dSPACE Release

New Features and Migration

Release 7.0 – November 2010
How to Contact dSPACE

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

How to Contact dSPACE Support
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
  • TargetLink Support:
support.tl@dspace.de
  +49 5251 1638-700
• Use the dSPACE Installation Manager:
  • On your dSPACE DVD at \Tools\InstallationManager
  • Via Start – Programs – dSPACE Installation Manager (after installation of the dSPACE software)
  • At http://www.dspace.com/goto?im

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

Software Updates and Patches
dSPACE strongly recommends that you download and install the most recent patches for your current dSPACE installation. Visit http://www.dspace.com/goto?support for software

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 2000 - 2010 by:
dSPACE GmbH
Rathenaustrialle 26
33102 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 9

Overview of dSPACE Release 7.0 11
   General Enhancements and Changes 12
   Product Version Overview 14
   New Product Key Features 17
   Migrating to dSPACE Release 7.0 21

AutomationDesk 23
   New Features of AutomationDesk 3.2 23
   Migrating to AutomationDesk 3.2 26

Automotive Simulation Models (ASM) 29
   ASM Base InCylinder Blockset 31
      Migrating to ASM Base InCylinder Blockset 1.2 31
   ASM Diesel Exhaust Blockset 32
      Migrating to ASM Diesel Exhaust Blockset 1.1.2 32
   ASM Diesel InCylinder Blockset 33
      Changes in the ASM Diesel InCylinder Demo Model 33
      Migrating to ASM Diesel InCylinder Blockset 1.1 33
   ASM Drivetrain Basic Blockset 34
      New Features of ASM Drivetrain Basic Blockset 1.4 34
      Migrating to ASM Drivetrain Basic Blockset 1.4 34
   ASM Electric Components Blockset 35
      New Features of ASM Electric Components Blockset 2.0 35
      Changes in the ASM Electric Components Demo Model 36
      Migrating to ASM Electric Components Blockset 2.0 36
   ASM Engine Diesel Blockset 37
      Changes in the ASM Engine Diesel Demo Model 37
      Migrating to ASM Engine Diesel Blockset 1.4 37
   ASM Engine Gasoline Basic Blockset 38
      Changes in the ASM Engine Gasoline Basic Demo Model 38
      Migrating to ASM Engine Gasoline Basic Blockset 1.3.5 38
   ASM Engine Gasoline Blockset 39
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Changes in the ASM Engine Gasoline Demo Model</td>
<td>39</td>
</tr>
<tr>
<td>Migrating to ASM Engine Gasoline Blockset 2.2</td>
<td>39</td>
</tr>
<tr>
<td>ASM Environment Blockset</td>
<td>41</td>
</tr>
<tr>
<td>New Features of ASM Environment Blockset 1.5</td>
<td>41</td>
</tr>
<tr>
<td>Migrating to ASM Environment Blockset 1.5</td>
<td>41</td>
</tr>
<tr>
<td>ASM Gasoline InCylinder Blockset</td>
<td>43</td>
</tr>
<tr>
<td>Changes in the ASM Gasoline InCylinder Demo Model</td>
<td>43</td>
</tr>
<tr>
<td>Migrating to ASM Gasoline InCylinder Blockset 1.1</td>
<td>44</td>
</tr>
<tr>
<td>ASM Optimizer</td>
<td>46</td>
</tr>
<tr>
<td>New Features of ASM Optimizer 1.3</td>
<td>46</td>
</tr>
<tr>
<td>Migrating to ASM Optimizer Blockset 1.3</td>
<td>47</td>
</tr>
<tr>
<td>ASM Parameterization Tool</td>
<td>49</td>
</tr>
<tr>
<td>New Features of the ASM Parameterization Tool 1.5.1</td>
<td>49</td>
</tr>
<tr>
<td>ASM Traffic Blockset</td>
<td>50</td>
</tr>
<tr>
<td>Changes in the ASM Traffic Demo Model</td>
<td>50</td>
</tr>
<tr>
<td>Migrating to ASM Traffic Blockset 1.1.4</td>
<td>50</td>
</tr>
<tr>
<td>ASM Trailer Blockset</td>
<td>51</td>
</tr>
<tr>
<td>New Features of ASM Trailer Blockset 1.3</td>
<td>51</td>
</tr>
<tr>
<td>Changes in the ASM Trailer Demo Model</td>
<td>51</td>
</tr>
<tr>
<td>Migrating to ASM Trailer Blockset 1.3</td>
<td>51</td>
</tr>
<tr>
<td>ASM Truck Blockset</td>
<td>52</td>
</tr>
<tr>
<td>New Features of ASM Truck Blockset 1.2</td>
<td>52</td>
</tr>
<tr>
<td>Changes in the ASM Truck Demo Model</td>
<td>52</td>
</tr>
<tr>
<td>Migrating to ASM Truck Blockset 1.2</td>
<td>53</td>
</tr>
<tr>
<td>ASM Turbocharger Blockset</td>
<td>54</td>
</tr>
<tr>
<td>Migrating to ASM Turbocharger Blockset 1.5</td>
<td>54</td>
</tr>
<tr>
<td>ASM Vehicle Dynamics Blockset</td>
<td>55</td>
</tr>
<tr>
<td>New Features of ASM Vehicle Dynamics Blockset 1.6</td>
<td>56</td>
</tr>
<tr>
<td>Changes in the ASM Vehicle Dynamics Demo Model</td>
<td>56</td>
</tr>
<tr>
<td>Migrating to ASM Vehicle Dynamics Blockset 1.3</td>
<td>56</td>
</tr>
<tr>
<td>ControlDesk</td>
<td>59</td>
</tr>
<tr>
<td>New Features of ControlDesk 3.7</td>
<td>59</td>
</tr>
<tr>
<td>ControlDesk Next Generation</td>
<td>61</td>
</tr>
<tr>
<td>New Features of ControlDesk Next Generation (ControlDesk 4.0)</td>
<td>62</td>
</tr>
<tr>
<td>New Project and Experiment Features</td>
<td>63</td>
</tr>
<tr>
<td>New Bus Navigator for CAN, LIN, and FlexRay Bus Communication</td>
<td>63</td>
</tr>
<tr>
<td>New Features of Platform Management and Platforms/Devices</td>
<td>65</td>
</tr>
<tr>
<td>New Visualization Features</td>
<td>68</td>
</tr>
</tbody>
</table>
New Instrument Features .......................................................... 70
New Measurement and Recording Features ................................. 73
New Data Set Management Features ........................................... 76
New Features of the ECU Diagnostics Module .............................. 77
New Signal Editor ...................................................................... 81
New Failure Simulation Module .................................................... 82
New Automation Features ............................................................ 85
Further Enhancements with ControlDesk Next Generation .......... 86
Migrating to ControlDesk Next Generation (ControlDesk 4.0) ...... 88
Migrating to ControlDesk Next Generation ................................ 88
New Features of the Variable Editor .............................................. 89
General Enhancements ............................................................... 89
dSPACE ECU Flash Programming Tool ....................................... 91
New Features of the dSPACE ECU Flash Programming Tool 2.2 .. 91
dSPACE FlexRay Configuration Package ...................................... 93
New Features of dSPACE FlexRay Configuration Package 2.4 . . 93
Migration to dSPACE FlexRay Configuration Package 2.4 ........ 95
ModelDesk .............................................................................. 97
New Features of ModelDesk 2.3 .................................................. 97
RTI/RTI-MP and RTLib ............................................................... 99
New Features of RTI/RTI-MP and RTLib ...................................... 99
Migration Aspects of RTI/RTI-MP and RTLib .............................. 102
RTI AUTOSAR Package ............................................................ 103
New Features of the RTI AUTOSAR Package 1.2 ...................... 103
RTI Bypass Blockset .................................................................. 107
New Features of the RTI Bypass Blockset 2.7.2 ......................... 107
RTI CAN MultiMessage Blockset ................................................ 109
New Features of the RTI CAN MultiMessage Blockset 2.5.2 .. 109
Migrating to RTI CAN MultiMessage Blockset 2.5.2 ............... 110
RTI LIN MultiMessage Blockset .................................................. 111
New Features of the RTI LIN MultiMessage Blockset 1.8 .......... 111
Migrating to RTI LIN MultiMessage Blockset 1.8 .................... 112
### RTI Ethernet (UDP) Blockset  
New Features of the RTI Ethernet (UDP) Blockset 1.0  
113

### RTI FPGA Programming Blockset  
115
New Features of the RTI FPGA Programming Blockset 2.0  
Migrating to RTI FPGA Programming Blockset 2.0  
115

### SystemDesk  
119
New Features of SystemDesk 3.0  
120
New General Features  
120
Changed Modeling of ECU Software  
122
Modeling Systems  
125
Configuring ECUs  
127
Simulating Systems  
129
Importing and Exporting AUTOSAR Files  
132
Exchanging Data with TargetLink  
134
dSPACE Target for Offline Simulation 1.2  
137
Features Discontinued in SystemDesk 3.0  
138
Migrating to SystemDesk 3.0  
139
Migrating to SystemDesk 3.0  
139

### TargetLink  
141
New Features of TargetLink 3.2 and dSPACE Data Dictionary 3.2  
142
New Production Code Generation Features  
142
Online Parameter Modification  
143
Debugging in SIL Simulation Mode  
145
New TargetLink Sqrt Block  
145
Enhancements to the Target Simulation Module  
146
New TargetLink API commands  
148
Code Generator Options  
149
General Enhancements and Changes  
151
New AUTOSAR-Related Features  
154
Features of the TargetLink AUTOSAR Module  
154
Exchanging Software Component Containers with SystemDesk  
157
New Features of the dSPACE Data Dictionary  
160
New Key Features  
160
New and Modified DD MATLAB API Commands  
165
Migrating to TargetLink 3.2 and dSPACE Data Dictionary 3.2  
167
Migrating to dSPACE Data Dictionary 3.2  
169
How to Upgrade a Data Dictionary Without Include Files  
173
How to Upgrade a Data Dictionary With Include Files  
174
Contents

Obsolete and Changed Code Generator Options .................. 176
Changes in TargetLink API Functions............................... 178
Various Migration Aspects............................................. 180
AUTOSAR-Related Migration Aspects................................. 184

Compatibility Information 189
Supported MATLAB Releases........................................... 190
Operating System......................................................... 191
Limitations for Windows Vista/Windows 7............................ 192
Limitations for 64-bit Windows Operating Systems................. 194

Index 197
About This Document

This document informs you about the new features of all the dSPACE software products in Release 7.0. 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 7.0

<table>
<thead>
<tr>
<th>Objective</th>
<th>Gives you an overview of the new key features in Release 7.0, and also information about unchanged products and general instructions on migrating.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Where to go from here</th>
<th>Information in this section</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General Enhancements and Changes</strong></td>
<td>12</td>
</tr>
<tr>
<td><strong>Product Version Overview</strong></td>
<td>14</td>
</tr>
<tr>
<td><strong>New Product Key Features</strong></td>
<td>17</td>
</tr>
<tr>
<td><strong>Migrating to dSPACE Release 7.0</strong></td>
<td>21</td>
</tr>
</tbody>
</table>
General Enhancements and Changes

Objective

The following new features and changes concern several dSPACE products.

New installation concepts

With dSPACE Release 7.0, you can install RCP&HIL software, TargetLink, Model Compare, ControlDesk Next Generation and SystemDesk in different folders. This is because the software architecture has been redesigned to reduce dependencies between products. Software components which are used by several dSPACE products are now Windows-compliant installed in the \textit{Common Files} folder, for example Real-Time Testing, DCI Configuration Tool, and ECU Flash Programming Tool. Here, you find also the dSPACE HelpDesk now.

To prevent removal of common software components, they are not listed in the Control Panel's \textit{Add or Remove Programs} dialog.

In the Windows \textit{Start} menu, you will now find separate entries for each installed product family (RCP&HIL, TargetLink, Model Compare, ControlDesk Next Generation, SystemDesk). There is no longer a folder called \textit{dSPACE Tools}.

The new software architecture allows you to install the same TargetLink version multiple times. No switching is required between an activated and a deactivated installation. The settings required for the MATLAB connection can be made quickly in the new Installation Manager (for details, see below).

Different installations of Model Compare, ControlDesk Next Generation and SystemDesk also do not need to be switched via the Installation Manager. Only RCP&HIL software still requires the Installation Manager to activate another installation.

New version of dSPACE Installation Manager

The dSPACE Installation Manager 3.0 has enhanced functionality. In addition to its basic functionality (managing dSPACE installations), the tool now provides the following features.

\textbf{Linking MATLAB to dSPACE software} The dSPACE Installation Manager provides an easy way to integrate your MATLAB installation into a specific dSPACE installation. All the available MATLAB installations on your host PC and their integration statuses are listed.
Managing licenses  The available licenses on your host PC are shown with details such as their license types. In addition, you can access the dSPACE license management tools (dSPACE License Manager and dSPACE Floating Network License Manager).

Getting diagnostic information for contacting dSPACE Support  When contacting dSPACE Support, you need to provide the Support team with information about your dSPACE installation and the problems you may have. Some information required by dSPACE Support is extracted automatically by the dSPACE Installation Manager’s diagnostic feature. Other information must be collected manually.

After collecting all the information, the diagnostic feature creates an e-mail with an automatically generated ZIP file to send to the responsible Support team in your country.

Release update  The printed user documentation is not delivered with Release 7.0 if you receive the release as an update for your existing dSPACE release. Use the current online help, for example, dSPACE HelpDesk, to obtain information about new features, enhancements, and the current safety precautions regarding your products.
# Product Version Overview

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

<table>
<thead>
<tr>
<th>Product</th>
<th>dSPACE Release</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>6.4</td>
</tr>
<tr>
<td>AutomationDesk</td>
<td>2.3</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Automotive Simulation Models</td>
<td>2.1</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>CalDesk</td>
<td>3.0</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>ConfigurationDesk</td>
<td>3.0.1</td>
</tr>
<tr>
<td>ControlDesk</td>
<td>3.4</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>ControlDesk Next Generation</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>DCI Configuration Tool</td>
<td>2.1.4</td>
</tr>
<tr>
<td>Product</td>
<td>dSPACE Release</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td></td>
<td>6.4</td>
</tr>
<tr>
<td>dSPACE Data Dictionary</td>
<td>1.5.1</td>
</tr>
<tr>
<td>dSPACE CAN API</td>
<td>2.5</td>
</tr>
<tr>
<td>dSPACE ECU Flash Programming Tool</td>
<td>2.1</td>
</tr>
<tr>
<td>dSPACE FlexRay Configuration Package</td>
<td>2.1</td>
</tr>
<tr>
<td>dSPACE Target for Offline Simulation</td>
<td>1.0</td>
</tr>
<tr>
<td>Model Compare</td>
<td>2.0.1</td>
</tr>
<tr>
<td>ModelDesk</td>
<td>2.2</td>
</tr>
<tr>
<td>MotionDesk</td>
<td>2.1.3</td>
</tr>
<tr>
<td>MotionDesk Blockset</td>
<td>1.3.12</td>
</tr>
<tr>
<td>Real-Time Testing</td>
<td>1.6</td>
</tr>
<tr>
<td>RTI</td>
<td>6.3</td>
</tr>
<tr>
<td>RTI-MP</td>
<td>5.7</td>
</tr>
<tr>
<td>RTI AUTOSAR Package</td>
<td>1.0</td>
</tr>
<tr>
<td>RTI Bypass Blockset</td>
<td>2.6</td>
</tr>
<tr>
<td>RTI CAN Blockset</td>
<td>2.6.8</td>
</tr>
</tbody>
</table>

See **TargetLink** on page 141.

