

dSPACE Release

New Features and Migration

dSPACE Release 6.4 – June 2009

dSPACE

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 2009 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 6.4	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 6.4</i>	16
AutomationDesk	19
<i>New Features of AutomationDesk 2.3</i>	19
<i>Migrating to AutomationDesk 2.3</i>	21
Automotive Simulation Models (ASM)	23
All ASM Blocksets.....	25
<i>New Features of All ASM Blocksets</i>	25
ASM Base InCylinder Blockset.....	26
<i>New Features of ASM Base InCylinder Blockset 1.0.1</i>	26
<i>Migrating to ASM Base InCylinder Blockset 1.0.1</i>	26
ASM Diesel InCylinder Blockset.....	27
<i>Migrating to ASM Diesel InCylinder Blockset 1.0.1</i>	27
ASM Drivetrain Basic Blockset.....	28
<i>New Features of ASM Drivetrain Basic Blockset 1.6.1</i>	28
<i>Migrating to ASM Drivetrain Basic Blockset 1.6.1</i>	28
ASM Electric Components Blockset	29
<i>New Features of ASM Electric Components Blockset 1.0.2</i>	29
<i>Migrating to ASM Electric Components Blockset 1.0.2</i>	29
ASM Engine Diesel Blockset.....	30
<i>New Features of ASM Engine Diesel Blockset 1.3.2</i>	30
<i>Migrating to ASM Engine Diesel Blockset 1.3.2</i>	30
ASM Engine Gasoline Basic Blockset.....	31
<i>New Features of ASM Engine Gasoline Basic Blockset 1.3.2</i>	31
<i>Migrating to ASM Engine Gasoline Basic Blockset 1.3.2</i>	31
ASM Engine Gasoline Blockset	32
<i>New Features of ASM Engine Gasoline Blockset 2.1.2</i>	32

<i>Migrating to ASM Engine Gasoline Blockset 2.1</i>	32
ASM Environment Blockset	33
<i>Migrating to ASM Environment Blockset 1.4</i>	33
ASM Gasoline InCylinder Blockset	35
<i>Migrating to ASM Gasoline InCylinder Blockset 1.0.1</i>	35
ASM Optimizer	35
<i>New Features of ASM Optimizer 1.3</i>	35
ASMParameterization Tool	35
<i>New Features of the ASMParameterization Tool 1.4</i>	35
ASM Traffic Blockset	36
<i>Migrating to ASM Traffic Blockset 1.1.1</i>	36
ASM Traffic Operator Blockset.....	37
<i>New Blockset ASM Traffic Operator 1.0</i>	37
ASM Trailer	37
<i>New Blockset ASM Trailer 1.0</i>	37
ASM Turbocharger Blockset	38
<i>New Features of ASM Turbocharger Blockset 1.4.1</i>	38
<i>Migrating to ASM Turbocharger Blockset 1.4</i>	38
ASM Utilities Blockset	38
<i>Migrating to ASM Utilities Blockset 1.5.1</i>	38
ASM Vehicle Dynamics Blockset	39
<i>New Features of ASM Vehicle Dynamics Blockset 1.3</i>	39
<i>Migrating to ASM Vehicle Dynamics Blockset 1.3</i>	39

CalDesk	43
New Features of CalDesk 3.0	44
<i>New Automation Interface</i>	44
<i>New Project and Experiment Features</i>	45
<i>New Platform Management Features</i>	46
<i>New Variable Management Features</i>	47
<i>New Instruments and Instrument Features</i>	48
<i>New Visualization Features (Common to all Instruments)</i>	53
<i>New Measurement and Recording Features</i>	54
<i>New Data Set Management Features</i>	55
<i>New Features of the Variable Editor</i>	55
<i>New Features of the CalDesk ECU Diagnostics Module</i>	56
<i>New ECU Flash Programming Tool Features</i>	57
<i>Further Enhancements with CalDesk 3.0</i>	58
Migrating to CalDesk 3.0.....	59
<i>Migrating to CalDesk 3.0</i>	59
<i>How to Migrate a Project to CalDesk 3.0</i>	60
<i>Migrating CalDesk 2.1 Experiments with Curves</i>	62

<i>How to Migrate a Variable Editor Project (VEP File)</i>	63
<i>Automatic Reconnect Feature: Changed Default Settings</i>	64
ConfigurationDesk	65
Migrating to ConfigurationDesk 3.0.1	65
<i>Migration to ConfigurationDesk 3.0.1</i>	66
<i>How to Migrate a ConfigurationDesk Project with the RapidPro Projects Migrator</i>	68
ControlDesk	71
<i>New Features of ControlDesk 3.4</i>	71
dSPACE FlexRay Configuration Package	75
<i>New Features of dSPACE FlexRay Configuration Package 2.1</i> ..	75
ModelDesk	77
<i>New Features of ModelDesk 2.2</i>	77
MotionDesk	79
<i>New Features of MotionDesk 2.1.3</i>	79
RTI and RTLib	81
<i>New Features of RTI/RTI-MP and RTLib</i>	81
RTI AUTOSAR Package	85
<i>New Features of the RTI AUTOSAR Package 1.0</i>	85
RTI Bypass Blockset	87
<i>New Features of the RTI Bypass Blockset 2.6</i>	87
RTI CAN MultiMessage Blockset	91
<i>New Features of the RTI CAN MultiMessage Blockset 2.4.1</i>	91
RTI LIN MultiMessage Blockset	93
<i>New Features of the RTI LIN MultiMessage Blockset 1.7</i>	93
SystemDesk	95
<i>New Features of SystemDesk 2.0.4</i>	95

Compatibility Information	101
<i>Supported MATLAB Releases</i>	101
<i>Supported Operating Systems</i>	102
Index	105

About This Document

Contents

This document informs you about the new features of all the dSPACE software products in dSPACE Release 6.4. 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 6.4

Objective Gives you an overview of the new key features in dSPACE Release 6.4, 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 6.4</i>	16

General Enhancements and Changes

Objective	The following new features and changes concern several dSPACE products.
Microtec PowerPC C Compiler	<p>With dSPACE Release 6.4, the Microtec PowerPC C Compiler version 3.5 is used with the DS1005, DS1103, DS1104, and MicroAutoBox. It is fully compatible with Version 3.3.</p> <p>For detailed information on the compiler changes, refer to the release notes that you can find in <PPC-InstallationRoot>/relnotes.</p>
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 6.4 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.
Software support discontinued for DS4120	The DS4120 ECU Interface Board is supported by the dSPACE software only up to dSPACE Release 6.3. As of dSPACE Release 6.4, the software no longer supports the DS4120.

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			
	6.1	6.2	6.3	6.4
AutomationDesk	2.0.1	2.1	2.2	2.3 See <i>AutomationDesk</i> on page 19.
Automotive Simulation Models	1.5	1.6	2.0	2.1 See <i>Automotive Simulation Models (ASM)</i> on page 23.
CalDesk	2.0	2.0	2.1	3.0 See <i>CalDesk</i> on page 43.
ConfigurationDesk	1.3.1	1.3.2	3.0	3.0.1 See <i>ConfigurationDesk</i> on page 65.
ControlDesk	3.2.1	3.2.2	3.3	3.4 See <i>ControlDesk</i> on page 71.
dSPACE Data Dictionary	1.5	1.5.1	1.5.1	1.5.1
dSPACE FlexRay Configuration Package	1.8	1.10	1.10	2.1 See <i>dSPACE FlexRay Configuration Package</i> on page 75.
Model Compare	1.0	2.0	2.0.1	2.0.1
ModelDesk	2.0	2.1	2.1	2.2 See <i>ModelDesk</i> on page 77.
MotionDesk	2.1.2	2.1.2	2.1.2	2.1.3
RTI	6.0	6.1	6.2	6.3 See <i>RTI and RTLib</i> on page 81.
RTI-MP	5.4	5.5	5.6	5.7
RTI AUTOSAR Package	---	---	---	1.0 See <i>RTI AUTOSAR Package</i> on page 85.

Product	dSPACE Release			
	6.1	6.2	6.3	6.4
RTI Bypass Blockset	2.4	2.4.1	2.5	2.6 <i>See RTI Bypass Blockset on page 87.</i>
RTI CAN Blockset	2.6.5	2.6.6	2.6.7	2.6.8
RTI CAN MultiMessage Blockset	2.2	2.3	2.4	2.4.1 <i>See RTI CAN MultiMessage Blockset on page 91.</i>
RTI LIN MultiMessage Blockset	1.5	1.5.1	1.6	1.7 <i>See RTI LIN MultiMessage Blockset on page 93.</i>
RTI RapidPro Control Unit Blockset	1.7	1.8	1.9	1.9.1
SystemDesk	1.0	1.1	2.0	2.0.4 <i>See SystemDesk on page 95.</i>
TargetLink	2.3	3.0	2.3.1 3.0	2.3 2.3.1 (enhancements based on TargetLink 2.3) 3.0 3.0.1 (enhancements based on TargetLink 3.0) For information on TargetLink 3.0.1, see the <i>ReadMe.txt</i> file on the dSPACE DVD.
Variable Editor	1.1	1.1	1.2	1.3 <i>See CalDesk on page 43.</i>

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"> ■ You can now exclude elements from execution. ■ You can now choose between several report layouts providing overview information and statistics. ■ You can now do operations that were formerly restricted to administrators. ■ There is a new library to manage Real-Time Testing (RTT) sequences. ■ The Remote Calibration (COM) library comes with two new blocks. ■ The Remote Diagnostics (COM) library comes with two new blocks. ■ The COM API now supports all the library-specific data objects. <p>The new key features of Real-Time Testing are:</p> <ul style="list-style-type: none"> ■ You can now handle CAN messages on MicroAutoBox. ■ A new library to access the RS232 interface on a DS1005 or DS1006. ■ You can now restart data replay in RTT sequences. <p>For details on the new features, refer to <i>New Features of AutomationDesk 2.3</i> on page 19.</p>
Automotive Simulation Models (ASM)	<p>The new ASM blocksets and tools of ASM are:</p> <ul style="list-style-type: none"> ■ A new model to simulate a trailer for vehicle dynamics, named ASM Trailer. <p>For details on the new blocksets and tools, refer to <i>Automotive Simulation Models (ASM)</i> on page 23.</p>
CalDesk	For details on the new features, refer to <i>New Features of CalDesk 3.0</i> on page 44.
ControlDesk	<p>The new key features of ControlDesk are:</p> <ul style="list-style-type: none"> ■ Supporting new hardware: MicroAutoBox 1401/1507, DS4004 HIL I/O Board and DS802 PHS Link Board. ■ Enhancement of data capturing. ■ The Bus Navigator with new design and features replaces the CAN Navigator.

For details on the new features, refer to *New Features of ControlDesk 3.4* on page 71.

dSPACE FlexRay Configuration Package

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

- Supporting FIBEX 3.0.
- The configuration of the variable description file (TRC file) generation was enhanced.

The new key features of dSPACE FlexRay Configuration Blockset are:

- You can now send and receive PDUs in your real-time model.
- You can send and receive wakeup patterns.

For details on the new features, refer to *New Features of dSPACE FlexRay Configuration Package 2.1* on page 75.

ModelDesk

The new key features of ModelDesk are:

- You can parameterize ASM models for engines (gasoline basic, gasoline and diesel) and a trailer.
- ModelDesk has a Traffic Editor to simulate traffic situations such as vehicles overtaking and changing lanes based on ASM Traffic models. With the Traffic Editor, you can parameterize the models more comfortable than with the Excel-based ASM Traffic Creator.
- ModelDesk is available with different licenses so you can buy only the functionality required for your simulation.

For details on the new features, refer to *New Features of ModelDesk 2.2* on page 77.

MotionDesk

The new key feature of MotionDesk is:

- MotionDesk is available with different licenses so you can buy only the functionality required for your visualization.

For details on the new features, refer to *New Features of MotionDesk 2.1.3* on page 79.

RTI, RTI-MP and RTLib

The new key features of RTI, RTI-MP and RTLib are:

- The RTI-MP Blockset was extended by the RTI Gigalink Blockset. It provides blocks for implementing data transfer between different real-time applications using DS1005 and/or DS1006 boards.
- Two new options are supported to reduce the contents of a variable description file. You can include *Simulink.Parameter* and *Simulink.Signal* objects only, and you can exclude all the variables of a subsystem from the generated variable description file.

- Real-time testing can now be enabled only globally for an RTI-MP model.
- There is a new function that allows you to adapt target-specific settings of an RTI-MP model according to the currently active RTI platform.
- There is a new function that allows you to read version information from a model file without opening the model.
- RTI and RTLib support the new DS4004 HIL Digital I/O Board.
- RTI and RTLib support the new MicroAutoBox variant 1401/1507 that provides bus and ECU interfaces for pure bypassing and gatewaying applications.

For details on the new features, refer to *New Features of RTI/RTI-MP and RTLib* on page 81.

RTI AUTOSAR Package

The **RTI AUTOSAR Package** is a dSPACE software product for implementing atomic software components and compositions in a Simulink environment and executing them on dSPACE real-time hardware. It consists of the *RTI AUTOSAR Interface Generator* and the *RTI AUTOSAR Interface Blockset*.

RTI Bypass Blockset

The new key features of the RTI Bypass Blockset are:

- Variable definition by external block inputs
- Running external tools from within the RTI Bypass Blockset
- Using additional FlexRay buffers for XCP command processing
- Support of XCP protocol version 1.1 and `IF_DATA_XCPplus`

The new key feature of the RTI Bypass Blockset MATLAB API is:

- Using the RTI Bypass Blockset without A2L files

For details on the new features, refer to *New Features of the RTI Bypass Blockset 2.6* on page 87.

