockup clutch control.

A new parameter (Map_Trq_Clutch_Red) is added to the block. The parameter is initialized with a linear relationship, so the former behavior of the block remains unchanged.

Release 6.2

The friction model was improved with additional saturation of the static torque.

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The unit of the damping parameters was corrected from [Nm/rad/s] to [Nm/(rad/s)].

History of the LOCKUP_CLUTCH_CONTROL Block

Release 2020-B

The unit of the control signal was changed from [0_1] to [%] and a first-order-dynamics filter was added to the control signal to restore the previous behavior.

Related topics

References

Lockup Clutch Control 97

History of the LOCKUP_CLUTCH_RIGID Block

Release 2015-A

To simulate the kiss point, a new parameter (**Map_Trq_Clutch_Red**) was introduced. The parameter describes the clutch nominal torque capacity as a function of the clutch actuation signal: i.e., clutch pedal position or lockup clutch control.

A new parameter (**Map_Trq_Clutch_Red**) is added to the block. The parameter is initialized with a linear relationship, so the former behavior of the block remains unchanged.

History of the MANIFOLD_PRESSURE Block

Release 6.5

The new block is based on a look-up table that calculates the manifold pressure from the accelerator pedal position. It is needed to simulate the manifold pressure as an input signal for the brake booster.

Related topics

References

Manifold Pressure 121

History of the MASS_FORCES_AND_TORQUES Block

Release 6.2	The tags at the fromblocks for the inertias of all wheels were corrected to FL, FR, RL and RR. Before, all inertias had been taken from the front left wheel.
Related topics	References Mass Forces and Torques 279

History of the MC_PHERSON_STRUT Block

Release 2015-A	The block has a new outport for partial derivative calculation of stabilizer, spring and damper kinematics.
Release 2013-A	New inport SW_Rotation (currently unused) was added in order to have the same interface as the other suspension model. All outport signals are now Simulink array instead of column vector.
Release 7.4	The Steering rod displacement and Steering rod velocity inports are now 2-dimensional signals to enable the left and right wheels to be steered independently.
Release 6.4	Additional signals were added to ASMSignalBus.

History of the REAR_DIFFERENTIAL Block

Release 2016-B	The global GoToFrom connection of the main reduction ratio intended for the CLUTCH_4WD block was deleted. New inports were added for an external initialization of the speed integrators.
Release 7.3	To support the multi-instance feature, this drivetrain block requires a MaskInitialization function to ensure unique Goto/From connection for data exchange with stabilization. This is added during migration.

Release 7.1

New input ports for an external torque introduced on the cage (Trq_Ext_Cage[Nm]), an additional inertia of parts connected to the cage (Inertia_Ext_Cage[kgm2]) and the parking pawl (Ctrl_Parking_Pawl[0|1]) were added. The main reduction ratio and the torque distribution were added to the ASMSignalBus. The bus to the stabilization now contains the inertia for each separate part and for all ratios.

Release 7.0

The new omega_Cage[rad|s] outport and Inertia_Ext_InputShaft[kgm2] inport were added. This makes it easier to use this block in other drivetrain structures.

Release 6.2

The friction model was improved with additional saturation of the static torque.

The differential lock was redesigned. You can select the parts to be connected by the lock clutch with the new Sw_DifferentialLock_Mode parameter. It is now possible to simulate a part-time all-wheel drive with appropriate parameters.

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The Mux block was replaced by the BusCreator block for creating bus signals.

The unit of the damping parameters was corrected from [Nm/rad/s] to [Nm/(rad/s)].

History of the REAR_DIFFERENTIAL_RIGID Block

Release 2017-A

The SW_DrivetrainMode[1|2|3] inport was renamed Sw_Axle_Driven[0Undriven|1Driven]. During migration, this change is compensated.

History of the RELATIVE_POSITION_x Block

Release 2013-A

A new inport called **SW_ROTATION** was added.

You can choose from two different rotation interpretations for the suspension:

- Gamma-Beta-Alpha
- Toe-Camber-Sideview

Related topics

References

Relative Position Front..... 137
Relative Position Rear..... 229

History of the RELATIVE_VELOCITIES_x Block

Release 7.4

The **Steering rod velocity** inport is now a 2-dimensional signal to enable the left and right wheels to be steered independently.

Related topics

References

Front Wheels Relative Velocities..... 135
Rear Wheel Relative Velocities..... 227

History of the RIGID_AXLE Block

Release 2016-A

There was a bug fix for the suspension kinematics calculation of the rigid axle.

Release 2013-A

New inport **SW_Rotation** (currently unused) was added in order to have the same interface as the other suspension model.

Release 7.4

The **Steering rod displacement** and **Steering rod velocity** inports are now 2-dimensional signals to enable the left and right wheels to be steered independently.

Release 6.5	This block has the new **WheelPos_Offset[x;y;z][m]** inport. This inport is connected to a constant block with a value of [0,0,0]. The inport can be used to set an offset for the wheel position of suspension kinematics.
Release 6.4	Additional signals were added to **ASMSignalBus**.

History of the SEMI_TRAILING_ARM Block

Release 2013-A	New inport **SW_Rotation** (currently unused) was added in order to have the same interface as the other suspension model.
Release 7.4	The **Steering rod displacement** and **Steering rod velocity** inports are now 2-dimensional signals to enable the left and right wheels to be steered independently.
Release 6.5	This block has the new **WheelPos_Offset[x;y;z][m]** inport. This inport is connected to a constant block with a value of [0,0,0]. The inport can be used to set an offset for the wheel position of suspension kinematics.
Release 6.4	Additional signals were added to **ASMSignalBus**.

History of the SENSOR_MOTION Block

Release 2015-A	The **SENSOR_MOTION** block now also calculates the horizontal sensor signals. There is a new Rotation Matrix inport for the calculation of horizontal sensor signals.
Release 6.5	A fourth sensor position was added. The parameter can be parameterized in ModelDesk.
Release 6.1	There are now two lateral acceleration signals available at the **SENSOR_MOTION** block. One lateral acceleration signal ignores the acceleration of gravity (**a_y_Sensor_CoorSys_V** signal), the other lateral acceleration signal takes the effect of the acceleration of gravity into account

(a_y_Total_Sensor_CoorSys_V signal). Both signals are part of the ASMSignalBus.

When you connect an ABS/ESP system (i.e. an ECU) to the sensor's signals, for example, for hardware-in-the-loop (HIL) simulation, you must be careful to use the correct lateral acceleration from the SENSOR_MOTION block. If the ECU itself compensates the acceleration of gravity, connect it to the a_y_Sensor_CoorSys_V signal. If it does not, connect it to the a_y_Total_Sensor_CoorSys_V signal.

History of the SHAFT_RIGID Block

Release 2016-B	A new parameter was added to describe the rotation damping. A new inport was added for an external initialization of the speed integrator.
Release 2015-A	This new block represents a general shaft. The block calculates the shaft speed from the shaft inertia and the applied torques.

History of the SHAFT_TGCL4WD Block

Release 7.3	To support the multi-instance feature, this drivetrain block requires a MaskInitialization function to ensure unique Goto/From connection for data exchange with stabilization. This is added during migration.
Release 7.1	New drivetrain variants for Front Wheel Front Engine Vehicle with Optional 4WD Extension. In addition to the existing drivetrain variants All Wheel Drive, Front Wheel Drive and Rear Wheel Drive, the ASM Vehicle Dynamics Model now also supports front wheel front engine and front wheel front engine with 4WD extension drivetrain. To use the new variant, you can replace the subsystem of the existing drivetrain variant by a prepared demo subsystem. This block was added to support the new variants.

History of the SHAFT_x Block

Release 2016-B

The names of the inports and the outports are made modular.