See **dSPACE ECU Flash Programming Tool** on page 91.

See **dSPACE FlexRay Configuration Package** on page 93.

See **dSPACE Target for Offline Simulation** on page 137.

See **ModelDesk** on page 97.

See **RTI/RTI-MP and RTLib** on page 99.

See **RTI AUTOSAR Package** on page 103.

See **RTI Bypass Blockset** on page 107.
# Overview of dSPACE Release 7.0

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.

<table>
<thead>
<tr>
<th>Product</th>
<th>dSPACE Release</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>6.4</td>
</tr>
<tr>
<td>RTI CAN MultiMessage Blockset</td>
<td>2.4.1</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>RTI LIN MultiMessage Blockset</td>
<td>1.7</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>RTI RapidPro Control Unit Blockset</td>
<td>1.9.1</td>
</tr>
<tr>
<td>RTI FPGA Programming Blockset</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>RTI Ethernet (UDP) Blockset</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>SystemDesk</td>
<td>2.0.4</td>
</tr>
<tr>
<td>TargetLink</td>
<td>2.3</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Variable Editor</td>
<td>1.3</td>
</tr>
</tbody>
</table>

1) Up to dSPACE Release 6.6, the dSPACE ECU Flash Programming Tool was part of the CalDesk product setup. As of dSPACE Release 7.0, there is a separate setup for the dSPACE ECU Flash Programming Tool.
## 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
The new key features of AutomationDesk are:
- New automation libraries:
  - ControlDesk NG Access
  - CANstress (available on demand)
  - CANscope (available on demand)
- Enhancements to the following libraries:
  - Remote Diagnostics (COM)
  - HIL API (support of signals and stimulus)
  - Test Framework
  - Report
  - Evaluation
- Enhancements to the COM API
- Enhancements to the project handling

For details on the new features, refer to New Features of AutomationDesk 3.2 on page 23.

### Automotive Simulation Models (ASM)
The ASM blocksets have been extended:
- The ASM Vehicle Dynamics Blockset have a torque-based steering model for driver assistance systems.
- The ASM Environment Blockset now has a road converter tool to create ASM roads from road measurement data or from GPS data.
- The ASM Electric Components Blockset supports multicell batteries.

For details on the extensions, refer to Automotive Simulation Models (ASM) on page 29.

### CalDesk
ControlDesk Next Generation is the successor to CalDesk.
For details on the new features of ControlDesk Next Generation compared with CalDesk 3.0.1, refer to New Features of ControlDesk Next Generation (ControlDesk 4.0) on page 62.

### ControlDesk
The new key features of ControlDesk are:
- Support of MicroAutoBox II improved
Support of DS802 PHS Link Board improved
ControlDesk’s Bus Navigator supports the monitoring and logging of LIN bus communication.

For details on the new features, refer to New Features of ControlDesk 3.7 on page 59.

ControlDesk Next Generation
ControlDesk Next Generation is the successor to ControlDesk and CalDesk.
- For information on the different working concepts in ControlDesk Next Generation compared with ControlDesk 3.x, refer to the ControlDesk Next Generation Migration Guide.
- For details on the new features of ControlDesk Next Generation compared with CalDesk 3.0.1, refer to New Features of ControlDesk Next Generation (ControlDesk 4.0) on page 62.

dSPACE ECU Flash Programming Tool
The new key features of the dSPACE ECU Flash Programming Tool are:
- ECU flash programming via XCP on Ethernet
- dSPACE Flash Kernel Configuration Tool for modifying configuration settings of a flash kernel

For details on the new features, refer to New Features of the dSPACE ECU Flash Programming Tool 2.2 on page 91.

dSPACE FlexRay Configuration Package
The new key features of the dSPACE FlexRay Configuration Tool are:
- Several configurations can be created for implementing several FlexRay buses in one real-time model
- The FIBEX file of the configuration can be updated
- Graphical user interface is redesigned

The new key features of the dSPACE FlexRay Configuration Blockset are:
- Several FlexRay buses can be implemented in one real-time model, for example, to model a gateway.
- New variants of MicroAutoBox II are supported
- Hardware access function for MicroAutoBox are improved

For details on the new features, refer to New Features of dSPACE FlexRay Configuration Package 2.4 on page 93.
ModelDesk

The new key feature of ModelDesk is:
- ModelDesk can plot signals of the Automotive Simulation Models during simulation. The signals can be recorded and saved to MAT files for comparing different simulation results. The management of simulation results is done via ModelDesk’s Project Manager.

For details on the new features, refer to New Features of ModelDesk 2.3 on page 97.

RTI, RTI-MP and RTLib

The new key features of RTI, RTI-MP and RTLib are:
- Microtec Compiler 3.7
- Support of the new I/O features of MicroAutoBox II (new variants with DS1511 and DS1512 I/O boards, new USB Flight Recorder, performance improvement for bus support)
- Enhancements to the Gigalink Blockset and RTLib functions
- TRC file enhancements
- Support of MATLAB R2010b

For details on the new features, refer to New Features of RTI/RTI-MP and RTLib on page 99.

RTI CAN MultiMessage Blockset

The new key features of the RTI CAN MultiMessage Blockset are:
- Support of MicroAutoBox II
- Support of Bus Navigator in ControlDesk Next Generation

For details on the new features, refer to New Features of the RTI CAN MultiMessage Blockset 2.5.2 on page 109.

RTI LIN MultiMessage Blockset

The new key features of the RTI LIN MultiMessage Blockset are:
- Support of MicroAutoBox II
- LIN bus monitoring with the Bus Navigator
- Support of Bus Navigator in ControlDesk Next Generation

For details on the new features, refer to New Features of the RTI LIN MultiMessage Blockset 1.8 on page 111.

RTI Ethernet (UDP) Blockset

A new RTI blockset for modeling communication via an Ethernet interface using the UDP/IP protocol.

For details on the new features, refer to New Features of the RTI Ethernet (UDP) Blockset 1.0 on page 113.
## Overview of dSPACE Release 7.0

### RTI FPGA Programming Blockset

The new key feature of the RTI FPGA Programming Blockset is:
- Extended Xilinx® software support

For details on the new features, refer to *New Features of the RTI FPGA Programming Blockset 2.0* on page 115.

### TargetLink

The new key features of TargetLink are:
- Support of *compiler abstraction* macro definitions for source code elements such as functions, variables, and pointers as defined by AUTOSAR to make source code platform-independent.
- *Online Parameter Modification* to change the parameter values of the simulation application.
- *Debugging in SIL simulation mode* to check whether code branches are executed or why the generated production code does not behave as expected.

For details on the new features and migration aspects, refer to *TargetLink* on page 141.
## Migrating to dSPACE Release 7.0

### Objective
After you install Release 7.0, some additional steps may be necessary.

### Migrating from dSPACE Release 6.6
There are no general migration steps to be done. Product-specific migration steps are usually done automatically by the products. For exceptions, refer to the product-specific migration descriptions.

### Migrating from dSPACE Release 6.5 or earlier
To migrate from dSPACE Release 6.5 or earlier to Release 7.0, you also have to perform the migration steps of the intervening dSPACE Releases. All of the required migration steps can be done with Release 7.0 installed.

### Example
For example, if you want to migrate from dSPACE Release 6.3 to Release 7.0, you have to perform the migration steps described in:
1. New Features and Migration of dSPACE Release 6.4
2. New Features and Migration of dSPACE Release 6.5
3. New Features and Migration of dSPACE Release 6.6
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:
- 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.
AutomationDesk

Where to go from here

Information in this section

<table>
<thead>
<tr>
<th>New Features of AutomationDesk 3.2</th>
<th>23</th>
</tr>
</thead>
<tbody>
<tr>
<td>Migrating to AutomationDesk 3.2</td>
<td>26</td>
</tr>
</tbody>
</table>

New Features of AutomationDesk 3.2

New automation libraries

**ControlDesk NG Access library** AutomationDesk provides the ControlDesk NG Access library, which supports some basic features of the automation API of ControlDesk Next Generation (ControlDesk 4.0). This lets you run automation tasks in ControlDesk Next Generation with the help of an AutomationDesk project.

For further information, refer to ControlDesk NG Access ([AutomationDesk Library Reference](#)).

**CANscope library** AutomationDesk provides the CANscope library to remote-control the CANscope Evaluation Software (version 3.0). CANscope is a hardware device for CAN buses from Vector Informatik GmbH. It records and evaluates signal levels on CAN buses.

The CANscope library is not included in the standard distribution. It is available on demand.

For further information, refer to CANscope ([AutomationDesk Library Reference](#)).
**CANstress library**  AutomationDesk provides the CANstress library to remote-control the CANstressD/CANstressDR Configuration Software (version 2.1). CANstressD and CANstressDR are devices from Vector Informatik GmbH which can initiate digital disturbances on the CAN bus.

The CANstress library is not included in the standard distribution. It is available on demand.

For further information, refer to CANstress ([AutomationDesk Library Reference](#)).

**Enhancements to the libraries**

<table>
<thead>
<tr>
<th>Main Library</th>
<th>DataContainer elements can now be nested.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remote Diagnostics (COM)</td>
<td>The Remote Diagnostics (COM) library provides the new SyncPDUService block to execute a service synchronously using the PDU instead of symbolic data. The execution result includes the interpreted data.</td>
</tr>
<tr>
<td></td>
<td>For further information, refer to Remote Diagnostics (COM) (<a href="#">AutomationDesk Library Reference</a>).</td>
</tr>
<tr>
<td>HIL API</td>
<td>The HIL API library provides two new data objects (SignalGenerator and SignalDescriptionSet) and one new block (CreateSignalGenerator) for supporting signal handling and stimulus. You can apply the methods specified in the ASAM AE HIL standard, for example, to load a stimulus file (STI file) created with ControlDesk Next Generation.</td>
</tr>
<tr>
<td></td>
<td>For further information, refer to HIL API (<a href="#">AutomationDesk Library Reference</a>).</td>
</tr>
<tr>
<td>Test Framework</td>
<td>The Test Framework library provides the new SetTestStepCounter block to set the test step counter to a specific start value.</td>
</tr>
<tr>
<td></td>
<td>The bodies of TestSequence, TestStepGroup, Test, and TestStep provide the new NameInReport data object to specify a custom element name in the report.</td>
</tr>
<tr>
<td></td>
<td>For further information, refer to Test Framework (<a href="#">AutomationDesk Library Reference</a>).</td>
</tr>
</tbody>
</table>
The Report library provides the new AddTreeNode block to add a specific node to the navigation tree in the report. The report itself now contains information on the operation modes of the built-in libraries, the paths to the opened custom libraries and the termination state of the executed sequences.

You can find the new CustomizedPlots demo project in %DSPACE_ROOT%\Demos\AutomationDesk\ReportLibrary, which shows you how to customize the layout of a plot.

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

Evaluation Some of the converter blocks in the Evaluation library provide the new ForceStrictlyIncreasing data object to manipulate input signals that do not provide strictly increasing data. There are some predefined methods for manipulation.

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

Enhancements to the COM API

The AutomationDesk COM API provides the following enhancements:

- Modifications in the AutomationDesk user interface and the AutomationDesk Automation Server are now synchronized in both directions.
- The new OperationMode property gets or sets the operation mode of a built-in library to online, offline, or offline recording.
- The interface of the DT57LogicalLink object has been changed. The ComPrimitives and Services methods now return the new LogicalLinkChildBase object.
- The interface of the D3LogicalLink object has been changed. The ControlPrimitives and Services methods now return the new LogicalLinkChildBase object.
- The interface of the MC3LogicalLink object has been changed. The Collectors and Characteristics methods now return the new LogicalLinkChildBase object.

Enhancements to project handling

The following changes enhance AutomationDesk’s usability:

- There are new options in the Find and Find Inconsistencies dialogs.
- You can remove all results from a project with one click.
- You can locate the template of an instantiated element in the library.
- The AutomationDesk Python Editor now uses four spaces instead of a tab. This improves compatibility with the PythonWin Editor.
■ You can enable and disable automation blocks in multiselect mode.
■ You can insert hyperlinks in the descriptions of a block.
■ There is an additional namespace, called _INFO_, for read-only block attributes.
■ The layout of the AutomationDesk user interface can be saved to a view set.

Migrating to AutomationDesk 3.2

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.

Before you open an older project with the new AutomationDesk version, ensure the following preconditions are fulfilled:
■ You must create backups of the project and of the linked custom libraries.
■ AutomationDesk must be running properly. There must not be any error messages.
■ The built-in libraries, required custom libraries and other packages must be correctly loaded.