RTI LIN MultiMessage Blockset

The new key features of RTI LIN MultiMessage Blockset are:

- Support of slave node configuration services compliant with LIN 2.0 and higher.

For details on the new features, refer to *New Features of the RTI LIN MultiMessage Blockset 1.7* on page 93.

SystemDesk

For details on the new features, refer to *New Features of SystemDesk 2.0.4* on page 95.

Variable Editor For details on the new features, refer to *New Features of CalDesk 3.0* on page 44.

Migrating to dSPACE Release 6.4

Objective After you install dSPACE Release 6.4, some additional steps may be necessary.

Migrating from dSPACE Release 6.3 There are no general migration steps to be done. The required product-specific migration steps are usually done automatically by the product. For exceptions, refer to the product-specific migration descriptions.

Migrating from dSPACE Release 6.2 or earlier To migrate from dSPACE Release 6.2 or earlier to dSPACE Release 6.4, 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 6.4 installed.

Example For example, if you want to migrate from dSPACE Release 6.0 to dSPACE Release 6.4, you have to perform the migration steps described in:

1. New Features and Migration of dSPACE Release 6.1
2. New Features and Migration of dSPACE Release 6.2
3. New Features and Migration of dSPACE Release 6.3
4. Finally, the migration steps described above.

Previous release documents The New Features and Migration documents for previous releases are available via Internet and on the dSPACE DVD:

- Download them from <http://www.dspace.com/goto?migration>.
- Read them from the dSPACE DVD (see the \Doc folder). The PDF files are called `NewFeaturesAndMigrationxx.pdf`, 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

Where to go from here


Information in this section

<i>New Features of AutomationDesk 2.3</i>	19
<i>Migrating to AutomationDesk 2.3</i>	21

New Features of AutomationDesk 2.3

AutomationDesk 2.3

Excluding elements from execution You can now disable automation blocks to exclude them from execution. Because you can also disable sequences and folders in the Project Manager, you can also use the feature to select a set of tests, etc.

For further information, refer to *Basics of Sequence Execution* ( *AutomationDesk Guide*).

Enhancements to the report generation In the Report page of the Properties dialog, you can now specify report layouts to add overview information and statistics to a report's contents by selecting the related style sheet.


You can choose between:

- A brief report to get only an overview of the quality state, including overview information and statistics.
- A detailed report to get all the available information from the result, including report headers, all the specified attributes, overview information and statistics.

- A classic report to get the information known from older AutomationDesk versions, including report headers and the specified attributes, but no overview information and statistics.


Each of the report layouts now takes the different requirements for standard sequences and test sequences from the Test Framework library into account. If you choose a report layout for the Test Framework, you get statistics on the executed tests and test steps.

The layout of plots has been improved, for example, the distance between the frame and the plot has been enlarged, and time axes for multi plots are aligned to the origin.

For further information, refer to *Basics of Report Generation* ( *AutomationDesk Guide*).


Enhancements to the access rights of standard users Some operations in AutomationDesk no longer require administrator rights, for example, creating a new custom library, and changing the execution mode of a library.

New Real-Time Testing library The Real-Time Testing library provides automation blocks for managing RTT sequences.

For further information, refer to *Real-Time Testing* ( *AutomationDesk Library Reference*).

Enhancements to the Remote Diagnostics (COM) library

- The new SyncHexService block executes a synchronous service specified directly by raw data (the request PDU) instead of symbolic data that must be mapped to the related raw data in the ODX database.
- The new CreateOfflineResults block stores a diagnostic result that you got in online execution mode to a Python object that you can reuse in offline execution mode.

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

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

- The data objects in the Remote Diagnostics (COM) library: System, Project, VehicleInformation, LogicalLink, ControlPrimitive, Service and SingleJob. Results can only be created.
- The data objects in the DTS7 library: System, Project, VehicleInformation, LogicalLink, ComPrimitive and Service. Results can only be created.

- The data objects in the MATLAB Access library: MATLAB and MATFile.
- The FailurePattern data object in the Failure Simulation Access library.
- The RS232Configuration data object in the RS232 library.

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

The demo scripts for implementing a user interface in Python are available for Python 2.2.1 and Python 2.5.1. You can find them in `<InstallationRoot>\Demos\AutomationDesk\API\Operator_Python_<x.y.z>`.

Real-Time Testing 1.6

CAN messages on MicroAutoBox Since Real-Time Testing 1.6, you can send and receive CAN messages on MicroAutoBox.

RS232 interface You can access the RS232 interfaces of DS1005 and DS1006 processor boards.

Restart of data streaming It is now possible to restart data streaming within an RTT sequence. The RTT sequence has to be reloaded beforehand.

MTest

MTest support is discontinued with dSPACE Release 6.4.

Migrating to AutomationDesk 2.3

General migration aspects

If you open an AutomationDesk project with a newer AutomationDesk version, the software automatically detects whether migration is necessary. If you click OK in the message dialog, the migration is started. If you also want to continue working with the old project, you should not overwrite it with the migrated project, because the versions are not downward compatible. Save the migrated project to another path or name.

You need not do any manual migration, except for the following point.

Migrating from AutomationDesk 1.x to AutomationDesk 2.x

The serialization of a project structure to the file system has been totally changed with AutomationDesk 2.x. Automatic migration covers only elements that are handled by the AutomationDesk project.

If you have added a file or folder to an AutomationDesk project structure in the file system manually using AutomationDesk 1.x, and you migrate from AutomationDesk 1.x to AutomationDesk 2.x, the new AutomationDesk project does not contain that file or folder. You must copy the file or folder to the new AutomationDesk project structure in the file system to manually make the files available to your project.

For example, the MainLibraryExamples.zip project contains an ExternalMaterial folder which you must copy to the migrated project manually.


Automotive Simulation Models (ASM)

Where to go from here

Information in this section

<i>All ASM Blocksets</i>	25
<i>ASM Base InCylinder Blockset</i>	26
<i>ASM Diesel InCylinder Blockset</i>	27
<i>ASM Drivetrain Basic Blockset</i>	28
<i>ASM Electric Components Blockset</i>	29
<i>ASM Engine Diesel Blockset</i>	30
<i>ASM Engine Gasoline Basic Blockset</i>	31
<i>ASM Engine Gasoline Blockset</i>	32
<i>ASM Environment Blockset</i>	33
<i>ASM Gasoline InCylinder Blockset</i>	35
<i>ASM Optimizer</i>	35
<i>ASMParameterization Tool</i>	35
<i>ASM Traffic Blockset</i>	36
<i>ASM Traffic Operator Blockset</i>	37
<i>ASM Trailer</i>	37
<i>ASM Turbocharger Blockset</i>	38
<i>ASM Utilities Blockset</i>	38
<i>ASM Vehicle Dynamics Blockset</i>	39

Information in other sections

Migrating ASM Models ( *ASM User Guide*)

Provides general information on the migration process of ASM models.

All ASM Blocksets

New Features of All ASM Blocksets

ModelDesk support

The parameterization of ASM Engine Gasoline Basic, ASM Engine Gasoline and ASM Engine Diesel is now supported by ModelDesk. Parameter sets generated in ASMPparameterization can be exported to ModelDesk projects, organized in experiments and downloaded to the simulation platforms. In ModelDesk, ASM models can be configured and parameters can be changed, saved, downloaded, etc. Parameters can also be exported back to MATLAB INI files.

Related topics

Basics

- *New Features of ModelDesk 2.2* on page 77

ASM Base InCylinder Blockset

New Features of ASM Base InCylinder Blockset 1.0.1

WALL_HEAT block	Scalings for the heat flow coefficient have been added. These scalings can be different for expansion and compression. The scaling for the wall temperature has been separated into different scalings for compression and expansion.
------------------------	---

Migrating to ASM Base InCylinder Blockset 1.0.1

CRANK_MECHANISM block	The engine speed used for operating point definition is now filtered with a PT1 element.
------------------------------	--

INJECTOR_CONTINUOUS block	The injection quantity for operating point definition is set to zero on engine standstill.
----------------------------------	--

RAIL block	The limits were missing in the integrator for the rail pressure. They have been added.
-------------------	--

ASM Diesel InCylinder Blockset

Migrating to ASM Diesel InCylinder Blockset 1.0.1

**SOFT_ECU_INCYLINDER_DI
ESEL block**

In the `SOFT_ECU_INCYLINDER_DIESEL` block, the anti-windup implementation of the EGR rate and rail pressure controller has been improved.

ASM Drivetrain Basic Blockset

New Features of ASM Drivetrain Basic Blockset 1.6.1

TORQUE_CONVERTER block

The TORQUE_CONVERTER block has been improved so that it can also handle negative pump speeds. The default parameterization for dragging the engine has also been optimized.

Migrating to ASM Drivetrain Basic Blockset 1.6.1

SOFT_ECU_TRANSMISSION block

The SOFT_ECU_TRANSMISSION block has been renamed SOFT_ECU_TRANSMISSION_BASIC. There is no functional change.

TORQUE_CONVERTER block

In the new TORQUE_CONVERTER block, there are only changes inside the library block which are transferred automatically. If you want to use the new default parameterization, you have to transfer it manually to the old projects from the demo project.

GEARBOX_AT block

The real-time path to the Const_Inert_Gear_In and Const_Inert_Clutch in GEARBOX_AT parameters has been changed.

GEARBOX_MT block

The real-time path to the Const_Inert_Gear_In and Const_Inert_Clutch in GEARBOX_MT parameters has been changed.

ASM Electric Components Blockset

New Features of ASM Electric Components Blockset 1.0.2

BATTERY block	Thermal power can now be set from an external source. The State-of-Charge model (SOC) can now be set initially and reset from an external source during simulation.
----------------------	--

Migrating to ASM Electric Components Blockset 1.0.2

HALF_BRIDGE_INVERTER block	Signal label in ASMSignalBus changed.
THREE_PHASE_INVERTER block	Signal label in ASMSignalBus changed.
BRUSHLESS_DC_MACHINE _ALPHA_BETA block	Signal label in ASMSignalBus changed.
SEPARATELY_EXCITED_DC_MACHINE block	Signal label in ASMSignalBus changed. The changed signal labels of the ASMSignalBus have also been changed in all corresponding Simulink Bus Selector blocks, so no additional migration steps are required.

ASM Engine Diesel Blockset

New Features of ASM Engine Diesel Blockset 1.3.2

COMMON_DIESEL_PARAM ETERS block For using the same real-time code with different cylinder counts, a new parameter, `Const_num_Cyl_vector`, has been added to the `COMMON_DIESEL_PARAMETERS` block. This parameter is a vector of length 20. For each cylinder, there is a vector element containing a value of 1. For example, if there are four cylinders, the first four vector elements each contain a 1, the remaining elements contain a 0.

SOFT_ECU_DIESEL block The post injection management for DPF regeneration has been modified so that the injection quantity distribution is set to zero for injections beginning after a given post injection limit angle if the DPF regeneration is deactivated.

The maximum torque characteristic uses end values instead of being extrapolated when the input engine speed is out of range.

COOLER block The energy balance has been corrected.

Migrating to ASM Engine Diesel Blockset 1.3.2

INJECTOR block The parameters for the number of cylinders and the number of injections have been replaced by the maximal number of cylinders and the maximal number of injections.

UNIT_INJECTOR block The parameters for the number of cylinders and the number of injections have been replaced by the maximal number of cylinders and the maximal number of injections.

SOFT_ECU_DIESEL block The parameters for the number of cylinders and the number of injections have been replaced by the maximal number of cylinders and the maximal number of injections.

A new parameter has been added to specify the post injection limit angle.

ASM Engine Gasoline Basic Blockset

New Features of ASM Engine Gasoline Basic Blockset 1.3.2

COMMON_GASOLINE_PARAMETERS block

For using the same real-time code with different cylinder counts, a new parameter, `Const_num_Cyl_vector`, has been added to the `COMMON_GASOLINE_PARAMETERS` block. This parameter is a vector of length 20. For each cylinder, there is a vector element containing a value of 1. For example, if there are four cylinders, the first four vector elements each contain a 1, the remaining elements contain a 0.

Migrating to ASM Engine Gasoline Basic Blockset 1.3.2

SOFT_ECU_GASOLINE_BASIC block

The parameter number of cylinders has been removed.

CATALYST block

The `Const_T1_lambda_sens_a_Cat`, `Const_T2_lambda_sens_a_Cat`, `Const_T1_lambda_sens_b_Cat`, and `Const_T2_lambda_sens_b_Cat` parameters have been moved from the PT2 mask to a constant block.

ASM Engine Gasoline Blockset

New Features of ASM Engine Gasoline Blockset 2.1.2

COMMON_GASOLINE_PARAMETERS block

For using the same real-time code with different cylinder counts, a new parameter, `Const_num_Cyl_vector`, has been added to the `COMMON_GASOLINE_PARAMETERS` block. This parameter is a vector of length 20. For each cylinder, there is a vector element containing a value of 1. For example, if there are four cylinders, the first four vector elements each contain a 1, the remaining elements contain a 0.

Migrating to ASM Engine Gasoline Blockset 2.1

SOFT_ECU_GASOLINE block

The parameters of the number of cylinders and the number of injections have been replaced by the maximal number of cylinders and the maximal number of injections.

DIRECTINJECTOR block

The parameters of the number of cylinders and the number of injections have been replaced by the maximal number of cylinders and the maximal number of injections.

CATALYST block

The `Const_T1_lambda_sens_a_Cat`, `Const_T2_lambda_sens_a_Cat`, `Const_T1_lambda_sens_b_Cat`, and `Const_T2_lambda_sens_b_Cat` parameters have been moved from the PT2 mask to a constant block.