Further signals were added to the ASMSignalBus.

Release 7.3

To support the multi-instance feature, this drivetrain block requires a MaskInitialization function to ensure unique Goto/From connection for data exchange with stabilization. This is added during migration.

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The Mux block was replaced by the BusCreator block for creating bus signals.

The unit of the damping parameters was corrected from [Nm/rad/s] to [Nm/(rad/s)].

History of the SHIFT_STRATEGY Block

Release 2020-B

The block has a new inport: Enable_v_Vehicle_Init. To keep the model behavior unchanged, it is connected to constant block during migration.

The new inport is used to enable drivetrain initialization for scenarios with an initial vehicle speed.

History of the SOFT_ECU_POWERSTEERING Block

Release 2017-B

The TrqSpring_SteeringColumn[Nm] inport was renamed to TrqSensor_Steering[Nm].

Release 2015-B

The Bus2Vector block was added to avoid the "Bus signal treated as vector" warning.

Release 2014-B

The SOFT_ECU_POWERSTEERING block was added to the Soft ECU blockset to provide the assistive power steering torque to the steering model.

History of the SOFT_ECU_TRANSMISSION Block

Release 2016-B

The SOFT_ECU_TRANMISSION block was modularized. The old functionalities and controller strategies were subdivided into several modular blocks.

You can now modify and extend the controller structure according to your needs. The new implementation offers also the possibility to integrate a third-party controller strategy.

The following blocks were added to the library:

- CLUTCH_ENGAGEMENT_CONTROL
- LOCKUP_CLUTCH_CONTROL
- SHIFT_LOCK_CONTROL
- SHIFT_STRATEGY
- SOFTECU_TRANSMISSION_SETUP
- TORQUE_INTERVENTION_CONTROL
- TIP_SHIFT_CONTROL

Because of its completely new structure, the previous SOFT_ECU_TRANSMISISON block is not migrated to the new implementation. Instead, the previous block is saved in FormerVersions, and the link is changed to this block during migration.

Release 2016-A

It is now possible to stimulate the SOFT_ECU_TRANSMISSION block with stimulus or reference gear and clutch pedal position signals.

The stimulus gear and clutch pedal are forwarded to the output without further actions. Depending on whether the AMT switch is active or not, the clutch pedal signal is used for the AMT clutch or the lockup clutch, respectively.

The reference gear is used as a set value that affects the other outputs.

These features are deactivated per default and can be activated using a switch in the demo model.

During migration, the new inports are connected to dummy values. However, you can still connect the inports with the related signals from the model to use the block functionality.

Release 2015-A

The SOFT_ECU_TRANSMISSION block was completely revised and new approaches were implemented. The new block offers a better overview, a more robust implementation and several new functionalities.

The new block offers the following features:

- Robust automatic shift strategy with gear skipping functionality.
- Revised implementation of TipTronic control.
- New lockup clutch control strategy
- Clutch and accelerator pedals control for automated manual transmissions.

The block cannot be automatically migrated, due to the new added block parameter. Therefore, during migration, the link to the SOFT_ECU_TRANSMISSION block is changed to the former implementation in FormerVersions/SOFT_ECU_TRANSMISSION_8_0.

To use the new implementation, the SOFT_ECU_TRANSMISSION block, drag it from the ASM VehicleDynamics Library to the model. However, a hand adaptation of the inports, outports and parameters is necessary.

Release 2013-B

A reset inport was added. The simulation step size was added as a parameter.

The FurtherShiftTables ground blocks were replaced by constant zero blocks to avoid dimension problems in Simulink.

Release 2013-A

An initial gear is set in case of an initial velocity at maneuver start. The block interface in Simulink has not been modified.

Release 6.5

The handling of the parking pawl was redesigned. The old Pos_ParkingBrakePedal[%] output was renamed Ctrl_Parking_Pawl[0|1]. This output is active if the selector lever is in "P". During migration, a gain of 50 is added to this outport to achieve the same behavior as before, where Pos_ParkingBrakePedal[%] was set to 50 when the selector lever was in "P". This output can also be manually connected to the Ctrl_Parking_Pawl[0|1] input of the CENTRAL_DIFFERENTIAL block to add the parking pawl functionality to the model.

Release 6.2

The dimension of the shift tables was increased to handle up to 25 gears.

The shift tables were renamed from Map_AccPedal_down shiftFactor to Map_omega_Down shift and from Map_AccPedal_up shiftFactor to Map_omega_Up shift. In addition the dimension of the shift tables was increased to handle up to 25 gears.

The ASM_SHIFTABLE_LOOKUP block from the ASM Utilities library is used to evaluate the shift tables. This solves the problems that occur with the multiple use of local mask variables.

History of the SOFT_ECU_TRANSMISSION_10_0 Block

Release 2016-B

The block cannot be migrated automatically. Therefore, during migration, the link to the block is changed to the former version:`FormerVersions/SOFT_ECU_TRANSMISSION_10_0`.

History of the STABI_COMP_FRONT Block

Release 2015-A

This subsystem is new. It calculates the stabilizer compliance effects.

History of the STABILISATION Block

Release 7.3

To support the multi-instance feature, this drivetrain block requires a MaskInitialization function to ensure unique Goto/From connection for data exchange with stabilization. This is added during migration.

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (**SIM.StepSize** parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The **Mux** block was replaced by the **BusCreator** block for creating bus signals.

The unit of the damping parameters was corrected from [Nm/rad/s] to [Nm/(rad/s)].

History of the STABILISATION_FF_FF4WD Block

Release 7.3

To support the multi-instance feature, this drivetrain block requires a MaskInitialization function to ensure unique Goto/From connection for data exchange with stabilization. This is added during migration.

Release 7.1

New drivetrain variants for Front Wheel Front Engine Vehicle with Optional 4WD Extension.

In addition to the existing drivetrain variants All Wheel Drive, Front Wheel Drive and Rear Wheel Drive, the ASM **Vehicle Dynamics** Model now also supports front wheel front engine and front wheel front engine with 4WD extension drivetrains. To use the new variant, you can replace the subsystem of the existing drivetrain variant by a prepared demo subsystem.

This block was added to support the new variants.

Related topics

References

Stabilization.....68

History of the STABILISATION_TO_TRANSMISSION Block

Release 7.3

To support the multi-instance feature, this drivetrain block requires a MaskInitialization function to ensure unique Goto/From connection for data exchange with stabilization. This is added during migration.

Related topics

References

Stabilization.....68

History of the STARTER Block

Release 7.1

The signal label for starter inertia was corrected from **Signal1** to **Inertia_Starter[kgm2]**.

Release 6.5

The **STARTER** block is now consistent with the **STARTER** block of the ASM **Drivetrain Basic** Blockset. The started inertia was introduced as a new parameter, which is assigned to the new **Inertia_Started[kgm2]** output if the starter is active. To avoid changes in the model behavior, the new parameter value is set to zero during migration.

History of the START_BUTTON Block

Release 2018-B	A new START_BUTTON block was added to the SoftECU subsystem to simulate an engine start button.