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

Migrating from AutomationDesk 1.x to AutomationDesk 2.x or 3.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 or 3.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 manually to make it 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

<table>
<thead>
<tr>
<th>Information in this section</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASM Base InCylinder Blockset</td>
</tr>
<tr>
<td>ASM Diesel Exhaust Blockset</td>
</tr>
<tr>
<td>ASM Diesel InCylinder Blockset</td>
</tr>
<tr>
<td>ASM Drivetrain Basic Blockset</td>
</tr>
<tr>
<td>ASM Electric Components Blockset</td>
</tr>
<tr>
<td>ASM Engine Diesel Blockset</td>
</tr>
<tr>
<td>ASM Engine Gasoline Basic Blockset</td>
</tr>
<tr>
<td>ASM Engine Gasoline Blockset</td>
</tr>
<tr>
<td>ASM Environment Blockset</td>
</tr>
<tr>
<td>ASM Gasoline InCylinder Blockset</td>
</tr>
<tr>
<td>ASM Optimizer</td>
</tr>
<tr>
<td>ASM Parameterization Tool</td>
</tr>
<tr>
<td>ASM Traffic Blockset</td>
</tr>
<tr>
<td>ASM Trailer Blockset</td>
</tr>
<tr>
<td>ASM Truck Blockset</td>
</tr>
<tr>
<td>ASM Turbocharger Blockset</td>
</tr>
<tr>
<td>ASM Vehicle Dynamics Blockset</td>
</tr>
</tbody>
</table>
Information in other sections

Migrating ASM Models (ASM User Guide)
Provides general information on the migration process of ASM models.
ASM Base InCylinder Blockset

Migrating to ASM Base InCylinder Blockset 1.2

<table>
<thead>
<tr>
<th>Block</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>INTAKE_MANIFOLD_</strong></td>
<td>This new block has been introduced to solve issues with correct cylinder filling. For details, refer to Intake Manifold Boost (<a href="#">ASM Base InCylinder Reference</a>).</td>
</tr>
<tr>
<td><strong>BOOST block</strong></td>
<td></td>
</tr>
<tr>
<td><strong>INJECTOR_CONTINUOUS</strong></td>
<td>The handling of injection signals around 360° crank angle has been improved. The injection time is now estimated by integration instead of being calculated from the crank angle difference and engine speed. This avoids inaccuracy with fast-changing engine speeds. The t_inj and phi_inj signals have been added to the ASMSignalBus.</td>
</tr>
<tr>
<td><strong>EXHAUST_VALVE block</strong></td>
<td>There is a switch block after the optimization maps. Now you can easily switch between the map and the constant value to investigate parameterization using ControlDesk.</td>
</tr>
<tr>
<td><strong>INTAKE_VALVE block</strong></td>
<td>There is a switch block after the optimization maps. Now you can easily switch between the map and the constant value to investigate parameterization using ControlDesk.</td>
</tr>
<tr>
<td><strong>WALL_HEAT block</strong></td>
<td>There is a switch block after the optimization maps. Now you can easily switch between the map and the constant value to investigate parameterization using ControlDesk.</td>
</tr>
</tbody>
</table>
ASM Diesel Exhaust Blockset

Migrating to ASM Diesel Exhaust Blockset 1.1.2

SWITCHES_EXHAUST SYSTEM block

The text of the mask prompt has been corrected.
ASM Diesel InCylinder Blockset

Where to go from here

<table>
<thead>
<tr>
<th>Information in this section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Changes in the ASM Diesel InCylinder Demo Model</td>
</tr>
<tr>
<td>Migrating to ASM Diesel InCylinder Blockset 1.1</td>
</tr>
</tbody>
</table>

Changes in the ASM Diesel InCylinder Demo Model

Parameterization function

The parameterization functions have been updated. Now the `asm_table_generator` function is used for table generation. This makes it easier to read and to modify the extrapolation.

Comparison with measurement data

The model has been extended with an option to compare simulation results and measurement data. You have to specify only the number of the engine operating points from the steady state measurement used for ASM Parameterization. Then the engine speed and injection quantity are set automatically according to the engine operating point. You can easily compare all the other measured variables including the in-cylinder pressure curve in ControlDesk.

Migrating to ASM Diesel InCylinder Blockset 1.1

SOFTAPU block

The handling of injection signals around 360° crank angle has been improved.

HEAT_RELEASE_CHMELA block

Behind the optimization maps, a switch block is used. Now you can easily switch between the map and the constant value to investigate parameterization using ControlDesk.

HEAT_RELEASE_ARRHENIUS block

Behind the optimization maps, a switch block is used. Now you can easily switch between the map and the constant value to investigate parameterization using ControlDesk.
ASM Drivetrain Basic Blockset

Where to go from here

Information in this section

New Features of ASM Drivetrain Basic Blockset 1.4 34
Migrating to ASM Drivetrain Basic Blockset 1.4 34

New Features of ASM Drivetrain Basic Blockset 1.4

Driving cycle

The driving cycle JC08 is now included as a demo cycle.

Migrating to ASM Drivetrain Basic Blockset 1.4

SOFT_ECU_TRANSMISSION_BASIC block

The parameterization MDL structure has been changed from MDL.SoftECU.SoftECUTransmission to MDL.SoftECU.SoftECUTransmissionBasic. This makes the parameter independent of the parameter of the SOFT_ECU_TRANSMISSION in the vehicle dynamics model. The renaming is automatically generated in the postmigrate variant.

Driving cycle

The JC08 driving cycle is now included as a demo cycle. If desired, you can copy it to existing projects from the current demo projects of your installation.
ASM Electric Components Blockset

Where to go from here

<table>
<thead>
<tr>
<th>Information in this section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Features of ASM Electric Components Blockset 2.0</td>
<td>35</td>
</tr>
<tr>
<td>Changes in the ASM Electric Components Demo Model</td>
<td>36</td>
</tr>
<tr>
<td>Migrating to ASM Electric Components Blockset 2.0</td>
<td>36</td>
</tr>
</tbody>
</table>

New Features of ASM Electric Components Blockset 2.0

**BATTERY block**

The diffusion behavior is simulated by an additional RC circuit.

To simulate the loss voltage behavior of the battery, the resistance of the model is parameterized via a look-up table which depends on the charge state and the battery current.

All the parameter values for the different resistances, the inductance and capacitances must be parameterized for one battery cell.

The loss current, the inductance, the double layer and the diffusion behavior can be enabled or disabled via a switch.

If you want to change back to the former battery implementation, you can find the former block at subsystem Accumulator/FormerVersions in the ASM_ElectricComponents_lib library. Just drag the former block to your Simulink model.
Changes in the ASM Electric Components Demo Model

ModelDesk plotting
The demo model now supports plotting of simulation data in ModelDesk. The `ModelDesk_Plotting` block has been added to the `MDL_UserInterface` subsystem.

Migrating to ASM Electric Components Blockset 2.0

STARTER block
Internal block adaptations have been performed without any functional changes.
The `KI_50_active[0|1]` signal label in ASMSignalBus has changed to `Sw_StarterReq[0Off|1On]`. As all the relevant Simulink Bus Selector blocks automatically contain the new signal label, no additional migration steps are required.

ALTERNATOR block
Internal block adaptations have been performed without any functional changes.

BATTERY block
As the new implementation of the battery model changed a lot and parameters cannot be migrated automatically, the link to the battery library is changed to the former implementation (FormerVersion/BATTERY_4_0 subsystem) during the migration of older ASM models. Thus, the simulation behavior is not changed. If you want to use the new battery implementation, just drag the `BATTERY` block from the `ASM_ElectricComponent_lib` Simulink library to your model and adapt the new parameters to your needs.
ASM Engine Diesel Blockset

Where to go from here

Information in this section

Changes in the ASM Engine Diesel Demo Model

ModelDesk plotting

The demo model now supports plotting of simulation data in ModelDesk. The ModelDesk_Plotting block has been added to the MDL_UserInterface subsystem.

Migrating to ASM Engine Diesel Blockset 1.4

INTAKE_MANIFOLD block

The integrator lower limits have been set to zero. Summation blocks have been inserted to allow vector-based massflow inputs.

EXHAUST_MANIFOLD block

The integrator lower limits have been set to zero. Summation blocks have been inserted to allow vector-based massflow inputs.

INTERCOOLER block

Internal block adaptations have been performed without any functional changes. Vector-based massflow calculations are now possible.

EGRCOOLER block

Internal block adaptations have been performed without any functional changes. Vector-based massflow calculations are now possible.

INJECTOR block

The calculation of the post injection quantity is now cumulated for all cylinders, instead of representing the mean injection quantity for only one cylinder.

New Features and Migration  November 2010
ASM Engine Gasoline Basic Blockset

Where to go from here

<table>
<thead>
<tr>
<th>Information in this section</th>
<th>38</th>
</tr>
</thead>
<tbody>
<tr>
<td>Changes in the ASM Engine Gasoline Basic Demo Model</td>
<td>38</td>
</tr>
<tr>
<td>Migrating to ASM Engine Gasoline Basic Blockset 1.3.5</td>
<td>38</td>
</tr>
</tbody>
</table>

Changes in the ASM Engine Gasoline Basic Demo Model

ModelDesk plotting

The demo model now supports plotting of simulation data in ModelDesk. The ModelDesk_Plotting block has been added to the MDL_UserInterface subsystem.

Migrating to ASM Engine Gasoline Basic Blockset 1.3.5

INTAKE_MANIFOLD block

The integrator lower limits have been set to zero. Summation blocks have been inserted to allow vector-based massflow inputs.

THROTTLE block

Saturations have been removed to allow backflow through the throttle and consequently pressure balancing in the intake manifold during engine standstill.

WALL_FILM block

The integrator for the cumulated wall film mass is reset by toggling the wall film mode switch from off to on.
ASM Engine Gasoline Blockset

Where to go from here

Information in this section

| Changes in the ASM Engine Gasoline Demo Model | 39 |
| Migrating to ASM Engine Gasoline Blockset 2.2 | 39 |

Changes in the ASM Engine Gasoline Demo Model

ModelDesk plotting

The demo model now supports plotting of simulation data in ModelDesk. The ModelDesk_Plotting block has been added to the MDL_UserInterface subsystem.

Migrating to ASM Engine Gasoline Blockset 2.2

<table>
<thead>
<tr>
<th>Block</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMBUSTION_TORQUE_CI</td>
<td>The evaluation of the mean temperature has been modified together with some internal block adaptations without any functional changes.</td>
</tr>
<tr>
<td>COMBUSTION_TORQUE_SI</td>
<td>The evaluation of the mean temperature has been modified together with some internal block adaptations without any functional changes.</td>
</tr>
<tr>
<td>INTAKE_MANIFOLD block</td>
<td>The integrator lower limits have been set to zero. Summation blocks have been inserted to allow vector-based massflow inputs.</td>
</tr>
<tr>
<td>EXHAUST_MANIFOLD block</td>
<td>The integrator lower limits have been set to zero. Summation blocks have been inserted to allow vector-based massflow inputs.</td>
</tr>
<tr>
<td>THROTTLE block</td>
<td>Saturations have been removed to allow backflow through the throttle and consequently pressure balancing in the intake manifold during engine standstill.</td>
</tr>
<tr>
<td>INTERCOOLER block</td>
<td>Internal block adaptations have been performed without any functional changes. Vector-based massflow calculations are now possible.</td>
</tr>
</tbody>
</table>
### EGRCOOLER block
Internal block adaptations have been performed without any functional changes. Vector-based massflow calculations are now possible.

### Related topics
- Basics
- *Migrating ASM Models* ([ASM User Guide](#))
ASM Environment Blockset

Where to go from here

Information in this section

| New Features of ASM Environment Blockset 1.5 | 41 |
| Migrating to ASM Environment Blockset 1.5 | 41 |

New Features of ASM Environment Blockset 1.5

RoadConverter

A tool to create ASM roads from road measurement data or from GPS data is provided. For details, refer to ASM RoadConverter (ASM User Guide).

ROAD block

It is now possible to download an unlimited number of road segments. For details, refer to ASM DownloadRoad (ASM User Guide).

Migrating to ASM Environment Blockset 1.5

BASIC_ROADS block

The rate-limiter sample modes at slope and lateral slope signals have been changed from "continuous" to "inherited". Thus, the signal is of discrete sample time and not of continuous one. Now the ASM_Truck model can be used in combination with ASM_VehicleDynamics Operator.

CONTROLLER block

The Const_dens_Air parameter was renamed to Const_Dens_Air. The Const_Cof_Cw_x_Vehicle parameter was removed. This information is now provided as an input signal.

The new Coef_Cw_x_Vehicle[] input was added. During migration a subsystem with a constant block is connected to the new input and its constant value is set to the value of MDL.Environment.Driver.LongitudinalController.Controller.Const_Cof_Cw_x_Vehicle.
For correct simulation results, the VehicleDynamics.Aerodynamics.Coefficients.Cw_x[] signal from the ASMSignalBus should be connected to the new Coef_Cw_x_Vehicle[] import. This must be done manually. For models created with dSPACE Release 6.6, refer to the ASM_VehicleDynamics demo model as an example of how to connect the bus signal using the Environment interface block (ASM_VehicleDynamics/MDL/Environment/EnvironmentInterface_In).

If you use ModelDesk automation, note that the Environment.CONTROLLER.Const_Coef_Cw_x_Vehicle parameter was removed. The Environment.CONTROLLER.Const_dens_Air parameter was renamed to Environment.CONTROLLER.Const_Dens_Air.

ROAD block

The calculation of the contact points in the ROAD Simulink S-function has been improved. This improvement also reduces the time taken by contact point calculation.

GEAR_SHIFTER block

It is now possible to set the Implement logic signals as boolean data (vs. double) option in the Operator Simulink models.

v_ROAD_REF block

It is now possible to set the Implement logic signals as boolean data (vs. double) option in the Operator Simulink models.

LATERAL_CONTROL1 block

Setting the preview time to zero previously caused invalid simulation results (division by zero). This has been corrected.
ASM Gasoline InCylinder Blockset

Where to go from here

Information in this section

<table>
<thead>
<tr>
<th>Changes in the ASM Gasoline InCylinder Demo Model</th>
<th>43</th>
</tr>
</thead>
<tbody>
<tr>
<td>Migrating to ASM Gasoline InCylinder Blockset 1.1</td>
<td>44</td>
</tr>
</tbody>
</table>

Changes in the ASM Gasoline InCylinder Demo Model

Engine operation point

The engine operation point definition has been changed from engine speed and accelerator pedal position to engine speed and relative air mass. This affects the model, all parameterization functions which generate engine operation point-dependent maps, and the optimizer project.

Parameterization function

The parameterization functions have been updated. Now the asm_table generator function is used for table generation. This makes it easier to read and to modify the extrapolation.

Comparison with measurement data

The model has been extended with an option to compare simulation results and measurement data. You have to specify only the number of the engine operation point from the steady state measurement used for ASMPararameterization. Then the engine speed and relative air mass are set automatically according to the engine operation point. You can easily compare all the other measured variables including the in-cylinder pressure curve in ControlDesk.