Related topics

Basics

- [Migrating ASM Models \(ASM User Guide\)](#)

ASM Environment Blockset

Migrating to ASM Environment Blockset 1.4

AMBIENT block

The AMBIENT block was added to the ASM_Environment_lib. In previous releases this block existed without any library link. No migration steps are necessary.

BASIC_ROADS block

This block is now dynamically sized to be able to handle more tires for truck and trailer.

The calculation of the z-position was wrong if the lateral and longitudinal slopes were used. This bug has been fixed.

Some inports have been renamed.

Old Inport Name	New Inport Name
Pos_CP_CoorSys_E[FL;FR;RL;RR][x;y][m]	Pos_CP_CoorSys_E[:,x;y][m]
Fric_Coeff_BasicRoad[FL;FR;RL;RR][:]	Fric_Coeff_CP_BasicRoad[:,:]
Sw_Tire_Parameter_Set_BasicRoad[FL;FR;RL;RR][1 2 3 4]	Sw_Tire_Parameter_Set_CP_BasicRoad[:,1 2 3 4]

KEY_STATES block

An inport was renamed.

Old Inport Name	New Inport Name
Sw_Key[0off 1on]	Key[-1_2]

LATERAL_CONTROL1 block

The lateral controller had problems with steering trucks. This has been fixed now. No migration steps are necessary.

MANEUVER_SCHEDULER block

In the MANEUVER_SCHEDULER block, only minor bugfixes have been done. There was a wrong unit conversion in hold segments. The maneuver segment length sometimes differed by 1 ms.

User signals are now held in brake state.

No migration steps are necessary.

ROAD block

This block is now dynamically sized to be able to handle more tires for truck and trailer. In previous versions the ASM_Environment_lib and ASM_Traffic_lib contained a separate version of the ROAD block. Now the ROAD block of the ASM_Environment_lib can handle the traffic features as well.

It is now possible to define a certain lateral offset from the road centerline. Outputs for lateral offset, road width and current road segment were added.

An inport signal was renamed.

Old Inport Name	New Inport Name
Pos_CP_CoorSys_E[FL;FR;RL;RR][x;y][m]	Pos_CP_CoorSys_E[:,x;y][m]

Some output signals were renamed.

Old Output Name	New Output Name
Pos_z_CP_CoorSys_E[FL,FR,RL,RR][m]	Pos_z_CP_CoorSys_E[:,m]
Fric_Coeff[FL,FR,RL,RR][:]	Fric_Coeff_CP[:,:]
UnitVec_z_CP_CoorSys_E[FL,FR,RL,RR][x y z]	UnitVec_z_CP_CoorSys_E[:,x;y;z]
v_Wind_Coor_Sys_E[x,y,z][m/s]	v_Wind_Coor_Sys_E[x;y;z][m/s]
Sw_Tire_Parameter_Set[FL,FR,RL,RR][1 2 3 4]	Sw_Tire_Parameter_Set_CP[:,1 2 3 4]

The block has 3 new inports: LateralOffset_Vehicle[m], s_Fellows[m], d_Fellows[m].

The block has 3 new outputs: Fellow_Signals, Slope_Lateral_Road[deg], RoadInfo.

ASM Gasoline InCylinder Blockset

Migrating to ASM Gasoline InCylinder Blockset 1.0.1

SELECTFUELINJECTION block The SELECTFUELINJECTION block has been added to the library.

ASM Optimizer

New Features of ASM Optimizer 1.3

Migrating optimization projects A migration feature has been added so that prior optimization projects can be used with the current release. The optimization projects and the optimization models are automatically migrated.

Execute Optimizer The file names to save OP_Data after each step (preprocessing, optimization, postprocessing) and the file name to load OP_Data in preprocessing can be selected separately.

Wallheat Optimization The optimization of the wall heat is separated into two tasks to optimize the wall heat flux for compression and expansion individually.

ASMParameterization Tool

New Features of the ASMParameterization Tool 1.4

Parameter Export to ModelDesk You can now export an engine parameter set created in ASMParameterization to a ModelDesk project.

ASM Traffic Blockset

Migrating to ASM Traffic Blockset 1.1.1

ModelDesk access	<p>From this version on, ASM Traffic is parameterized with ModelDesk. The following blocks are prepared for ModelDesk access:</p> <ul style="list-style-type: none"> ■ FELLOW_PARAMETERS ■ RADARSENSOR_3D ■ SENSOR_x_PARAMETERS ■ SENSOR_SELECTOR ■ TRAFFIC_SCHEDULER
TRAFFIC_SCHEDULER block	<p>The traffic scenario parameter is now included in the MDL struct (old: TrafficMatrix, new: MDL.Traffic.TrafficScheduler.Scenario).</p>
TRAFFIC_ROAD_SIGNALS block	<p>There were some strange yaw angle oscillations in the fellow vehicles if relative lateral deviation was used close to standstill. This bug has been fixed.</p>
ROAD block	<p>The ROAD block was removed from the ASM_Traffic_lib. During model migration the library link is changed to the source block ASM_Environment_lib/Environment/Road/ROAD.</p>
Converting traffic scenarios from the TrafficCreator	<p>Traffic scenarios which were created with the TrafficCreator (xls) can be converted for use with ModelDesk.</p> <p>The converter uses the scenario XLS file and the corresponding M file. The following points are required for correct functioning:</p> <ul style="list-style-type: none"> ■ The first sheet in the XLS file is used for conversion. ■ It is assumed that the XLS file and M file are both available and have the same name. ■ The M file is generated from the XLS file. It is assumed that the M file was created with the current data. <p>For scenario conversion, perform the following steps:</p> <ol style="list-style-type: none"> 1. In MATLAB, call <code>asm_traffic_convert_xls2mod</code>. 2. In the File Selection dialog, choose the scenarios to be converted. 3. In ModelDesk, import the converted scenarios.

ASM Traffic Operator Blockset

New Blockset ASM Traffic Operator 1.0

New blockset

ASM Traffic is now available as an operator version. This version is prepared for simulations in Simulink.

ASM Trailer

New Blockset ASM Trailer 1.0

New blockset

ASM Trailer is an add-on to ASM VehicleDynamics. It simulates the dynamics and coupling effects of a trailer hitched to a vehicle. The model represents the trailer's longitudinal, lateral, and vertical dynamics, and simulates suspension kinematics and compliance, tire-road friction forces and moments, brakes, and hitch.

ASM Turbocharger Blockset

New Features of ASM Turbocharger Blockset 1.4.1

SHAFT_TC block	The effect of friction has been added by means of a speed-dependent torque map.
-----------------------	---

Migrating to ASM Turbocharger Blockset 1.4

TURBINE_SAEJ922 block	The unused reference pressure and temperature parameters have been deleted from the block mask. The saturation for mass has been removed to allow a flow in negative flow direction as well.
------------------------------	---

ASM Utilities Blockset

Migrating to ASM Utilities Blockset 1.5.1

UNIT_VECTOR block	Mux and demux have been removed from this block.
ROTATION_MATRIX_ANGLE block	The rotation matrix calculation from the yaw, pitch and roll angle with the rotation sequence z, y and x has been improved.
UNIT_VECTORS block	Two muxes have been removed from this block.

ASM Vehicle Dynamics Blockset

New Features of ASM Vehicle Dynamics Blockset 1.3

SUSPENSION_COMPLIANCE block	Suspension compliance has been 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.
SUS_KIN_SYM_3DOF block, SUS_KIN_ASYM_3DOF block	Two new library blocks have been added to the ASM_VehicleDynamics library: Symmetric Suspension Kinematics 3-DoF and Asymmetric Suspension Kinematics 3-DoF. The suspension kinematics 3-DoF 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.
STEERING block	The block has been extended so that it can also be used with an external electronic 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.
TORQUE_CONVERTER block	The TORQUE_CONVERTER block has been improved so that it can also handle negative pump speeds. The default parameterization for dragging the engine has also been optimized.
BRAKE_DISC block	A new inport for the parking brake function has been added.

Migrating to ASM Vehicle Dynamics Blockset 1.3

SUSPENSION_COMPLIANCE blockset	In the new SUSPENSION_COMPLIANCE blockset, new parameters for the extended compliance have been added.
MC_PHERSON_STRUT block	Additional signals have been added to ASMSignalBus.

SEMI_TRAILING_ARM block	Additional signals have been added to ASMSignalBus.
RIGID_AXLE	Additional signals have been added to ASMSignalBus.
TIRE_MODEL_TMEASY block	<ul style="list-style-type: none">■ The lower and upper bound calculation of Bore Torque have been 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 has been 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.
CONTACT_POINT_CALCULATION block	The Pos_CP_CoorSys_E output signal is now a 1D-array Simulink signal type instead of a row/column matrix.
COORDINATE_TRANSFORMATION block	<ul style="list-style-type: none">■ 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 has been improved.
VEHICLE_MOVEMENT_INFO_CAR block	The internal signal bus has been restructured. This will improve the initialization time for Simulink simulation.
CLUTCH block	The extrapolation for the Map_Stiffness lookup-table has been changed from constant to linear.
STEERING block	In the new STEERING block, the additional inputs are connected to default values. They have been chosen so that the change does not affect the simulation result. No additional migration steps are required.
TORQUE_CONVERTER block	In the new TORQUE_CONVERTER block, there are only changes inside the Library block (see <i>New Features of ASM Vehicle Dynamics Blockset 1.3</i> on page 39). If you want to use the new default parameterization, you must transfer it to old projects from the demo project manually.

BRAKE_DISC block

The additional inputs are connected to default values. They have been chosen so that the change does not affect the simulation result. No additional migration steps are required.

**VEHICLE_MOTION_CAR
block**

Some Simulink Reshape blocks have been deleted.

There is a new inport for virtual test bench mode switch.

CalDesk

Where to go from here

Information in this section

<i>New Features of CalDesk 3.0</i>	44
<i>Migrating to CalDesk 3.0</i>	59

New Features of CalDesk 3.0


Where to go from here

Information in this section

<i>New Automation Interface</i>	44
<i>New Project and Experiment Features</i>	45
<i>New Platform Management Features</i>	46
<i>New Variable Management Features</i>	47
<i>New Instruments and Instrument Features</i>	48
<i>New Visualization Features (Common to all Instruments)</i>	53
<i>New Measurement and Recording Features</i>	54
<i>New Data Set Management Features</i>	55
<i>New Features of the Variable Editor</i>	55
<i>New Features of the CalDesk ECU Diagnostics Module</i>	56
<i>New ECU Flash Programming Tool Features</i>	57
<i>Further Enhancements with CalDesk 3.0</i>	58

New Automation Interface

Features of CalDesk's automation interface CalDesk lets you automate most of its features via its new automation interface:

- Creating a new project and experiment
- Adding and configuring a platform/device
- Adding and configuring layouts and instruments
- Connecting variables to instruments
- Configuring measurements and recordings
- Accessing and editing data sets
- Customizing the user interface
- For details and instructions on using the automation interface, refer to *Automating CalDesk* ( *CalDesk Calibration Guide*).
- For reference information, refer to the *CalDesk API Reference*.

**WARNING****Risk of serious injury and/or property damage**


Using the CalDesk software can have a direct effect on networked electronic systems connected to it.

Improper or negligent use can result in serious personal injury and/or property damage. This risk also exists when CalDesk is used via the automation interface.

New Project and Experiment Features

Version control support

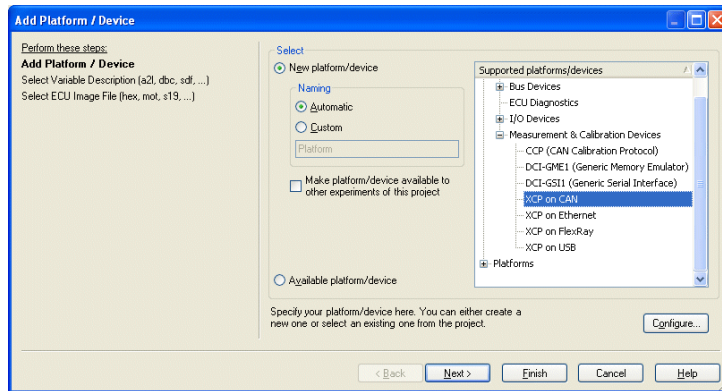
CalDesk 3.0 lets you put projects under version control. The CalDesk version control interface supports version control systems that provide a Microsoft Source Code Control (SCC) client. The SCC client access includes a standardized set of operations, but the details of the version control systems vary.

For details and instructions, refer to *Using a Version Control System* ( *CalDesk Calibration Guide*).

New Platform Management Features

Structured platform/device selection

CalDesk provides a structured platform/device selection in the Add Platform / Device dialog. The platforms/devices are structured according to their different use cases.



Support of Kvaser Leaf Professional and LAPcan II

CalDesk now also supports the Leaf Professional and LAPcan II interfaces from Kvaser.

For lists of the supported interfaces, refer to *Supported CAN Interfaces* (📖 *CalDesk Calibration Guide*) and *Supported LIN Interfaces* (📖 *CalDesk Calibration Guide*).

Observing variables

CalDesk can observe variables on DS1005 and MicroAutoBox platforms to detect value changes even if they are not caused by CalDesk, and even if no measurement is currently running.