History of the STEERING Block

Release 2016-A	The Simulink Unit Delay block was replaced with a memory block to avoid some warnings in MATLAB.
Release 2015-B	The Bus2Vector block was added to avoid the "Bus signal treated as vector" warning.
Release 2013-A	The Torque inport of PowerSteering is now connected to Torque spring column steering. First-order dynamics were added in the calculation of steering friction.
Release 7.2	The Sw_SteeringRod inport was extended by state 3: ExtDisplDispl_dt, in which the steering rod displacement and the steering rod speed at the steering subsystem are replaced by external ones. This mode can be used if the ASM Vehicle Dynamics Model is used in combination with a real steering test bench. The Displ_SteeringRod_External[m] inport was renamed SteeringRod_External[Displ;Displ_dt][m;m|s]. In addition, its dimension was changed from 1-D to 2-D. It is now a vector containing the steering rod displacement and steering rod speed that come from external sources such as measurements from a real steering test bench.
Release 7.1	The unit of the steering angle was corrected from [rad] to [deg].
Release 7.0	The new inport Trq_Driver[Nm] was added to set the driver torque from outside. In addition, the new inport Sw_DriverSteeringMode[1Angle|2Torque] was added to specify whether to use the steering wheel angle or torque from the driver as input to the steering system. If the driver's torque is used as input to the steering system, the steering column calculation is ignored. Signals representing the spring torque and spring stiffness at the steering column were added to the ASMSignalBus.

The ASMSignalBus hierarchy was changed: the ForceSteeringRod entry was replaced with ForcesAndTorques. The new bus hierarchy is migrated automatically.

Release 6.4

The block was extended so that it can also be used with an external electric power steering model. Additional outputs and inputs have therefore been added. For more information, see the embedded help of the block. The Angle_SteeringWheel and Angle_SteeringGear signals are added to the ASMSignalBus.

In the new STEERING block, the additional inputs are connected to default values. They were chosen so that the change does not affect the simulation result. No additional migration steps are required.

Release 6.2

The transient friction coefficient was put in the block's mask and is now accessible from ModelDesk. Before, the parameter was set to constant below the mask. A new Const_m_steering_rod parameter was added to the summation of the wheel masses to take the mass of the steering rod itself into account.

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The Const_Angel_Max_SteeringWheel parameter was renamed Const_Angle_Max_SteeringWheel.

This block now has an inport for resetting its integrators.

History of the SUSCOMP_2D_REAR Block

Release 2018-B

The torque about the y-axis was added to the compliance calculation.

Release 2014-B

This new suspension compliance block can be used to parameterize a compliance which also depends on the force and torque at the left wheel and the right wheel.

History of the STEERING_TIE_ROD Block

Release 2014-A	The STEERING_TIE_ROD block can be used to simulate the forces acting at the left and right steering tie rods. A 2-D look-up table has to be parameterized for this describing the geometric relation between the steering tie rod and the steering rack as a function of the steering rod displacement and the vertical wheel displacement. SUBFRAME block.

History of the STEERING_VARIABLE_RATIO Block

Release 2016-A	The Simulink Unit Delay block was replaced with a memory block to avoid some warnings in MATLAB.
Release 2013-B	The dimensions of the Map_i_Steering and Map_phi_SteeringColumn look-up tables were increased from [25,1] to [60,1]. These extensions are done automatically during initialization via a call of the steering postmigrate variant, for example, in the go function.
Release 7.2	The Displ_SteeringRod_External[m] inport was renamed SteeringRod_External[Displ;Displ_dt][m;m|s]. In addition, its dimension was changed from 1-D to 2-D. It is now a vector containing the steering rod displacement and steering rod speed that come from external sources such as measurements from a real steering test bench.
Release 7.1	The unit of the steering angle was corrected from [rad] to [deg].
Release 7.0	The new inport Trq_Driver[Nm] was added to set the driver torque from outside. In addition, the new inport Sw_DriverSteeringMode[1Angle|2Torque] was added to specify whether to use the steering wheel angle or torque from the driver as input to the steering system. If the driver's torque is used as input to the steering system, the steering column calculation is ignored. Signals representing the spring torque and spring stiffness at the steering column were added to the ASMSignalBus. The ASMSignalBus hierarchy was changed: the ForceSteeringRod entry was replaced with ForcesAndTorques. The new bus hierarchy is migrated automatically.

Release 6.6	A variable steering ratio can be simulated with this block. Two one-dimensional look-up tables have to be parameterized, defining the steering ratio and the angle of the steering column as functions of the displacement of the steering rod. Note that the variable steering ratio block is not supported by ModelDesk 2.2 (as included in Release 6.6).

History of the STEERING_3DOF_VARIABLE_RATIO Block

Release 2018-A	The possibility to simulate variable steering column stiffness was added.
Release 2017-B	wasThe new SW_DriverArm parameter has been added. The reset logic for System states was updated to facilitate plausible switching from steering mode 3 to 2. The new TrqSensor_Steering[Nm] signal was added to switch the sensor torque based on EPS motor location. A signal specification was added to avoid Simulink signal dimension error.
Release 2017-A	All the Memory blocks were replaced with MemoryWithReset blocks. Resets were added to all triggered subsystems in the friction elements to reset the friction elements with global resets.
Release 2016-A	There was a bug fix for the steering system end-limits simulation. The Simulink **Unit Delay** block was replaced with a memory block to avoid some warnings in MATLAB.
Release 2015-B	The **Bus2Vector** block was added to avoid the "Bus signal treated as vector" warning. The calculation of the exponential spring friction element at the steering rod and steering column was corrected to avoid "Not a Number" generation.
Release 2015-A	It is now possible to configure the location of a steering column spring damper system with respect to the two cardan joints. It is now also possible to configure the offset angle between the steering wheel zero degree mark and the upper cardan joint. These features can be configured using the **SW_LocationColumnStiffness** and **Const_InitialAngle_UpperCardanJoint** variables.

Two new parameters were added to the block:

- SW_LocationColumnStiffness to configure the position of the steering column spring damper system with respect to cardan joints
- Const_InitialAngle_UpperCardanJoint to configure the offset angle between the steering wheel 0deg mark and the upper cardan joint

During migration, the two variables are initialized with the values 2deg or 0deg, respectively so the former behavior of the block remains unchanged.

Release 2014-B

The STEERING_3DOF_VARIABLE_RATIO block is an improvement of the existing steering block with three inertial masses, corresponding force elements, and new friction elements. Compared to the 1DOF model, the on-center handling behavior of a vehicle is better represented by the 3DOF model. Also, the block can be parameterized with variable steering ratios.

History of the SUBFRAME Block

Release 2018-A

A problem with the linked subsystem in the ASM operator version was fixed.

Release 2016-A

There was a bug fix for the additional lateral and longitudinal displacement due to the angle gamma-compliance based on the lateral force.

Release 2014-A

The SUBFRAME block was extended to include the displacements in longitudinal direction.

The block was extended to include the displacements in longitudinal direction. The parameters have also been shifted to a lower level inside the model structure.

Release 7.4

The elastokinematics of a subframe are now used in addition to the suspension compliances. The SUBFRAME block can be used in the model as an option and in parallel to the suspension compliance block.

History of the SUSCOMP_RIGID_SYM_x Block

Release 2015-B

A new parameter was added to the block: Map_Angle_Gamma_Axle_F_y. It takes the axle rotation due to lateral force into account.

Release 2015-A	A bug concerning the mirrored compliance right wheel was fixed.
Release 2013-B	A new suspension compliance model was added for a rigid axle. With this new model you can parameterize a compliance effect of a rigid axle.

History of the SUSCOMP_2D_FRONT Block

Release 2018-B	The torque about the y-axis was added to the compliance calculation.
Release 2014-B	This new suspension compliance block can be used to parameterize a compliance which also depends on the force and torque at the left wheel and the right wheel.

History of the SUSFORCES_ANC_SPRING Block

Release 7.1	An ancillary spring can now be used in addition to the main spring. The ancillary spring accounts for additional stiffnesses of real suspension joints, which are not made of ideal revolute joints but contain elastic parts. This block calculates the ancillary spring forces as functions of the ancillary spring displacements.

History of the SUSKIN_ANC_SPRING Block

Release 7.1	An ancillary spring can now be used in addition to the main spring. The ancillary spring accounts for additional stiffnesses of real suspension joints, which are not made of ideal revolute joints but contain elastic parts. This block calculates the ancillary spring displacements as functions of the wheel lift and/or steering rod displacement.

