

dSPACE Release

New Features and Migration

dSPACE Release 6.3 – November 2008

How to Contact dSPACE

Mail:	dSPACE GmbH Technologiepark 25 33100 Paderborn Germany
Tel.:	+49 5251 1638-0
Fax:	+49 5251 66529
E-mail:	info@dspace.de
Web:	http://www.dspace.com http://www.caldesk.com

How to Contact dSPACE Support

There are different ways to contact dSPACE Support:

- Visit our Web site at <http://www.dspace.com/goto?support>
- Send an e-mail or phone:
 - General Technical Support:
support@dspace.de
+49 5251 1638-941
 - SystemDesk Support:
support.systemdesk@dspace.de
+49 5251 1638-996
 - CalDesk Support:
support.caldesk@dspace.de
+49 5251 1638-363
 - TargetLink Support:
support.tl@dspace.de
+49 5251 1638-700
- Use the dSPACE Support Wizard:
 - On your dSPACE DVD at `\Diag\Tools\dSPACESupportWizard.exe`
 - Via Start – Programs – dSPACE Tools (after installation of the dSPACE software)
 - At <http://www.dspace.com/goto?supportwizard>

You can always find the latest version of the dSPACE Support Wizard here.
dSPACE recommends that you use the dSPACE Support Wizard to contact dSPACE Support.

Important Notice

This document contains proprietary information that is protected by copyright. All rights are reserved. Neither the documentation nor software may 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.

© Copyright 2008 by:
dSPACE GmbH
Technologiepark 25
33100 Paderborn
Germany

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

AutomationDesk, CalDesk, ConfigurationDesk, ControlDesk, SystemDesk and TargetLink are registered trademarks of dSPACE GmbH in the United States or other countries, or both. Other brand names or product names are trademarks or registered trademarks of their respective companies or organizations.

Contents

About This Document	7
Overview of dSPACE Release 6.3	9
<i>General Enhancements and Changes</i>	10
<i>Product Version Overview</i>	11
<i>New Product Key Features</i>	13
<i>Migrating to dSPACE Release 6.3</i>	16
AutomationDesk	17
<i>New Features of AutomationDesk 2.2</i>	17
Automotive Simulation Models (ASM)	21
ASM Diesel Exhaust Blockset	22
<i>New Blockset ASM Diesel Exhaust 1.0</i>	22
ASM Diesel InCylinder Blockset.....	23
<i>New Blockset ASM Diesel InCylinder 1.0</i>	23
ASM Drivetrain Basic Blockset.....	24
<i>Migrating to ASM Drivetrain Basic Blockset 1.2.1</i>	24
ASM Electric Components Blockset	25
<i>Migrating to ASM Electric Components Blockset 1.1</i>	25
ASM Engine Diesel Blockset.....	28
<i>Migrating to ASM Engine Diesel Blockset 1.3.1</i>	28
ASM Engine Gasoline Basic Blockset.....	29
<i>Migrating to ASM Engine Gasoline Basic Blockset 2.1.1</i>	29
ASM Environment Blockset.....	30
<i>New Features of ASM Environment Blockset 1.3.1</i>	30
ASM Gasoline InCylinder Blockset	31
<i>New Blockset ASM Gasoline InCylinder 1.0</i>	31
ASM Optimizer.....	32
<i>New ASM Optimizer 1.0</i>	32
ASM Traffic Blockset.....	33
<i>New Blockset ASM Traffic 1.1</i>	33

ASM Turbocharger Blockset	34
<i>New Features of ASM Turbocharger Blockset 1.4</i>	34
<i>Migrating to ASM Turbocharger Blockset 1.4</i>	34
ASM Vehicle Dynamics Blockset	36
<i>New Features of ASM Vehicle Dynamics Blockset 1.2.1</i>	36
<i>Migrating to ASM Vehicle Dynamics Blockset 1.2.1</i>	36
CalDesk	37
New Features of CalDesk 2.1	38
<i>New Project and Experiment Features</i>	38
<i>New Devices And Device Management Features</i>	39
<i>New Variable Management Features</i>	41
<i>New Instrument Features</i>	43
<i>New Visualization Features (Common to all Instruments)</i>	44
<i>New Measurement and Recording Features</i>	45
<i>New Data Set Management Features</i>	46
<i>New Features of the Variable Editor</i>	47
<i>New Features of the CalDesk ECU Diagnostics Module</i>	48
<i>Further Enhancements with CalDesk 2.1</i>	49
Migrating to CalDesk 2.1	50
<i>Migrating to CalDesk 2.1</i>	50
<i>How To Migrate a CalDesk 2.0 Experiment with Tunable Parameters</i>	51
ControlDesk	53
<i>New Features of ControlDesk 3.3</i>	53
dSPACE FlexRay Configuration Package	55
<i>New Features of dSPACE FlexRay Configuration Package 1.12</i>	55
RTI and RTLib	57
<i>New Features of RTI 6.2 and RTLib</i>	57
RTI Bypass Blockset	59
<i>New Features of the RTI Bypass Blockset 2.5</i>	59
RTI CAN MultiMessage Blockset	61
<i>New Features of the RTI CAN MultiMessage Blockset 2.4</i>	61
RTI RapidPro Control Unit Blockset	63
<i>New Features of the RTI RapidPro Control Unit Blockset 1.9</i>	63

SystemDesk	65
New Features of SystemDesk 2.0.....	66
<i>Support for AUTOSAR 2.1 and AUTOSAR 3.0</i>	66
<i>Measurement and Calibration</i>	66
<i>Network Communication and Bus Support</i>	69
<i>Basic Software and ECU Configuration</i>	71
<i>SystemDesk Simulation Module</i>	75
Migrating to SystemDesk 2.0.....	78
<i>Migrating to SystemDesk 2.0</i>	78
Compatibility Information	81
<i>Supported MATLAB Releases</i>	81
<i>Supported Operating Systems</i>	82
Index	85

About This Document

Contents

This document informs you about the new features of all the dSPACE software products in dSPACE Release 6.3. It also gives you an overview of software products with no or minor changes. There are instructions on migrating from older dSPACE releases, especially from older product versions, if required.

Overview of dSPACE Release 6.3

Objective Gives you an overview of the new key features in dSPACE Release 6.3, and also information about unchanged products and general instructions on migrating.

Where to go from here Information in this section

<i>General Enhancements and Changes</i>	10
<i>Product Version Overview</i>	11
<i>New Product Key Features</i>	13
<i>Migrating to dSPACE Release 6.3</i>	16

General Enhancements and Changes

Objective	The following new features concern several dSPACE products.
New documentation features	Since dSPACE Release 6.2, the <i>New Features and Migration</i> document that you are reading contains information about all the dSPACE software products. There are no more separate documents for RCP & HIL software, TargetLink, and CalDesk.
Release update	The printed user documentation is not delivered with dSPACE Release 6.3 if you receive the release as an update for your existing dSPACE release. Use the current online help, for example, dSPACE HelpDesk, to obtain information about new features, enhancements, and the current safety precautions regarding your products.

Product Version Overview

Objective

The following table is an extract from product version histories showing the product versions of the current release and of three older releases. If a product has new features, there is a link to the brief description in this document.

Product	dSPACE Release			
	6.0	6.1	6.2	6.3
AutomationDesk	2.0	2.0.1	2.1	2.2 See <i>AutomationDesk</i> on page 17.
Automotive Simulation Models	1.4	1.5	1.6	2.0 See <i>Automotive Simulation Models (ASM)</i> on page 21.
CalDesk	1.4.1	2.0	2.0	2.1 See <i>CalDesk</i> on page 37.
ConfigurationDesk	1.3	1.3.1	1.3.2	3.0
ControlDesk	3.2	3.2.1	3.2.2	3.3 See <i>ControlDesk</i> on page 53.
dSPACE Data Dictionary	1.4	1.5	1.5.1	1.5.1
dSPACE FlexRay Configuration Package	1.8	1.8	1.10	1.12 See <i>dSPACE FlexRay Configuration Package</i> on page 55.
Model Compare	----	1.0	2.0	2.0.1
ModelDesk	2.0	2.0	2.1	2.1
MotionDesk	2.1.2	2.1.2	2.1.2	2.1.2
RTI	5.6	6.0	6.1	6.2 See <i>RTI and RTLib</i> on page 57.
RTI-MP	5.3	5.4	5.5	5.6
RTIBypass Blockset	2.3	2.4	2.4.1	2.5 See <i>RTI Bypass Blockset</i> on page 59.
RTI CAN Blockset	2.6.4	2.6.5	2.6.6	2.6.7
RTI CAN MultiMessage Blockset	2.1	2.2	2.3	2.4 See <i>RTI CAN MultiMessage Blockset</i> on page 61.

Product	dSPACE Release			
	6.0	6.1	6.2	6.3
RTI LIN MultiMessage Blockset	1.4	1.5	1.5.1	1.6
RTI RapidPro Control Unit Blockset	1.6	1.7	1.8	1.9
SystemDesk	1.0	1.0	1.1	2.0 See <i>SystemDesk</i> on page 65.
TargetLink	2.2.1	2.3	3.0	3.0 2.3.1 (enhancements based on TargetLink 2.3) For information on TargetLink 2.3.1, see the ReadMe file on the dSPACE DVD.
Variable Editor	1.0	1.1	1.1	1.2 See <i>CalDesk</i> on page 37.

If you have not updated regularly, refer to the *New Features and Migration* documents for the dSPACE releases listed above for information about the new features and necessary migration steps.

New Product Key Features

Objective	This is an overview of each product's new key features. For detailed information, refer to the product-specific sections.
AutomationDesk	<p>The new key features of AutomationDesk are:</p> <ul style="list-style-type: none">■ The Sequence Builder provides a debugger for executing a sequence in debug mode.■ With the new Evaluation library, you can insert blocks in your sequence to automatically evaluate the execution results.■ Enhancements to the Main Library, the Report library and the Remote Calibration library.■ Extension to the COM API with regard to data objects, for example, you can now create and access all the data objects in the Main Library.■ Enhancements to the Version Control Interface■ Enhanced usability and performance■ In Real-Time Testing, MicroAutoBox is now supported as new real-time hardware. <p>For details on the new features, refer to <i>New Features of AutomationDesk 2.2</i> on page 17.</p>
Automotive Simulation Models (ASM)	<p>The new ASM blocksets and tools of ASM are:</p> <ul style="list-style-type: none">■ ASM Diesel Exhaust Blockset■ ASM Diesel InCylinder Blockset■ ASM Gasoline InCylinder Blockset■ ASM Optimizer■ ASM Traffic Blockset <p>For details on the new blocksets and tools, refer to <i>Automotive Simulation Models (ASM)</i> on page 21.</p>
CalDesk	For details on the new features, refer to <i>New Features of CalDesk 2.1</i> on page 38.