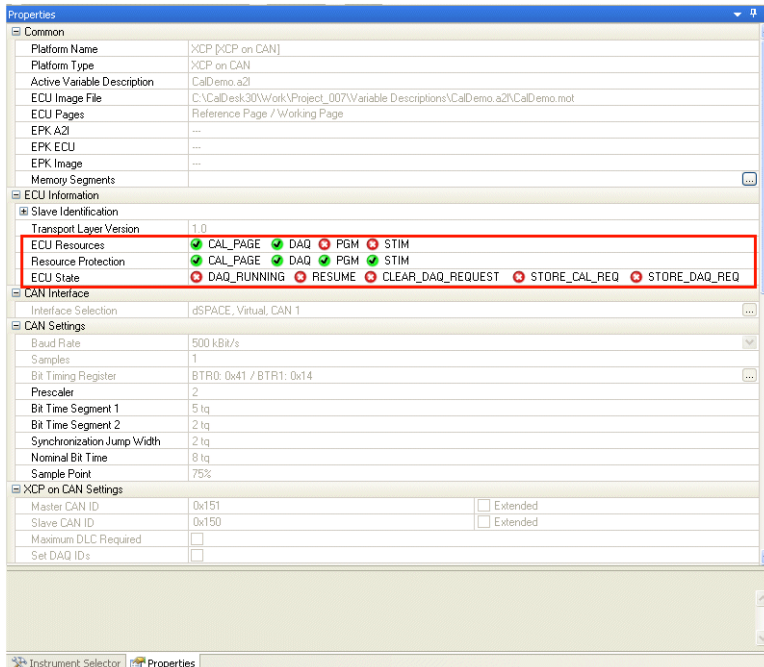
CalDesk's variable observer functionality can be used for these purposes:

- Detecting parameter value changes of adaptive parameters
- Detecting parameter value changes caused from outside CalDesk, for example, by using ControlDesk or AutomationDesk
- Performing data acquisition of measurement variables without using a measurement raster to reduce bandwidth and measurement buffer size

For details, refer to *Observing Variables* (📖 *CalDesk Calibration Guide*).

Platform/device configuration via Properties controlbar

Platforms/devices can now be configured via CalDesk's Properties controlbar. The controlbar also gives you quick information on the current ECU state, etc.



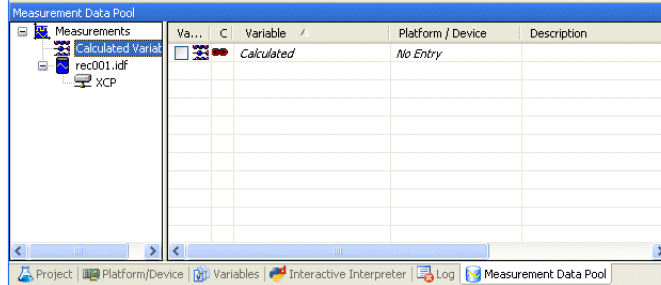
New Variable Management Features

Improvements for calculated variables

CalDesk 3.0 comes with improvements for calculated variables:

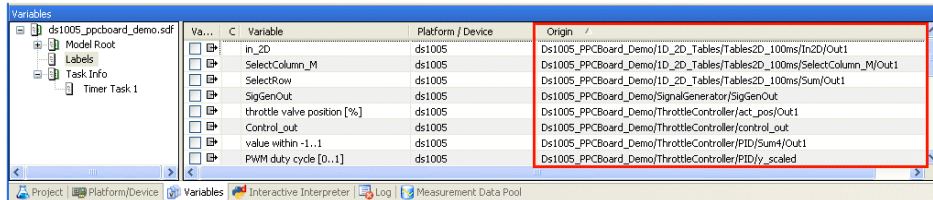
- If you export a CalDesk experiment containing calculated variables, the variables and their associated formulas are also exported. Calculated variables do not have to be exported and imported separately any more.

- You can create calculated variables based on variables in recorded data files. This allows you to also use calculated variables during postprocessing.



Display of Simulink labels in the Variable Browser

CalDesk's Variable Browser now displays the Simulink label of variables, and the variable's path in the Simulink model. This allows you to unambiguously identify Simulink labels, since they can reside in several subsystems of the Simulink model.



New Instruments and Instrument Features

Information in this topic

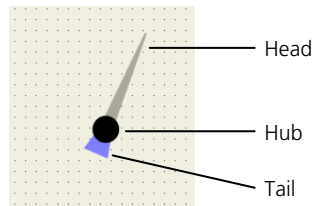
New instruments on page 49

- Animated Needle on page 49
- Check Button on page 49
- Display on page 49
- Frame on page 50
- Invisible Switch on page 50
- MultiState Display on page 50
- Numeric Input on page 50
- OnOff Button on page 51
- Push Button on page 51
- Radio Button on page 51
- Selection Box on page 52
- Static Text on page 52
- Plotter Array on page 52

New instruments

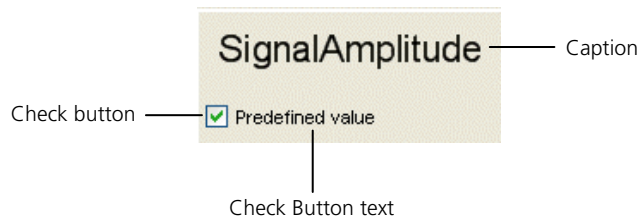
CalDesk 3.0 provides the following new instruments:

Animated Needle An instrument for displaying the value of a connected variable by a needle deflection.



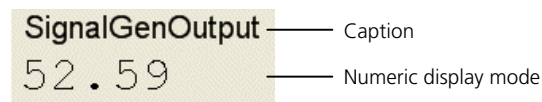
For details, refer to *Animated Needle* ( *CalDesk Calibration Reference*).

Check Button An instrument for setting the value of the connected parameter to a predefined value when you activate the check button (On value) and when you clear it (Off value).



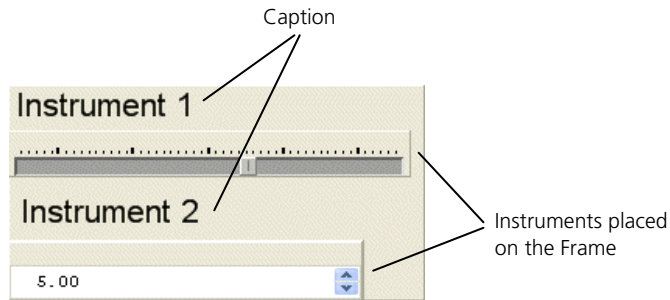
For details, refer to *Check Button* ( *CalDesk Calibration Reference*).

Display An instrument for displaying variable values numerically.



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

Frame An instrument for adding a background frame to a layout, for example, to visualize an instrument group.

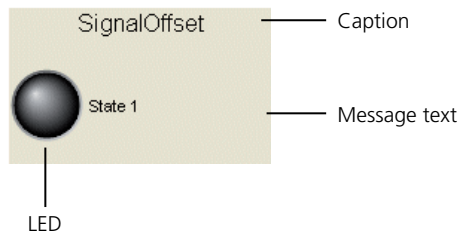


For details, refer to *Frame* ( *CalDesk Calibration Reference*).

Invisible Switch An instrument for defining an area that is sensitive to mouse operations.

For details, refer to *Invisible Switch* ( *CalDesk Calibration Reference*).


MultiState Display An instrument for displaying the value of a variable as an LED state and/or as a message text.



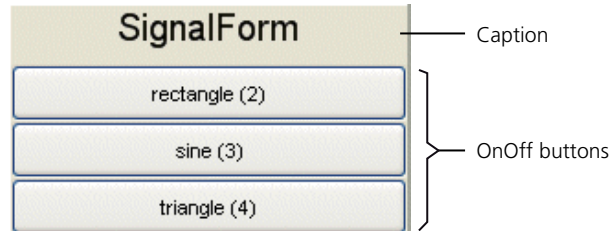
For details, refer to *MultiState Display* ( *CalDesk Calibration Reference*).


Numeric Input An instrument for displaying and setting the value of the connected variable numerically.



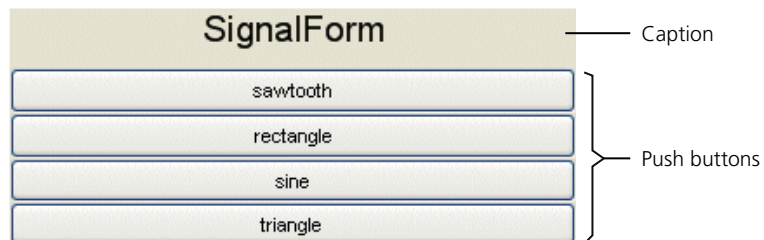
For details, refer to *Numeric Input* ( *CalDesk Calibration Reference*).

OnOff Button An instrument for setting the value of the connected parameter to a predefined value when you press the button (On value) and when you release the button (Off value).



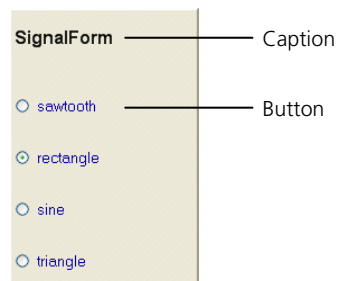
For details, refer to *OnOff Button* ( *CalDesk Calibration Reference*).

Push Button An instrument for setting the value of the connected parameter by push buttons.



For details, refer to *Push Button* ( *CalDesk Calibration Reference*).

Radio Button An instrument for displaying and setting the value of the connected parameter by radio buttons.



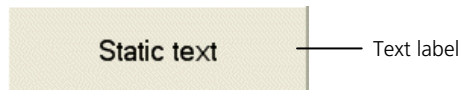
For details, refer to *Radio Button* ( *CalDesk Calibration Reference*).

Selection Box An instrument for selecting a text-value entry and setting the respective numerical value for the connected variable.



For details, refer to *Selection Box* ( *CalDesk Calibration Reference*).

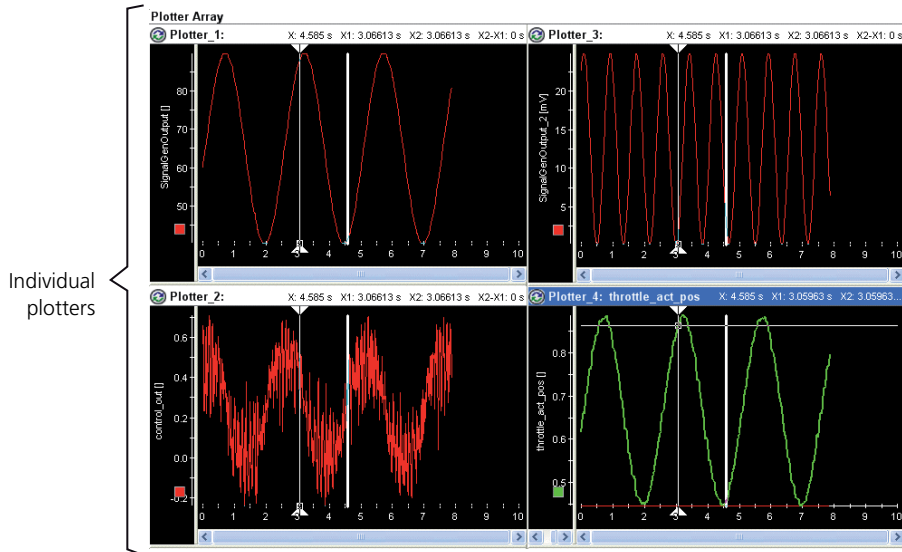
Static Text An instrument for displaying explanations or inscriptions on the layout.



For details, refer to *Static Text* ( *CalDesk Calibration Reference*).

Plotter Array

You can group several Plotters via the *Plotter Array*. You can customize the number and layout of the Plotters.

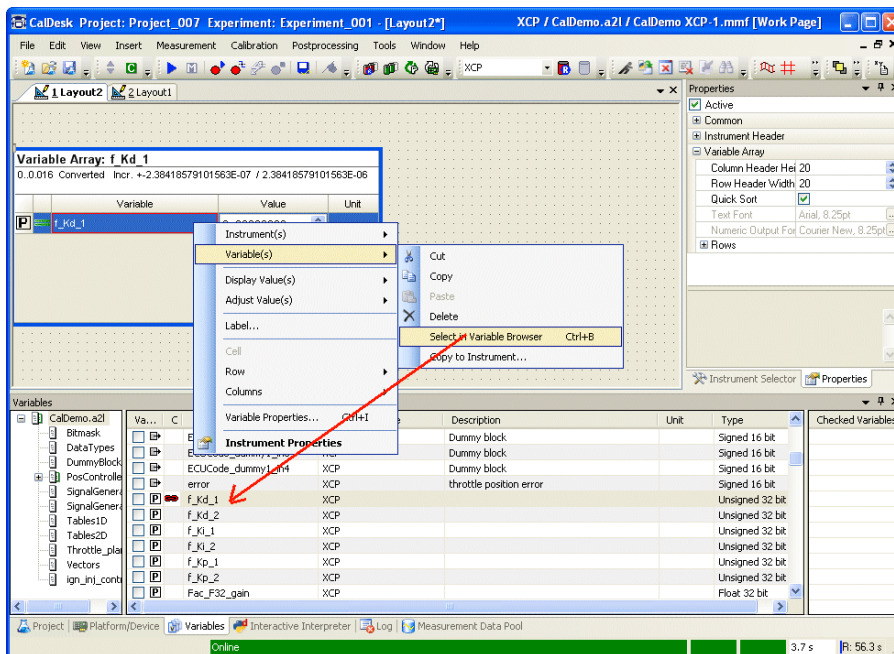


For details, refer to *Plotter Array* ( *CalDesk Calibration Reference*).

New Visualization Features (Common to all Instruments)

Highlighting variables in the Variable Browser

CalDesk now lets you highlight variables visualized in instruments in the Variable Browser.



Disabling instruments

You can enable and disable each instrument individually. If disabled, an instrument is grayed out on the layout, and the instrument properties are read-only.