History of the SUSKIN_x_ASYM_3DOF Block

Release 2013-A	A new inport called SW_ROTATION was added. You can choose from two different rotation interpretations for the suspension: ▪ Gamma-Beta-Alpha ▪ Toe-Camber-Sideview
Release 6.4	Two new library blocks were added to the ASM_VehicleDynamics library: SYMMETRIC_SUSPENSION_KINEMATICS_3DOF and ASYMMETRIC_SUSPENSION_KINEMATICS_3DOF. The SUSPENSION_KINEMATICS_3DOF can be parameterized according to the three degrees of freedom, for example, displacement z-wheel, displacement opposite z-wheel, and displacement steering rod. Hence, a steerable rigid axle or steerable rear axle can be modeled with these blocks.

History of the SUSKIN_FRONT_ASYM_3DOF Block

Release 2015-A	The block has a new outport for partial derivative calculation of stabilizer, spring and damper kinematics.
Release 7.4	The Steering rod displacement and Steering rod velocity inports are now 2-dimensional signals to enable the left and right wheels to be steered independently.

History of the SUSKIN_FRONT_SYM_3DOF Block

Release 2015-A	The block has a new outport for partial derivative calculation of stabilizer, spring and damper kinematics.
Release 7.4	The Steering rod displacement and Steering rod velocity inports are now 2-dimensional signals to enable the left and right wheels to be steered independently.

History of the SUSKIN_REAR_ASYM_3DOF Block

Release 7.4	The **Steering rod displacement** and **Steering rod velocity** inports are now 2-dimensional signals to enable the left and right wheels to be steered independently.
Release 6.5	This block has the new **WheelPos_Offset[x;y;z][m]** inport. This inport is connected to a constant block with a value of [0,0,0]. The inport can be used to set an offset for the wheel position of suspension kinematics.

History of the SUSKIN_REAR_SYM_3DOF Block

Release 7.4	The **Steering rod displacement** and **Steering rod velocity** inports are now 2-dimensional signals to enable the left and right wheels to be steered independently.
Release 6.5	This block has the new **WheelPos_Offset[x;y;z][m]** inport. This inport is connected to a constant block with a value of [0,0,0]. The inport can be used to set an offset for the wheel position of suspension kinematics.

History of the SUSKIN_RIGID_SYM Block

Release 2015-B	There was a bug fix for the Taylor series arcsin calculation (wrong sign at the third Taylor series).
Release 2015-A	The block has a new outport for partial derivative calculation of stabilizer, spring and damper kinematics.
Release 2013-B	With this new suspension kinematics model, a rigid axle is modeled that provides more convenient parameterization than the **SUSKIN_SYM_3DOF** model.

History of the SUSKIN_RIGID_TRUCK_x Block

Release 2016-B	The block has a new suspension kinematics model for, e.g., a truck rigid axle with recirculation ball steering system.

History of the SUSKIN_x_SYM_3DOF Block

Release 2013-A	A new inport called SW_ROTATION was added. You can choose from two different rotation interpretations for the suspension: ▪ Gamma-Beta-Alpha ▪ Toe-Camber-Sideview
Release 6.4	Two new library blocks were added to the ASM_VehicleDynamics library: SYMMETRIC_SUSPENSION_KINEMATICS_3DOF and ASYMMETRIC_SUSPENSION_KINEMATICS_3DOF. The SUSPENSION_KINEMATICS_3DOF can be parameterized according to the three degrees of freedom, for example, displacement z-wheel, displacement opposite z-wheel, and displacement steering rod. Hence, a steerable rigid axle or steerable rear axle can be modeled with these blocks.

History of the SUSPENSION_COMPLIANCE Block

Release 7.2	Suspension compliance behaviors on the left and right sides are mirrored correctly for opposite forces.
Release 7.1	The behavior of the compliance table for the right side of the suspension was mirrored for the left side.
Release 6.4	Suspension compliance was extended with an additional look-up table with opposite forces and torques as input. All wheel displacement and rotation due to compliance are then superimposed on the suspension kinematics. Now it is possible to parameterize the compliance for (semi-) dependent suspension.

History of the SUSPENSION_COMPLIANCE_FRONT Block

Release 7.3

The suspension compliance effects of the forces and torques of the opposite wheels were moved to the new SUSPENSION_COMPLIANCE_OPP_x subsystems. These new subsystems are added and connected automatically during the migration process.

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

This block now has an inport for resetting its integrators.

History of the SUSPENSION_COMPLIANCE_OPP_x Block

Release 2013-B

A new Pos_Wheel[Left;Right] inport was added. This inport will be used only for the new Suspension Compliance Rigid Axle Symmetric.

Release 2013-A

A new inport called SW_ROTATION was added.

You can choose from two different rotation interpretations for the suspension:

- Gamma-Beta-Alpha
- Toe-Camber-Sideview

History of the SUSPENSION_COMPLIANCE_REAR Block

Release 7.3

The suspension compliance effects of the forces and torques of the opposite wheels were moved to the new SUSPENSION_COMPLIANCE_OPP_x subsystems. These new subsystems are added and connected automatically during the migration process.

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

This block now has an inport for resetting its integrators.

History of the SUSPENSION_COMPLIANCE_x Block

Release 2013-B

A new Pos_Wheel[Left;Right] inport was added. This inport will be used only for the new Suspension Compliance Rigid Axle Symmetric.

Release 2013-A

A new inport called SW_ROTATION was added.

You can choose from two different rotation interpretations for the suspension:

- Gamma-Beta-Alpha
- Toe-Camber-Sideview

Release 6.4

In the new SUSPENSION_COMPLIANCE blockset, new parameters for the extended compliance were added.

History of the SUSPENSION_FORCE_KINEMATICS Block

Release 7.1

New inports for an ancillary spring were added.

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

History of the SUSPENSION_FORCE_KINEMATICS_FRONT Block

Release 2017-A	Two signals were added to the ASMSignalBus: AxleLoad_Left and AxleLoad_Right.
Release 6.2	The inport names were changed from F_FL/FR_Damper_dDispl_FL/FR_Damper[N|(m|s)] to F_FL/FR_Damper_dDispl_FL/FR_Damper_dt[N|(m|s)].
Related topics	References Suspension Force Kinematics Front 152

History of the SUSPENSION_FORCE_KINEMATICS_REAR Block

Release 2017-A	Two signals were added to the ASMSignalBus: AxleLoad_Left and AxleLoad_Right.
Release 6.2	The inport names were changed from F_RL/RR_Damper_dDispl_RL/RR_Damper[N|(m|s)] to F_RL/RR_Damper_dDispl_RL/RR_Damper_dt[N|(m|s)].
Related topics	References Suspension Force Kinematics Rear 222

History of the SUSPENSION_FORCES_ACTIVE Block

Release 6.6	The suspension forces look-up tables were extended by a second dimension. The force tables now depend not only on spring/damper/stabilizer displacements, but also on control signals, for example, control current from ECU. Thus, active suspension types can be simulated. Note that the suspension forces blocks are not supported by ModelDesk 2.2 (as included in Release 6.6).

History of the SUSPENSION_FORCES_FRONT Block

Release 6.2

The outport names were changed from F_FL/FR_Damper_dDispl_FL/FR_Damper[N|(m|s)] to F_FL/FR_Damper_dDispl_FL/FR_Damper_dt[N|(m|s)]. The related signal labels were changed accordingly at ASMSignalBus.

History of the SUSPENSION_FORCES_REAR Block

Release 6.2

The outport names were changed from F_RL/RR_Damper_dDispl_RL/RR_Damper[N|(m|s)] to F_RL/RR_Damper_dDispl_RL/RR_Damper_dt[N|(m|s)]. The related signal labels were changed accordingly at ASMSignalBus.