Optimization of cylinder filling

The model has been extended with the new INTAKE_MANIFOLD_BOOST block. For details on the block, see the block documentation. The optimization project has been changed so that the p_InManBoost map of this block is optimized instead of the flow coefficient of the intake valve.
### Migrating to ASM Gasoline InCylinder Blockset 1.1

<table>
<thead>
<tr>
<th>Block/Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>COMMON_INCYLINDER_GASOLINE_PARAMETERS block</strong></td>
<td>The new <code>Const_m_Air_ref</code> parameter has been added. In the postmigrate variant, the parameter is copied from the SoftECU InCylinder Gasoline.</td>
</tr>
<tr>
<td><strong>DIRECTINJECTOR_CONTINUOUS block</strong></td>
<td>The handling of injection signals around 360° crank angle has been improved. The injection time is now estimated by integration instead of being calculated from the crank angle difference and engine speed. This avoids inaccuracy with fast-changing engine speeds. The <code>t_inj</code> and <code>phi_inj</code> signals have been added to the ASMSignalBus.</td>
</tr>
<tr>
<td><strong>PORTINJECTOR_CONTINUOUS block</strong></td>
<td>The handling of injection signals around a 360° crank angle has been improved. The injection time is now estimated by integration instead of being calculated from the crank angle difference and engine speed. This avoids inaccuracy with fast-changing engine speeds. The <code>t_inj</code> and <code>phi_inj</code> signals have been added to the ASMSignalBus.</td>
</tr>
<tr>
<td><strong>SOFTAPU block</strong></td>
<td>The capture window, number of expected injections and duration below which injections are ignored can now be specified for direct and port injection separately. The handling of injection signals around a 360° crank angle has been improved. Related adaptions of the parameterization have been added to the postmigrate function.</td>
</tr>
<tr>
<td><strong>SOFT_ECU_INCYLINDER_GASOLINE block</strong></td>
<td>The unit of <code>Map_phi_ign_opt</code> has been changed from ( \phi_{\text{ign}} \text{[rad]} = f(n_{\text{Engine}[\text{rad/s}], \text{rel air mass}[\text{]}]) ) to ( \phi_{\text{ign}} \text{[deg]} = f(n_{\text{Engine}[\text{rpm}], \text{rel air mass}[\text{]}]) ). The parameterization has been adapted accordingly in postmigrate function.</td>
</tr>
<tr>
<td><strong>HEATRELEASE_VIBE block</strong></td>
<td>The block has been redesigned so that the parameter calculation from Csallner and from optimizer maps are now located in separate enabled subsystems. Furthermore, a trigger hold block is used to keep the parameter fix during a combustion cycle. Behind the optimization maps, a switch block is used. Now you can easily switch between the map and the constant value to investigate parameterization using ControlDesk.</td>
</tr>
</tbody>
</table>
The unit of the ignition angle has been changed from \([\text{rad}]\) to \([\text{deg}]\). This influences the related input port and the following parameters:

- \text{Const\_phi\_ign\_Ref}
- \text{Const\_phi\_ID\_Ref}
- \text{Const\_phi\_CD\_Ref}
- \text{Const\_phi\_BurnDuration}
- \text{Const\_phi\_IgnitionDelay}
- \text{Map\_phi\_BurnDuration}
- \text{Map\_phi\_IgnitionDelay}

All these parameters are scaled in the postmigrate variant.

Special adaptations are required in the following cases:

- If you only restart the postprocessing of the optimizer, no changes are necessary. The generated maps will still contain the values in radians. They are converted from radians to degrees by the postmigrate variant.
- If you perform the optimization again for these parameters and then generate new initialization files, they will already be in degrees. In this case you have to modify the related postmigrate variant file at: _asmmigratepost\IniFiles\mig500\incylgas\mig500\heat\release\vibe.m. At the end of the file is a marked section with the scaling of each parameter from radians to degrees which has to be removed or commented out if the initialization files already have degrees as the unit.

\textbf{CATALYST}

Instead of \(\lambda\), \(1/\lambda\) is now delayed with a second-order delay. This improves the behavior during start up, where the old implementation resulted in too-high \(\lambda\) values.
ASM Optimizer

Where to go from here

Information in this section

<table>
<thead>
<tr>
<th>New Features of ASM Optimizer 1.3</th>
<th>46</th>
</tr>
</thead>
<tbody>
<tr>
<td>Migrating to ASM Optimizer Blockset 1.3</td>
<td>47</td>
</tr>
</tbody>
</table>

New Features of ASM Optimizer 1.3

Performance

The performance has been improved, especially for experiments with a number of operating points.

Reading in-cylinder measurements from MAT files

In-cylinder measurement data can now also be read from MAT files. This improves performance, especially when a high number of operating points is used. For a detailed description, refer to ASM Optimizer User Guide.

Task Parameters page

The parameters page has been redesigned. The edit fields have been rearranged. Two new input fields have been added for specifying the name and the unit of an optimized parameter. This name and unit are transferred to the related fields of the generated maps.

Postprocessing

The postprocessing has been redesigned. It is now possible to specify separate axes for each generated map. The default axis of the engine operation point and therefore of the generated maps of the in-cylinder gasoline demo has changed from Engine speed and accelerator pedal position to engine speed and relative air mass. Map generation is now done by a user-editable script which calls the `asm_tablegen generator` function. You can define interpolation and extrapolation. Some buttons and checkboxes have been renamed to make their functions clearer.
Migrating to ASM Optimizer Blockset 1.3

Task-parameters page

Default values are set for the new fields to keep behavior similar.

Postprocessing

The settings for the axes are automatically set to ensure identical behavior to that before migration. The script for extrapolation is generated from the extrapolation information (set fixed point, extrapolation value for lower/upper x-axis, y-axis) included in the project before migration. A backward compatibility flag has been added to the call of \texttt{asm_tablegenerator} so that \texttt{asm_tablegenerator} uses the same extrapolation algorithm as the previously used \texttt{dsutil_tablegenerator} function. It is recommended to remove this compatibility flag and modify the extrapolation according to your needs.

Special migration for Gasoline InCylinder experiments

There are some changes in the Gasoline InCylinder model which also affect the optimization project.

\textbf{Parameter change in the SOFT_APU block} The Map\_Capture\_Window parameter has been copied to Map\_Capture\_Window\ Direct and Map\_Capture\_Window\ Port. The make single cylinder variant script is therefore automatically adapted after a backup is created. If this fails, compare the backup with the file of the current demo project of your installation and merge manually.

\textbf{Parameter change in the HEAT\_RELEASE\_VIBES block} The units of the Map\_phi\_BurnDuration.v and Map\_phi\_IgnitionDelay.v parameters have been changed from radian to degree. If you only restart postprocessing, no changes are necessary. The generated maps will still contain the values in radians. They are converted from radians to degrees by the postmigrate variant. If you perform the optimization for these parameters again and then generate new initialization files, they will already be in degrees. In this case you have to modify the related postmigrate variant file at: 

\textbackslash \texttt{_asm\_migrate\_post\_\ini\_files\_mig500\_incyl\_gas\_mig500\_heat\_release\_vibe.m}.

At the end of the file is a marked section with the scaling of each parameter from radians to degrees which has to be removed or commented out if the initialization files already have degrees as the unit.
**Engine operating point**  The engine operating points and therefore the axes of the generated maps still have the old definition of engine speed and accelerator pedal position. They are not changed during migration of the project. Automatic migration is not possible because changing this would also require changes in the definition of the engine operating point in the model and all the engine operating point-dependent maps in the ASM Parameterization project.
ASM Parameterization Tool

New Features of the ASM Parameterization Tool 1.5.1

| Performance | The performance for generating initialization files has been improved. The execution time has been decreased, the number of decimal places in the initialization files has been increased, and the tabbing of table entries has been changed. |

---

---
ASM Traffic Blockset

Where to go from here

<table>
<thead>
<tr>
<th>Information in this section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Changes in the ASM Traffic Demo Model</td>
</tr>
<tr>
<td>Migrating to ASM Traffic Blockset 1.1.4</td>
</tr>
</tbody>
</table>

Changes in the ASM Traffic Demo Model

ModelDesk plotting

The demo model now supports plotting of simulation data in ModelDesk. The ModelDesk_Plotting block has been added to the MDL_UserInterface subsystem.

Migrating to ASM Traffic Blockset 1.1.4

COORDINATE_TRANSFORMATION block

Sensor calculation performance was improved.

SENSOR_POSITION block

Sensor calculation performance was improved.

NEAREST_POINT block

Sensor calculation performance was improved.

NEAREST_SURFACE block

Sensor calculation performance was improved.

RADARSENSOR_3D block

Previously, the horizontal angle calculation was saturated after driving multiple laps of closed roads. This has been corrected.
ASM Trailer Blockset

Where to go from here

Information in this section

- New Features of ASM Trailer Blockset 1.3
- Changes in the ASM Trailer Demo Model
- Migrating to ASM Trailer Blockset 1.3

New Features of ASM Trailer Blockset 1.3

TIRE_MODEL_TMEASY

The TMEasy tire model has been extended with bore torque calculation at very low or zero vehicle speed, for example, during parking maneuvers. This ensures the generation of a proper parking torque response to steering at very low or zero speed. For the calculation of this extension, refer to Torques Calculation (ASM Vehicle Dynamics Addendum).

Changes in the ASM Trailer Demo Model

ModelDesk plotting

The demo model now supports plotting of simulation data in ModelDesk. The ModelDesk_Plotting block has been added to the MDL_UserInterface subsystem.

Migrating to ASM Trailer Blockset 1.3

TIRE_MODEL_TMEASY block

Bore torque calculation at very low or zero vehicle speed was improved. For details, refer to New Features of ASM Trailer Blockset 1.3 on page 51.

WHEEL_SPEED block

The new omega_Wheel_Init[rads] inport has been added to set the initial wheel speed from outside.
ASM Truck Blockset

Where to go from here

<table>
<thead>
<tr>
<th>Information in this section</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>New Features of ASM Truck Blockset 1.2</td>
<td>52</td>
</tr>
<tr>
<td>Changes in the ASM Truck Demo Model</td>
<td>52</td>
</tr>
<tr>
<td>Migrating to ASM Truck Blockset 1.2</td>
<td>53</td>
</tr>
</tbody>
</table>

New Features of ASM Truck Blockset 1.2

TIRE_MODEL_TMEASY

The TMEasy tire model has been extended with bore torque calculation at very low or zero vehicle speed, for example, during parking maneuvers. This ensures the generation of a proper parking torque response to steering at very low or zero speed. For the calculation of this extension, refer to Torques Calculation (ASM Vehicle Dynamics Addendum).

Changes in the ASM Truck Demo Model

ModelDesk plotting

The demo model now supports plotting of simulation data in ModelDesk. The ModelDesk_Plotting block has been added to the MDL_UserInterface subsystem.

New demo model

The demo project has been expanded by a new model variant. The former variant ASM_Truck has been renamed to ASM_TruckSemitrailer. The new variant ASM_Truck contains a truck model with a torsional frame and without a trailer. The variants can be activated by changes in the go.m script.
Migrating to ASM Truck Blockset 1.2

<table>
<thead>
<tr>
<th>Block</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIRE_MODEL_TMEASY</td>
<td>Bore torque calculation at very low or zero vehicle speed was improved. For details, refer to New Features of ASM Truck Blockset 1.2 on page 52.</td>
</tr>
<tr>
<td>WHEEL_SPEED</td>
<td>The new omega.Wheel_Init[rad/s] input has been added to set the initial wheel speed from outside.</td>
</tr>
</tbody>
</table>
ASM Turbocharger Blockset

Migrating to ASM Turbocharger Blockset 1.5

MAPS_TC

The map-based turbocharger block now uses the model ambient conditions as initialization values. The compressor output pressure is not influenced below a parameterized engine speed.
ASM Vehicle Dynamics Blockset

<table>
<thead>
<tr>
<th>Where to go from here</th>
<th>Information in this section</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Features of ASM Vehicle Dynamics Blockset 1.6</td>
<td>56</td>
</tr>
<tr>
<td>Changes in the ASM Vehicle Dynamics Demo Model</td>
<td>56</td>
</tr>
<tr>
<td>Migrating to ASM Vehicle Dynamics Blockset 1.3</td>
<td>56</td>
</tr>
</tbody>
</table>
New Features of ASM Vehicle Dynamics Blockset 1.6

**TIRE_MODEL_TMEASY**
The TMEasy tire model has been extended with bore torque calculation at very low or zero vehicle speed, for example, during parking maneuvers. This ensures the generation of a proper parking torque response to steering at very low or zero speed. For the calculation of this extension, refer to Torques Calculation ([ASM Vehicle Dynamics Addendum](#)).

Changes in the ASM Vehicle Dynamics Demo Model

**ModelDesk plotting**
The demo model now supports plotting of simulation data in ModelDesk. The ModelDesk_Plotting block has been added to the ModelDesk_Plotting subsystem.

Migrating to ASM Vehicle Dynamics Blockset 1.3

**ASM_VehicleDynamics demo model**
The error in the *w_Vehicle_CoG_dt[xyz][rad|s²]* block has been corrected from "…[m|s²]" into "[rad|s²]". The block is used in ASM_VehicleDynamics - MDLUserInterface - VehicleDynamics - MDL_DISP - Accelerations. If there are any ControlDesk layouts referring to this block, the connection has to be renewed manually.

**STEERING block**
The new import *Trq_Driver[Nm]* has been added to set the driver torque from outside. In addition, the new import *Sw_DriverSteeringMode[1Angle2Torque]* has been added to specify whether to use the steering wheel angle or torque from the driver as input to the steering system. If the driver’s torque is used as input to the steering system, the steering column calculation is ignored.

Signals representing the spring torque and spring stiffness at the steering column have been added to the ASMSignalBus. The ASMSignalBus hierarchy has been changed: the ForceSteeringRod entry has been replaced with ForcesAndTorques. The new bus hierarchy is migrated automatically.
<table>
<thead>
<tr>
<th>Block Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>STEERING_VARIABLE_RATIO block</strong></td>
<td>The new import <code>Trq_Driver[Nm]</code> has been added to set the driver torque from outside. In addition, the new import `Sw_DriverSteeringMode[1Angle</td>
</tr>
<tr>
<td><strong>WHEEL_SPEED block</strong></td>
<td>The new `omega_Wheel_Init[FL;FR;RL;RR][rad</td>
</tr>
<tr>
<td><strong>TIRE_MODEL_TMEASY block</strong></td>
<td>Bore torque calculation at very low or zero vehicle speed was improved. For details, refer to <em>New Features of ASM Vehicle Dynamics Blockset 1.6</em> on page 56.</td>
</tr>
<tr>
<td><strong>VEHICLE_MOTION_CAR block</strong></td>
<td>The new <code>q_dt_Init[10x1]</code> inport has been added to set initial vehicle velocities, like initial vehicle longitudinal speed, and wheels’ relative velocities with regard to the vehicle body. The new <code>q_Init[4x1]</code> inport has been added to set initial wheels’ position relative to the vehicle body.</td>
</tr>
<tr>
<td><strong>CENTRAL_DIFFERENTIAL block</strong></td>
<td>The new `omega_Cage[rad</td>
</tr>
<tr>
<td><strong>FRONT_DIFFERENTIAL block</strong></td>
<td>The new `omega_Cage[rad</td>
</tr>
<tr>
<td><strong>REAR_DIFFERENTIAL block</strong></td>
<td>The new `omega_Cage[rad</td>
</tr>
</tbody>
</table>
ControlDesk

New Features of ControlDesk 3.7

<table>
<thead>
<tr>
<th>MicroAutoBox II support</th>
<th>The ControlDesk support of MicroAutoBox II was enhanced:</th>
</tr>
</thead>
<tbody>
<tr>
<td>■ ControlDesk can access USB mass storage devices which are connected to MicroAutoBox II and used for flight recording. You can read the BIN files saved during flight recording, convert them into MAT or CSV files and delete them from the USB mass storage device. For details, refer to How to Upload Flight Recorder Data Written to a USB Mass Storage Device (MicroAutoBox II only) (<a href="#">ControlDesk Experiment Guide</a>).</td>
<td></td>
</tr>
<tr>
<td>■ A property page displays information on the FPGA module of MicroAutoBox II.</td>
<td></td>
</tr>
<tr>
<td>■ A property page displays the time which a MicroAutoBox II is in operation.</td>
<td></td>
</tr>
</tbody>
</table>
There are limitations for using MicroAutoBox II:

- Hot plugging is not supported for MicroAutoBox II:
  When you reconnect MicroAutoBox II to the host PC, the MicroAutoBox II platform remains in the disconnected state. To reconnect MicroAutoBox II, use the Refresh Platform Connection command.
  For details, refer to Notes and Tips on Working with MicroAutoBox in a Vehicle (MicroAutoBox II Hardware Installation and Configuration).