Refer to *Active Property* (*CalDesk Calibration Reference*).

Highlighting instruments without connections

You can highlight the instruments on the current layout that are not connected to variables.

Refer to *Highlight Missing Connections* (*CalDesk Calibration Reference*).

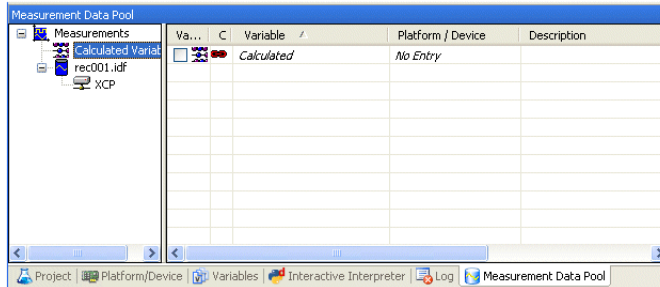
Using the same layouts for rapid prototyping and ECU calibration

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

New Measurement and Recording Features

Using calculated variables for postprocessing

You can create calculated variables based on variables in recorded data files. This allows you to also use calculated variables during postprocessing.



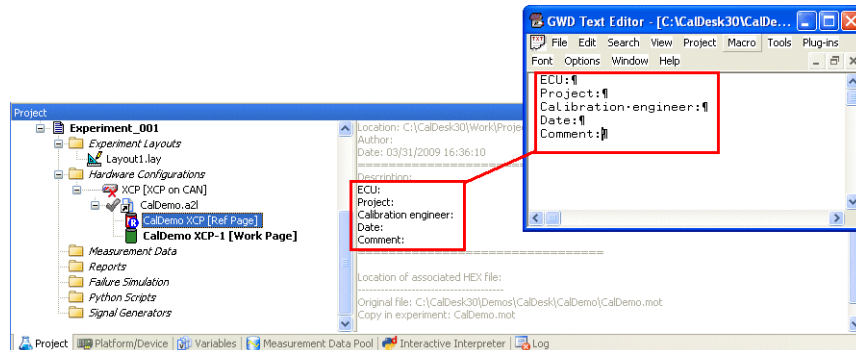
Description for measurement files shown in the Project Manager

CalDesk's Project Manager now displays the description of a measurement data file when you select the file in the Project Manager.

New Data Set Management Features

Defining description templates

CalDesk 3.0 allows you to use templates for data set descriptions. The template is a user-defined text file.



For details, refer to *How to Create a Description Template* ( *CalDesk Calibration Guide*).

New Features of the Variable Editor

The *Variable Editor 1.3* in CalDesk 3.0 provides the following new feature:


Updating A2L files by map files directly in CalDesk


You can now also update address information in an A2L file via map files when you use the Variable Editor as an integrated component of CalDesk.

For details and instructions, refer to *Working with Symbol Addresses and Map Files* ( *Variable Editor Guide*).

Structuring variables in groups via variable links

In the hierarchy tree of the Variable Editor, all the variables of an A2L file are displayed in the root node of the A2L file. You can create groups to organize the variables. Groups represent the functions (FUNCTION keyword) in an A2L file, and collect variables that belong to the same functionality.

Variable links You can copy a variable, and now paste it into another variable group as a *variable link*. The copied variable appears in the groups as a reference to the original. Variable references are marked by an arrow, for example,  indicates a reference to a parameter. No new variable is created when you paste a variable as a link.

For details and instructions, refer to *How to Cut, Copy, and Paste Variables* ( *Variable Editor Guide*).

New Features of the CalDesk ECU Diagnostics Module


Support for the Automatic Reconnect feature

CalDesk's ECU Diagnostics device now also supports the *Automatic Reconnect* feature.

For details on the feature, refer to *Reconnecting to Platform/Device Hardware Automatically* ( *CalDesk Calibration Guide*).


Support for Kvaser CAN interfaces

CalDesk's ECU Diagnostics device now supports Kvaser CAN interfaces.

For a list of the supported CAN interfaces, refer to *Supported CAN Interfaces* ( *CalDesk Calibration Guide*).



OBD support

The CalDesk ECU Diagnostics Module now also supports the OBD (On-Board Diagnostics) diagnostic protocol for CAN-based ECU diagnostics.


For an overview of the supported diagnostic protocols, refer to *Basics of ECU Diagnostics Devices* ( *CalDesk Calibration Guide*).


Configuring the read service selection in the Fault Memory Instrument

You can now configure which services for reading environment data, clearing single fault memory entries and clearing all fault memory entries are selectable for a logical link in CalDesk's Fault Memory Instrument.

For details, refer to *Identifying Services and Parameters for CalDesk Functions* ( *CalDesk Calibration Guide*) and *Logical Links Properties* ( *CalDesk Calibration Reference*).

Configuring COMPARAMs dynamically

You can configure communication parameters (COMPARAMs) dynamically via the `ProtocolParameterSet` control primitive. This allows you, for example, to disable the cyclic transmission of `TesterPresent` indications to check the reaction of the connected ECU. For details, refer to *Conventions in Connection with ODX Databases* ( *CalDesk Calibration Reference*).

To configure communication parameters dynamically, you have to select the `Show Control Primitives` property for the Diagnostics instrument for this. For details, refer to *Tree View Properties* ( *CalDesk Calibration Reference*).

New ECU Flash Programming Tool Features

The *dSPACE ECU Flash Programming Tool 2.1* in CalDesk 3.0 provides the following new feature:

Programming the ECU flash memory without using a boot check function

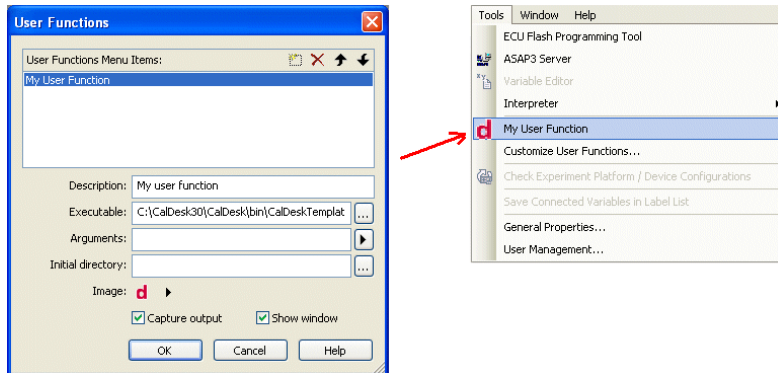
Via DCI-GSI1, you can also perform ECU flash programming without using a boot check function. This allows you to program the flash memory of brain-dead ECUs, i.e., ECUs with an empty flash memory or invalid flash memory contents.


For details, refer to *Principle of ECU Flash Programming with the DCI-GME1 or DCI-GSI1* ( *ECU Flash Programming*).

Further Enhancements with CalDesk 3.0

Integrating user functions

CalDesk allows you to embed external applications or additional functions as *user functions*. A user function appears as a new button in CalDesk's toolbar and a new menu item in CalDesk's Tool menu.



For details and instructions, refer to *Adding User Functions to CalDesk* ( *CalDesk Calibration Guide*).

Migrating to CalDesk 3.0

Where to go from here

Information in this section

<i>Migrating to CalDesk 3.0</i>	59
<i>How to Migrate a Project to CalDesk 3.0</i>	60
<i>Migrating CalDesk 2.1 Experiments with Curves</i>	62
<i>How to Migrate a Variable Editor Project (VEP File)</i>	63
<i>Automatic Reconnect Feature: Changed Default Settings</i>	64

Migrating to CalDesk 3.0

CalDesk

To migrate to CalDesk 3.0 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	3.0
From Version ...								
1.2.0 ...	No	No	No	Yes ¹⁾	Yes ¹⁾	Yes ¹⁾	Yes ¹⁾	Yes ²⁾
1.2.1 ...	–	No	No	Yes ¹⁾	Yes ¹⁾	Yes ¹⁾	Yes ¹⁾	Yes ²⁾
1.2.2 ...	–	–	No	Yes ¹⁾	Yes ¹⁾	Yes ¹⁾	Yes ¹⁾	Yes ²⁾
1.3.0 ...	–	–	–	Yes ¹⁾	Yes ¹⁾	Yes ¹⁾	Yes ¹⁾	Yes ²⁾
1.4.0 ...	–	–	–	–	No	Yes ¹⁾	Yes ¹⁾	Yes ²⁾
1.4.1 ...	–	–	–	–	–	Yes ¹⁾	Yes ¹⁾	Yes ²⁾
2.0 ...	–	–	–	–	–	–	Yes ¹⁾	Yes ²⁾
2.1 ...	–	–	–	–	–	–	–	Yes ²⁾³⁾

¹⁾ For the migration steps, refer to <http://www.dspace.com/goto?VersionsCal>.

²⁾ Refer to *How to Migrate a Project to CalDesk 3.0* on page 60.

³⁾ Refer to *Migrating CalDesk 2.1 Experiments with Curves* on page 62.

Variable Editor

To migrate to Variable Editor 1.3 on CalDesk 3.0 and reuse existing Variable Editor projects, refer to *How to Migrate a Variable Editor Project (VEP File)* on page 63.

How to Migrate a Project to CalDesk 3.0

If a project was created before CalDesk 3.0, you have to migrate it to reuse it (and its experiments) in CalDesk 3.0.

Possible methods


You can migrate a project in two ways:

- Using CalDesk. Refer to Method 1.
- Using the *CalDesk Project Converter*. Refer to Method 2.

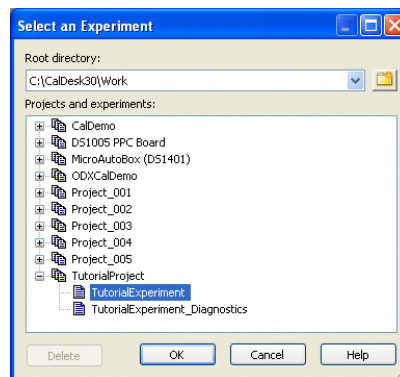


A project (and its experiments) last saved with CalDesk 1.3 or before cannot be migrated *directly* using the methods described below. Instead, you first have to migrate such a project to CalDesk 1.4 or 1.4.1, and then use one of the methods described below.

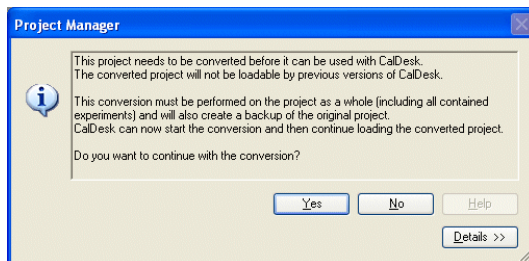
Method 1**To migrate a project to CalDesk 3.0 via CalDesk**

- 1 Start CalDesk 3.0.
- 2 From the File menu, select Open - Project + Experiment, or press **Ctrl+Shift+O**, or click .

CalDesk opens the Select an Experiment dialog.



- 3 From the **Root** directory drop-down list, select the project root directory containing the project and experiment you want to migrate.
- 4 From the **Projects and experiments** list, select the project and experiment you want to migrate.
- 5 Click **OK**.
- 6 CalDesk displays the following dialog:

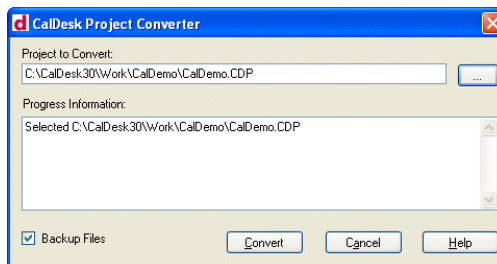


- 7 Click **Yes**.
CalDesk migrates the project and all its experiments, and then opens the selected experiment.

Method 2

To migrate a project to CalDesk 3.0 via CalDesk Project Converter

- 1 Run `CalDeskProjectConverter.exe`. After installation of CalDesk 3.0, it is located in the `%DSPACE_ROOT%\CalDesk\Bin` folder. The CalDesk Project Converter opens.
- 2 In the CalDesk Project Converter, select the project to be converted.



- 3 Click **Convert**.
CalDesk migrates the project and all its experiments.

Result

The project and all its experiments are ready for use with CalDesk 3.0.



Once a project is migrated, you can no longer use it with a CalDesk version earlier than 3.0. For this reason, CalDesk stores a backup of the original project in the project root folder during migration. You can use the backup project with a CalDesk version earlier than 3.0.

Related topics

HowTos

- [How to Open a Project and Experiment](#) ( *CalDesk Calibration Guide*)

References

- [Open Project + Experiment](#) ( *CalDesk Calibration Reference*)

Migrating CalDesk 2.1 Experiments with Curves

When you reuse an experiment created with CalDesk 2.1, the connections to curves are lost if the following conditions are true:

- The curves originate from an SDF file, that means, from entries made for Lookup Table (n-D) blocks in Simulink models for a DS1005 or MicroAutoBox platform.
- The curves were not defined as tunable parameters in the related Simulink model.
- You reload the variable description or add the variable description again.

To work with the curves in CalDesk 3.0, you have to reconnect them. The LOG file contains an entry for each connection that is lost.

How to Migrate a Variable Editor Project (VEP File)

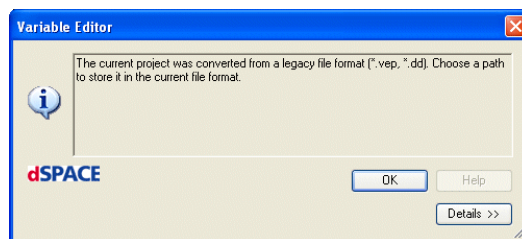
If a Variable Editor project (VEP file) was created before Variable Editor 1.3, you have to save it as a VDB file to reuse it in Variable Editor 1.3.