History of the SUSPENSION_KINEMATICS_x_ASYMMETRIC Block

Release 2013-A

A new inport called SW_ROTATION was added.

You can choose from two different rotation interpretations for the suspension:

- Gamma-Beta-Alpha
- Toe-Camber-Sideview

History of the SUSPENSION_KINEMATICS_x_SYMMETRIC Block

Release 2013-A

A new inport called SW_ROTATION was added.

You can choose from two different rotation interpretations for the suspension:

- Gamma-Beta-Alpha
- Toe-Camber-Sideview

History of the SUSPENSION_KINEMATICS_FRONT_ASYMMETRIC Block

Release 2015-A

The block has a new outport for partial derivative calculation of stabilizer, spring and damper kinematics.

Release 7.4	The **Steering rod displacement** and **Steering rod velocity** inports are now 2-dimensional signals to enable the left and right wheels to be steered independently.

History of the SUSPENSION_KINEMATICS_FRONT_SYMMETRIC Block

Release 2015-A	The block has a new outport for partial derivative calculation of stabilizer, spring and damper kinematics.
Release 7.4	The **Steering rod displacement** and **Steering rod velocity** inports are now 2-dimensional signals to enable the left and right wheels to be steered independently.

History of the SUSPENSION_KINEMATICS_REAR_ASYMMETRIC Block

Release 7.4	The **Steering rod displacement** and **Steering rod velocity** inports are now 2-dimensional signals to enable the left and right wheels to be steered independently.
Release 6.5	This block has the new **WheelPos_Offset[x;y;z][m]** inport. This inport is connected to a constant block with a value of [0,0,0]. The inport can be used to set an offset for the wheel position of suspension kinematics.

History of the SUSPENSION_KINEMATICS_REAR_SYMMETRIC Block

Release 7.4	The **Steering rod displacement** and **Steering rod velocity** inports are now 2-dimensional signals to enable the left and right wheels to be steered independently.
Release 6.5	This block has the new **WheelPos_Offset[x;y;z][m]** inport. This inport is connected to a constant block with a value of [0,0,0]. The inport can be used to set an offset for the wheel position of suspension kinematics.

History of the TEST_BENCH Block

Release 2015-A	The dynamometer inertia is now considered as a parameter. The block cannot be automatically migrated, due to the new added block parameter. Therefore, during migration, the link to the TEST_BENCH block is changed to the former implementation in FormerVersions/TEST_BENCH_1_0. To use the new implementation, drag the TEST_BENCH block from the ASM VehicleDynamics Library to the model. However a hand adaptation of the parameters is necessary.
Release 6.5	A new block was inserted for performing engine test bench tests. This block is equivalent to the TEST_BENCH block of the ASM DrivetrainBasic Blockset. If you set up a virtual vehicle including a detailed engine model, the TEST_BENCH block supports easy engine model testing.

History of the THROTTLE Block

Release 6.2	The idle speed controller has improved transient behavior between the idle speed controlled accelerator pedal position and the forwarded driver accelerator pedal position. The AngleThrottle[deg] variable was replaced by Pos_AccPedal[%]. The related input and output ports, labels and were changed. During the migration process AngleThrottle[deg] is replaced by Pos_AccPedal_out[%] in all bus selector blocks. The Const_Max_ThrottleAngel[deg] parameter is now unnecessary and was removed.

History of the TIRE_MF Block

Release 2019-A	A bug concerning the external tire radius connection was fixed.
Release 2018-B	The internal damping at low speed for slip x and for slip y was reduced.
Release 2018-A	All internal memories are reset during reset or in disabled state.

History of the TIRE_MODEL_MAGIC_FORMULA Block

Release 7.1	The sign of the "lateral slip" signal output in **ASMSignalBus** for the tire on the right side was corrected.
Release 6.3	The right tire behavior (force Fy, torque Mz and torque Mx) is now mirrored to the left tire behavior. Torque My in the **Magic Formula** tire model is now calculated with the correct sign.

History of the TIRE_MODEL_MAGIC_FORMULA_DUMMY_x Block

Release 2013-B	The inport name was changed from **Angle_Alpha_Tire[rad]** to **Sin_Angle_Alpha_Tire[]**.

History of the TIRE_MODEL_MAGIC_FORMULA_x Block

Release 2016-B	A new signal for the circumference of unloaded tire radius was added to the block in the signal bus.
Release 2016-A	There was a bug fix for the external tire radius, which was connected to the wrong signal. The Simulink **Signal Specification** block was added to support the simplified initialization mode in Simulink.
Release 2015-A	The bus signals are now used as a vector.
Release 2014-B	Two new inports for tire parameters are added which depend on the longitudinal velocity. The **ScalingParameter.LFZ0** parameter in the XML file (for ModelDesk) was renamed **ScalingParameters.LFZO**.
Release 2013-B	The tire force calculation at standstill considers the friction coefficient. The inport name was changed from **Angle_Alpha_Tire[rad]** to **Sin_Angle_Alpha_Tire[]**. The trigonometry sin calculation was removed.

Release 2013-A	Tire dynamic calculation was vectorized.
Release 7.4	Two new inports were added to the blocks for the tire radius to be changed by an external signal.
Release 6.2	The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask. **Note** The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` . The Mux block was replaced by the BusCreator block for creating bus signals. This block now has an inport for resetting its integrators.

History of the TIRE_MODEL_TMEASY Block

Release 7.1	The signal label of the front right tire was changed from v_Circumf_dyn to v_Circumf_Dyn to be compatible with the operator version.
Release 7.0	The TMEasy tire model was extended with bore torque calculation at very low or zero vehicle speed, for example, during parking maneuvers. This ensures the generation of a proper parking torque response to steering at very low or zero speed. Bore torque calculation at very low or zero vehicle speed was improved.
Release 6.5	Three user scaling parameters were added to the TMEasy tire model. You can scale the effect of the camber angle on the lateral force, the effect of the vertical angular velocity of the wheel on the vertical torque (bore torque), and the effect of the camber angle on the torque about the longitudinal axis of the wheel (tipping torque). During migration, three user scaling parameters with the default value "1" are added to the TMEasy tire model.
Release 6.4	The lower and upper bound calculation of Bore Torque were improved.

A smooth transition of the Tipping Torque calculation from loaded wheel to unloaded wheel is performed in the **TMEasy** model.

The normalizing factors calculation was changed. The normalizing factors take characteristic properties of the longitudinal and lateral tire force characteristics into account. With the new calculation, the normalizing factor will be nearly equal to one if the longitudinal and the lateral tire characteristics do not differ too much.

History of the TORQUE_INTERVENTION_CONTROL Block

Release 2020-B

The block has a new inport: **Enable_v_Vehicle_Init**. To keep the model behavior unchanged, it is connected to constant block during migration.

The new inport is used to enable drivetrain initialization for scenarios with an initial vehicle speed.

Related topics

References

Torque Intervention Control..... 105

History of the TIRE_MODEL_TMEASY_DUMMY_x Block

Release 2013-B

The inport name was changed from **Angle_Alpha_Tire[rad]** to **Sin_Angle_Alpha_Tire[]**.

History of the TIRE_MODEL_TMEASY_x Block

Release 2017-B

The calculation in the blocks was corrected: The bore torque is reduced to zero during forward movement.

Release 2016-B

Division by zero is now avoided if the friction coefficient is zero.