If several MicroAutoBox II are members of a multiconnect group (specified in the DSGROUPS.ini), the firmware update is limited:

- The update fails if at least one of them is running in secure mode. For details on the secure mode, refer to Checking MicroAutoBox (MicroAutoBox II Hardware Installation and Configuration).
- The update fails if at least one of them has a host interface firmware with a revision earlier than 1.4.
- Only the firmware of the working board is updated.

### DS802 support

There is a new property page for the DS802 that displays:

- The port states of the DS802 board
- A tree view of the ports and the I/O boards connected to them.

### Firmware update

The scoutcmd command line program can update the firmware of dSPACE real-time boards. Thus, you can update the firmware of the boards via command line in a DOS box without starting ControlDesk. For details, refer to Updating or Downgrading the Firmware Via Command Line (ControlDesk Experiment Guide).

### Bus Navigator

**Monitoring and Logging of LIN bus communication** The Bus Navigator supports filtered and unfiltered monitoring and logging of LIN bus communication. For details, refer to Monitoring, Logging and Replaying a Bus Communication (ControlDesk Experiment Guide).

### Related topics

- References
  - Refresh Platform Connection (ControlDesk Reference)
# ControlDesk Next Generation

<table>
<thead>
<tr>
<th>Where to go from here</th>
<th>Information in this section</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>New Features of ControlDesk Next Generation</strong>&lt;br&gt;(ControlDesk 4.0)</td>
</tr>
<tr>
<td></td>
<td>ControlDesk Next Generation is the successor to ControlDesk and CalDesk. You get information on the features of ControlDesk Next Generation compared with CalDesk 3.0.1.</td>
</tr>
<tr>
<td></td>
<td><strong>Migrating to ControlDesk Next Generation</strong>&lt;br&gt;(ControlDesk 4.0)</td>
</tr>
<tr>
<td></td>
<td>ControlDesk Next Generation (ControlDesk 4.0) is the successor to ControlDesk and CalDesk. You get information on migrating from ControlDesk 3.x and from CalDesk. are the features of ControlDesk Next Generation compared with CalDesk 3.0.1.</td>
</tr>
<tr>
<td></td>
<td><strong>New Features of the Variable Editor</strong>&lt;br&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Information in other sections:

- **ControlDesk Next Generation Migration Guide**
  - Explains migration from CalDesk/ControlDesk to ControlDesk Next Generation.

---

New Features and Migration  November 2010  61
New Features of ControlDesk Next Generation (ControlDesk 4.0)

ControlDesk Next Generation is the successor to ControlDesk and CalDesk.

Since the technological development of ControlDesk Next Generation is based on CalDesk, however, there are conceptual differences between ControlDesk 3.x and ControlDesk Next Generation. For this reason, the new features described below are enhancements and changes with regard to CalDesk 3.0.1 (and not to ControlDesk 3.x).

<table>
<thead>
<tr>
<th>Where to go from here</th>
<th>Information in this section</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Project and Experiment Features</td>
<td>63</td>
</tr>
<tr>
<td>New Bus Navigator for CAN, LIN, and FlexRay Bus Communication</td>
<td>63</td>
</tr>
<tr>
<td>New Features of Platform Management and Platforms/Devices</td>
<td>65</td>
</tr>
<tr>
<td>New Visualization Features</td>
<td>68</td>
</tr>
<tr>
<td>New Instrument Features</td>
<td>70</td>
</tr>
<tr>
<td>New Measurement and Recording Features</td>
<td>73</td>
</tr>
<tr>
<td>New Data Set Management Features</td>
<td>76</td>
</tr>
<tr>
<td>New Features of the ECU Diagnostics Module</td>
<td>77</td>
</tr>
<tr>
<td>New Signal Editor</td>
<td>81</td>
</tr>
<tr>
<td>New Failure Simulation Module</td>
<td>82</td>
</tr>
<tr>
<td>New Automation Features</td>
<td>85</td>
</tr>
<tr>
<td>Further Enhancements with ControlDesk Next Generation</td>
<td>86</td>
</tr>
</tbody>
</table>
New Project and Experiment Features

ControlDesk Next Generation is the successor to ControlDesk and CalDesk.

The new features described below are enhancements and changes with regard to CalDesk 3.0.1 (and not to ControlDesk 3.x).

<table>
<thead>
<tr>
<th>Projects/experiments based on ControlDesk CDX files</th>
</tr>
</thead>
<tbody>
<tr>
<td>You can define a new project or a new experiment in ControlDesk Next Generation based on an experiment created with ControlDesk 3.x. Layouts contained in the ControlDesk 3.x experiment are migrated automatically to the new ControlDesk Next Generation project.</td>
</tr>
<tr>
<td>For details, refer to New Project + Experiment from CDX File / New Experiment from CDX File (ControlDesk Next Generation Reference).</td>
</tr>
</tbody>
</table>

A ControlDesk Next Generation project/experiment cannot be reloaded in ControlDesk 3.x.

For further migration aspects, refer to Migrating to ControlDesk Next Generation on page 88.

New Bus Navigator for CAN, LIN, and FlexRay Bus Communication

ControlDesk Next Generation is the successor to ControlDesk and CalDesk.

The new features described below are enhancements and changes with regard to CalDesk 3.0.1 (and not to ControlDesk 3.x).
The **Bus Navigator** is a new software component which lets you handle CAN messages, LIN frames, and FlexRay PDUs that can be configured by blocks of the RTI CAN MultiMessage Blockset (RTICANMM), blocks of the RTI LIN MultiMessage Blockset (RTILINMM), and settings in the dSPACE FlexRay Configuration Package. With the **Bus Navigator**, you can:

- Create Bus Instruments to view RX messages/frames/PDUs and configure TX messages/frames/PDUs.
- Trigger the transmission of TX messages/frames/PDUs.
- Select several messages/frames/PDUs in the Bus Navigator tree and, for example, open them in one go.
- Specify the transmission status of a CAN controller on dSPACE CAN hardware (for example, on a DS4302 CAN Interface Board) and its TX messages.
- Monitor raw and physical data and log the raw data of CAN messages and LIN frames.
- Replay logged CAN bus communications.
- Monitor, log, and replay CAN bus communications via a PC-based CAN interface, such as the dSPACE DCI-CAN1 or a Vector CANcardXL.

For details, refer to **Handling Bus Communication** ([ControlDesk Next Generation Guide]).
New Features of Platform Management and Platforms/Devices

ControlDesk Next Generation is the successor to ControlDesk and CalDesk.

The new features described below are enhancements and changes with regard to CalDesk 3.0.1 (and not to ControlDesk 3.x).

Supported platforms/devices

- In addition to the DS1005 PPC Board and MicroAutoBox support, ControlDesk Next Generation now also supports the following platforms:
  - DS1006 Processor Board
  - DS1103 PPC Controller Board
  - DS1104 R&D Controller Board
  - Multiprocessor System

ControlDesk Next Generation supports the multicore DS1006 Processor Board. You can register a multicore DS1006 Processor Board as a multiprocessor system or a DS1006 single-processor platform.

For further information on the supported platforms/devices, refer to Platform/Device Descriptions (ControlDesk Next Generation Reference).

- ControlDesk Next Generation does not support the following devices:
  - CAN Data Output
  - CSM ScanMess Modules
  - DCI-GME
  - IMC CANSAS Modules
  - IPETRONIK SIM Modules
  - RapidPro
  - XCP on USB

ControlDesk Next Generation's support for these devices is inactive by default. To activate support, contact dSPACE.
### Registering platform hardware

After installing a single dSPACE processor or controller board or a multiprocessor system, you must register it to make it known to ControlDesk Next Generation. You do not need to register boards that support the plug & play feature (MicroAutoBox connected via DS81x Link Board, DS1104).

Platform hardware can be registered independently of ControlDesk Next Generation experiments. ControlDesk Next Generation creates a platform for each item of registered hardware. The platforms are displayed in the Platform/Device Manager and can be added to experiments later on. The registration data is stored in the recent platform configuration, and the Platform/Device Manager can remember the configuration when ControlDesk Next Generation is restarted.

For further information, refer to How to Register dSPACE Real-Time Hardware (ControlDesk Next Generation Guide).

ControlDesk Next Generation lets you manage your recent platform configuration. You can remove elements from the recent platform configuration, hide registered platforms in ControlDesk Next Generation and import/export the configuration of registered hardware.

For further information, refer to Manage Recent Platform Configuration (ControlDesk Next Generation Reference).

### Assigning registered hardware to a platform in the experiment

To let ControlDesk Next Generation access the connected hardware correctly, you must assign it to an appropriate platform in the ControlDesk Next Generation experiment. ControlDesk Next Generation provides different ways of assigning the hardware:

- You can add a registered platform to the current experiment via the Add to Active Experiment command from within the Platform/Device Manager. The experiment does not need to contain a platform already.

  For further information, refer to How to Assign dSPACE Real-Time Hardware to a Platform (ControlDesk Next Generation Guide).

- You can assign registered hardware to an existing platform in the active experiment. You must specify the assignment settings during platform configuration (using the Configure Platform/Device command in the Project Manager).

  For further information, refer to How to Assign dSPACE Real-Time Hardware to a Platform (ControlDesk Next Generation Guide) and How to Configure a Multiprocessor System Platform (ControlDesk Next Generation Guide).
When you add a new platform to the experiment using the Add Platform/Device command, you can assign a registered platform if one is available. ControlDesk Next Generation lists all platforms that are registered in the system, but not yet assigned to an experiment, according to platform type.

For further information, refer to How to Add a Platform/Device to an Experiment (ControlDesk Next Generation Guide).

The Platform/Device Manager now lists the platforms/devices available in your system that are currently connected to the platform/device hardware or corresponding simulators, i.e., all the platforms that were registered in ControlDesk Next Generation or that are connected to your host PC without having to be registered (MicroAutoBox connected via DS81x Link Board, and DS1104). Platforms/devices that are contained in the current project but are currently disconnected are not listed. So, the Platform/Device Manager can display items even without any ControlDesk Next Generation project and experiment being open.

The Platform/Device Manager also provides information on each platform/device’s state and membership. If a platform/device belongs to an experiment, the name used for it in the experiment is displayed next to the membership icon.

For further information, refer to Platform/Device Manager (ControlDesk Next Generation Reference).

You can specify whether ControlDesk Next Generation should search for connected and registered platforms during startup.

For further information, refer to Platform Management Page (ControlDesk Next Generation Reference).
Platform name in experiment
When a platform/device is added to an experiment or project, an experiment name is specified for it. This name is displayed in the Project Manager, Platform/Device Manager, etc. You can change the default name when adding the platform/device.
Refer to Add Platform/Device (ControlDesk Next Generation Reference).

Start online calibration behavior for platforms
The default start online calibration behavior for platforms was changed from "Prompt user" to "Upload connected variables".
Refer to General Settings Properties (ControlDesk Next Generation Reference).

Support of FlexRay interfaces from Elektrobit
ControlDesk Next Generation now also supports the EB 61x0 (formerly BUSDOCTOR) FlexRay interface modules from Elektrobit.
For a list of the supported interfaces, refer to Supported FlexRay Interfaces (ControlDesk Next Generation Guide).

New Visualization Features
ControlDesk Next Generation is the successor to ControlDesk and CalDesk.
The new features described below are enhancements and changes with regard to CalDesk 3.0.1 (and not to ControlDesk 3.x).

Layout Navigator
ControlDesk Next Generation provides the new Layout Navigator. The Layout Navigator displays all the layouts of the currently active experiment. It gives you quick access to the context menu commands of each layout and can be used for switching between layouts.
For details, refer to Layout Navigator (ControlDesk Next Generation Reference).

**Instrument Navigator**

ControlDesk Next Generation provides the new Instrument Navigator. The Instrument Navigator displays a tree with all the instruments of the active layout and all the variables that are connected to them. The Instrument Navigator’s main function is easy selection of instruments in complex layouts.

You can perform a text search to highlight specific instruments or variables. Various buttons make it easy to browse the search results and select instruments or variables.

For details, refer to Instrument Navigator (ControlDesk Next Generation Reference).

**Creating a layout copy**

You can now create a new layout as a copy of the currently selected layout.

For details, refer to Create Layout Copy (ControlDesk Next Generation Reference).

**Switching the data source of a layout**

You can switch the variable connections of all instruments of a layout to another source (a recorded data file or the running measurement if available).

For details, refer to Switch Data Source (ControlDesk Next Generation Reference).
ControlDesk Next Generation lets you lock/unlock the editing of a layout. Locking layout editing protects a layout against unintentional changes. If layout editing is locked you cannot, for example, delete, add, or move instruments using the mouse. Some actions belonging to the variables of an instrument are also restricted. For example, you cannot add variables via drag and drop. Changes via the Property control bar are not restricted. This gives you the option to make intentional changes even in locked mode.

For details, refer to Locked Mode (ControlDesk Next Generation Reference).

New Instrument Features

ControlDesk Next Generation is the successor to ControlDesk and CalDesk.

The new features described below are enhancements and changes with regard to CalDesk 3.0.1 (and not to ControlDesk 3.x).

Triggered Plotter

You can specify a start and a stop trigger for the Plotter display. If the display is triggered, the time axis does not show the original time stamps of the running measurement, but starts with 0 when the start trigger condition is met. When the stop trigger condition is met, the Plotter display is frozen.

The following illustrations show how the Plotter display changes when the triggered mode is enabled. In both cases the same triggers are activated for the associated measurement raster, which means that parts of the signal are transferred to the host PC only if they meet the trigger conditions of the measurement raster.
If the Plotter display is not triggered, the Plotter shows the incoming data stream for the signals continuously, with the original time stamps on the x-axis.

If the Plotter display is triggered, only that part of the data stream that meets the trigger conditions is displayed. The time stamps on the x-axis begin with 0.

In the example, the original time stamp of the first displayed measurement point is shown on the left side above the x-axis.

For further information, refer to How to Specify a Trigger for the Plotter Display (ControlDesk Next Generation Guide)
Quick Access to Instrument Properties

In the Properties controlbar, some instruments now have a Quick Access category containing a selection of important properties.