Method

To migrate a Variable Editor project (VEP file)

- 1 Start Variable Editor 1.3.
- 2 From the Variable Editor's File menu, select Open Variable Editor Project.
The Open Variable Editor Project dialog is opened.
- 3 In the dialog, select VEP in the Files of type drop-down list, then select the project you want to migrate. Click Open.
- 4 From the Variable Editor's File menu, select Save Variable Editor Project.

The Variable Editor displays the following dialog:



- 5 Click OK.
The Save Variable Editor Project As dialog is opened.
- 6 Specify the name and path of the project, then click Save.
The Variable Editor saves the project as a VDB file.

Result

The project is ready for use with Variable Editor 1.3.



Once a Variable Editor project is migrated and saved as a VDB file, you can no longer use it with a Variable Editor version earlier than 1.3. However, the original VEP file is not removed and can be used with a Variable Editor version earlier than 1.3.

Automatic Reconnect Feature: Changed Default Settings

Resuming measurement automatically when switching the ECU off and on again

Since version 2.1, CalDesk provides the *Automatic Reconnect* feature for automatically reconnecting to device/platform 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.

For details on the feature, refer to *Reconnecting to Platform/Device Hardware Automatically* ( *CalDesk Calibration Guide*).

Automatic Reconnect now enabled by default

The table below shows the default settings in CalDesk 2.1 and 3.0.

Version	Automatic Reconnect Behavior Default Setting	Resume Online Calibration Behavior Default Setting
CalDesk 2.1	Disconnect Platform/Device ¹⁾	-
CalDesk 3.0	Resume measurement and calibration ²⁾³⁾	Ignore differences ⁴⁾

¹⁾ CalDesk does not try to reestablish the logical connection for the platform/device, but sets the platform/device state to disconnected.

²⁾ CalDesk periodically tries to reestablish the logical connection. When the platform/device is reconnected to the platform/device hardware, CalDesk resumes measurement and online calibration.

³⁾ The ECU Diagnostics device is an exception: The default setting remains "Disconnect Platform/Device" for this device.

⁴⁾ Online calibration is started without a data consistency check and without equalizing the contents of the memory pages and the mirrored memory. CalDesk lets you calibrate parameters even if the parameter values on the connected hardware and on the host PC are different. *Note:* You must ensure data integrity yourself.

The default settings are applied to platforms/devices in the following cases:

- You add a new platform/device to the experiment.
- You reuse a project created with CalDesk 2.0 or earlier in CalDesk 3.0.

ConfigurationDesk

Migrating to ConfigurationDesk 3.0.1

Objective

Applications created with ConfigurationDesk versions prior to version 1.2. must be migrated to achieve upward compatibility with ConfigurationDesk 3.0.1.

Where to go from here

Information in this section

<i>Migration to ConfigurationDesk 3.0.1</i>	66
<i>How to Migrate a ConfigurationDesk Project with the RapidPro Projects Migrator</i>	68

Migration to ConfigurationDesk 3.0.1

Objective

Applications created with ConfigurationDesk 1.1.x cannot be used with ConfigurationDesk 3.0.1 as they are incompatible with this ConfigurationDesk version.

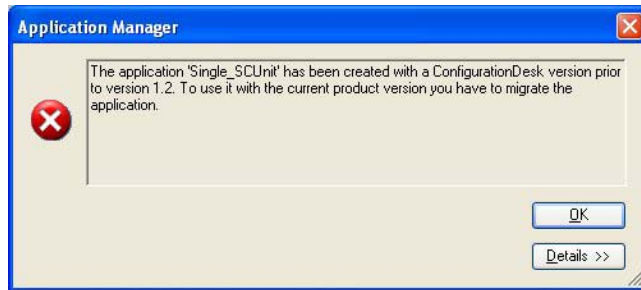
Upward compatibility

ConfigurationDesk can convert these applications to achieve upward compatibility. Downward compatibility is not implemented.

Application Created with ...	Compatible with ConfigurationDesk 1.1.2 or Earlier	Compatible with ConfigurationDesk 3.0.1
ConfigurationDesk 1.1.2 or earlier	Yes	Yes, after migration
ConfigurationDesk 1.2 or later	No	Yes

Migrating an old project/application

When you open a project that contains an old application, or when you import an old application, ConfigurationDesk prompts you to migrate the application:



If you cancel the conversion, the application and its old subfolders are not shown in the Project Manager.

After an old project has been loaded and converted, you can configure its applications and add new ones.

Creating application based on old hardware data

When you are creating a new application and the Project Wizard prompts you to add the hardware topology, you cannot use an old HTF file as the data source. The same applies when the Project Wizard prompts you to add the configuration settings. You cannot use an old HCF file as a data source either. It is therefore recommended to migrate only complete projects or applications.



If you only have an old HTF or an old HCF file which is part of a project or an (exported) application, contact dSPACE Support.

Files affected by migration

An old application (ConfigurationDesk 1.1.2 or earlier) comprises a hardware topology file (HTF file) and in some cases a hardware configuration file (HCF file) depending on whether configuration settings were specified. During migration, ConfigurationDesk converts an HTF file to a new HTF file. An HCF file is copied, and the copy is converted to a CDS file. The project's folder structure is not affected by migration. Thus, all the files are stored in the same subfolders as before.



For ConfigurationDesk 1.1.2 or earlier, the parameters of the user-configurable circuits of a module are stored in HCF files. Since ConfigurationDesk 1.2 these parameters are stored in HTF files. As a result, if an application that does not comprise an HCF file is converted, the parameters of the user-configurable circuits are lost.






CDL files (application) and CDP files (project) are also modified during migration.

How to Migrate a ConfigurationDesk Project with the RapidPro Projects Migrator

Objective Projects and applications created with ConfigurationDesk 1.1.x are not compatible with ConfigurationDesk 3.0.1. You must therefore migrate them with ConfigurationDesk's RapidPro Projects Migrator.

RapidPro Projects Migrator To facilitate migration, ConfigurationDesk provides the RapidPro Projects Migrator. It is a stand-alone tool which migrates projects and applications created with ConfigurationDesk 1.1.x to ConfigurationDesk 3.0.1. You can find the RapidPro Projects Migrator at
 %DSPACE_ROOT%\ConfigurationDesk\RapidProProjectsMigrator\Bin\RapidProProjectsMigrator.exe.

Symbols used in the RapidPro Projects Migrator The RapidPro Projects Migrator provides several symbols to indicate the status of the migration. The following symbols are used:

Symbol	Description
	The project was already migrated or does not need to be migrated to use it with ConfigurationDesk 3.0.1.
	The project needs to be migrated to use it with ConfigurationDesk 3.0.1.
	The migration was successful.
	The migration failed, but the affected project is not defective. It was automatically restored as all the projects to be migrated are backed up before migration. If migration is successful, the backup files are automatically deleted.
	The project is defective or is not a ConfigurationDesk project.

You can also find this list of symbols on the Help tab in the RapidPro Projects Migrator.

Migrating several projects The RapidPro Projects Migrator allows you to migrate several projects at a time. If you do so, migration may take some time.

Recommendation

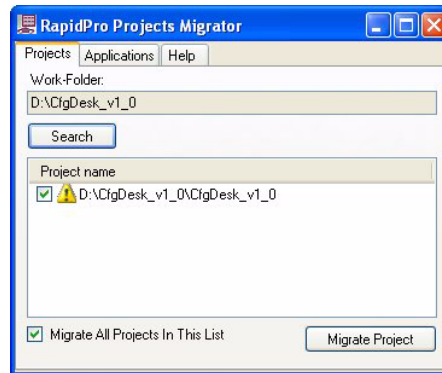
It is recommended to back up the projects you want to migrate. This is for safety purposes only.


Method**To migrate a ConfigurationDesk Project with the RapidPro Projects Migrator**

- 1 Start the RapidPro Projects Migrator.
- 2 On the Projects page, search for the work folder where the project(s) you want to migrate are located.




If you want to migrate applications only, switch to the Applications page of the RapidPro Projects Migrator.



The project located in the selected work folder is displayed in the Project name list. An automatically activated checkbox next to the project name indicates that the project will be considered for migration. The symbol  next to the project name indicates that the project needs to be migrated.

- 3 Click Migrate Project.

Result

The selected project is successfully migrated, which is indicated in the Project name list by the following symbol .



The project can now be used with ConfigurationDesk 3.0.1.


ControlDesk

New Features of ControlDesk 3.4

ControlDesk


Supporting new hardware ControlDesk supports the new boards: DS4004 HIL I/O Board and DS802 PHS Link Board. The Platform Manager displays the new boards when they are installed in a modular system based on DS1005 or DS1006. Board properties, for example, version information, are also displayed.

For information on the features of the DS4004, refer to *DS4004 Features*.

For information on use scenarios, installation notes, etc., of the DS802, refer to *Partitioning a Modular System with the DS802 PHS Link Board* ( *DS100x Hardware Installation and Configuration Guide*).

Data Capturing The data capturing of ControlDesk was enhanced:

- You can preserve the capture variables and capture settings when the simulation model (SDF file) is replaced.
- You can export and import capture variables.
- You can now remove all, selected, or invalid capture variables.
- Stream To Disk data acquisition has a new autaname mode.

Refer to *Working with Data Captures* ( *ControlDesk Experiment Guide*).


Variable description file You can now use variable description files (TRC files) which contain strings longer than 128 characters, for example, block and variable names.

Bus Navigator

CAN Navigator renamed Bus Navigator The CAN Navigator, ControlDesk's component to handle CAN messages, has been renamed to Bus Navigator.


Enhancements of the Bus Navigator tree

- The Bus Navigator tree displays additional nodes for the hardware system (single- or multi-processor system), the application loaded to the processor(s), bus monitoring, data logging, and replaying logged data.
- You can modify the order of the different node types in the Bus Navigator to represent the hierarchical dependencies of the nodes.
- All the icons of the node types are updated.

For further information on the structure of the Bus Navigator tree, refer to *Structure of the Bus Navigator Tree* ( *ControlDesk Experiment Guide*).

Enhancements of bus communication logging


- Bus communication logging is independent of bus monitoring, i.e. you can log bus communication and write the logged data to a log file without defining bus monitoring.
- When bus communication logging is selected, a logger node is added to the Bus Navigator tree. The logger node is added below the node that data logging was selected for. Logger nodes can be added to system, hardware, controller, or filter nodes.
- Each logger node lets you configure settings for data logging that apply to the selected node only.

For further information on data logging, refer to *Monitoring, Logging and Replaying a CAN Bus Communication* ( *ControlDesk Experiment Guide*).

Enhancements of bus monitoring


- When bus monitoring is selected, a monitor node is added to the Bus Navigator tree. The monitor node is added below the node that monitoring was selected for. Monitor nodes can be added to system, hardware, controller, or filter nodes.
- The bus communication of different CAN controllers can now be monitored in one go. For example, you can add an unfiltered monitor node to the system node in the Bus Navigator tree, and in the generated monitor list, the monitored data can be sorted by controller, etc..
- Each monitor node lets you configure local monitoring settings that apply to the selected node only.

- Monitoring can be started:
 - Separately for each monitor node of the Bus Navigator tree.
 - Automatically for activated monitor nodes if monitoring was started for a higher-level monitor node.
 - Globally for all the monitor nodes of the Bus Navigator.

For further information on bus monitoring, refer to *Monitoring, Logging and Replaying a CAN Bus Communication* ( *ControlDesk Experiment Guide*).

Enhancements of bus replay

- When bus replay is selected, a replay node is added to the Bus Navigator tree. The replay node is added below the controller node that replay was selected for.
- Replaying the raw data of a logged CAN bus communication is now supported by MicroAutoBox.
- Replaying can be started separately for each activated replay node or globally for all the activated replay nodes of the Bus Navigator tree.

For further information on bus replay, refer to *Monitoring, Logging and Replaying a CAN Bus Communication* ( *ControlDesk Experiment Guide*).


Exporting databases to an XML file You can export the real-time databases (RTDB) or user-defined databases (UDDB) that are available in the Bus Navigator tree to an XML file. This XML file can be used to define a new UDDB, for example.

dSPACE FlexRay Configuration Package


New Features of dSPACE FlexRay Configuration Package 2.1


FlexRay Configuration Tool


Supported FIBEX version The FlexRay Configuration Tool now supports FIBEX 3.0.

Configuring the TRC file A variable description file (TRC file) is used by ControlDesk to access the signals and frames if they are configured for TRC file access. In the FlexRay Configuration Tool, you can configure several signals to be included in the TRC file. For details, refer to *Basics of Code Generation* ( *FlexRay Configuration Tool Guide*).

RTI FlexRay Configuration Blockset

Sending and Receiving PDUs PDU-based modeling is possible if the FlexRay configuration has specified PDUs (protocol data units) which comprise several signals. You can handle several signals using one Simulink block when using PDU-based modeling. For details, refer to *PDU-Based Modeling* ( *FlexRay Configuration Features*).


Wakeup pattern You can send a wakeup pattern to the FlexRay bus using the RTIFLEXRAYCONFIG TX WAKEUP block. For details, refer to *How to Send Wakeup Patterns* ( *FlexRay Configuration Features*).

You can use a wakeup pattern received by the FlexRay bus to trigger a subsystem. For details, refer to *How to Use FlexRay-Specific Interrupts* ( *FlexRay Configuration Features*).