Release 2016-A	There was a bug fix for the external tire radius, which was connected to the wrong signal. The Simulink **Signal Specification** block was added to support the simplified initialization mode in Simulink.
Release 2015-A	The incorrect signal name **F_x_Dyn_RR_Tire_CoorSys_CP[N]** was fixed.
Release 2013-B	The tire force calculation at standstill considers the friction coefficient. The inport name was changed from **Angle_Alpha_Tire[rad]** to **Sin_Angle_Alpha_Tire[]**, as **sin(angle_alpha)** is used instead of **angle_alpha[rad]**.
Release 2013-A	The Simulink If subsystem was used instead of the enable subsystem. The default parameters in bore torque calculation were changed.
Release 7.4	Two new inports were added to the blocks for the tire radius to be changed by an external signal. The scaling parameters are connected to the slip camber and overturning torque calculation.
Release 6.2	The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask. **Note** The simulation step size (**SIM.StepSize** parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` . The **Mux** block was replaced by the **BusCreator** block for creating bus signals.

History of the TIRE_VEL_SAMPLING_POINT Block

Release 2014-B	The Magic Formula tire model can be used to parameterize different tire slip curves depending on the longitudinal tire velocity. Four velocity sampling points can be defined in this block.

History of the TORQUE_CONVERTER Block

Release 2020-B	The grounds inside the block are replaced with constants during migration to avoid problems with underspecified signal dimensions.
Release 2017-A	The Stabilisation_Out outport was renamed TO_STABILIZATION[] for unification purposes.
Release 6.4	The TORQUE_CONVERTER block was improved so that it can also handle negative pump speeds. The default parameterization for dragging the engine has also been optimized. In the new TORQUE_CONVERTER block, there are only changes inside the Library block. If you want to use the new default parameterization, you must transfer it to old projects from the demo project manually.
Release 6.2	The unit of the damping parameters was corrected from [Nm/rad/s] to [Nm/(rad/s)].
Release 6.1	The input of the efficiency map was corrected. It was changed from Ω_{Pump}/ $\Omega_{Turbine}$ to $\Omega_{Turbine}$/ Ω_{Pump}. This improves vehicle start-up.

History of the TORQUE_INTERVENTION_FAST_ENGINE_BASIC Block

Release 2020-B	The block is renamed to ESP_FAST_TORQUE_SET.
Release 7.2	The soft ECU engine gets torque requests from different ECUs and systems, for example, an ESP ECU, an ACC ECU or a hybrid management system. These requests depend on the dynamic states of the vehicle, which require the engine torque to decrease or increase. The engine ECU can control engine torque slowly, for example, by setting the accelerator pedal position, or fast, for example, by controlling the injection and ignition. An idle speed control is also included in the soft ECU. The model consists of the following library blocks: ▪ TORQUE_INTERVENTION_FAST_ENGINE_BASIC ▪ TORQUE_INTERVENTION_SLOW_ENGINE_BASIC ▪ IDLE_SPEED_CONTROL_ENGINE_BASIC

Related topics

References

ESP Fast Torque Set.....86

History of the TORQUE_INTERVENTION_SLOW_ENGINE_BASIC Block

Release 2020-B

The block is renamed to TORQUE_INTERVENTION_BASIC.

Release 7.2

The soft ECU engine gets torque requests from different ECUs and systems, for example, an ESP ECU, an ACC ECU or a hybrid management system. These requests depend on the dynamic states of the vehicle, which require the engine torque to decrease or increase. The engine ECU can control engine torque slowly, for example, by setting the accelerator pedal position, or fast, for example, by controlling the injection and ignition. An idle speed control is also included in the soft ECU.

The model consists of the following library blocks:

- TORQUE_INTERVENTION_FAST_ENGINE_BASIC
- TORQUE_INTERVENTION_SLOW_ENGINE_BASIC
- IDLE_SPEED_CONTROL_ENGINE_BASIC

History of the TRANSFER_GEARBOX Block

Release 2017-A

New inports and outports were added: FROM_STABILIZATION[] and TO_STABILIZATION[]. During migration, these ports are connected with the corresponding Goto and From blocks.

Release 7.3

To support the multi-instance feature, this drivetrain block requires a MaskInitialization function to ensure unique Goto/From connection for data exchange with stabilization. This is added during migration.

Release 7.1

New drivetrain variants for Front Wheel Front Engine Vehicle with Optional 4WD Extension.

In addition to the existing drivetrain variants All Wheel Drive, Front Wheel Drive and Rear Wheel Drive, the ASM Vehicle Dynamics Model now also supports front wheel front engine and front wheel front engine with 4WD extension drivetrains.

To use the new variant, you can replace the subsystem of the existing drivetrain variant by a prepared demo subsystem.

This block was added to support the new variants.

History of the TRANSMISSION_TO_STABILISATION Block

Release 7.3

To support the multi-instance feature, this drivetrain block requires a MaskInitialization function to ensure unique Goto/From connection for data exchange with stabilization. This is added during migration.

Release 6.2

The unit of the damping parameters was corrected from [Nm/rad/s] to [Nm/(rad/s)].

History of the VEHICLE_DYNAMICS_VARIANT_SWITCHES Block

Release 2020-B

New signals have been added to the ASMSignalBus.

Related topics

References

Vehicle Dynamics Variant Switches..... 296

History of the VEHICLE_MASS_AND_ADDITIONAL_LOADS Block

Release 2018-B

Three inports were added to the block:

- M_Body
- PosVec_CoG_Body[x;y;z]
- Inertia_Body[3x3]

Release 6.5

The calculation of Steiner's theorem was improved.

History of the VEHICLE_MOTION_1_0 Block

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

History of the VEHICLE_MOTION_CAR Block

Release 7.0

The new q_dt_Init[10x1] inport was added to set initial vehicle velocities, like initial vehicle longitudinal speed, and wheels' relative velocities with regard to the vehicle body. The new q_Init[4x1] inport was added to set initial wheels' position relative to the vehicle body.

Release 6.4

Some Simulink blocks were deleted.

There is a new inport for virtual test bench mode switch.

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

This VEHICLE_MOTION_CAR subsystem was given four new inports: F_External_CoorSys_V[x;y;z][N], Trq_External_CoorSys_V[x;y;z][Nm], Stabilization_F_External[3x6] and Stabilization_Trq_External[3x6]. Thus, external forces and torques acting on the vehicle's reference system can be taken into account in the generalized forces and torques calculation and in the mass matrix of the system.

During migration, the new external forces and torques inports are connected automatically to constant zero vectors and matrices.

Related topics

References

Vehicle Motion 287

History of the VEHICLE_MOVEMENT_INFO_1_0 Block

Release 6.2

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

History of the VEHICLE_MOVEMENT_INFO_CAR Block

Release 2020-B

New signals have been added to the ASMSignalBus.

Release 2018-B

The RotationMatrix_V_to_E[3x3] signal was added to the signal bus.

Release 2017-A

The vehicle sideslip angle calculation is set to 0 if the vehicle velocity is low.

Release 2016-A

A vehicle acceleration signal including gravitation was added to ASMSignalBus.

Release 2015-B

The "Not a number avoidance" at the calculation of the vehicle sideslip angle was removed. The calculation now uses the atan2 function from Simulink. Therefore, the division by zero is already avoided.

Release 2015-A

The ASMProperties were corrected from VEHICLE_MOVEMENT_INFO to VEHICLE_MOVEMENT_INFO_CAR.