ControlDesk

The new key features of ControlDesk are:

- Working with experimental messages and user-defined databases in the CAN Navigator
- Enabling and disabling instruments
- Setting instruments to read-only

For details on the new features, refer to *New Features of ControlDesk 3.3* on page 53.

dSPACE FlexRay Configuration Package

The new key features of the dSPACE FlexRay Configuration Tool are:

- Specifying a frame-specific identifier for use in checksum calculation
- Specifying the frame membership of send-startup-sync frames

The new key features of dSPACE FlexRay Configuration Blockset are:

- A new RTI block for sending a wakeup pattern
- A new RTI block for reading status information of specified frames

The new FlexRay Replay Script Generator tool has the key feature:

- Generating Python scripts for replaying a FlexRay bus communication

For details on the new features, refer to *New Features of dSPACE FlexRay Configuration Package 1.12* on page 55.

RTI and RTLib

The new key features of RTI and RTLib are:

- The Single Edge Nibble Transmission (SENT) protocol can be used on a DS2211.
- RTI1401 configuration supports Real-Time Testing
- External simulation is discontinued

For details on the new features, refer to *New Features of RTI 6.2 and RTLib* on page 57.

RTI Bypass Blockset

The new key features of the RTI Bypass Blockset are:

- Support of bypassing via XCP on FlexRay
- MATLAB API to configure RTI Bypass blocks

For details on the new features, refer to *New Features of the RTI Bypass Blockset 2.5* on page 59.

RTI CAN MultiMessage Blockset

The new key feature of the RTI CAN MultiMessage Blockset is:

- Support of experimental messages for user-defined databases

For details on the new features, refer to *New Features of the RTI CAN MultiMessage Blockset 2.4* on page 61.

SystemDesk

The new key features of SystemDesk are:

- SystemDesk supports:
 - AUTOSAR Release 2.1 with Versions 2.1.2 and 2.1.4
 - AUTOSAR Release 3.0 with Versions 3.0.0 and 3.0.2
- New SystemDesk Simulation Module
 - Open- and closed-loop simulation of automotive software architectures on the PC
 - Simulation on virtual functional bus (VFB) level
 - CAN bus simulation
 - Stimulation of software component ports
 - Simulation of basic software modules such as NVRAM and DEM
 - Integration of Simulink® models as atomic software components, and including them in a simulation
- Support of measurement and calibration
 - Modeling measurement access to data elements, etc.
 - Modeling calibration access to scalar parameters.
 - Creating measurement variables and calibration parameters, and exporting the variables to an A2L file via the SystemDesk RTE Generation Module.
- Improved bus support and network communication
 - Support for FlexRay
 - Import of COM configurations
- Improved support for configuring ECUs and integrating basic software modules

For details on the new features, refer to *New Features of SystemDesk 2.0* on page 66.

Variable Editor

The new key features of Variable Editor are:

- Command line automation interface for updating variable addresses in A2L files

For details on the new features, refer to *New Features of CalDesk 2.1* on page 38.

Migrating to dSPACE Release 6.3

Objective	After you install dSPACE Release 6.3, some additional steps may be necessary.
Migration from dSPACE Release 6.2	There are no general migration steps to be done. For product-specific migration steps, refer to the product section.
Migrating from dSPACE Release 6.1 or earlier	To migrate from dSPACE Release 6.1 or earlier to dSPACE Release 6.3, you also have to perform the migration steps of the intervening dSPACE Releases. All of the required migration steps can be done with dSPACE Release 6.3 installed.
Example	<p>For example, if you want to migrate from dSPACE Release 6.0 to dSPACE Release 6.3, you have to perform the migration steps described in:</p> <ol style="list-style-type: none">1. New Features and Migration of dSPACE Release 6.02. New Features and Migration of dSPACE Release 6.13. New Features and Migration of dSPACE Release 6.24. Finally, the migration steps described above.
Previous release documents	<p>The New Features and Migration documents for previous releases are available via Internet and on the dSPACE DVD:</p> <ul style="list-style-type: none">■ Download them from http://www.dspace.com/goto?migration.■ Read them from the dSPACE DVD (see the \Doc folder). The PDF files are called <code>NewFeaturesAndMigrationxx.pdf</code>, where xx stands for the release number.



Until dSPACE Release 6.2, the new features and migration steps for RCP & HIL software, CalDesk and TargetLink were described in separate documents.


For the new features and migration documents of previous CalDesk versions, refer to <http://www.dspace.com/goto?VersionsCal>.


AutomationDesk

New Features of AutomationDesk 2.2

AutomationDesk 2.2

Sequence Debugger AutomationDesk now provides a debugger for executing a sequence in debug mode. You can set breakpoints at each automation block and execute the sequence step by step to examine the current values of the sequence.

For further information, refer to *Executing Sequences in Debug Mode* ( *AutomationDesk Guide*).

New Evaluation library The Evaluation library provides automation blocks for preparing specific results from your sequence as evaluation signals. These signals can be modified by various arithmetic operations. You can compare an evaluation signal with a reference signal to decide whether the execution result is correct. The evaluation results are automatically added to the report. Intermediate signals can be explicitly added to the report as plots. For further information, refer to *Evaluation* ( *AutomationDesk Library Reference*).


Enhancements to the Main Library

- There is a new block for printing messages to the Output Viewer. A setting in the General Properties dialog allows you to deactivate the output of all Print blocks in your session.
- There are new blocks for working with the contents of the Output Viewer. You can copy the contents to a String data object and clear the Output Viewer.

For further information, refer to *Main Library* ( *AutomationDesk Library Reference*).


Enhancements to the Remote Calibration library

- There is a new block for switching between the working page and the reference page of a calibration system.
- There is a new block for exiting a calibration session explicitly.

For further information, refer to *Remote Calibration* ( *AutomationDesk Library Reference*).

Enhancements to the Report library

- There is a new block for adding the image of the sequence to the report.
- There is a new block for creating subreports for specific sections within your sequence.

For further information, refer to *Report* ( *AutomationDesk Library Reference*).

Enhancements to the COM API You can now execute TestSequences from the Test Framework.

You can now create and access the following data objects using the COM API:

- All the data objects in the Main Library: not only the main data types, but also List, Dictionary, Tuple, Variant, Condition, and DataContainer.
- The data objects in the Platform Access library: Platform, Variable, CaptureResult
- The data objects in the Remote Calibration (COM) library: System, Project, LogicalLink, Characteristic, Collector, and Measurement.
- The Color data object in the Report library.

For further information, refer to *AutomationDesk API Reference*.

Enhancements to the Version Control Interface The Version Control Interface now also supports *Subversion* from Tigris.org (Open Source).

Enhancements to usability There are some improvements that make working with AutomationDesk more convenient:

- The recent file list contains ten entries instead of four.
- The Change Children Order dialog contains a function for sorting all the entries in ascending or descending order.
- New color schemes are available, for example, for users who prefer a subtle look.

- The Report properties contain an option to create separate subreports for each sequence executed in the project. In some cases, this boosts report generation performance.

Real-Time Testing 1.5

New hardware support Since Real-Time Testing 1.5, MicroAutoBox is supported as new real-time hardware.


Automotive Simulation Models (ASM)

Where to go from here

Information in this section

<i>ASM Diesel Exhaust Blockset</i>	22
<i>ASM Diesel InCylinder Blockset</i>	23
<i>ASM Drivetrain Basic Blockset</i>	24
<i>ASM Electric Components Blockset</i>	25
<i>ASM Engine Diesel Blockset</i>	28
<i>ASM Engine Gasoline Basic Blockset</i>	29
<i>ASM Environment Blockset</i>	30
<i>ASM Gasoline InCylinder Blockset</i>	31
<i>ASM Optimizer</i>	32
<i>ASM Traffic Blockset</i>	33
<i>ASM Turbocharger Blockset</i>	34
<i>ASM Vehicle Dynamics Blockset</i>	36

Information in other sections

<i>Migrating ASM Models</i> ( <i>ASM User Guide</i>) Provides general information on the migration process of ASM models.
--

ASM Diesel Exhaust Blockset

New Blockset ASM Diesel Exhaust 1.0

New blockset

ASM Diesel Exhaust is an add-on to ASM EngineDiesel or ASMEngineDiesel InCylinder.

Exhaust systems containing Diesel Oxidation Catalyst (DOC) and Diesel Particulate Filter (DPF) are popular forms of aftertreatment in diesel engines. A DOC reduces CO and unburned hydrocarbons, and a DPF reduces the particulates (soot). Some recent systems use Selective Catalytic Reduction (SCR) technology to reduce nitrogen oxides. SCR has the advantage of running continuously without affecting engine operation (no increase in fuel consumption) and is usually installed either without or downstream of a DPF.

Models for these exhaust systems are provided by ASM Diesel Exhaust.



To use ASM Diesel Exhaust with SCR, you need at least a DS1005GX.

ASM Diesel InCylinder Blockset

New Blockset ASM Diesel InCylinder 1.0

New blockset

The Diesel Engine InCylinder Simulation Package is an open Simulink model for developing and testing electronic control units (ECUs). The model performs torque generation by calculating the in-cylinder combustion process according to a phenomenological approach. In-cylinder pressure is simulated in real time by means of a zero-dimensional thermodynamic approach. The combustion process simulation can handle multiple injection patterns such as pre-, main, and post-injection. The gas dynamical behavior of the air path and the exhaust path is implemented as a single-zone system with the manifold pressure, temperature, and mass calculation. The intake and exhaust valves are modeled as isentropic orifices. The ASM diesel engine in-cylinder model can be used in combination with real controllers in a hardware-in-the-loop environment. A DS1006 processor is mandatory for real-time simulation.

ASM Drivetrain Basic Blockset

Migrating to ASM Drivetrain Basic Blockset 1.2.1

General changes

Simulation Step Size parameter The parameterization of the Simulation Step Size parameter has been moved from the workspace to the mask. Now all the parameters of a library block are parameterized via the mask. This applies to the following blocks:

- GEARBOX_AT
- GEARBOX_MT

ASM Electric Components Blockset

Migrating to ASM Electric Components Blockset 1.1

Renamed and removed mask parameters in the BATTERY block

Renamed parameters The following parameters have been renamed:

Old Parameter Name	New Parameter Name
Map_soc_CellFactor	Map_Factor_CellVoltage
Const_C_Block_nom	Const_Capacity_nom
Map_Kn_current_temp	Map_Factor_Capacity
Const_Number_of_Cells	Const_n_Cells
Const_mbat	Const_m_Batt
Const_thermalCapacity	Const_Capacity_Thermal
Map_R_Batt_Temp_Discharge	Map_R_Discharge
Const_Lbat_Discharge	Const_L_Discharge
Map_R_Batt_Temp_Charge	Map_R_Charge
Const_Lbat_Charge	Const_L_Charge
Const_Tn	Const_Temp_nom
Const_I0	Const_I_Loss_nom
Const_A_surface	Const_A_Surface
Const_eps	Const_Factor_ThermalEmission
Const_cMr	Const_c_MainReaction

The renamed parameters are not changed in the initialization M-file (Simulation.current\AutomotiveElectricalSystem\IniFiles\asm_automotiveelectricalsystem_ini.m OR Simulation.current\ElectricDriveFullHybrid\IniFiles\asm_electricdriveclosedloop_ini.m) automatically. The values of the renamed parameters are therefore not initialized during the initialization procedure. To update the initialization, set the new parameter names in the initialization M-file manually. Replace the following strings in the initialization M-file:

MDL.ElectricComponents.AutomotiveComponents.Battery.Map_soc_CellFactor

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Map_Factor_CellVoltage

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_C_Block_nom

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_Capacity_nom

MDL.ElectricComponents.AutomotiveComponents.Battery.Map_Kn_current_temp

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Map_Factor_Capacity

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_Number_of_Cells

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_n_Cells

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_mbat

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_m_Batt

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_thermalCapacity

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_Capacity_Thermal

MDL.ElectricComponents.AutomotiveComponents.Battery.Map_R_Batt_Temp_Discharge

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Map_R_Discharge

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_Lbat_Discharge

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_L_Discharge

MDL.ElectricComponents.AutomotiveComponents.Battery.Map_R_Batt_Temp_Charge

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Map_R_Charge

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_Lbat_Charge

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_L_Charge

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_Tn

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_Temp_nom

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_I0

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_I_Loss_nom

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_A_surface

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_A_Surface

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_eps

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_Factor_ThermalEmission

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_cMr

rename as

MDL.ElectricComponents.AutomotiveComponents.Battery.Const_c_MainReaction

Removed parameters The Const_Boltzmann parameter has been removed.