ModelDesk

New Features of ModelDesk 2.2

Traffic Editor

The Traffic Editor is ModelDesk's new component to create and edit traffic scenarios for a vehicle simulated by an ASM model (ASM vehicle). A traffic scenario specifies the movements of traffic participants and static objects around the ASM vehicle. The movements or positions of the traffic participants and other objects can be specified relative to the ASM vehicle or to the road. In traffic scenarios, you can simulate traffic situations such as vehicles overtaking and changing lanes. For details on the Traffic Editor, refer to *Traffic Scenario Creation* ( *ModelDesk Guide*).


New supported ASM models

The following ASM models are now supported by ModelDesk:

- ASM Engine Gasoline Basic
- ASM Engine Gasoline
- ASM Engine Diesel
- ASM Trailer
- ASM Traffic

For details on the parameters of the new supported ASM models, refer to *ModelDesk ASM Parameters Reference*.

New license concept

ModelDesk has a new license concept. This allows to match the ModelDesk version to the ASM model which is parameterized. This removes irrelevant functionalities. A low-priced version of ModelDesk allows to parameterize ASM models for Simulink simulation. For details, refer to *ModelDesk Licenses* ( *ModelDesk Guide*).

MotionDesk


New Features of MotionDesk 2.1.3

New license concept

MotionDesk is now available with different licenses.

MotionDesk Offline The license is a low-priced version which allows to use MotionDesk in a Simulink simulation. This license does not allow to use MotionDesk in real-time simulation.

MotionDesk HW-Access The license allows to use MotionDesk in a Simulink simulation and a real-time simulation with dSPACE hardware.

For details on the licenses, refer to *MotionDesk Licenses* ( *MotionDesk 3-D Online Visualization Guide*).

RTI and RTLib

New Features of RTI/RTI-MP and RTLib

RTI Gigalink Blockset

The RTI-MP Blockset was extended by the RTI Gigalink Blockset. It provides blocks for implementing data transfer between different real-time applications using DS1005 and/or DS1006 boards. Signals can be transmitted to processor boards of the same type or of different types. Thus, it is possible to connect a DS1006 multiprocessor system with a DS1005 processor board running a single-processor application. Note that all processor boards must be equipped with Gigalink modules. The consistency of data transfer is ensured by the unsynchronized swinging buffer protocol.

The RTI Gigalink Blockset contains the following blocks:

Gigalink_Send block To send signals to a specified Gigalink channel.

Gigalink_Receive block To receive signals from a specified Gigalink channel.

Gigalink_Status block To return the synchronization status of a specified Gigalink.

For details on the RTI Gigalink Blockset, refer to *RTI Gigalink Blockset Reference* ( *RTI and RTI-MP Implementation Reference*).

Including objects in the variable description file

A new variable description file option is available, named `Include only Simulink.Parameter` and `Simulink.Signal` objects with global storage class. This option lets you significantly reduce the time needed for code generation as it disables all the other variable description file options. It includes only parameters and signals in the variable description file which reference a `Simulink.Parameter` or a

Simulink.Signal object in the MATLAB workspace. For details, refer to *Real-Time Workshop Dialog (Model Configuration Parameters Dialogs)* ([📖 RTI and RTI-MP Implementation Reference](#)) and for RTI-MP to *Variable Description File Options Page (CPU Options Dialog)* ([📖 RTI and RTI-MP Implementation Reference](#)).

Excluding variables from the variable description file

RTI and RTI-MP now provide the Apply subsystem omission tags option. It can be used as an alternative to the TRC Exclusion block from the Extras library to exclude a subsystem and its contents from the generated variable description file. It can be used together with the TRC Exclusion block in the same model. For details, refer to *Real-Time Workshop Dialog (Model Configuration Parameters Dialogs)* ([📖 RTI and RTI-MP Implementation Reference](#)) and for RTI-MP to *Variable Description File Options Page (CPU Options Dialog)* ([📖 RTI and RTI-MP Implementation Reference](#)).

Enabling real-time testing

The option **Enable real-time testing** is no longer a CPU-specific option. It can now be selected or deselected globally for the RTI-MP model. The option was therefore moved from the CPU Options dialog of the Build Options page to the Main page of the Multiprocessor Setup dialog. Selecting the option enables real-time testing globally for all CPUs in the RTI-MP model.

Migrating RTI-MP models Earlier RTI-MP models which have the option **Enable real-time testing** selected for at least one CPU are migrated automatically. The real-time testing option is enabled globally for these RTI-MP models.

Switching target-specific settings of RTI-MP models

RTI-MP provides a new function named `rtimp_targetswitch`. It can be used to adapt target-specific settings of an RTI-MP model according to the currently active RTI platform.


Examples of target-specific settings are the RTI-MP model's system target file and the selected platform type in the Multiprocessor Setup dialog. You can use this function, for example, before starting the build process with RTI-MP to ensure that target-specific settings are up-to-date regarding the current RTI platform.

The supported RTI platforms for this function are: RTI1005 and RTI1006. For details, refer to *rtimp_targetswitch* ([📖 RTI and RTI-MP Implementation Reference](#)).

Getting version information

A new function named `rti_md1versionget` is available. It reads version information from a model file and displays the results in the MATLAB Command Window. The version information valid at the moment the model was last saved comprises:

- Simulink version
- RTI version
- RTI platform
- dSPACE Release

This function does not open the model, it is only for checking version information quickly. For details, refer to `rti_md1versionget` ( *RTI and RTI-MP Implementation Reference*).

DS4004 support

RTI and RTLib support the new DS4004 HIL Digital I/O Board. The I/O board comprises 96 digital I/O channels with signal conditioning arranged in three identical ports. Each channel is software-configurable as digital I/O or timing I/O. The board can be connected to up to six independent external supply voltages (two per port). Timing I/O is supported only by RTLib functions. For details on the I/O board, refer to *DS4004 Features*.

RTI AUTOSAR Package

New Features of the RTI AUTOSAR Package 1.0

RTI AUTOSAR Package 1.0 provides the following features:

Integrating AUTOSAR software components with the RTI AUTOSAR Package

The RTI AUTOSAR Package is a dSPACE software product for integrating atomic software components (atomic SWCs) and compositions in a Simulink environment. You can combine integrated SWCs with other Simulink/Stateflow models. The main purpose of the package is to work with SWCs and use RTI features and the dSPACE RCP and HIL hardware.

The package consists of the *RTI AUTOSAR Interface Generator* and the *RTI AUTOSAR Interface Blockset*. For user documentation of the RTI AUTOSAR Package, refer to the *RTI AUTOSAR Package Document*.

The RTI AUTOSAR Package supports:

- AUTOSAR Release 3.0 with Version 3.0.2
- AUTOSAR Release 3.1 with Version 3.1.0

Generating software component containers with the RTI AUTOSAR Interface Generator

The *RTI AUTOSAR Interface Generator* is a software product for generating an SWC container with integration code for embedding an atomic SWC or composition in an S-function. The generator provides a user interface that lets you import AUTOSAR files and select implementations of atomic software components or compositions for generating an SWC container.

Integrating AUTOSAR software components with the RTI AUTOSAR Interface Blockset

The *RTI AUTOSAR Interface Blockset* lets you integrate AUTOSAR software components in a Simulink model. The blockset consists of an SWC block that lets you select an SWC and start building its S-function. The SWC block lets you open a report file that shows information on port usage for connecting the SWC block to its environment in the surrounding Simulink environment.

Simulating Simulink models with software components

You can simulate models with atomic software components and compositions in a Simulink environment on the host computer or build them as real-time applications for dSPACE's real-time hardware.

You can measure simulation RTE variables such as the following with dSPACE experiment software during real-time simulation:

- Data elements for sender-receiver communication
- Operation arguments for client-server communication
- Interrunnable variables for interrunnable communication

Supporting the AUTOSAR standard

The RTI AUTOSAR Package lets you integrate atomic software components and compositions in Simulink models that use the following AUTOSAR concepts:

- Unqueued sender-receiver communication
- Synchronous client-server communication with getter or setter operations
- Interrunnable communication


RTI Bypass Blockset


New Features of the RTI Bypass Blockset 2.6

RTI Bypass Blockset

External variable definition The RTI Bypass Blockset provides external variable configuration. If enabled, additional Simulink block ports are added to the Read, Write, Upload and Download blocks which allow you to define variable properties, like the variable addresses and data types, during run time by external block inputs.

Starting external tools from within the RTI Bypass Blockset

The RTI Bypass Blockset lets you configure up to five external tools to be started by a single click each via the RTIBYPASS_SETUP block user interface. For details, refer to *External Tools Page (RTIBYPASS_SETUP_BLx)* ( *RTI Bypass Blockset Reference*).

New configuration option for XCP on FlexRay interfaces The RTI Bypass Blockset now lets you specify the FlexRay buffers to be used for command processing. This can reduce the initialization time and increase the upload/download bandwidth via the FlexRay bus. For details, refer to *Buffers Configuration Page (RTIBYPASS_SETUP_BLx for XCP on FlexRay)* ( *RTI Bypass Blockset Reference*).

Support of XCP protocol version 1.1 and IF_DATA XCPplus

The RTI Bypass Blockset now supports the XCP 1.1 standard and the IF_DATA entry IF_DATA XCPplus. IF_DATA XCPplus allows you to describe multiple instances of one and the same transport layer.

For further information, refer to the *Interface Description Data Reference*.

Renamed bypass interface, and support of further AML files

The formerly known bypass interface Nexus/MPC55xx has been renamed JTAG/Nexus. The IF_DATA entries belonging to this bypass interface have also been renamed:

- IF_DATA JTAG_NEXUS_BYPASS (for service-based bypassing via JTAG/Nexus)
- IF_DATA cPATCH_JTAG_NEXUS_BYPASS (for code patch-based bypassing via JTAG/Nexus)

In connection with the renaming of the IF_DATA entries, the RTI Bypass Blockset now supports the relevant AML files `JTAG_NEXUS_Bypass.aml` and `cPATCH_JTAG_NEXUS_Bypass.aml`.

For compatibility reasons, the old IF_DATA entries `IF_DATA NEXUS_MPC55XX_BYPASS` and `IF_DATA cPATCH_NEXUS_MPC55XX_BYPASS` (and their relevant AML files) are still supported by the RTI Bypass Blockset. However, as of RTI Bypass Blockset 2.6 it is recommended to use the new IF_DATA entries.

RTI Bypass Blockset MATLAB API

Using the RTI Bypass Blockset without A2L file The ECU interface parameters, which are usually provided by A2L files, can be set via the RTI Bypass Blockset MATLAB API. This allows you to use the RTI Bypass Blockset with and without A2L files.

Easier to change ECU interface type The RTI Bypass Blockset MATLAB API comes with an improved automation API that allows you to easily change the ECU interface type (for example, from XCP on CAN to JTAG/OCDS) via automation scripts.

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

Working with models from earlier RTI Bypass Blockset versions 2.x

dSPACE Release 6.4 comes with RTI Bypass Blockset 2.6, which is compatible with earlier blockset versions 2.x. However, data management was changed compared to earlier 2.x versions of the RTI Bypass Blockset.

If you have a Simulink model built with an earlier blockset version 2.x and open it with RTI Bypass Blockset 2.6, the old data dictionary file (file name extension .dd) is deleted automatically and the new data dictionary file (.vdb) is recreated using the information stored in the Setup block as soon as you open and close the Setup block dialog by clicking OK, or open the Read/Write/Upload/Download block dialog and click the Fill Variable Selector button on the Variables page.

If you have a model that was saved with RTI Bypass Blockset 2.6 and want to use it with an earlier 2.x version of the RTI Bypass Blockset, the model's data dictionary file required for the earlier blockset version 2.x (file name extension .dd) is recreated as soon as you update the A2L files in the Setup block or open the Read/Write/Upload/Download block and click the **Fill Variable Selector** button on the Variables page. The data dictionary file created under RTI Bypass Blockset 2.6 (*.vdb) remains on disk.

To enable the RTI Bypass Blockset to recreate the data dictionary, the ASAM-MCD 2MC (A2L) files specified in the Setup block must be accessible at the specified location and must be unchanged.


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 a CalDesk version earlier than 3.0.

RTI CAN MultiMessage Blockset

New Features of the RTI CAN MultiMessage Blockset 2.4.1

Defining baud rate of a CAN controller via model input


The RTI CAN MultiMessage Blockset now lets you set the baud rate of a CAN controller via model input. You can add a Baudrate inport to the RTICANMM ControllerSetup block for this purpose, refer to *Options Page (RTICANMM ControllerSetup)* ( *RTI CAN MultiMessage Reference*).

RTI LIN MultiMessage Blockset

New Features of the RTI LIN MultiMessage Blockset 1.7

Support of slave node configuration services

The RTI LIN MultiMessage Blockset now supports slave node configuration services of LIN 2.0 and higher.

You can select slave nodes that are configured by RTILINMM to react to master request frames compliant with node configuration services such as assign NAD, conditional change NAD, assign frame ID, and assign frame ID range. The slave response frames of the selected slave nodes are transmitted on the bus by RTILINMM. The RTILINMM MainSetup block provides new pages that let you select slave nodes and specify initial node configuration parameters for them, refer to *Network Node Configuration Page (RTILINMM MainSetup)* ( *RTI LIN MultiMessage Reference*).

SystemDesk

New Features of SystemDesk 2.0.4

SystemDesk 2.0.4 provides the following new features:

Supported AUTOSAR Releases

SystemDesk 2.0.4 supports the following AUTOSAR Releases when importing or exporting data in the AUTOSAR format:

- AUTOSAR Release 2.1 with Versions 2.1.2 and 2.1.4
- AUTOSAR Release 3.0 with Versions 3.0.0, 3.0.2 and 3.0.4

Handling basic software tasks during RTE generation

The RTE generator in SystemDesk 2.0 does not distinguish between runnables of application software components and main processing functions of basic software components. For this reason, RTE code generated with SystemDesk 2.0 contains OS task bodies for runnables *and* for the main processing functions of basic software components. OS task bodies contain code to trigger runnables and main functions.