Release 6.4	The internal signal bus was restructured. This will improve the initialization time for Simulink simulation.
Release 6.2	The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask. **Note** The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

History of the WHEEL_SPEED Block

Release 2020-B	New signals have been added to the ASMSignalBus.
Release 2013-A	The reset type for calculating wheel rotation angle was changed from *Rising* to *Level*. Thus, the correct rotation angle is also calculated if the initial vehicle speed at the start is larger than zero. The continuous integrator was replaced with a discrete one.
Release 7.3	The parameters for the dynamics brake torques are parameterizable and available at the COMMON_VEHICLE_PARAMETERS block.
Release 7.1	A Switch block to reset the integrator of the rotation angle calculation was added.
Release 7.0	The new omega_Wheel_Init[FL;FR;RL;RR][rad|s] inport was added to set the initial wheel speed from outside.
Release 6.5	A fourth sensor position with the default value [0,0,0] was added to the subsystem.
Release 6.2	The friction model was improved with additional saturation of the static torque.

The simulation step size was added to the mask of this block. Now all the parameters of a library block are parameterized via the mask.

Note

The simulation step size (SIM.StepSize parameter) is not accessible from ModelDesk. To change this parameter, edit the `sim_ini.m` function in `%PROJECT_ROOT%\Simulation.current\IniFiles` .

The calculation of the wheel speeds is now implemented as a vector signal.

History of the Vehicle Dynamics Demo Model

Release 2020-B

Drivetrain model Initial values of the integrators are now connected to nonzero values to enable correct initialization for scenarios with an initial vehicle speed. A new flag is used to indicate this feature.

SHIFT_TORQUE_SET block This block is new. It enables torque intervention during gearshift.

ENGINE_OPERATION_BASIC block The block sets the correct engine state immediately when the start button is stimulated with the *starter on* state.

Release 2019-B

Creating geometrically optimal trajectories A Python script to generate geometrically optimal trajectories is added to the ModelDesk project.

New parameter set The ModelDesk project now contains the *Luxury car* parameter set.

Reference speed calculation A new SPEED_PROFILER block is introduced to include the driving style characterization (driver's performance envelope) in the reference speed calculation. For more information, refer to Speed Profiler (ASM Environment Reference 🕮).

Release 2018-A

ASM Driver A new longitudinal driver model based on a different control strategy is introduced. The new model simulates a more realistic pedal actuation. The driver model also adapts the vehicle velocity to the road height profile. This ensures a stable vehicle guidance when driving uphill and downhill without losing road contact. Refer to Driver (ASM Environment Reference 🕮).

ASM Road The LANE_SENSOR block was replaced with a new LANE_SENSOR_DYNAMIC block. The new block has several new features, such as:

- Configurable interpretation of lane borders.
- Detection of virtual lanes on junctions.

- Configurable measurement points on the vehicle from which the distances to the lane markings are accurately measured.

The new block also lets you specify the number of lanes to the left and to the right of the current lane that are to be considered by the lane sensor.

Release 2017-B

Objects_DataStreams The MDLUserInterface (Environment/MDL_PAR) was extended by a new block to set object states for objects with multiple states, e.g., traffic lights or variable message signs.

MD_Instrumentation The MD_Instrumentation block in the MotionDesk Interface was extended by data streams to set the states for objects with multiple states, e.g., traffic lights or variable message signs.

Release 2017-A

Drivetrain stabilization The drivetrain stabilization was revised and a new concept for the information exchange between the ASM blocks and the stabilization was introduced.The previous stabilization blocks were changed to former versions and new blocks were added. The new concept offers a unified and modular interface for the information exchange with the stabilization. In addition, it introduces the possibility to delete the stabilization and thus facilitates the integration of third-party models.

The new concept was built with the following new library blocks:

- DRIVETRAIN_STABILIZATION
- FROM_STABILIZATION
- TO_STABILIZATION
- FROM_ASM
- TO_ASM

Release 2016-B

The engine and transmission soft ECUs are interconnected, so that an engine torque intervention can be realized during the gearshift process. The amount of the intervened torque can be parameterized in the TORQUE_INTERVENTION_CONTROL block from the SoftECU_Transmission.

Release 2016-A

Drivetrain demo In the Drivetrain subsystem, the CRANKSHAFT, Transmission and Final_Drive_Assembly models can now easily be copied to the model via the Open Demos buttons and drag & drop.

During the copy process, you are asked to confirm the replacement of the existing demo. If you select OK, the model is replaced and the inports and outports are automatically connected. If you select Cancel, a normal copy process is performed. This feature is only available in new ASM demos.

Brake hydraulics demo The BRAKE_HYDRAULICS model can now easily be copied to the model via the Open ASM_Brakehydraulics_lib button and drag & drop.

During the copy process, you are asked to confirm the replacement of the existing demo. If you select OK, the model is replaced and the inports and

outports are automatically connected. If you select **Cancel**, a normal copy process is performed. This feature is only available in new ASM demos.

Release 2016-A The wheel camber angle is now also animated in MotionDesk.

Release 2014-B

Suspension Compliance The new **SUSCOMP_2D_FRONT** block is used as default suspension compliance for the front axle.

The new **SUSCOMP_2D_REAR** block is used as default suspension compliance for the rear axle.

Steering The new **STEERING_3DOF_VARIABLE_RATIO** model is used as the default steering model. The **SOFT_ECU_POWERSTEERING** block was added to the **Soft ECU** subsystem. The **MDL_IN** block was extended to control the assistive torque provided to the steering model.

Drivetrain Due to the new rigid drivetrain approach, new shortcuts were added inside the **Drivetrain** subsystem.

These shortcuts open the following drivetrain-related demos for both flexible and rigid drivetrain:

- Crankshaft demos
- Transmission System demos
- Final Drive Assembly demos

Release 2014-A

The input ports of **SuspensionFront** and **SuspensionRear** were modified so that subframe blocks can be implemented more easily in these subsystems.

Release 7.0

The demo model now supports plotting of simulation data in ModelDesk. The **ModelDesk_Plotting** block was added to the **MDL_UserInterface** subsystem.

Appendix

Bibliography

List of literature

The following literature provides more details:

[Bos96] Robert Bosch GmbH: Automotive Handbook. 4th edition, 1996.

[Bre02] Brendecke T.: Virtuelle Echtzeitumgebung für Getriebesteuergeräte mit Hardware-In-the-Loop. Aachen: Shaker, 2002.

[Con99] Connair K.M. et al: Development of a Common Vehicle Model for Chassis Control Design. Society of Automotive Engineers, Inc, 1999.

[dSP99] dSPACE GmbH: dSPACE Simulator-ESP Test System. Application Note, Version 2.0, 1999.

[Fri94] Frik S.: Untersuchungen zur erforderlichen Modellkomplexität bei der Fahrdynamiksimulation. VDI Reihe 12, No. 208. VDI Verlag, 1994.

[Gen97] Genta G.: Motor Vehicle Dynamics. World Scientific Publishing Co. Pte. Ltd, 1997.

[Hal01] Halfmann C.: Adaptive Semiphysikalische Echtzeitsimulation der Kraftfahrzeugdynamik im bewegten Fahrzeug. Düsseldorf: VDI, 2001.

[Hol01] Holzmann H.: Adaptive Kraftfahrzeugdynamik Echtzeitsimulation mit Hybrid Modellen. Düsseldorf: VDI, 2001.

[Jed98] Jedrkowiak A., et al: Implementation of real-time simulation with HIL on a DEC ALPHA workstation. Robert Bosch GmbH and Gerhard Mercator Universität Duisburg, 1998.

[Kie05] Kiencke U., Nielsen L.: Automotive Control Systems - For Engine, Driveline and Vehicle. Berlin: Springer, 2005.

[Moh03] Mostafa M.: Real-Time Vehicle Dynamics Simulation for HIL-Testing of VDC-ECUs. Master's Thesis, dSPACE GmbH, 2003.

[Pac02] Pacejka H. B.: Tire and Vehicle Dynamics. Society of Automotive Engineers, Inc. 2002.

[Ril02] Rill G.: Vehicle Dynamics. Fachhochschule Regensburg, Lecture Notes, 2002.