ASM Engine Diesel Blockset

Migrating to ASM Engine Diesel Blockset 1.3.1

SOFT_ECU_DIESEL block

The parameter which describes the relative portions of the injections (Const_inj_rel) is not accepted as a column (original) and as a row vector (new). This adaptation is required due to adaptations in ASM Parameterization.

EXHAUST_MANIFOLD block

Because the ForwardEuler S-function can now be used for V-engines, too, problems can occur with the signal width when calling the **Update Diagram** command (**Ctrl+D**) for models generated in Release 5.4 or earlier. To avoid these problems, a fixed width for the EngOP[*rpm*][*mm3/cyc*] input has been specified. A **SignalSpecification** block has been added.

ASM Engine Gasoline Basic Blockset

Migrating to ASM Engine Gasoline Basic Blockset 2.1.1

WallFilm

A reset feature has been added to the integrator for calculating the wall film fuel mass. This reset is enabled if the wall film model is switched on by an external switch.

ASM Environment Blockset

New Features of ASM Environment Blockset 1.3.1

Enhanced ROAD block

In some cases, task overruns occurred in driving on the road. The clothoid segment calculation was modified to avoid these overruns.

**MANEUVER_SCHEDULER
block**

Tables and MAT files are now interpreted correctly when user maneuvers are used.

ASM Gasoline InCylinder Blockset

New Blockset ASM Gasoline InCylinder 1.0

New blockset

The Gasoline Engine InCylinder Simulation Package is an open Simulink model for developing and testing electronic control units (ECUs). The model performs torque generation by calculating the in-cylinder combustion process according to a phenomenological approach. In-cylinder pressure is simulated in real time by means of a zero-dimensional thermodynamic approach. The combustion process simulation can handle port and direct injection systems. The gas dynamical behavior of the air path and the exhaust path is implemented as a single-zone system with the manifold pressure, temperature, and mass calculation. The intake and exhaust valves are modeled as isentropic orifices. Variable valve timing and lift can be handled. The ASM gasoline engine in-cylinder model can be used in combination with real controllers in a hardware-in-the-loop environment. A DS1006 processor is mandatory for real-time simulation.

ASM Optimizer

New ASM Optimizer 1.0

New tool

ASM Optimizer is for adjusting diesel and gasoline engine in-cylinder models to the in-cylinder engine pressure measurement curves. ASM Optimizer finds the optimal model parameters for the gas exchange and the combustion process by comparing the simulation results with the measurement data.

ASM Traffic Blockset

New Blockset ASM Traffic 1.1

New blockset

ASM Traffic is an add-on to ASM VehicleDynamics. It simulates the movement of fellow vehicles for testing driver assistant systems especially ACC. The test vehicle can be equipped with multiple sensors for object detection. ASM Traffic was previously provided as an engineering solution. Automatic migration of previous versions of ASM Traffic is not supported. Contact ASM Support.

ASM Turbocharger Blockset

Where to go from here

Information in this section

<i>New Features of ASM Turbocharger Blockset 1.4</i>	34
<i>Migrating to ASM Turbocharger Blockset 1.4</i>	34

New Features of ASM Turbocharger Blockset 1.4

New support for ASM Packages

The turbocharger library was enhanced and now supports the following ASM Packages:

- Diesel Engine Simulation Package
- Gasoline Engine Simulation Package
- Diesel Engine InCylinder Simulation Package
- Gasoline Engine InCylinder Simulation Package

When the turbocharger demo model is copied, the target model has to be selected in a graphical user interface.

Enhanced support of flow calculation

The TURBINE, TURBINE_SAEJ922, and WASTEGATE blocks now support the calculation of the flow of different mass fractions (fuel, air, and exhaust) and the enthalpy flow, which is required by the InCylinder models.

Migrating to ASM Turbocharger Blockset 1.4

Mass components and enthalpy flow

The InCylinder models require the mass flow subdivided into the mass components fuel, air, and exhaust. The enthalpy flow is also required.

New input ports have therefore been added:

- With the specific enthalpy:
 - $h_ExhMan[J/kg]$

- With the mass fractions of fuel, air, and exhaust in the exhaust manifold:
 - XsiFuel_ExhMan[0_1]
 - XsiAir_ExhMan[0_1]
 - XsiExh_ExhMan[0_1]

New output ports have been created for:

- The enthalpy flow:
 - Hdot_...[J/s]
- The mass flow of fuel, air, and exhaust:
 - mdot_Fuel...[kg/s]
 - mdot_Air...[kg/s]
 - mdot_Exh...[kg/s]

These new ports are not required in the mean value engine model. They are therefore connected with a dummy value or terminated during the migration process.

These changes affect the following blocks:

- TURBINE
- WASTEGATE
- TURBINE_SAEJ922

These changes also affect the ports of the turbocharger demo models. When the new turbocharger demo models are copied for the first time to an ASM Engine Diesel or the ASM Engine Gasoline model that was created with Release 6.2 or earlier, the ports have to be terminated by hand. Use the ASM Engine Diesel or the ASM Engine Gasoline demo model of Release 6.3. as an example.

SHAFT block

The saturation of the shaft speed is now performed by the Integrator block and not by an additional saturation block.

Maps_TC block

The control signal for the turbocharger (Ctrl_TC[0_1]) has been added to the ASMSignalBus block.

ASM Vehicle Dynamics Blockset

Where to go from here

Information in this section

<i>New Features of ASM Vehicle Dynamics Blockset 1.2.1</i>	36
<i>Migrating to ASM Vehicle Dynamics Blockset 1.2.1</i>	36

New Features of ASM Vehicle Dynamics Blockset 1.2.1

New parameter sets

The following new parameter sets are now available for the Vehicle Dynamics demo model in ModelDesk:

- SmallCar
- Van
- SUV

Model Ini files for these variants are not provided, but can be generated in ModelDesk.

TIRE_MODEL_MAGIC_FORMULA block

The right tire behavior (force F_y , torque M_z and torque M_x) is now mirrored to the left tire behavior. Torque M_y in the Magic Formula tire model is now calculated with the correct sign.

CONTACT_POINT_CALCULATION block

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.

Migrating to ASM Vehicle Dynamics Blockset 1.2.1

CONTACT_POINT_CALCULATION block

A new inport for the reset signal has been added.

CalDesk

Where to go from here

Information in this section

<i>New Features of CalDesk 2.1</i>	38
<i>Migrating to CalDesk 2.1</i>	50

New Features of CalDesk 2.1

Where to go from here

Information in this section

<i>New Project and Experiment Features</i>	38
<i>New Devices And Device Management Features</i>	39
<i>New Variable Management Features</i>	41
<i>New Instrument Features</i>	43
<i>New Visualization Features (Common to all Instruments)</i>	44
<i>New Measurement and Recording Features</i>	45
<i>New Data Set Management Features</i>	46
<i>New Features of the Variable Editor</i>	47
<i>New Features of the CalDesk ECU Diagnostics Module</i>	48
<i>Further Enhancements with CalDesk 2.1</i>	49

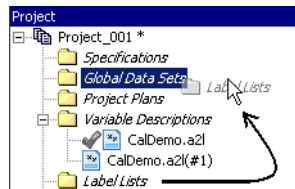
New Project and Experiment Features

Information in this topic

Sorting items on page 38
Global devices in a separate folder on page 39

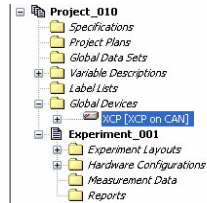
Sorting items

In the **Project Manager**, you can sort items that are on the same hierarchy tree level by dragging them to new positions.



Global devices in a separate folder


In the project structure, global devices are now shown in a separate folder.



New Devices And Device Management Features

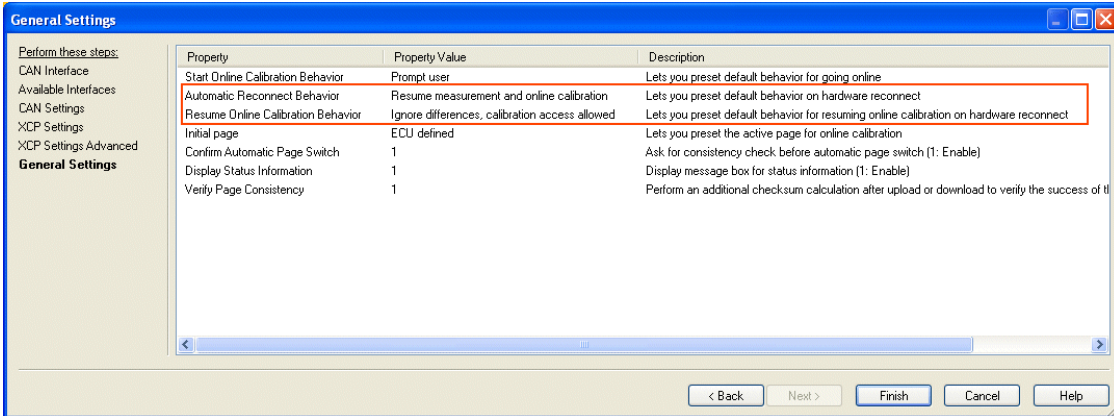
XCP on FlexRay

CalDesk 2.1 provides the new *XCP on FlexRay device*. This lets you access an ECU with XCP connected to the CalDesk PC via FlexRay. You can use the XCP on FlexRay device to access the ECU for measurement and calibration purposes via XCP (*Universal Measurement and Calibration Protocol*). To configure the device, you can specify an A2L file containing a link to the associated FIBEX file. CalDesk then adopts the FlexRay configuration settings contained in the FIBEX file automatically.