Collapsing and Expanding the Properties Tree

The Properties controlbar provides buttons to expand or collapse categories and properties.

For more information, refer to *Properties / Instrument Properties* (ControlDesk Next Generation Reference).
New Measurement and Recording Features

ControlDesk Next Generation is the successor to ControlDesk and CalDesk.

The new features described below are enhancements and changes with regard to CalDesk 3.0.1 (and not to ControlDesk 3.x).

New Measurement Configuration controlbar

ControlDesk Next Generation has a Measurement Configuration controlbar which lets you access all the variables selected for measurement and recording, and make configuration settings for measurements and recordings.

For further information, refer to Configuring Measurement and Recording (ControlDesk Next Generation Guide).
Configuring measurement and recording via Properties controlbar

Measurement and recordings can now be configured via the Properties controlbar. The controlbar also gives you quick information on the current capture states of measurement rasters, the specified trigger conditions, etc.

For further information, refer to Measurement/Recording-Related Properties (ControlDesk Next Generation Reference).

Some configuration settings for measurement and recording can be specified via the Measurement Configuration controlbar and via the Properties controlbar, others can be set via one controlbar only.

Now that the Measurement Configuration and the Properties controlbars have been introduced, the Configure Measurement dialog is no longer available.
Triggered measurement on dSPACE real-time hardware

With ControlDesk Next Generation, you can perform triggered measurements on dSPACE real-time hardware. By this, measurement can be started and stopped by triggers. You can assign a start and a stop trigger to each measurement raster. Data flows between the dSPACE real-time hardware and the host PC only if the trigger conditions are met. Performing triggered measurement reduces measurement data throughput.

In ControlDesk Next Generation, triggered measurement is the default mode for measurements on dSPACE platforms.

For details, refer to Configuring Triggered Measurement on dSPACE Platforms (ControlDesk Next Generation Guide).

Observing variables on dSPACE platforms

The variable observer functionality is now supported and always enabled for the following platforms:
- DS1005 PPC Board
- DS1006 Processor Board
- DS1103 PPC Controller Board
- DS1104 R&D Controller Board
- MicroAutoBox
- Multiprocessor System

Variable observation is performed for parameters and measurement variables that are visualized in single-shot instruments (all instruments except for the Plotter) on a layout, but that are not on the measurement signal list.
Observing variables means reading their values cyclically from the hardware and displaying the current values in ControlDesk Next Generation, even if no measurement is currently running. For variable observation, it is sufficient to start online calibration. The variable observer allows you to add further variables to single-shot instruments while online calibration is started. The variables will be observed without having to stop and restart online calibration.

For further information, refer to Observing Variables on dSPACE Platforms (ControlDesk Next Generation Guide).

**No recorded data layout**

The recorded data layout for displaying recorded data is no longer required. ControlDesk Next Generation provides only one layout type, which can be used for visualizing a current measurement and visualizing recorded data.

When you open a CalDesk project/experiment containing recorded data layouts (RDL files), ControlDesk Next Generation changes their layout type and saves them as LAY files.

### New Data Set Management Features

ControlDesk Next Generation is the successor to ControlDesk and CalDesk.

The new features described below are enhancements and changes with regard to CalDesk 3.0.1 (and not to ControlDesk 3.x).

<table>
<thead>
<tr>
<th>Support of data sets for MP systems</th>
<th>ControlDesk Next Generation supports data sets for a multiprocessor system. The Project Manager lists data sets separately for each member of the multiprocessor system.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creating data set from open layouts</td>
<td>You can create a new sub data set that contains all the parameters that are connected to instruments on currently opened layouts. For details, refer to Create Data Set from Open Layouts (ControlDesk Next Generation Reference).</td>
</tr>
</tbody>
</table>
Creating initial data sets for SDF files

If the variable description of a dSPACE platform contains initial parameter values, you can add an initial data set via the context menu of the variable description (SDF file).

For details, refer to Create Data Set(s) (ControlDesk Next Generation Reference).

New Features of the ECU Diagnostics Module

ControlDesk Next Generation is the successor to ControlDesk and CalDesk.

The new features described below are enhancements and changes with regard to CalDesk 3.0.1 (and not to ControlDesk 3.x).

Revised device configuration wizard

The Configure Device wizard for configuring an ECU Diagnostics device in ControlDesk Next Generation was partially revised, especially the wizard pages for selecting the physical interfaces for the logical links.

For details, refer to How to Configure an ECU Diagnostics Device (ControlDesk Next Generation Guide).
Device configuration via Properties controlbar

ECU Diagnostics devices can now also be configured via the Properties controlbar.

![Properties controlbar screenshot](image)

ECU communication via PDU information

The Diagnostics instrument lets you perform communication with the ECU based on raw protocol data units (PDUs). When you select a service and configure its parameters, the Diagnostics instrument displays the resulting request PDU. Alternatively, you can edit the request PDU before the service is executed. The ECU response is interpreted by ControlDesk Next Generation, if the necessary information is available in the ODX database.

For further information, refer to Basics of ECU Diagnostics (ControlDesk Next Generation Guide).
Extended XML configuration file

When defining the ODX diagnostics database for an ECU Diagnostics device, you can add an XML configuration file to the database. This XML configuration file comprises the service configurations for the logical links.

ControlDesk Next Generation lets you use XML configuration files containing more than one configuration (FUNCTION_SET element). This allows you to specify service configurations individually for each logical link. For further information, refer to Identifying Services and Parameters for ControlDesk Functions (ControlDesk Next Generation Guide).

New configuration options

Checking the parameter value limits in the Diagnostics instrument

ControlDesk Next Generation lets you specify to check the parameter value limits in the Diagnostics instrument. With the check enabled, only values within the allowed value range can be used. With the check disabled, all the values valid for the current data type are permitted. In both cases, the computation method specified in the ODX database is applied to the parameter values.

Checking the ECU connection state

ControlDesk Next Generation lets you specify to check the ECU connection state cyclically. If enabled, a message is transmitted cyclically to the ECU to check whether ECU communication is available. If disabled, no message is transmitted and it is not possible to check whether ECU communication is available or interrupted.

For further information on these configuration options, refer to Diagnostics Settings Properties (ControlDesk Next Generation Reference).

Enhanced ODX demo project

The ODXCalDemo project which comes with ControlDesk Next Generation can simulate ECU diagnostics with a dynamic ECU. You can access the dynamic CalDemo ECU via a virtual CAN channel.

For further information, refer to Demos for ControlDesk (ControlDesk Next Generation Guide).

Configuring the ECU Diagnostics device via the automation interface

Control Desk Next Generation lets you automate the configuration of the ECU Diagnostics device via its automation interface.

You can automate the following features:
- Selecting the ODX database
- Selecting the vehicle
- Selecting logical link(s)
Selecting and configuring the physical connection used for a logical link

For further information, refer to Automating ControlDesk (ControlDesk Next Generation Guide).

Using an experiment for an MCD 3D automation project

If you automate your ECU diagnostics tasks via the ASAM-MCD 3D compatible interface of ControlDesk Next Generation, you can access a ControlDesk Next Generation experiment to read out the configuration of the ECU Diagnostics device and make it the basis for your MCD 3D automation.

For further information, refer to Automating ControlDesk’s Diagnostics Features (ControlDesk Next Generation MCD 3 Automation Guide).
New Signal Editor

ControlDesk Next Generation is the successor to ControlDesk and CalDesk.

The new features described below are enhancements and changes with regard to CalDesk 3.0.1 (and not to ControlDesk 3.x).

Graphical interface for signal generation

The Signal Editor is a software module for the graphical definition and execution of signal generators for stimulating model variables of real-time applications.

For details, refer to Using the Signal Editor (ControlDesk Next Generation Guide).
### Creating arbitrary signal shapes

The Signal Editor lets you create and arrange arbitrary signal shapes. It provides controlbars and commands to configure, display, manage, and store signals in *signal description sets*. You can graphically edit and configure the signals in the working area.

You can use the editor’s *Signal Selector*, a kind of library with predefined segments, and imported numerical data to specify the signal shapes.

### Performing stimulus generation

The Signal Editor supports stimulus generation for all dSPACE platforms that can run Real-Time Testing (RTT) sequences.

You can use your configured signal description sets as *signal generators* to stimulate model variables of real-time applications running on dSPACE real-time hardware. You can map model variables of a real-time application to the signals of a signal generator via drag & drop and perform the stimulus control on connected real-time hardware via commands provided by the Signal Editor.

---

### New Failure Simulation Module

ControlDesk Next Generation is the successor to ControlDesk and CalDesk.

The new features described below are enhancements and changes *with regard to CalDesk 3.0.1* (and not to ControlDesk 3.x).
Graphical interface for failure simulation control

The Failure Simulation module lets you handle and set failure types in a graphical environment to control failure simulation on a connected dSPACE real-time hardware. You can simulate failures in the wiring of an electronic control unit (ECU), for example, an ECU pin short-circuited to ground or the battery voltage, or an ECU pin that is not connected (cable break).

For details, refer to Failure Simulation (ControlDesk Next Generation Guide).
Managing and controlling pin failures

The Failure Simulation module provides controlbars and context menu commands to control the failure insertion units of dSPACE hardware.

Failure patterns are used to manage pin failures. A pin failure describes a failure which can be simulated on a specific ECU pin. You can open and display failure patterns in Failure Pattern windows in the working area, see example below.

Failure Pattern windows visualize failure patterns and the statuses of the failure simulation hardware, and allow the failure patterns to be edited and saved. A failure pattern is described by a table containing all the pins that have to be failure-simulated at the same time. You can add pins to the table via drag & drop from the Failure Simulation controlbar.

You can activate and deactivate a failure pattern in the Failure Pattern window or in the Failure Simulation Set window. The Failure Simulation Set window gives you a clearly arranged tree view of all the failure patterns of a failure simulation system.
New Automation Features

ControlDesk Next Generation is the successor to ControlDesk and CalDesk.

The new features described below are enhancements and changes with regard to CalDesk 3.0.1 (and not to ControlDesk 3.x).

Using events

ControlDesk Next Generation lets you link the execution of Python code to the occurrence of a specific event, such as the activating of an experiment or the value change of a variable in an instrument.

For more information, refer to Using ControlDesk Events (ControlDesk Next Generation Guide).

Enhanced object model

The object model of ControlDesk Next Generation’s automation interface is enhanced compared with CalDesk 3.0.1. The object model now supports the automation of view sets, failure simulation, and multiprocessor platforms, for example. For details, refer to the ControlDesk Next Generation Measurement Data API Reference.

Automating ECU diagnostics tasks

For new automation features in connection with ECU diagnostics, refer to New Features of the ECU Diagnostics Module on page 77.
Further Enhancements with ControlDesk Next Generation

ControlDesk Next Generation is the successor to ControlDesk and CalDesk.

The new features described below are enhancements and changes with regard to CalDesk 3.0.1 (and not to ControlDesk 3.x).

### View sets

ControlDesk Next Generation lets you save the current configuration of the controlbars, toolbars and main menu of the application as a view set. You can create various view sets and switch between them.

View sets can easily be switched via a specific View Sets toolbar:

For details, refer to How to Customize the Screen Arrangement (ControlDesk Next Generation Guide).

### Automatic start and stop of CalDemo ECU

To work with the CalDemo and ODXCalDemo demo projects, the CalDemo ECU (CalDemo.exe) must be running. The CalDemo ECU is started/stopped automatically when you open/close the CalDemo or ODXCalDemo demo project of ControlDesk Next Generation.

For further information on the demo projects, refer to Demos for ControlDesk (ControlDesk Next Generation Guide).
ControlDesk Next Generation provides various demo projects, such as the CalDemo project, which demonstrates ControlDesk’s measurement and calibration features. When you start ControlDesk Next Generation the first time, all the demo projects are copied automatically to the user-specific documents folder.

The documents folder is a folder that serves as a repository for user-specific documents. The location of the Documents folder depends on the operating system. For Windows XP, for example, it is \Documents and Settings\<User>\My Documents\dSPACE\ControlDesk NG\<VersionNumber>.

For more information on the demo projects, refer to Demos for ControlDesk (ControlDesk Next Generation Guide).
Migrating to ControlDesk Next Generation (ControlDesk 4.0)

ControlDesk Next Generation (ControlDesk 4.0) is the successor to ControlDesk and CalDesk. You get information on migrating from ControlDesk 3.x and from CalDesk.

Migrating to ControlDesk Next Generation

<table>
<thead>
<tr>
<th>Migrating from</th>
<th>To migrate from ControlDesk 3.x to ControlDesk Next Generation and reuse existing ControlDesk experiments in ControlDesk Next Generation, you have to carry out additional migration steps. For details, refer to Migrating from ControlDesk 3.x to ControlDesk Next Generation (ControlDesk Next Generation Migration Guide).</th>
</tr>
</thead>
<tbody>
<tr>
<td>ControlDesk 3.x</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Migrating from</th>
<th>To migrate from CalDesk to ControlDesk Next Generation and reuse existing experiments last saved with CalDesk, you may have to carry out additional migration steps. For information on the migration steps, refer to Migrating from CalDesk to ControlDesk Next Generation (ControlDesk Next Generation Migration Guide).</th>
</tr>
</thead>
<tbody>
<tr>
<td>CalDesk</td>
<td></td>
</tr>
</tbody>
</table>
New Features of the Variable Editor

General Enhancements

Support of ASAP2 1.6

The Variable Editor supports variable description files based on the ASAM MCD-2MC (ASAP2) V 1.6 standard.
dSPACE ECU Flash Programming Tool

New Features of the dSPACE ECU Flash Programming Tool 2.2

Up to dSPACE Release 6.6, the dSPACE ECU Flash Programming Tool was part of the CalDesk product setup. As of dSPACE Release 7.0, there is a separate setup for the dSPACE ECU Flash Programming Tool. You get information on the features of the dSPACE ECU Flash Programming Tool compared with dSPACE ECU Flash Programming Tool 2.1.2 included in CalDesk 3.0.1.

| ECU flash programming via XCP on Ethernet | The dSPACE ECU Flash Programming Tool now also supports ECU flash programming via XCP on Ethernet. Refer to Supported ECU Interface Types (ECU Flash Programming). |

| Configuring the flash kernel | The dSPACE ECU Flash Programming Tool comes with the dSPACE Flash Kernel Configuration Tool, which provides information on a flash kernel and lets you modify the flash kernel configuration settings. The configurable settings vary according to the ECU interface type the flash kernel is specified for.  
For further information, refer to Configuring the Flash Kernel (ECU Flash Programming). |
dSPACE FlexRay Configuration Package

New Features of dSPACE FlexRay Configuration Package 2.4

<table>
<thead>
<tr>
<th>FlexRay Configuration Package</th>
<th>Information in this section</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Features of dSPACE FlexRay Configuration Package 2.4</td>
<td>93</td>
</tr>
<tr>
<td>Migration to dSPACE FlexRay Configuration Package 2.4</td>
<td>95</td>
</tr>
</tbody>
</table>

The following features are new for the FlexRay Configuration Tool and the FlexRay Configuration Blockset.