[Ril94] Rill G.: Simulation von Kraftfahrzeugen. Braunschweig/Wiesbaden: Vieweg, 1994.

[Ril97] Rill G.: Vehicle Modeling for Real-time Applications. RBCM-J. of the Braz. SOC. Mechanical Sciences, Vol XIX, No. 2, 1997, pp.192-206.

[Sch01] Schütte H. et al: Testsysteme im Steuergeräte-Entwicklungsprozess. Automotive Electronics, 2001.

A

additional loads 284
ADDITIONAL_CP_CALCULATIONS_TRAILER block 260
AERODYNAMICS block 108
ancillary spring
  displacement 199
  forces 143
automatic gearbox 378

B

bibliography 467
Brake block 363
brake booster 112
BRAKE_DISC block 117

C

CENTRAL_DIFFERENTIAL block 44
CENTRAL_DIFFERENTIAL_RIGID block 49
clutch 4WD 24
CLUTCH block 18
CLUTCH_4WD block 24
CLUTCH_4WD_RIGID block 24, 27
CLUTCH_ENGAGEMENT_CONTROL block 95
CLUTCH_RIGID block 22
Common Program Data folder 14
COMMON_VEHICLE_PARAMETERS block 124
contact point calculation 260, 261
contact point coordinate system 275
CONTACT_POINT_CALCULATION_xy block 261
coordinate systems 275
coordinate transformation 275
CRANK_SHAFT block 36
CRANK_SHAFT_Rigid block 38

D

DIFFERENTIAL_RIGID block 49
differentials 44, 49
  rigid 49
displacement sensor 129
Documents folder 14
DRUM_BRAKE_2D block 120
DRUM_BRAKE_FRONT block 119
DRUM_BRAKE_REAR block 119

E

earth coordinate system 275
Engine Basic sublibrary 73
ENGINE_BASIC_3_0 block 374
ENGINE_OPERATION_BASIC block 84
ENINGE block 73
ESP_FAST_TORQUE_SET block 86
ESP_TORQUE_INTERVENTION_FAST_3_0 block 376
ESP_TORQUE_INTERVENTION_SLOW_3_0 block 377

F

front suspension 365
  compliance 145
  forces 138
front suspension kinematics asymmetric, 3DOF 169
front suspension kinematics symmetric 174
front suspension kinematics symmetric, 3DOF 183
front wheels relative velocities 135
FRONT_DIFFERENTIAL block 44
FRONT_DIFFERENTIAL_RIGID block 49

G

gearbox
  automatic transmission 53, 55
  manual transmission 56, 58
gearbox automatic transmission
  rigid 55
gearbox manual transmission
  rigid 58
GEARBOX_AT block 53
GEARBOX_AT_RIGID block 55
GEARBOX_MT block 56
GEARBOX_MT_RIGID block 58

I

IDLE_SPEED_CONTROL_ENGINE_BASIC block 88

L

loads 284
Local Program Data folder 14
LOCKUP_CLUTCH block 29
LOCKUP_CLUTCH_CONTROL block 97
LOCKUP_CLUTCH_RIGID block 30

M

Magic Formula tire model 266
Magic Formula tire model 6.1 268
manifold pressure 121
MANIFOLD_PRESSURE block 121
mass torque 280
MASS_FORCES_AND_TORQUES block 279
MC_PHERSON_STRUT block 179

R

rear suspension 367
  compliance 233
  forces 230
rear suspension kinematics asymmetric, 3DOF 222
rear suspension kinematics symmetric 209
rear suspension kinematics symmetric, 3DOF 222
rear wheel relative velocities 227
REAR_DIFFERENTIAL 44
REAR_DIFFERENTIAL_RIGID block 49
RELATIVE_POSITION_FRONT block 137
RELATIVE_POSITION_REAR block 229
RELATIVE_VELOCITIES_FRONT block 135
RELATIVE_VELOCITIES_REAR block 227
residual acceleration 279
RIGID_AXLE block 218
ROTATION_SWITCH block 130

S

SEMI_TRAILING_ARM block 213
SENSOR_MOTION block 281
SHAFT_xy block 64
shafts 64
SHIFT_LOCK_CONTROL block 99
SHIFT_STRATEGY block 100
SHIFT_TORQUE_SET block 89
soft ECU transmission 382
SOFT_ECU_TRANSMISSION block 378
SOFT_ECU_TRANSMISSION_10_0 block 382
SOFTECU_TRANSMISSION_SETUP block 103
Stabilization blocks 68
STABILIZATION blocks 68
stabilizer compliance 160
START_BUTTON block 80
STARTER block 40
STEERING block 237
steering tie rod 243
STEERING_VARIABLE_RATIO block 257
SUBFRAME block 131
SUSFORCES_FRONT_ANC_SPRING block 143
SUSFORCES_REAR_ANC_SPRING block 236
SUSKIN_FRONT_ASYM_3DOF block 169
SUSKIN_FRONT_ASYM_ANC_SPRING block 199
SUSKIN_FRONT_ASYM_ANC_SPRING_3DOF block 199
SUSKIN_FRONT_SYM_3DOF block 183
SUSKIN_FRONT_SYM_ANC_SPRING block 199
SUSKIN_FRONT_SYM_ANC_SPRING_3DOF block 199
SUSKIN_REAR_ASYM_ANC_SPRING block 225
SUSKIN_REAR_ASYM_ANC_SPRING_3DOF block 225
SUSKIN_REAR_SYM_ANC_SPRING block 225
SUSKIN_REAR_SYM_ANC_SPRING_3DOF block 225
Suspension block 364
Suspension Compliance Rigid Axle 148
Suspension Kinematics Rigid Axle 194
SUSPENSION_COMPLIANCE_FRONT block 145
SUSPENSION_COMPLIANCE_OPP_FRONT block 145
SUSPENSION_COMPLIANCE_REAR block 233
SUSPENSION_FORCE_KINEMATICS_FRONT block 152
SUSPENSION_FORCE_KINEMATICS_REAR block 222
SUSPENSION_FORCES_ACTIVE_x block 141
SUSPENSION_FORCES_FRONT block 138
SUSPENSION_FORCES_REAR block 230
SUSPENSION_KINEMATICS_FRONT block 174

SUSPENSION_KINEMATICS_FRONT_ASYMMETRIC block 163
SUSPENSION_KINEMATICS_REAR_ASYMMETRIC block 203
SUSPENSION_KINEMATICS_REAR_SYMMETRIC block 209
SuspensionFront block 365
SuspensionRear block 367

T

THROTTLE_2_0 block 390
TIP_SHIFT_CONTROL block 103
tire forces 262
tire model 265
  EasyToUse 271
  Magic Formula 266
  TMeasy 271
tire model 6.1
  Magic Formula 268
TIRE_FORCES_AND_TORQUES_VECTORS block 262
TIRE_MODE block 264
TIRE_MODEL_MAGIC_FORMULA block 268
TIRE_MODEL_TMEASY block 271
TMeasy
  tire model 271
TMeasy tire model 271
TORQUE_CONVERTER block 31
TORQUE_CONVERTER_RIGID block 34
TORQUE_INTERVENTION_BASIC block 90
TORQUE_INTERVENTION_CONTROL block 105
transfer gearbox 60, 62
TRANSFER_GEARBOX block 60
TRANSFER_GEARBOX_RIGID block 62

V

vehicle mass 284
vehicle movement 369
Vehicle movement block 369
vehicle reference coordinate system 275
VEHICLE_AND_WHEEL_VELOCITIES block 283
VEHICLE_MASS_AND_ADDITIONAL_LOADS block 284
VEHICLE_MOTION_CAR block 287

W

wheel coordinate system 275
wheel speed 293
WHEEL_SPEED block 293