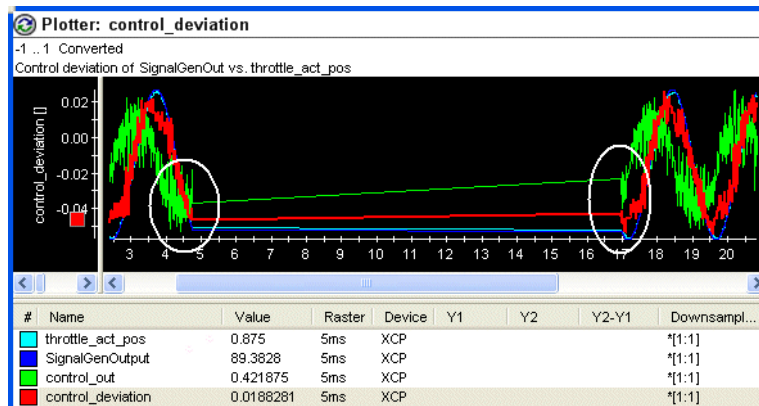
For configuration instructions, refer to *How to Configure an XCP on FlexRay Device* ( *CalDesk Calibration Guide*).


Resuming measurement automatically when switching the ECU off and on again

CalDesk provides the *Automatic Reconnect* feature for automatically reconnecting to device hardware, for example, when the ignition is turned off and on, or when the physical connection between the CalDesk PC and the ECU is temporarily interrupted. For example, you can use CalDesk for measuring correlated data from both the ECU and a simulation platform in HIL test scenarios that require ECU off/on transitions.



The illustration below shows a measurement in a CalDesk Plotter. The connection to the ECU is interrupted temporarily during the measurement. The Plotter interpolates the measurement linearly for periods of time for which no measurement data is available. After the ECU is switched on again, CalDesk automatically reconnects to the ECU, and measurement continues.



For details on specifying the Automatic Reconnect feature, refer to *Configure Device* ( *CalDesk Calibration Reference*).


dSPACE ECU Flash Programming Tool as a self-contained tool

You can now install the dSPACE ECU Flash Programming Tool without having to install CalDesk.

Opening the dSPACE ECU Flash Programming Tool via device context menu

You can now open the dSPACE ECU Flash Programming Tool directly via the context menu of the calibration devices supported by the tool:

- DCI-GME1
- DCI-GSI1
- XCP on CAN
- XCP on USB

Refer to *How to Start the dSPACE ECU Flash Programming Tool* ( *CalDesk Calibration Guide*).

New Variable Management Features

Sorting labels in the label list

You can sort the labels in the label list manually by moving them up and down and automatically sort them alphabetically.


Label list with connected variables

You can save a label list containing the names of the variables that are connected to an instrument.

Using wildcards for buffered search

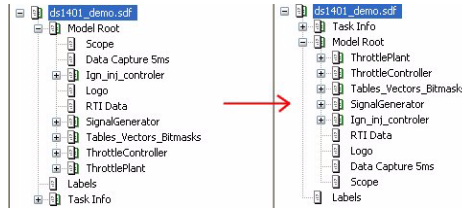
When you are searching for variables, you can use the wildcards * and ? in the keystroke sequence of a buffered search.



For details, refer to *How to Search for a Variable Using Buffered Search* ( *CalDesk Calibration Guide*).

Display subsystems first

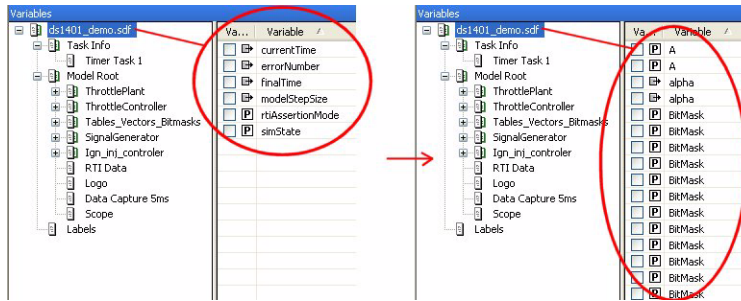
Lets you specify to display subsystems (items with a +–symbol) in the hierarchy tree first.



For details, refer to *Variables Page* ( *CalDesk Calibration Reference*).

Collect variables from subgroups in root

You can specify to display all the variables of a variable description in the variable list when you select the root node in the hierarchy tree.

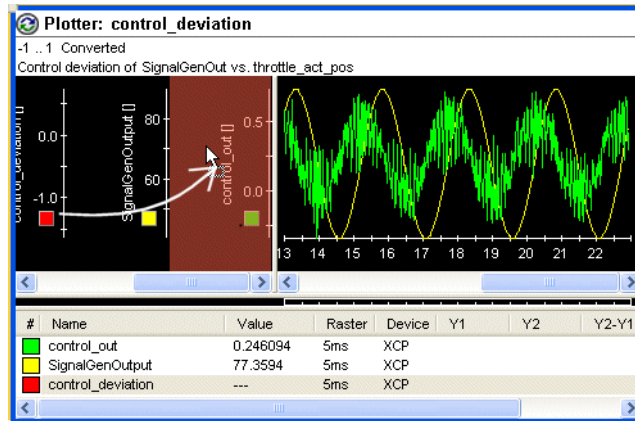


For details, refer to *Variables Page* ( *CalDesk Calibration Reference*).

New Instrument Features

Moving signals in a Plotter

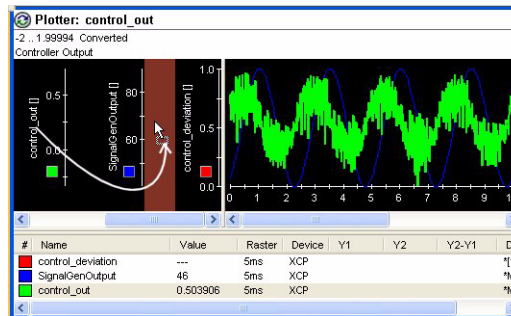
You can move or copy a signal to another axis within the Plotter via drag and drop.



For details, refer to *Basics of Handling the Plotter* ( CalDesk Calibration Guide).

Moving y-axes in a Plotter

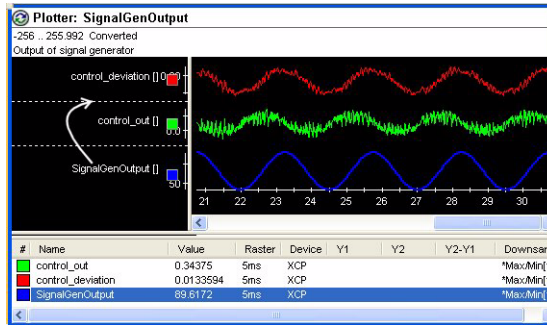
You can move a y-axis in a Plotter by dragging and dropping its label.



For details, refer to *Basics of Handling the Plotter* ( CalDesk Calibration Guide).

Changing the stack sequence

If you have selected the *stacked* view for the Plotter's y-axes, you can change the stack sequence by dragging an axis label up or down.



For details, refer to *Analyzing and Postprocessing Measured and Recorded Data in CalDesk* (*CalDesk Calibration Guide*).

New Visualization Features (Common to all Instruments)

Using the same layouts for rapid control prototyping and ECU calibration

You can reuse layouts with instruments that have connections to variables of an SDF file (prototyping device) in an ECU calibration project (calibration device with an A2L file). For details, refer to *Advanced: How to Export/Import Layouts* (*CalDesk Calibration Guide*).

Improved handling of fixed content size

You can specify fixed sizes for your layouts to ensure that all instruments are displayed without scrolling. For details, refer to *Variables Page* (*CalDesk Calibration Reference*).

Dedicated keyboard shortcuts for individual layouts

You can activate individual layouts (the first nine layouts counted from the left) via dedicated keyboard shortcuts (**ALT+1...9**).



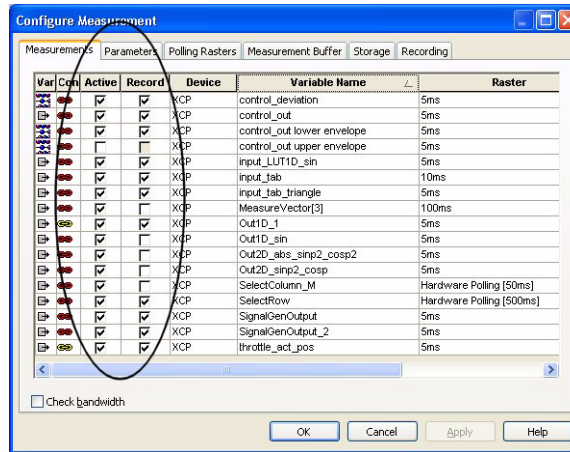
Improved shortcuts for deleting instruments and variables in instruments


The **Delete** keyboard shortcut deletes selected variable(s) in an instrument. If the last variable has been removed, **Delete** removes the empty instrument. You can remove an instrument with all its variables in one step by pressing **Ctrl+Delete**.

New Measurement and Recording Features

Option to activate/deactivate recording for selected variables

The measurement signal list has checkboxes for activating the measurement and recording of variables. Variables with measurement activated are included in measurement runs, and you can then activate/deactivate recording for them. For example, you can view a lot of signals in instruments and record only a few of them.



For details refer to *How to Activate Variables for Measurement and Recording* ( CalDesk Calibration Guide).

New Data Set Management Features

Export and Import of data sets in CDFX format

You can export and import data sets in the CDFX (ASAM Calibration Data Format V2.0) file format.

Comparing complex parameters


You can now view the differences in a complex value, such as a curve or map. Cells with differences are highlighted. The comparison value is shown in each cell's tooltip.

Parameter Name	CalDemo XCP.mmf	CalDemo XCP-1.m	New Data Set
abs_sinp2_cosp2_...	
airmass_x_table	
DummyAirMass	0.0048828125	0.0048828125	
DummyOmega	20	20	
f_Kd_1	0.00099992752075	0.00100111981364	1.396240234 1.745361328 2
f_Kd_2	0.00099992752075	0.00099992752075	0.984375 0.984375
f_Kl_1	1	0.9765625	0.9453125 0.9765625
f_Kl_2	3	0.8046875	0.828125 0.8046875
f_Kp_1	0.10000610351562	0.10010920808437	0.65625 0.65625
f_Kp_2	0.29998779296875	0.29998779296875	0.4296875 0.4296875
Fac_F32_gain	1	1	0.1640625 0.1640625
Fac_I16_gain	1	1	
Fac_I32_gain	1	1	
Fac_B_gain	1	1	


Comments for comparison reports

When you generate a comparison report, you can add a description text to it.

Reimporting calibration data to a Simulink® or TargetLink model

CalDesk 2.1 provides the command line tool `DSExp2M` to reimport a calibrated data set (with its current values) to a Simulink® or a TargetLink model. For details refer to *How to Export a Data Set* ( *CalDesk Calibration Guide*).

Saving new application image under a different name

When you create a new application image, CalDesk now lets you save the SDF file under a different name. The associated MAP, TRC and PPC files are automatically named after the application image file. For details, refer to *How to Create an Application Image* ( *CalDesk Calibration Guide*).