**Multiple FlexRay buses** You can now implement several FlexRay buses in one real-time model, for example, to model a gateway. You can use up to four FlexRay configurations in one real-time model depending on the platform type.

You must create a FlexRay configuration for each FlexRay bus which you want to implement in your real-time model, using the FlexRay Configuration Tool. The FlexRay configurations are the basis for the automatically generated FlexRay models. For details, refer to *How to Create Configurations for Multiple Buses* (FlexRay Configuration Tool Guide)
After generating the FlexRay models for each configuration, you can implement the real-time model. For details, refer to *Modeling Several FlexRay Buses on One dSPACE Real-Time System* ([FlexRay Configuration RTI Reference]).

**Bus Navigator support** The FlexRay Configuration Package generates configuration files for the Bus Navigator in ControlDesk Next Generation, which can now be used for generating layouts for RX and TX PDUs/frames.

**FlexRay Configuration Tool**

- **Updating the FIBEX file** If the FIBEX file which you created your configuration with has changed, you can update the configuration using the new FIBEX file. For details, refer to *How to Update the FIBEX File* ([FlexRay Configuration Tool Guide]).

- **Restructured graphical user interface** The graphical user interface was restructured. All settings can now be made in the *General Properties* dialog. The dialog consists of four pages to specify general settings and the settings for hardware configuration, CRC configuration, and code generation. For details, refer to *General Page* ([FlexRay Configuration Tool Reference]).

  Code generation is now started by a tool button.

**RTI FlexRay Configuration Blockset**

- **Supporting new versions of MicroAutoBox** The blockset now supports the following versions of MicroAutoBox:
  - MicroAutoBox in the following variants:
    - 1401/1505/1506
    - 1401/1505/1507
    - 1401/1507
  - MicroAutoBox II in the following variants:
    - 1401/1505/1507
    - 1401/1507
    - 1401/1511/1512

  To use MicroAutoBox II, choose DS1401 as processor board type and DS1512 as I/O board type on the *Hardware page* of the *Project Settings* dialog.

- **Hardware access of MicroAutoBox** The hardware access functions were improved for all versions of MicroAutoBox.
## Migration to dSPACE FlexRay Configuration Package 2.4

<table>
<thead>
<tr>
<th>Migrating multiprocessor models</th>
<th>After migrating FlexRay multiprocessor models based on several FlexRay configurations, you must assign the following blocks to a RTIFLEXRAYCONFIG UPDATE block manually:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>- RTIFLEXRAYCONFIG STATUS</td>
</tr>
<tr>
<td></td>
<td>- RTIFLEXRAYCONFIG INTERRUPT</td>
</tr>
<tr>
<td></td>
<td>- RTIFLEXRAYCONFIG CONTROLLER STOP</td>
</tr>
<tr>
<td></td>
<td>- RTIFLEXRAYCONFIG CONTROLLER RESTART</td>
</tr>
<tr>
<td></td>
<td>- RTIFLEXRAYCONFIG TX WAKEUP</td>
</tr>
<tr>
<td></td>
<td>- RTIFLEXRAYCONFIG ERROR HOOK INTERRUPT</td>
</tr>
<tr>
<td></td>
<td>- RTIFLEXRAYCONFIG COM EVENT CONTROL</td>
</tr>
<tr>
<td></td>
<td>- RTIFLEXRAYCONFIG COM CYCLIC CONTROL</td>
</tr>
<tr>
<td></td>
<td>- RTIFLEXRAYCONFIG TIMETABLE CONTROL</td>
</tr>
</tbody>
</table>

| Migrating MicroAutoBox applications | The hardware access functions were improved for all versions of MicroAutoBox. To use the improved function, you must rebuild existing real-time applications. |

| Using the multiple-bus option in existing projects | If you use the multiple-bus option, the TRC file is extended by ConfigID groups containing the configuration IDs of the buses. If it is changed for existing projects, the TRC paths of the variables are changed. Data connections to ControlDesk instruments gets lost and must be reconnected. |

### Related topics

- RTIFLEXRAYCONFIG UPDATE ([FlexRay Configuration RTI Reference](#))
ModelDesk

New Features of ModelDesk 2.3

### Plotting signals

ModelDesk can plot signals of the ASM model during real-time simulation (single processor system) and Simulink simulation. The signals can be recorded and saved to MAT files, so it is possible to compare signals of different simulation runs or other customer reference data. The management of simulation results is done via ModelDesk’s Project Manager.

Using ModelDesk’s Project Manager you can manage the settings for plotting and simulation results. Both are available for all the experiments of a ModelDesk project, so you can use the same settings in different experiments or compare simulation results of different experiments.

For details on plotting signals, refer to [Plotting Simulation Results](#) (ModelDesk Guide).

### Generating model initialization files

ModelDesk can generate initialization files for ASM models. Up to now, ModelDesk could generate the files for the whole ASM model only. Now you can also generate the files for parts of the ASM model on different levels (parameter set, main component, navigation page, or parameter page).
RTI/RTI-MP and RTLib

Where to go from here

Information in this section

| New Features of RTI/RTI-MP and RTLib | 99 |
| Migration Aspects of RTI/RTI-MP and RTLib | 102 |

New Features of RTI/RTI-MP and RTLib

Microtec PowerPC Compiler 3.7

With the current dSPACE release, the Microtec PowerPC compiler version has changed from 3.5 to 3.7. The updated compiler has the following main advantages:

- More optimizations that reduce the size of generated code
- Significantly improved compile time for large applications
- Improved compiler memory usage, so that large applications that previously could not be compiled with optimizations enabled can now complete.

The compiler versions are code-compatible, so you do not have to recompile the existing object files and libraries.

MicroAutoBox II

The current dSPACE Release provides software support for the new I/O features of MicroAutoBox II 1401/1511 and MicroAutoBox II 1401/1511/1512.

The new I/O features are:

- A/D conversion
  - 16 parallel A/D converters supporting burst conversion mode
- D/A conversion
  4 parallel D/A converters
- Digital I/O Unit
  40 digital input channels and 40 digital output channels that can be used for:
  - Bit I/O
  - PWM signal generation (PWM)
  - Square-wave signal generation (FREQ)
  - PWM signal measurement (PWM2D)
  - Frequency measurement (F2D)
  - Incremental encoder interface
    (only RTLib support)

The CAN modules of the new MicroAutoBox 1401/1511 and the FlexRay modules of the new MicroAutoBox 1401/1511/1512 comes with higher performance.

Because of the new I/O features and the larger number of MicroAutoBox variants, the RTI1401 blockset has been restructured. There are now hardware-specific blocksets providing only the RTI blocks which are supported by your MicroAutoBox variant.

MicroAutoBox II comes with two new RTI blocksets:
- RTI Ethernet (UDP) Blockset
  Provides access to the Ethernet I/O interface (MicroAutoBox II only) and the ECU interface (MicroAutoBox and MicroAutoBox II, LVDS Ethernet link cable required), and allows you to implement communication via the UDP/IP protocol.
- USB Flight Recorder
  Lets you perform long-term data acquisitions and save the data to an USB mass storage device connected to MicroAutoBox II. It also works with a higher performance than the flash-memory based Flight Recorder.

For further information, refer to MicroAutoBox Features.
### Gigalink Blockset and RTLib functions

**Gigalink Blockset**

Up to now, it was only possible to connect Gigalink Send and Receive blocks to ports which are not in the multiprocessor topology of an RTI-MP model. With the new Gigalink Blockset, this limitation has been removed. In an RTI-MP model, you can use the Gigalink Send and Receive blocks to configure a Gigalink in the MP topology. This allows you, for example, to implement the communication of several applications which run on a multicore DS1006 board via virtual Gigalinks.

**Gigalink access functions**

The `dsgl_ptr_get` function was replaced by `dsgl_write_ptr_get` and `dsgl_read_ptr_get`. These functions perform the different buffer handling of physical and virtual Gigalinks used with the DS1005 and the multicore DS1006 board.

### Number of host services

The Data Capture block provides 31 host services to capture data. By default, host services #28 to #31 are now reserved for monitoring features when using CAN or LIN bus support. If you do not use bus support, you can enable the reserved services by setting the Allow usage of reserved host service numbers with Data Capture blocks option in the RTI general build options page of the Configuration Parameters dialog.

Models which use host services #28 … #31 must be migrated.

### TRC file enhancements

The properties of Simulink.Signal and Simulink.Parameter objects are now used for TRC file generation.

- You can access the following properties via your experimentation software:
  - Description
  - DocUnits
  - Min and Max (represented by a range in the TRC file)

Signals and parameters of boolean data type in Simulink are now treated as boolean value by your experimentation software.

For further information, refer to Available Variables in the Variable Description File (RTI and RTI-MP Implementation Reference).

### New features coming with MATLAB R2010b

The following new features are supported. Note the limitations.

- Signals consisting of arrays of nonvirtual buses can be handled. A Simulink bus itself can also contain bus arrays. RTI supports this feature, but a bus array is not generated to the TRC file.
The new Variant Subsystem block contains several child subsystems but only one subsystem is executed during simulation. RTI supports this feature, but there must be no RTI blocks in any subsystem, whether activated or deactivated.

The Constant block now supports a bus object as data type. This allows you to use a structured parameter as Constant value. RTI supports this feature, but a structured parameter in a Constant block is not generated to the TRC file.

The Largest Atomic Size option in the Hardware Implementation page of the Configuration Parameters dialog is supported. If the Ensure deterministic data transfer option of the Rate Transition block is cleared, and the signal size is less than or equal to the specified largest atomic size, the code generator removes double-buffering and semaphore protection from the generated code. This optimization reduces RAM and ROM consumption and improves execution speed.

For further information, refer to Hardware Implementation Dialog (Model Configuration Parameters Dialogs) (RTI and RTI-MP Implementation Reference).

The following new features are not supported:

- The File generation control option in the Simulink Preferences dialog is not supported.

There are conflicts with some RTI blocksets, if the specified Code generation folder differs from the working folder used by these blocksets. Therefore, the RTI build process stops with an error message, if a Code generation folder was specified.

- The MATLAB Distributed Computing Server (MDCS) software is not supported.

When you work with referenced models, the MDCS software can be used for parallel builds distributed across remote workers.

### Migration Aspects of RTI/RTI-MP and RTLib

**Migration aspects of MicroAutoBox II**

Applications compiled with MicroAutoBox do not have to be recompiled for MicroAutoBox II if the same I/O boards are used.

MicroAutoBox II 1401/1511 and MicroAutoBox II 1401/1511/1512 require boot firmware version 3.0.
RTI AUTOSAR Package

New Features of the RTI AUTOSAR Package 1.2

**Supported AUTOSAR Releases**
The RTI AUTOSAR Package supports:
- AUTOSAR Release 3.1 with Version 3.1.0 (continued)
- AUTOSAR Release 3.0 with Version 3.0.2 (continued)
- AUTOSAR Release 2.1 with Versions 2.1.2 and 2.1.4 (continued)

**Improved support of client-server communication**
In client-server communication, a server provides operations which can be called by a client.
The new version of the RTI AUTOSAR Package offers improved support for client-server communication. You can additionally integrate software components in Simulink that contain operations with one or more `in`, `out`, or `inout` parameters.

**Support of mode management**
Mode management involves switching an ECU, or a functional unit within an ECU, between its possible operating states.
You can now integrate mode-using software components, called mode users, in Simulink. You can simulate the mode users’ reaction on mode changes and test the mode user in all possible operating states by connecting mode switch events and mode switch interfaces to them. You have to model the mode management in Simulink and create function calls for the mode user’s mode switch events. The RTI AUTOSAR Package also supports mode users with mode disabling dependencies.
Support of calibration

You can interactively control calibration parameters (calprms) using dSPACE’s ControlDesk while the ECU application is running.

You can now integrate software components in Simulink with the following calibration parameters:

- Shared calprms, i.e., calprms that are defined in the internal behavior of a software component and shared by all instances of the software component.

- Calprm values, i.e., calibration parameters that are communicated via ports with interfaces as defined by AUTOSAR.

In the Simulink model, you can access the calibration parameters of an integrated software component via a MATLAB workspace variable. This lets you control calibration parameters at simulation start and define different simulation scenarios.

The calibration parameters are added to a hierarchically structured application TRC file and can be accessed by ControlDesk. The illustration below shows an example TRC file with shared calprms.

Support of measurements

The new version of the RTI AUTOSAR Package generates a TRC file that comprises data elements, operation arguments, and interrunnable variables of imported software components. Using dSPACE’s ControlDesk, you can interactively measure variables while the ECU application is running.

The RTI AUTOSAR Package supports scalar calibration parameters and calibration parameters that consist of one vector of values (1-dimensional array). Multidimensional calibration parameters, such as curves and maps are not supported.

The RTI AUTOSAR Package supports scalar calibration parameters and calibration parameters that consist of one vector of values (1-dimensional array). Multidimensional calibration parameters, such as curves and maps are not supported.
<table>
<thead>
<tr>
<th>Related topics</th>
<th>Basics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Features of the RTI AUTOSAR Package ([link](RTI AUTOSAR Package Document))</td>
</tr>
</tbody>
</table>
RTI Bypass Blockset

New Features of the RTI Bypass Blockset 2.7.2

<table>
<thead>
<tr>
<th>RTI Bypass Blockset</th>
<th>Discontinuation of Variable Editor support</th>
<th>Working with models from earlier RTI Bypass Blockset versions 2.x</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The RTI Bypass Blockset no longer supports the Variable Editor. You cannot access the Variable Editor via the RTI Bypass Blockset.</td>
<td>The current release comes with RTI Bypass Blockset 2.7.2, which is compatible with earlier blockset versions 2.x. However, data management was changed compared to the RTI Bypass Blockset version 2.5 or earlier.</td>
</tr>
</tbody>
</table>

If you have a Simulink model built with blockset version 2.5 or earlier and open it with RTI Bypass Blockset 2.7.2, the old data dictionary file (file name extension .dd) is replaced by a new Data dictionary file (.vdb) 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 or 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 2.7.2 and want to use it with RTI Bypass Blockset 2.5 or earlier, the model’s Data dictionary file required for the earlier blockset version (file name extension .dd) is recreated as soon as you update the A2L files in the Setup block or open the Read, Write, Upload or Download block and click the Fill Variable Selector button on the Variables page. The Data dictionary file created under RTI Bypass Blockset 2.7.2 (*.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.
# New Features of the RTI CAN MultiMessage Blockset 2.5.2

<table>
<thead>
<tr>
<th>New supported platform</th>
<th>The RTI CAN MultiMessage Blockset supports MicroAutoBox II.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Support of Bus Navigator in ControlDesk Next Generation</td>
<td>In addition to the Bus Navigator in ControlDesk, the RTI CAN MultiMessage Blockset also supports the Bus Navigator in ControlDesk Next Generation. You can handle CAN messages configured with the RTI CAN MultiMessage Blockset with the Bus Navigator in ControlDesk and with the Bus Navigator in ControlDesk Next Generation. For further information, refer to <em>New Bus Navigator for CAN, LIN, and FlexRay Bus Communication</em> on page 63.</td>
</tr>
<tr>
<td>Bus Navigator support always enabled</td>
<td>Bus Navigator support now is always enabled in the RTI CAN MultiMessage Blockset, so layout generation using the Bus Navigator is always possible. The checkbox to enable or disable Bus Navigator support was removed from the Experimental Software page, so you can no longer change this setting.</td>
</tr>
</tbody>
</table>
Bus monitoring always enabled

Up to RTI CAN MultiMessage Blockset 2.5.1, it was necessary to activate CAN bus monitoring with the Bus Navigator and specify the host service in the RTICANMM GeneralSetup block. As of RTI CAN MultiMessage Blockset 2.5.2, bus monitoring with the Bus Navigator is always enabled and always uses host service 31. You cannot change these settings in the blockset. The host service used for CAN bus monitoring is displayed for information purposes. Refer to Main Page (RTICANMM GeneralSetup) (RTI CAN MultiMessage Reference).