However, OS task bodies for the main processing functions of basic software components should not be created by an RTE code generator, but by a Basic Software Scheduler (SchM).

The RTE code generator of SystemDesk 2.0.4 now treats main processing functions like this:

- During RTE code generation with SystemDesk 2.0.4, runnables are identified as main processing functions of basic software components *if the runnables are not triggered by RTE events*, regardless of whether RTE events are assigned to wait points of the runnables.

- During RTE code generation with SystemDesk 2.0.4, no OS task bodies are created for runnables identified as main processing functions of basic software components.




If you map runnables and main processing functions to the same OS task, you get an error message during RTE code generation.

DBC import harmonized with EB tresos®

As of this SystemDesk version, the DBC import is harmonized with the DBC import of EB tresos®. SystemDesk maps the signals in a DBC file to SystemDesk elements during DBC file import in the same way as EB tresos®:

For each signal in the DBC file, SystemDesk creates a system signal. For signals in the DBC file with the same name, SystemDesk names the system signals like this: the system signal created for the signal belonging to the CAN message with the lowest message ID has the same name as the signal in the DBC file. The system signal created for the signal belonging to the CAN message with the next message ID has the same name as the signal in the DBC file plus the "_1" suffix, and so on.

For each system signal, SystemDesk creates a signal instance with the system signal name, and adds the signal instance to the corresponding IPDU.

For details, refer to *Mapping of Imported Database Files* ( *SystemDesk Guide*).

Support of the RTE API functions Rte_Ports, Rte_NPorts, and Rte_Port

RTE API functions Rte_Ports, Rte_NPorts, and Rte_Port With this version of SystemDesk the RTE generator additionally supports the following indirect port API functions:

RTE API Function	Return Type	Description
Rte_Ports_<i>_<R/P> ¹⁾	Rte_PortHandle_<i>_<R/P> ¹⁾	Provides the handle of an array of ports with identical interfaces for index-based access to the ports.
Rte_NPorts_<i>_<R/P> ¹⁾	uint8	Provides the number of ports with identical interfaces that you can access via the Rte_Ports function for iterating over ports.

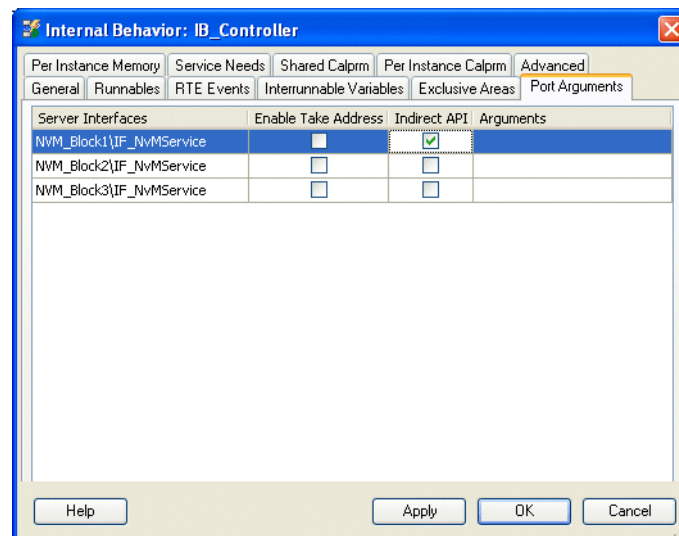
RTE API Function	Return Type	Description
Rte_Port_<p> ²⁾	Rte_PortHandle_<i>_<R/P> ¹⁾	Provides the handle of a single port for grouping sets of ports with identical interfaces.

¹⁾ i: Name of the interface; R/P: Denotes required or provided

²⁾ p: Name of the port

For reference information of RTE API functions, refer to the *AUTOSAR_RTE_SWS* document at <http://www.autosar.org>

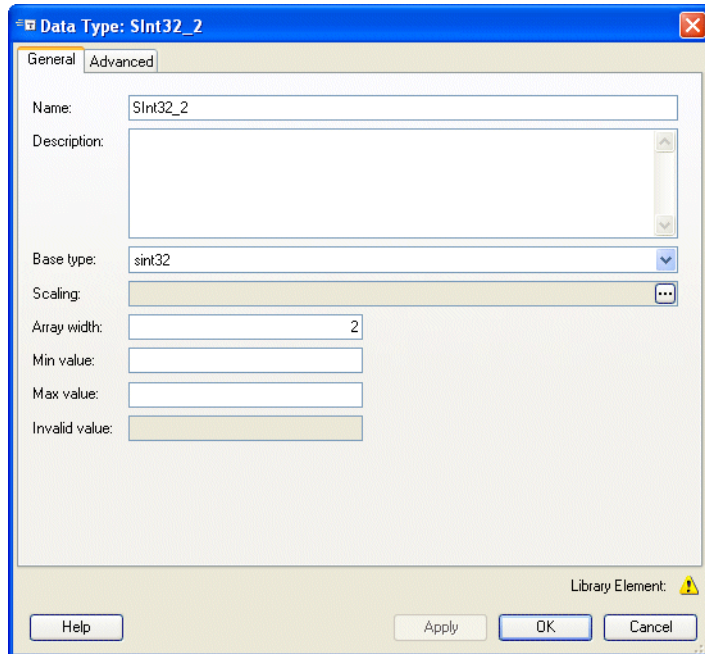
Generating indirect port API functions SystemDesk does not generate indirect port API functions by default. If you require their generation you have to select the Indirect API checkbox on the internal behavior's Port Arguments property page as shown in the illustration below.



With the settings above, indirect port API functions are generated for all the ports with the IF_NvMService interface, i.e., the ports NVM_Block1, NVM_Block2, NVM_Block3, when you generate the RTE. Select the **Enable Take Address** checkbox to prevent optimization of the indirect API functions by inlining for using function addresses in the software component code.

Support of defining scalar array data types

Defining scalar array data types With this version of SystemDesk the RTE generation module supports the definition of scalar array data types. You can define a scalar array as a data type by specifying the array width in the Data Type property dialog as shown in the illustration below.



With the settings above, the RTE code for data type definition and calling RTE API functions in a software component reads as follows:

```
// ### Rte_Type.h ###
typedef int32 SInt32_2[2];

// ### Swc.c ###
void my_runnable(void)
{
    SInt32_2 de1;
    de1[0] = 0;
    de1[1] = 1;
    pDe1 = Rte_Write_p1_de1(de1);
}
```

Defining the width of a data element / operation argument / calprm value

You can also specify the width of an interface element such as a data element, an operation argument, or a calprm value. SystemDesk then generates an array of the specified data type. If you specify the width of an interface element, the RTE code for data type definition and calling RTE API functions in a software component reads as follows.

```
// ### Rte_Type.h ###
typedef int32 SInt32_2;

// ### Swc.c ###
void my_runnable(void)
{
    SInt32_2 del[2];
    del[0] = 0;
    del[1] = 1;
    pDel = Rte_Write_pl_del(del);
}
```



RTE code generated with prior SystemDesk versions always reads as if you had specified the width of an interface element even if you had specified the width of a data type. Make sure that data type definitions of software components and the RTE match to avoid erroneous calls of RTE API functions in the software component code.

Compatibility Information

Where to go from here

Information in this section

<i>Supported MATLAB Releases</i>	101
<i>Supported Operating Systems</i>	102

Supported MATLAB Releases

Supported MATLAB releases

The table below shows which dSPACE software item supports which MATLAB release:

MATLAB Release...	Is Supported by dSPACE Release 6.4 Software Item...					
	RCP and HIL Software	TargetLink			Model Compare 2.0.1	SystemDesk 2.0.4
		2.3.1	3.0	3.0.1		
R2009a	Yes	No	No	Yes	No	Yes
R2008b	Yes	Yes	No	Yes	Yes	Yes
R2008a+	Yes	No	No	Yes	Yes	Yes
R2007b+	Yes	Yes	Yes	Yes	Yes	Yes
R2007a+	Yes	Yes	Yes	Yes	Yes	Yes
R2006b	Yes	Yes	Yes	Yes	Yes	Yes
R2006a+	No	Yes	Yes	Yes	Yes	Yes

For detailed information on compatibility, refer to <http://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 6.4 support which operating system:

Operating System...	Is Supported By...						
	RCP & HIL Software	TargetLink			CalDesk 3.0	Model Compare 2.0.1	SystemDesk 2.0.4
		2.3.1	3.0	3.0.1			
Windows 2000 Professional with Service Pack 4	No	Yes	Yes	Yes	No	No	No
Windows XP Professional (32-bit version) with Service Pack 3	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Windows Vista ¹⁾ (32-bit version) with the latest Service Pack	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Windows Vista ¹⁾ (64-bit version) ²⁾ with the latest Service Pack	Yes ³⁾	No	No	Yes	No	Yes	Yes

¹⁾ Only the editions Windows Vista Business, Ultimate, and Enterprise are supported. The editions Windows Vista Home and Starter are not supported.

²⁾ The 64-bit operating systems are supported by 32-bit software running in WoW (Windows-On-Windows64). 64-bit MATLAB versions are not supported.

³⁾ Not supported by the RTI AUTOSAR Package

Windows XP Professional x64 Edition is not supported by dSPACE Release 6.4.


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

Limitations when working with Windows Vista

The following limitations apply when you use Windows Vista in combination with dSPACE software:

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

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."`

Additional limitations when using Windows Vista (64-bit)

Some additional limitations apply when using Windows Vista (64-bit) with dSPACE software. Refer to `%DSPACE_ROOT%\Doc\Print\Vista64_Limitations.pdf`.

A

- ASM Base InCylinder Blockset
 - migrating 26
 - new features 26
- ASM blockset
 - new features 25
- ASM Diesel InCylinder Blockset
 - migrating 27
- ASM Drivetrain Basic Blockset
 - migrating 28
 - new features 28
- ASM Electric Components Blockset
 - migrating 29
 - new features 29
- ASM Engine Diesel Blockset
 - migrating 30
 - new features 30
- ASM Engine Gasoline Basic Blockset
 - migrating 31
 - new features 31
- ASM Engine Gasoline Blockset
 - migrating 32
 - new features 32
- ASM Environment Blockset
 - migrating 33
- ASM Gasoline InCylinder Blockset
 - migrating 35
- ASM Optimizer
 - new features 35
- ASM Traffic Blockset
 - migrating 36
- ASM Traffic Operator Blockset
 - new features 37
- ASM Trailer Blockset
 - new features 37
- ASM Turbocharger Blockset
 - migrating 38
 - new features 38
- ASM Utilities Blockset
 - migrating 38
- ASM Vehicle Dynamics Blockset
 - migrating 39
 - new features 39
- ASMParameterization tool
 - new features 35
- AutomationDesk
 - migration 21
 - new features 19

C

- CalDesk
 - migration 59
 - new features 44
- ConfigurationDesk
 - migration to Version 3.0.1 66

- ControlDesk
 - new features 71

D

- dSPACE FlexRay Configuration Package
 - new features 75

G

- general enhancements and changes 10

K

- key features 13

M

- Microtec PowerPC C Compiler 3.5 10
 - migrating
 - ASM Base InCylinder Blockset 26
 - ASM Diesel InCylinder Blockset 27
 - ASM Drivetrain Basic Blockset 28
 - ASM Electric Components Blockset 29
 - ASM Engine Diesel Blockset 30
 - ASM Engine Gasoline Basic Blockset 31
 - ASM Engine Gasoline Blockset 32
 - ASM Environment Blockset 33
 - ASM Gasoline InCylinder Blockset 35
 - ASM Traffic Blockset 36
 - ASM Turbocharger Blockset 38
 - ASM Utilities Blockset 38
 - ASM Vehicle Dynamics Blockset 39
 - to ConfigurationDesk 3.0.1 66
 - migrating a ConfigurationDesk project 68
 - AutomationDesk 21
 - CalDesk 59
 - ModelDesk
 - new features 77
 - MotionDesk
 - new features 79

N

- new features
 - ASM Base InCylinder Blockset 26
 - ASM blocksets 25
 - ASM Drivetrain Basic Blockset 28
 - ASM Electric Components Blockset 29
 - ASM Engine Diesel Blockset 30
 - ASM Engine Gasoline Basic Blockset 31
 - ASM Engine Gasoline Blockset 32
 - ASM Optimizer 35
 - ASM Traffic Operator Blockset 37
 - ASM Trailer Blockset 37

new features

- ASM Turbocharger Blockset 38
- ASM Vehicle Dynamics Blockset 39
- ASMParameterization tool 35
- AutomationDesk 19
- CalDesk 44
- ControlDesk 71
- dSPACE FlexRay Configuration Package 75
- ModelDesk 77
- MotionDesk 79
- Real-Time Testing 21
- RTI 81
- RTI AUTOSAR Package 85
- RTI Bypass Blockset 87
- RTI CAN MultiMessage Blockset 91
- RTI LIN MultiMessage Blockset 93

P

- product overview 11

R

- RapidPro Projects Migrator
 - migrating a Configuration Desk project 68
- Real-Time Testing
 - new features 21
- RTI
 - new features 81
- RTI AUTOSAR Package
 - new features 85
- RTI Bypass Blockset
 - new features 87
- RTI CAN MultiMessage Blockset
 - new features 91
- RTI LIN MultiMessage Blockset
 - new features 93

S

- supported MATLAB releases 101
- supported operating systems 102

V

- version history 11

W

- Windows Vista
 - limitations 102