New Features of the Variable Editor

Command line interface

The Variable Editor now provides a command line interface that lets you access several Variable Editor's functions without using the graphical user interface. For example, you can update the variable addresses of an A2L file via map files without a Variable Editor project. You can use the `variableeditor` command with options to let the Variable Editor carry out specific tasks.

```

C:\WINDOWS\system32\cmd.exe
C:\>variableeditor -h

dSPACE Variable Editor, Us 1.2 <C> 2008 by dSPACE GmbH
Usage: VariableEditor [Options]

[File.a2l]
-P <file.vep>           - Opens the Variable Editor Project file.vep
-I <Input A2L-File>    - Defines the input A2L file.
-O <Output A2L-File>   - Defines the output A2L file.
-D <A2L output path>  - Defines the A2L output path. For multiple A2L
project only.
-M <MAP-File>@<Parser-Type> - Defines the MAP-file(s) and the corresponding
MAP-file parser type.
-U                     - Performs an address update
-?                     - Displays this help screen.

Examples:
VariableEditor <A2Lfile.a2l>
Opens the A2L file <file.a2l> using the Variable Editor.
VariableEditor -P <file.vep>
Opens the Variable Editor project <file.vep>.
VariableEditor -P <UEProject.vep> -O <OutputA2L.a2l> -U
Opens the Editor project <file.vep>, performs an address update, and exports
the updated A2L file <Output.a2l>
VariableEditor -I <Input.a2l> -O <Output.a2l> -M <file.map>@<Parser-Type> -U
Opens the A2L File <Input.a2l>, imports the MAP file <file.map> and assigns
all the symbols by name, performs an address update, and exports the updated
A2L file <Output.a2l>

Supported MAP file parser types:
C16x_Tasking, CME555_DIAB, CME555_GHS, CST, ELF, H8_Hitachi, IEEF695, M32R_Gaio.

```

For details, refer to *Using the Command Line Interface* (📖 *Variable Editor Guide*).

Converting map file symbols to writable measurements automatically

If you create variables using map file symbols, the map file symbols are converted to writable measurements automatically by default.

New Features of the CalDesk ECU Diagnostics Module

Using several services for reading the ECU's fault memory

You can now use different services for reading the fault memory. This allows you to read different kinds of fault memory entries. CalDesk lets you select the read service to be used for each instance of the Fault Memory instrument.

Fault Memory Instrument

Status	Logical Link	Read Service	#DTCs	Read DTC Data	Update Rate [s]	Last Update
	DemoECU	DS_ReadDTCInformati	3	<input checked="" type="checkbox"/>	off	11:00:12

Logical Link	DTC #	Description
DemoECU	P0460	Fuel Level Sensor Circuit Malfunction
DemoECU	P0325	Knock Sensor 1 Circuit Malfunction
DemoECU	P0120	Throttle Position Sensor Circuit Malfunction

Parameter	Value
TestFailed	1
TestFailedThisMonitoringCycle	0
PendingDTC	0
ConfirmedDTC	1
TestNotCompleteSinceLastClear	0
TestFailedSinceLastClear	0
TestNotCompletedThisMonitoringCycle	1
WarningIndicatorRequested	0
DTC Extended Data Record Number	1
EngineCoolantTemperature	83.0
ThrottlePosition	10.2
EngineSpeed	3050.0
ManifoldAbsolutePressure	1.8

Update Selected LLS Clear Selected DTC Lock Update
Update All LLS Clear All DTCs Save to File


If you update the information on a logical link in one Fault Memory instrument, the information is also updated in all other Fault Memory instruments using the same read service that references the same logical link. This means that, multiple Fault Memory instruments are synchronized as long as the same read service is selected for them.

The different services must be specified in an XML configuration file. To make them available in CalDesk, the XML configuration file must be included in the ODX database during device configuration.

For details, refer to *Managing ECU Diagnostics Tasks* ( *CalDesk Calibration Guide*).

Further Enhancements with CalDesk 2.1

Full screen mode

CalDesk 2.1 provides a full screen mode which exploits the whole screen area for instrumentation. For details, refer to *Full Screen Mode* ( *CalDesk Calibration Reference*).

Migrating to CalDesk 2.1

Where to go from here

Information in this section

<i>Migrating to CalDesk 2.1</i>	50
<i>How To Migrate a CalDesk 2.0 Experiment with Tunable Parameters</i>	51

Migrating to CalDesk 2.1

To migrate to CalDesk 2.1 and reuse existing experiments, you may have to carry out additional migration steps. The table below shows the cases in which this is necessary.

	... To Version						
	1.2.1	1.2.2	1.3.0	1.4.0	1.4.1	2.0	2.1
From Version ...							
1.2.0 ...	No	No	No	Yes	Yes	Yes	Yes
1.2.1 ...	–	No	No	Yes	Yes	Yes	Yes
1.2.2 ...	–	–	No	Yes	Yes	Yes	Yes
1.3.0 ...	–	–	–	Yes	Yes	Yes	Yes
1.4.0 ...	–	–	–	–	No	Yes	Yes
1.4.1 ...	–	–	–	–	–	Yes	Yes
2.0 ...	–	–	–	–	–	–	Yes ¹⁾

¹⁾ Refer to *How To Migrate a CalDesk 2.0 Experiment with Tunable Parameters* on page 51.

For the migration steps, refer to <http://www.dspace.com/goto?VersionsCal>.

How To Migrate a CalDesk 2.0 Experiment with Tunable Parameters

To reuse a CalDesk 2.0 experiment that contains maps or curves that are defined as tunable parameters, you must renew all the connections of the affected parameters concerned.

Affected experiments

You must migrate an experiment if it fulfills all the following conditions:

- The experiment was created with CalDesk 2.0
- The experiment contains one or more devices with SDF variable description (DS1005, MicroAutoBox)
- The experiment contains connections to maps or curves that are defined as tunable parameters

Changed parameter names

The naming of maps and curves that are defined as tunable parameters has changed from CalDesk 2.0 to CalDesk 2.1. If you load an experiment that has such parameters, CalDesk cannot reestablish connections to tunable maps or curves.

Method

To migrate a CalDesk 2.0 experiment with tunable parameters

- 1 Open CalDesk 2.1.
- 2 Load the CalDesk 2.0 experiment to be reused.
- 3 Renew all the connections of maps and curves that are defined as tunable parameters (for example, create new connections to instruments or add the tunable parameters to the measurement signal list again).
- 4 Save the experiment.


Result


You can reuse the experiment with tunable parameters.

ControlDesk

New Features of ControlDesk 3.3

CAN Navigator

Dynamic configuration of CAN messages During run time, you can configure CAN messages which were not specified in the real-time model and activate them on the real-time platform. The number of these experimental messages is specified in the real-time model but they are independent of the database file specified for the RTICANMM MainBlock. Refer to *Working With Experimental Messages* ( *ControlDesk Experiment Guide*).

Working with user-defined databases A user-defined database (UDDB) is based on a DBC file specified for the RTICANMM MainBlock but its messages are specified in the CAN Navigator. Refer to *Working with User-Defined Databases* ( *ControlDesk Experiment Guide*).

ControlDesk

Enable/disable instruments It can be useful to disable currently inactive instruments when a simulation model with dual or multiple operation mode is used. For example, the instruments for the manual transmission can be disabled in the automatic transmission mode. It makes experimenting easier and more intuitive and prevents operating errors. Disabled instruments are grayed out and their values and checkbox selections are hidden.

You can set this feature either interactively via the instruments' property pages or automatically via Automation.


Make input instruments read-only You can make input instruments read-only to prevent them from being unintentionally modified. The instruments are dimmed but their values are visible.


You can set this feature either interactively via the instruments' property pages or automatically via Automation.

dSPACE FlexRay Configuration Package

New Features of dSPACE FlexRay Configuration Package 1.12

FlexRay Configuration Tool


Improved checksum calculation You can now specify identifiers for frames which can be used in checksum calculation. The identifiers are specified in a frame ID file and delivered to the checksum algorithm via a structure variable. For details, refer to *Using User-Defined Checksum Algorithms* ( *FlexRay Configuration Tool Guide*).

Monitoring send-startup-sync frames You can assign send-startup-sync frames to the Global RX Pool frame membership. This frame membership is used for receiving frames, but if necessary it could also be used to send a startup-sync frame if no more controllers are available and no other ECU in the network is sending second startup-sync frame when one is needed. For details, refer to *Building Frame Membership Groups* ( *FlexRay Configuration Tool Guide*).

XCP on FlexRay You can define an XCP master and configure XCP frames for the RTI Bypass Blockset. Refer to *New Features of the RTI Bypass Blockset 2.5* on page 59.

RTI FlexRay Configuration Blockset

RTIFLEXRAYCONFIG TX WAKEUP block You can specify the basic settings and trigger conditions for transmitting wakeup patterns on the FlexRay bus. For details, refer to *RTIFLEXRAYCONFIG TX WAKEUP* ( *FlexRay Configuration RTI Reference*).

RTIFLEXRAYCONFIG RX FRAME STATUS block You can monitor states of a specified FlexRay frame. For details, refer to *RTIFLEXRAYCONFIG RX FRAME STATUS* ( *FlexRay Configuration RTI Reference*).

FlexRay Replay Script Generator

The *FlexRay Replay Script Generator* is a new dSPACE software tool. It can be used to generate Python scripts for Real-Time Testing which replay the communication of a FlexRay bus. The communication must have been logged beforehand. For information on the whole workflow, refer to the *FlexRay Replay Script Generator Guide*.

RTI and RTLib


New Features of RTI 6.2 and RTLib

DS2211

The DS2211 RTLib functions were improved:

Supporting the SENT protocol SENT (Single Edge Nibble Transmission) is a protocol used between sensors and ECUs. It is used to transmit data of high resolution (10 bit or more) sensors as an alternative to an analog interface. You can implement a SENT receiver and SENT transmitter on a DS2211 using RTLib functions.

- SENT receiver:
 - Four independent channels (digital inputs 1 to 4).
 - Either all received messages are transferred to the real-time model or only the latest one.
 - Information about the message clock are transferred to the real-time model.
- SENT transmitter:
 - Five independent channels (digital outputs 1 to 5).
 - All nibbles are set by the user.
 - The checksum must be calculated within the real-time model. So it is possible to manipulate the checksum.
 - Several messages can be stored within a FIFO, so that a seamless transmission of several different messages is possible.
 - If only one message is in the FIFO, it will be repeated (optional).
 - The message clock can be set from the real-time model.

For details, refer to *Single Edge Nibble Transmission (SENT)* ( *DS2211 RTLib Reference*).

This feature is not supported by RTI.

RTI1401 configuration supports Real-Time Testing

In Real-Time Testing, MicroAutoBox is now supported as new real-time hardware. The RTI1401 configuration therefore provides the **Enable Real-Time Testing Support** option.

External simulation is discontinued

RTI/RTI MP support for the Simulink and Real-Time Workshop data exchange interface "External Mode" (External Simulation) is discontinued with this dSPACE Release.

RTI Bypass Blockset

New Features of the RTI Bypass Blockset 2.5

Support of bypassing via XCP on FlexRay

The RTI Bypass Blockset now supports service-based bypassing via XCP on FlexRay.

Bypassing via XCP on FlexRay is possible with the following dSPACE hardware:

- Modular system based on a DS1005 or DS1006 with DS4501 IP Carrier Board or DS4505 FlexRay Interface Board with FlexRay modules
- MicroAutoBox 1401/1505/1506 or 1401/1505/1507 with FlexRay modules

New configuration options for XCP interfaces

DAQ list priority specification The RTI Bypass Blockset now lets you configure priorities for XCP-based bypass interfaces. You can specify the DAQ list priority specification for each Read and Write block. Besides the default setting with internally defined standard priorities, you can specify to use the priority according to the service instance selected in the block, or enter a DAQ list priority manually.

Automation interface for RTI Bypass block configuration

RTI Bypass Blockset now comes with an automation programming interface (API) that allows you to configure and parameterize the blocks of the RTI Bypass Blockset. The API specifications are provided by MATLAB structures. The RTI Bypass Blockset MATLAB API can be used together with MATLAB standard M-functions to write batch scripts for the automated creation of bypass models and configuration of bypass interfaces and functions. It can also be used to configure RTI Bypass blocks from overlaid MATLAB blocks.

For details, refer to the *RTI Bypass Blockset MATLAB API Reference*.

Application note for the RTI Bypass Blockset available

RTI Bypass Blockset comes with an application note that describes the setup for performing function bypassing via XCP on FlexRay using the RTI Bypass Blockset. You are guided through setting up an XCP master using dSPACE FlexRay Configuration Package, implementing the bypass model under Simulink, and configuring the bypass interface and bypass functions using the RTI Bypass Blockset.

Refer to *RTI Bypass Blockset Application Note*.

Working with models from RTI Bypass Blockset versions 2.0, 2.1 and 2.2

dSPACE Release 6.3 comes with RTI Bypass Blockset 2.5, which is compatible with earlier blockset versions 2.x. However, the format of the data dictionary is no longer the same as in RTI Bypass Blockset Versions 2.0, 2.1 and 2.2. The data dictionaries of Simulink models built with blockset Versions 2.0, 2.1 and 2.2 are automatically converted to the format used by RTI Bypass Blockset 2.5.



If you have a model that was saved with RTI Bypass Blockset 2.5 and want to use it with an earlier 2.x version of the RTI Bypass Blockset, you must first delete the model's data dictionary (the name and path of the data dictionary can be found in the Info block) and import the ASAM-MCD 2MC (A2L) file again. The RTI Bypass Blockset then automatically creates a data dictionary in the appropriate format.

Limitation when using the Variable Editor


You cannot open the Variable Editor via the RTIBYPASS_SETUP block of the RTI Bypass Blockset if you have installed CalDesk 2.1 or 2.0. You can open the Variable Editor via the RTIBYPASS_SETUP block only if you have installed CalDesk 1.4.1 or earlier.

RTI CAN MultiMessage Blockset

New Features of the RTI CAN MultiMessage Blockset 2.4

Experimental messages for user-defined databases

The RTI CAN MultiMessage Blockset now supports experimental messages for user-defined databases (UDDB). You can work with UDDBs in the CAN Navigator of ControlDesk (refer to *New Features of ControlDesk 3.3* on page 53). Experimental messages are independent of the database file specified for the RTICANMM MainBlock and not included in the model's TRC file. You can configure these messages during run time.

You can specify experimental messages for UDDBs on the *Experimental Software Page (RTICANMM MainBlock)* ( *RTI CAN MultiMessage Reference*).

RTI RapidPro Control Unit Blockset

New Features of the RTI RapidPro Control Unit Blockset 1.9

RPCU_INJ_IGN_TPU block	The maximum number of injection and ignition pulses (engine control) has changed from 10 to 15.
RPCU_AABP_TPU block	The maximum number of angle-angle-based pulses (engine control) has changed from 10 to 15.

SystemDesk

Where to go from here

Information in this section

<i>New Features of SystemDesk 2.0</i>	66
<i>Migrating to SystemDesk 2.0</i>	78

New Features of SystemDesk 2.0

Where to go from here


Information in this section

<i>Support for AUTOSAR 2.1 and AUTOSAR 3.0</i>	66
<i>Measurement and Calibration</i>	66
<i>Network Communication and Bus Support</i>	69
<i>Basic Software and ECU Configuration</i>	71
<i>SystemDesk Simulation Module</i>	75

Support for AUTOSAR 2.1 and AUTOSAR 3.0

SystemDesk supports:

- AUTOSAR Release 2.1 with Versions 2.1.2 and 2.1.4
- AUTOSAR Release 3.0 with Versions 3.0.0 and 3.0.2


For details, refer to *Basics on AUTOSAR Import/Export* ( *SystemDesk Guide*).

Measurement and Calibration

SystemDesk 2.0 supports modeling measurement and calibration access according to the AUTOSAR standard.

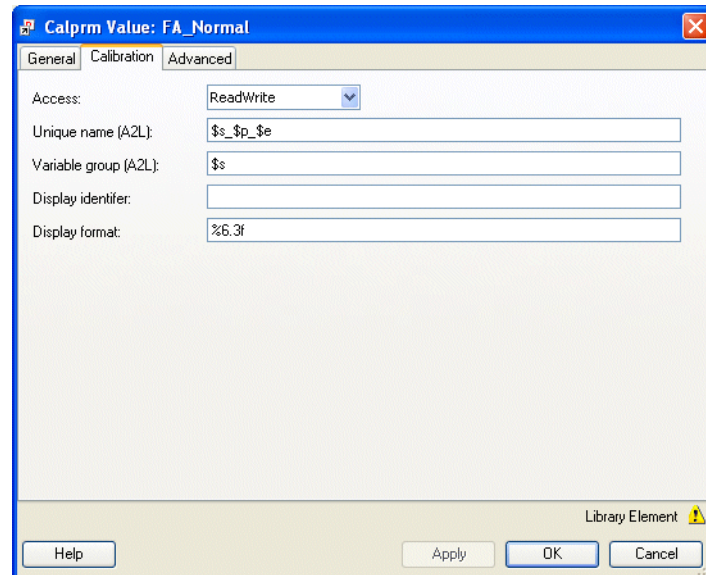
Modeling measurement and calibration in a software architecture

Measurement access In a software architecture, you can specify measurement access for data elements, operation arguments, and interrunnable variables. You can also specify measurement-specific properties to be included in an A2L file.

For details, refer to *Modeling Measurement Access* ( *SystemDesk Guide*).

Calibration access In a software architecture, you can specify calibration access for atomic software components by specifying calibration parameters via a *calprm interface* or via the internal behavior. You can also specify calibration-specific properties to be included in an A2L file.

As an example, the illustration below shows calibration-specific properties for the *FA_Normal* calibration parameter. Macros are used to specify the variable name and group.



SystemDesk 2.0 does not support defining multidimensional parameters, such as curves and maps.

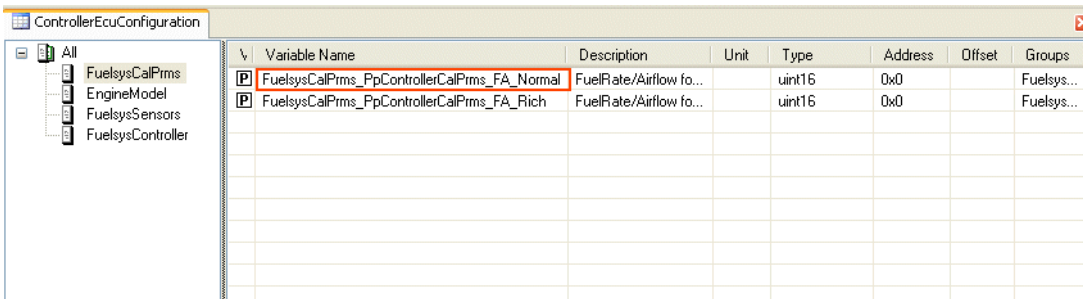
For details, refer to *Modeling Calibration Access* ( *SystemDesk Guide*).

RTE generation and A2L file export for measurement and calibration

During RTE generation, SystemDesk can create measurement variables and calibration parameters for the elements you have defined measurement and calibration access for.

After generating RTE code, you can export measurement and calibration variables to an A2L file. The A2L file contains the properties you have specified for them, such as minimum and maximum values. Physical addresses can be assigned via a map file. This allows measurement and calibration (MC) systems such as dSPACE's CalDesk to access these variables while the ECU application is running.

As an example, the illustration below displays a list of calibration parameters created during RTE generation. The list also contains an entry for the `FA_Normal` calibration parameter. To specify the variable name, each macro was replaced by the name of the software component (`$s`), the name of the port (`$p`) and the name of the calibration parameter (`$e`).




Variable Name	Description	Unit	Type	Address	Offset	Groups
FuelsysCalPrms_PpControllerCalPrms_FA_Normal	FuelRate/Airflow fo...		uint16	0x0		Fuelsys...
FuelsysCalPrms_PpControllerCalPrms_FA_Rich	FuelRate/Airflow fo...		uint16	0x0		Fuelsys...



For details, refer to *Basics on RTE Code Generation for Measurement and Calibration* ( *SystemDesk Guide*).

Network Communication and Bus Support

SystemDesk 2.0 provides the following new features for network communication.

Support for FlexRay

You can now import an existing FIBEX file into a communication matrix to specify network communication. Refer to *How to Add and Specify a Communication Matrix* ( *SystemDesk Guide*).

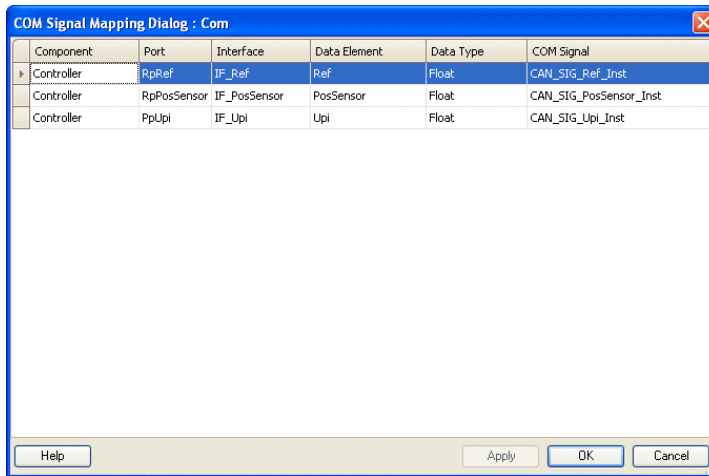
To map the network communication defined in a FIBEX file to a hardware topology, SystemDesk also lets you specify FlexRay buses and FlexRay ECU communication ports. Refer to *How to Add and Specify a Bus* ( *SystemDesk Guide*) and *How to Add and Specify an ECU* ( *SystemDesk Guide*).


Import of COM configurations

With SystemDesk 1.x, a communication matrix is required to specify the network communication. Data elements (and event messages) to be exchanged between ECUs have to be mapped to the network communication using the **Signal Mapping** editor. SystemDesk uses this information to generate a COM configuration.

Besides creating and updating a COM configuration based on a network communication, with SystemDesk 2.0 you can also *import a COM configuration*. This means you can generate RTE code *without specifying a network communication or importing a database file to the network communication*.

After you import a COM configuration, you have to map the contained COM signals to data elements for RTE code generation. SystemDesk provides the **COM Signal Mapping Dialog** for this purpose.



For instructions, refer to *How to Import COM Configurations* ( *SystemDesk Guide*).

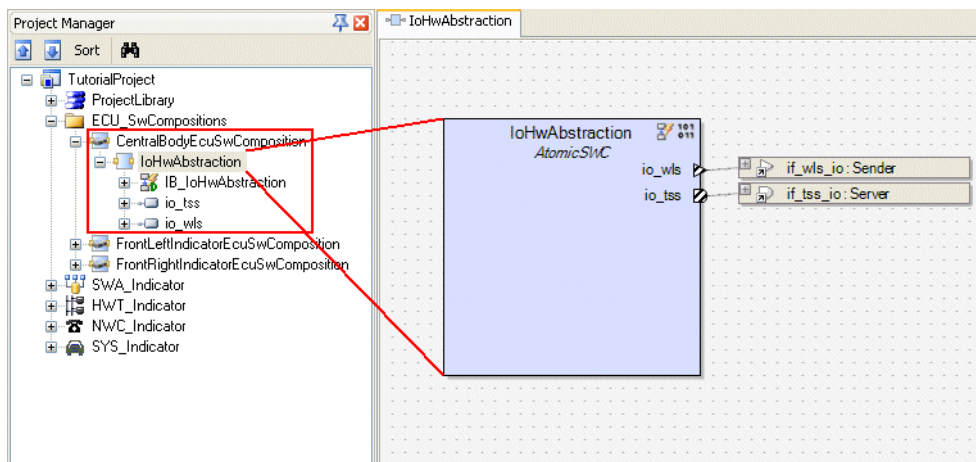
Basic Software and ECU Configuration

SystemDesk 2.0 provides improved support for configuring ECUs and integrating basic software modules.

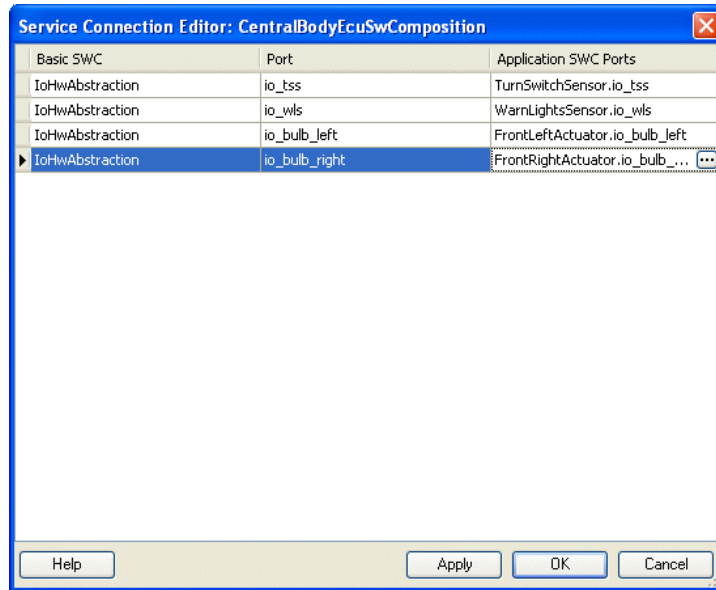
Graphical modeling of ECU software compositions

SystemDesk 2.0 supports the *graphical modeling of ECU software compositions*. An ECU software composition is an element which lets you specify all the basic software components of one ECU and their connections to the application software. This allows you to generate RTE code for the application software components to access the basic software.

As an example, the illustration below shows SystemDesk's **Project Manager** with a folder containing three ECU software compositions. The **CentralBodyEcuSwComposition** contains the **IoHwAbstraction** basic software component. The illustration also shows the graphical representation of this basic software component.



After assigning an ECU software composition to a specific ECU configuration, you can use the **Service Connection Editor** to connect application software components to the basic software components that are defined in the ECU software composition. This is a preparatory step for RTE code generation.



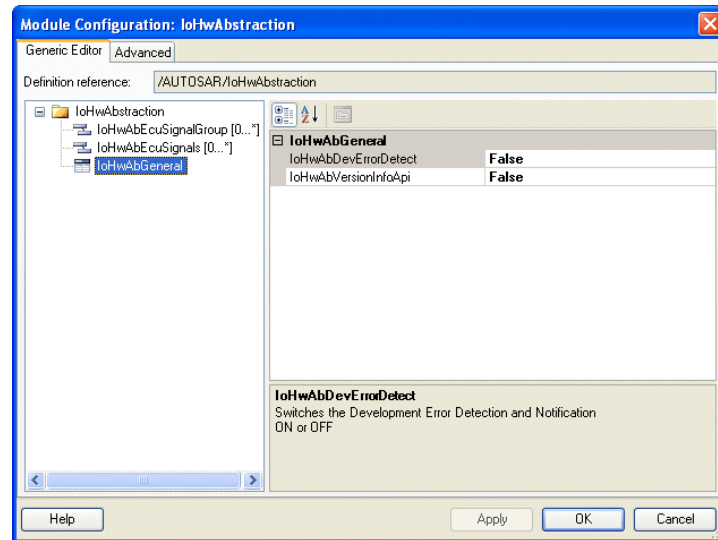
For details, refer to *Basics on ECU Software Compositions* ( *SystemDesk Guide*).


Support for basic software module configurations

SystemDesk 2.0 supports basic software module configurations and allows you to create, import, edit, and export them. SystemDesk provides the **BSW Module Editor** for editing configurations such as the *COM configuration* or the *NvM configuration*.

SystemDesk 2.0 supports all the basic software module configurations which are defined in an ECU configuration parameter definition file. Such definition files are part of the AUTOSAR standard or can be provided by third-party tools. The definition file is required as a template for module configurations and allows you to create and edit any module configuration.

The illustration below shows editing an `IoHwAbstraction` configuration with SystemDesk's `BSW Module Editor`. The parameters of the module configuration are defined in an ECU configuration parameter definition file of the AUTOSAR standard.



For details, refer to *Basics on Module Configurations* ( *SystemDesk Guide*).


Support for the NVRAM Manager

SystemDesk 2.0 supports the *NVRAM Manager*. The NVRAM Manager is a basic software module that provides access to nonvolatile memory such as an EEPROM or ECU flash memory.

SystemDesk lets you:

- Specify *NV block service needs* of atomic software components.
- Generate an *NvM* configuration based on the specified NV block service needs.
- Generate a basic software component based on the previously generated NvM configuration.

SystemDesk 2.0 also provides an implementation for the NVRAM Manager for you to simulate it with SystemDesk.

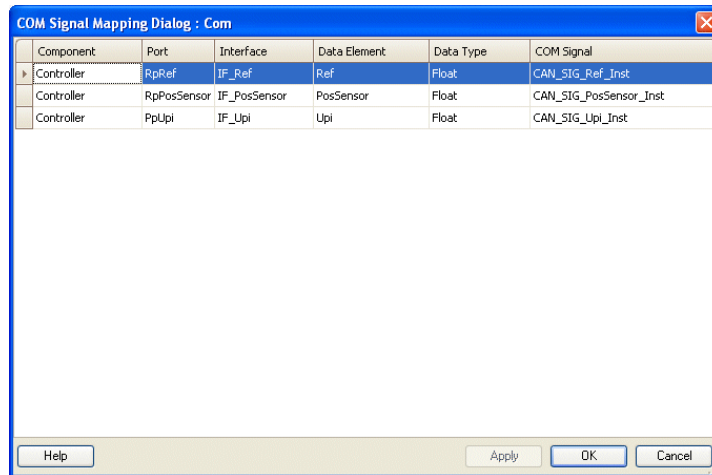
For details, refer to *Configuring the NVRAM Manager* ( *SystemDesk Guide*).


Import of COM configurations

With SystemDesk 1.x, a communication matrix is required to specify the network communication. Data elements (and event messages) to be exchanged between ECUs have to be mapped to the network communication using the **Signal Mapping** editor. SystemDesk uses this information to generate a COM configuration.


Besides creating and updating a COM configuration based on a network communication, with SystemDesk 2.0 you can also *import a COM configuration*. This means you can generate RTE code *without specifying a network communication or importing a database file to the network communication*.

After you import a COM configuration, you have to map the contained COM signals to data elements for RTE code generation. SystemDesk provides the **COM Signal Mapping Dialog** for this purpose.



For instructions, refer to *How to Import COM Configurations* ( *SystemDesk Guide*).

SystemDesk Simulation Module

SystemDesk 2.0 provides the new *Simulation Module* to perform a non-real-time offline simulation of a system on a developer PC using the original C code. You can use the module for open- and closed-loop SIL (software-in-the-loop) simulation of automotive software architectures. For details, refer to *Introduction to Simulating Systems* ( *SystemDesk Guide*).

Using original C code for SIL simulation


Simulation with SystemDesk is a SIL (software-in-the-loop) simulation and normally relies on C code for software components (SWCs) and the RTE (run-time environment). You have to attach code or object files to the implementations of the software components to be simulated. The C code of the SWCs and the generated code of the RTE are compiled and executed during simulation. Offline simulation with SystemDesk lets you detect software errors and verify diagnostics software at an early stage in the development process for networked ECUs and their software components.






Open-loop and closed-loop simulation

SystemDesk supports both open-loop and closed-loop simulation. You can use stimulus generators for open-loop simulation, or you can connect appropriate plant models to the software architecture for closed-loop simulation. SystemDesk allows the integration of Simulink®-based models. You can also import arbitrary models using C code and the AUTOSAR software component standard.

Testing basic software

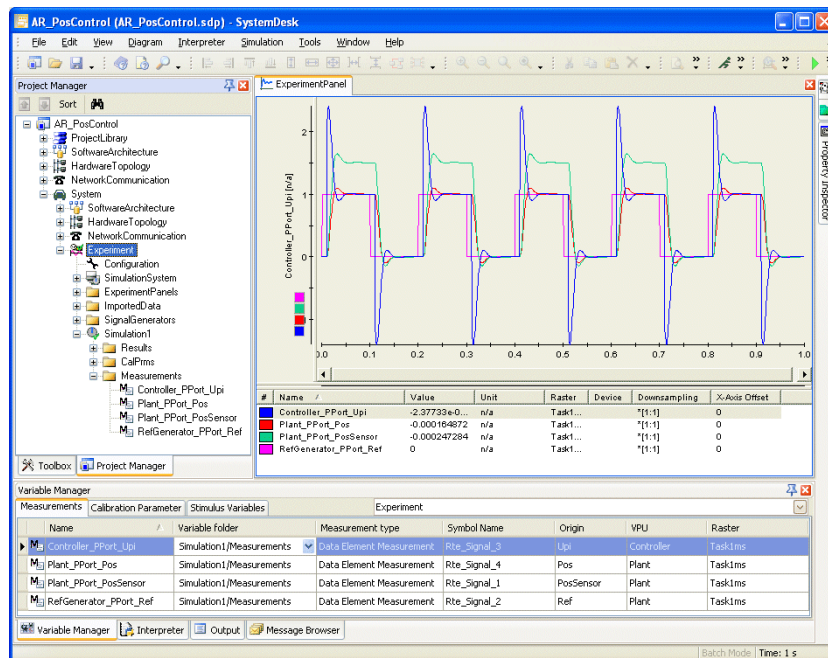
You can test application software together with basic software modules such as the AUTOSAR operating system, NVRAM and Error Manager for a realistic view of your system in an offline simulation. SystemDesk uses the AUTOSAR operating system simulation to simulate application scheduling behavior. SystemDesk lets you simulate all the basic software modules for which you can provide an implementation.

For details, refer to *Integrating Basic Software in a Simulation* ( *SystemDesk Guide*).

VFB system simulation	SystemDesk supports the Virtual Functional Bus (VFB) concept defined by AUTOSAR for system simulation on the logical level. You can use VFB system simulation to verify the dynamic system behavior at this early design stage, without having to specify a hardware topology or a network communication. The VFB system simulation is performed in an idealized environment. Effects of the underlying hardware are not simulated at this stage. For details, refer to <i>Basics on Simulating a VFB System</i> ( <i>SystemDesk Guide</i>).
CAN bus simulation	SystemDesk allows you to simulate effects caused by the communication buses. You can perform bus simulations to get a rough estimation of bus usage in this design stage and take communication delays into account when evaluating application timing behavior. Effects such as arbitration or bus capacities can be simulated for CAN buses. For details, refer to <i>Basics on Simulating a Bus</i> ( <i>SystemDesk Guide</i>).
Managing simulation runs	SystemDesk lets you configure experiments and simulations to manage your simulation tasks. You can create your simulations in the SystemDesk GUI, or you can use SystemDesk's automation interface for the test description. For details, refer to <i>Working with Experiments and Simulations</i> ( <i>SystemDesk Guide</i>).
Specifying variables	You can define measurement and stimulus variables for simulation as well as calibration parameters. For details, refer to <i>Specifying Variables for Simulation</i> ( <i>SystemDesk Guide</i>).
Running simulations	You can run one or more simulations at once. You can pause, restart, and stop simulation runs. You can also perform single-step simulation. You can change the calibration parameters of a simulation when a simulation run is paused. For details, refer to <i>Building and Running the Simulation Application</i> ( <i>SystemDesk Guide</i>).

Visualizing simulation runs

You can visualize the results of simulation runs in SystemDesk's Plotter. You can display measurement data in relation to the simulation time (time plot) or in relation to another measurement variable (XY plot). For details, refer to *Visualizing Simulation Runs in the Plotter* ([SystemDesk Guide](#)).



Importing and exporting measurement data

You can export simulation results and import measurement data of simulation results. For details, refer to *Working with Simulation Results* ([SystemDesk Guide](#)).

Software debugging on the C code level

SystemDesk supports the debugging of the code files used for the simulation. You can define breakpoints or perform single-step-simulation. You can use individual text messages to debug your code files or attach a C debugger such as Microsoft® Visual Studio .NET 2005 to SystemDesk. For details, refer to *Use Cases for Simulation* ([SystemDesk Guide](#)).

Migrating to SystemDesk 2.0

Migrating to SystemDesk 2.0

Reusing projects created with SystemDesk 1.0 or 1.1

To reuse a project created with SystemDesk 1.0, SystemDesk 1.0 with Simulation Module, or SystemDesk 1.1, you must export the project in the SystemDesk format in SystemDesk 1.1 Patch 1, and then import the SDXML file into a new project in SystemDesk 2.0.

Migration workflow This workflow shows the steps necessary for migrating a project to SystemDesk 2.0:

1. Install SystemDesk 1.1 Patch 1.
2. In SystemDesk 1.1 Patch 1, open the project created with SystemDesk 1.0, SystemDesk 1.0 with Simulation Module, or SystemDesk 1.1, which you want to reuse in SystemDesk 2.0.
3. Export the project in the SystemDesk format. This is done by calling the **Export - SystemDesk Format** command from the project's context menu in the **Project Manager**. The project and all its child elements are exported to the specified SDXML file.



To avoid data inconsistency, it is recommended to export the whole project and not only parts of it.


4. Change to SystemDesk 2.0.
5. Create a new project.
6. Import the SDXML file containing the exported project elements into the project. Call the **Import - SystemDesk Format** command from the project's context menu in the **Project Manager**. The imported data is added to the SystemDesk project under the project element.
7. Save the project.

Automating more than one version of SystemDesk


Automating a specific version of SystemDesk Suppose you have installed more than one version of SystemDesk, for example, SystemDesk 1.0 and 2.0. If you open a COM connection to SystemDesk, the version that is automated is by default the last one installed. You can also supply version information when opening the COM connection to automate earlier installed versions.

Automating SystemDesk after one version has been removed

Suppose you have removed a SystemDesk version and want to automate version of SystemDesk that is still installed. SystemDesk's automation feature has to be registered in the Microsoft Windows Registry. This is done during SystemDesk's installation process. If you remove a version of SystemDesk, the automation feature is unregistered, and you have to re-register it.


For basic information on automating more than one version of SystemDesk, refer to *Automating More Than One Version of SystemDesk* ( *SystemDesk Guide*).

Reusing automation scripts created for SystemDesk 1.0 or 1.1

SystemDesk's automation API has been changed with SystemDesk version 2.0. You have to migrate automation scripts that you have created for SystemDesk versions prior to SystemDesk 2.0 to the changed API. For migration information on changed and cancelled automation elements, refer to *Migrating Automation Scripts* ( *SystemDesk Guide*).

Reusing custom validation rules created for SystemDesk 1.0 or 1.1

When you define custom validation rules, you have to use the new `GetElementByGuid` automation method to access the element to be validated.

The `ElementPathToPythonInstruction` automation method used in custom validation rules created for SystemDesk 1.0 or 1.1 is no longer supported. To reuse custom validation rules, you have to adapt them to the new `GetElementByGuid` automation method. For example rules using the new method, refer to *Examples of Custom Validation Rules* ( *SystemDesk Guide*).



The `GetElementByGuid` automation method is intended for validation purposes only.

Compatibility Information

Where to go from here

Information in this section

<i>Supported MATLAB Releases</i>	81
<i>Supported Operating Systems</i>	82

Supported MATLAB Releases

Supported MATLAB releases

The RCP and HIL software, TargetLink, Model Compare, and SystemDesk in dSPACE Release 6.3 offer full compatibility with the following releases of MATLAB from The MathWorks:

- MATLAB R2008b
- MATLAB R2008a
- MATLAB R2007b+
- MATLAB R2007a+
- MATLAB R2006b
- MATLAB R2006a+

For detailed information on compatibility, refer to www.dspace.com/goto?Compatibility on the dSPACE Web site.

Supported Operating Systems

Supported operating systems

The following table shows which software items in dSPACE Release 6.3 support which operating system:

Operating System...	Is Supported By...					
	RCP & HIL Software	TargetLink 2.3.1	TargetLink 3.0	CalDesk 2.1	Model Compare 2.0.1	SystemDesk 2.0
Windows 2000 Professional with Service Pack 4	–	X	X	–	–	–
Windows XP Professional (32-bit version) with Service Pack 2 ¹⁾	X	X	X	X	X	X
Windows Vista ²⁾ (32-bit version) with Service Pack 1	X	X	X	–	X	X

¹⁾ It is recommended to use Service Pack 3. For the latest information on Windows XP with Service Pack 3, refer to <http://www.dspace.com/goto?winxpsp3>.

²⁾ Only Windows Vista Business, Ultimate, and Enterprise. Windows Vista Home and Starter are not supported.

Windows XP Professional x64 Edition and Windows Vista (64-bit version) are not supported by dSPACE Release 6.3.


For an overview of operating systems supported by dSPACE Releases and products, refer to www.dspace.com/goto?os_compatibility.

Limitations when working with Windows Vista (32-bit)

MATLAB support Under Windows Vista, the dSPACE software supports only MATLAB versions since MATLAB R2007a+.

dSPACE software Windows Vista is not supported by CalDesk.

Sleep mode not supported The dSPACE software does not support Windows Vista's sleep mode for power saving. When restarting the PC from the sleep mode, you must reboot it to restore communication with the dSPACE hardware.

To avoid the automatic sleep mode, you must disable it. Refer to *How to Disable Windows Vista's Sleep Mode* ( *Software Installation and Management Guide*).

Fast user switching not supported The dSPACE software does not support the fast user switching feature of Windows Vista.

Closing dSPACE software before PC shutdown The modified shutdown procedure of Windows Vista causes some required processes to be aborted although they are still being used by dSPACE software. To avoid data loss, you must terminate the dSPACE software manually before performing a PC shutdown.

Allowing communication via additional firewall rules During installation of the dSPACE software, two additional Windows Vista firewall rules are automatically installed. The first rule allows communication with a dSPACE expansion box, for example, AutoBox. The second rule allows MotionDesk to receive motion data from a network channel.

The rules are created by the following commands:

- `advfirewall firewall add rule name="dSPACE Net Service" service=any dir=in action=allow profile=any protocol=icmpv4:0, any description="Allow the dSPACE Net Service to connect to a dSPACE expansion box via network."`
- `advfirewall firewall add rule name="dSPACE MotionDesk" program="%dSPACE_ROOT%\MotionDesk\Bin\MotionDesk.exe" dir=in action=allow profile=any description="Allow dSPACE MotionDesk to receive motion data via network."`

A

ASM Diesel Exhaust Blockset
 new features 22

ASM Diesel InCylinder Blockset
 new features 23

ASM Drivetrain Basic Blockset
 migrating 24

ASM Electric Components Blockset
 migrating 25

ASM Engine Diesel Blockset
 migrating 28

ASM Engine Gasoline Basic Blockset
 migrating 29

ASM Environment Blockset
 new features 30

ASM Gasoline InCylinder Blockset
 new features 31

ASM Optimizer
 new features 32

ASM Traffic Blockset
 new features 33

ASM Turbocharger Blockset
 migrating 34
 new features 34

ASM Vehicle Dynamics Blockset
 migrating 36
 new features 36

AutomationDesk
 new features 17

C

CalDesk
 migration 50
 new features 38

ControlDesk
 new features 53

D

dSPACE FlexRay Configuration Package
 new features 55

G

general enhancements and changes 10

K

key features 13

M

migrating
 ASM Drivetrain Basic Blockset 24
 ASM Electric Components Blockset 25
 ASM Engine Diesel Blockset 28
 ASM Engine Gasoline Basic Blockset 29

migrating

ASM Turbocharger Blockset 34
 ASM Vehicle Dynamics Blockset 36

migration
 CalDesk 50
 SystemDesk 78

N

new features
 ASM Diesel Exhaust Blockset 22
 ASM Diesel InCylinder Blockset 23
 ASM Environment Blockset 30
 ASM Gasoline InCylinder Blockset 31
 ASM Optimizer 32
 ASM Traffic Blockset 33
 ASM Turbocharger Blockset 34
 ASM Vehicle Dynamics Blockset 36

AutomationDesk 17
 CalDesk 38
 ControlDesk 53
 dSPACE FlexRay Configuration
 Package 55
 Real-Time Testing 19
 RTI 57
 RTI Bypass Blockset 59
 RTI CAN MultiMessage Blockset 61
 SystemDesk 66

P

product overview 11

R

Real-Time Testing
 new features 19

RTI
 new features 57

RTI Bypass Blockset
 new features 59

RTI CAN MultiMessage Blockset
 new features 61

S

supported MATLAB releases 81
 supported operating systems 82

SystemDesk
 migration 78
 new features 66

V

version history 11

W

Windows Vista
 limitations 82