Changes to the RTICANMM ControllerSetup block

- The sample mode “3 sample per bit” is no longer available. The “1 sample per bit” mode is used instead.
- You can specify the baud rate settings for a CAN controller either via the Advanced Configuration page of the RTICANMM ControllerSetup block, or via an inport to the block. The RTI CAN MultiMessage Blockset does not allow to have the Advanced Configuration page active and add a baudrate inport at the same time.

For further information, refer to RTICANMM ControllerSetup (RTI CAN MultiMessage Reference).

Migrating to RTI CAN MultiMessage Blockset 2.5.2

Working with models from earlier RTI CAN MultiMessage Blockset versions

To reuse a model created with an earlier RTI CAN MultiMessage Blockset version, you must update the S-functions for all the contained RTICANMM blocks before you perform modifications to the CAN configuration.

To create new S-functions for all the RTICANMM blocks in your model in one step, you can select the Create S-Function for all CAN Blocks command from the Options menu of the RTICANMM GeneralSetup block.

For further information, refer to Limitations with RTICANMM (RTI CAN MultiMessage Reference).
RTI LIN MultiMessage Blockset

Information in this section

<table>
<thead>
<tr>
<th>New Features of the RTI LIN MultiMessage Blockset 1.8</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>Migrating to RTI LIN MultiMessage Blockset 1.8</td>
<td>112</td>
</tr>
</tbody>
</table>

New Features of the RTI LIN MultiMessage Blockset 1.8

New supported platform

The RTI LIN MultiMessage Blockset supports MicroAutoBox II.

Support of Bus Navigator in ControlDesk Next Generation

In addition to the Bus Navigator in ControlDesk, the RTI LIN MultiMessage Blockset now also supports the Bus Navigator in ControlDesk Next Generation. You can handle LIN frames configured with the RTI LIN MultiMessage Blockset with the Bus Navigator in ControlDesk and with the Bus Navigator in ControlDesk Next Generation.

For further information, refer to New Bus Navigator for CAN, LIN, and FlexRay Bus Communication on page 63.
Bus Navigator support always enabled

Bus Navigator support now is always enabled in the RTI LIN MultiMessage Blockset, so layout generation using the Bus Navigator is always possible. The checkbox to enable or disable Bus Navigator support was removed from the Experimental Software page, so you can no longer change this setting.

The removal of the checkbox makes the Experimental Software page obsolete, so it was also removed from the RTILINMM MainSetup block.

LIN bus monitoring with the Bus Navigator

The RTI LIN MultiMessage Blockset supports LIN bus monitoring with the Bus Navigator. You can monitor raw and physical data of LIN frames on a LIN bus, and log the raw data of LIN frames.

For further information, refer to Working with the Bus Navigator (ControlDesk Experiment Guide) or Working with the Bus Navigator (ControlDesk Next Generation Guide).

LIN bus monitoring is always activated and always uses host service 30. You cannot change these settings in the blockset. The host service used for LIN bus monitoring is displayed for information purposes. Refer to Main Page (RTILINMM GeneralSetup) (RTI LIN MultiMessage Reference).

Migrating to RTI LIN MultiMessage Blockset 1.8

Working with models from earlier RTI LIN MultiMessage Blockset versions

To reuse a model created with an earlier RTI LIN MultiMessage Blockset version, you must update the S-functions for all the contained RTILINMM blocks before you perform modifications to the LIN configuration.

To create new S-functions for all the RTILINMM blocks in your model in one step, you can select the Create S-Function for all LIN Blocks command from the Options menu of the RTILINMM GeneralSetup block.

For further information, refer to Limitations of RTI LIN MultiMessage Blockset (RTI LIN MultiMessage Reference).
RTI Ethernet (UDP) Blockset

New Features of the RTI Ethernet (UDP) Blockset 1.0

| Main features | The RTI Ethernet (UDP) Blockset is a Simulink® blockset for modeling communication via an Ethernet interface using the UDP/IP protocol. It gives you access to all external devices that also provide an Ethernet interface, such as another dSPACE board or a calibration device. The blockset provides RTI blocks for configuring the Ethernet interface by specifying its IP address and port number, and for sending and receiving data via the UDP/IP protocol. |
| Hardware support | The blockset supports the following interfaces, which are represented by their board types: |
| ETH Type 1 | The ETH Type 1 interface can be used with MicroAutoBox II. The communication is processed on the I/O Ethernet interface (ETH Type 1 module) of the board. |
| ECU Type 1 ETH | The ECU Type 1 interface can be used with MicroAutoBox and MicroAutoBox II. For connecting the ECU Interface to the Ethernet, an LVDS Ethernet link cable is required. With the link cable connected, the ECU Type 1 interface can be used as ECU Type 1 ETH interface. The communication is processed on the ECU interface (ECU Type 1 module). The MicroAutoBox variants provide a different number of Ethernet interfaces. |
### Supported UDP features and limitations

The blockset supports the following UDP features. Note that there are some limitations.

- The blockset supports transferring Ethernet packets via the UDP/IP protocol.
- You can define up to 4 sockets per module.
- Each socket can be used for bidirectional communication.
- Each socket can be configured with a maximum datagram size of 1472 bytes.
- Auto negotiation is supported.
- Data rates of 100 MBit/s are supported.
- Broadcasting is supported.
- Listening to any IP address and port is supported.
- IP fragmentation is not supported. Each UDP message is limited to the maximum Ethernet datagram size.
- DHCP is not supported.
- Routing is not supported. All participants in communication must be available in the same subnet.
RTI FPGA Programming Blockset

Limited availability outside Europe and Asia, please inquire.

Where to go from here

<table>
<thead>
<tr>
<th>Information in this section</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>New Features of the RTI FPGA Programming Blockset 2.0</td>
<td>115</td>
</tr>
<tr>
<td>Migrating to RTI FPGA Programming Blockset 2.0</td>
<td>116</td>
</tr>
</tbody>
</table>

New Features of the RTI FPGA Programming Blockset 2.0

Extended Xilinx® support

The RTI FPGA Programming Blockset now supports the Xilinx design tools in the versions 10.1.03 and 11.5.

This increases the number of supported MATLAB versions and operating systems for the blockset’s FPGA Interface.

<table>
<thead>
<tr>
<th>Xilinx Design Tools Version</th>
<th>Operating System</th>
<th>MATLAB Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.03</td>
<td>Windows XP Professional (32-bit version)</td>
<td>MATLAB R2007b+</td>
</tr>
<tr>
<td></td>
<td>Windows XP Professional (32-bit version)</td>
<td>MATLAB R2008a+</td>
</tr>
<tr>
<td></td>
<td>Windows Vista Business, Ultimate, and Enterprise (32-bit version)</td>
<td>MATLAB R2008b</td>
</tr>
<tr>
<td>11.5</td>
<td>Windows XP Professional (32-bit version)</td>
<td>MATLAB R2009a</td>
</tr>
<tr>
<td></td>
<td>Windows Vista Business, Ultimate, and Enterprise (32-bit version)</td>
<td>MATLAB R2009b SP1</td>
</tr>
</tbody>
</table>

The DS5203 FPGA Board supports both Xilinx design tools versions.
Preparation for MicroAutoBox II support

The RTI FPGA Programming Blockset has been prepared for use with MicroAutoBox 1401/1511/1512 as soon as the necessary FPGA hardware is available.

Related topics

Basics

- Migrating to RTI FPGA Programming Blockset 2.0 on page 116

Migrating to RTI FPGA Programming Blockset 2.0

Objective

There are different methods to migrate an existing model, depending on the blockset version used.

Available methods

- Migrating from RTI FPGA Programming Blockset 1.0 to 2.0
- Migrating from RTI FPGA Programming Blockset 1.1 to 2.0
- Migrating from RTI FPGA Programming Blockset 1.2 to 2.0

Migrating from RTI FPGA Programming Blockset 1.0 to 2.0

Because RTI FPGA Programming Blockset 1.0 (released with dSPACE Release 6.4) was not fully implemented, a model that you implemented with it must be migrated manually. You must replace each block of the RTI FPGA Programming Blockset with a new one to make the model compatible with the current dSPACE RTI environment for modeling, building and executing.

The update function of the script interface does not support RTI FPGA Programming Blockset 1.0.

Migrating from RTI FPGA Programming Blockset 1.1 to 2.0

If you have implemented your FPGA application using RTI FPGA Programming Blockset 1.1 (released with dSPACE Release 6.5), and you want to use it with RTI FPGA Programming Blockset 2.0 (released with dSPACE Release 7.0), you must update the FPGA framework. This involves only a few internal modifications that do not affect the blocks’ inputs and outputs or their parameters. You can use the script interface to update the FPGA framework.

To update the FPGA framework without changing the values of the block parameters

rtifpga_scriptinterface('FPGAFrameworkUpdate', <SimulinkHandle>)
The script handles all the subsystems in the model/subsystem which is specified by the Simulink handle. The parameters of the blocks are unchanged after updating to the current framework version.

Example: The following script updates the FPGA framework for any FPGA subsystems in the processor model called MyProcModel. The specified values of the block parameters are not changed.

```matlab
ProcModelHandle = get_param('MyProcModel','handle')
rtifpga_scriptinterface('FPGAFrameworkUpdate', ProcModelHandle)
```

To update the FPGA framework and reset the values of the block parameters to their initial values

```matlab
rtifpga_scriptinterface('FPGAFrameworkUpdate',<SimulinkHandle>, 'ReInit')
```

The script handles all the subsystems in the model/subsystem which is specified by the Simulink handle. The parameters of the blocks are reset to their initial values after updating to the current framework version.

Migrating from RTI FPGA Programming Blockset 1.2 to 2.0

No migration is necessary, except if you have updated from Xilinx design tools version 10.1.03 to 11.5. Then you must also update the framework, refer to Migrating from RTI FPGA Programming Blockset 1.1 to 2.0.
SystemDesk

Where to go from here

<table>
<thead>
<tr>
<th>Information in this section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Features of SystemDesk 3.0</td>
<td>120</td>
</tr>
<tr>
<td>Migrating to SystemDesk 3.0</td>
<td>139</td>
</tr>
</tbody>
</table>
New Features of SystemDesk 3.0

Where to go from here

<table>
<thead>
<tr>
<th>Information in this section</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>New General Features</td>
<td>120</td>
</tr>
<tr>
<td>Changed Modeling of ECU Software</td>
<td>122</td>
</tr>
<tr>
<td>Modeling Systems</td>
<td>125</td>
</tr>
<tr>
<td>Configuring ECUs</td>
<td>127</td>
</tr>
<tr>
<td>Simulating Systems</td>
<td>129</td>
</tr>
<tr>
<td>Importing and Exporting AUTOSAR Files</td>
<td>132</td>
</tr>
<tr>
<td>Exchanging Data with TargetLink</td>
<td>134</td>
</tr>
<tr>
<td>dSPACE Target for Offline Simulation 1.2</td>
<td>137</td>
</tr>
<tr>
<td>Features Discontinued in SystemDesk 3.0</td>
<td>138</td>
</tr>
</tbody>
</table>

New General Features

SystemDesk 3.0 has several new general features.

AUTOSAR Releases supported by SystemDesk 3.0

<table>
<thead>
<tr>
<th>Release</th>
<th>Version</th>
<th>Import</th>
<th>Export</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>3.1.4</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>3.1.2</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>3.1.0</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>3.0</td>
<td>3.0.6</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>3.0.4</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>3.0.2</td>
<td>X</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>3.0.0</td>
<td>X</td>
<td>–</td>
</tr>
<tr>
<td>2.1</td>
<td>2.1.4</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>2.1.2</td>
<td>X</td>
<td>–</td>
</tr>
</tbody>
</table>

1) New in SystemDesk 3.0
System-oriented modeling

This version of SystemDesk involves a major change in modeling practice.

Now your SystemDesk project is clearly separated into a library level and a system level to give users a new, well-structured workflow:

- The library level contains reusable AUTOSAR elements.
- The system level contains the software architecture, the hardware topology, and the network communication.

Software components and compositions modeled in the library can be directly integrated into a system and edited within it. This helps users keep track of everything, even in complex projects.

The illustration below shows an example.

![Diagram of SystemDesk project structure](image-url)
Changed Modeling of ECU Software

Modeling software architectures has been changed with regard to interaction with SystemDesk’s project library. Whenever you change elements in the project library, you now can update systems with the changed elements from the library. This helps you keep library and element instances consistent.

The illustration below shows the project library and instantiated elements in a system’s software architecture.
**Importing and exporting elements to and from the library**

SystemDesk now imports and exports software architecture elements to and from the project library. During export SystemDesk creates types or parts in the project library for each element to be exported if necessary. This lets you exchange AUTOSAR files smoothly with other AUTOSAR tools such as dSPACE’s behavior modeling tool TargetLink.

For details on working with SystemDesk's project library, refer to Basics on Working with the Project Library (SystemDesk Guide).

---

**Changed modeling of hardware topologies**

With SystemDesk 3.0, modeling and structuring a system’s hardware topology is now consistent with AUTOSAR. SystemDesk supports elements such as ECUs, channels and controllers for specifying a system’s hardware topology. SystemDesk supports CAN, LIN, and FIBEX. The illustration below shows an example of a CAN bus topology.

The above illustration shows a hardware topology consisting of the Controller and Plant ECUs that are connected by the CanChannel channel. Both ECUs have one controller for accessing the channel.
For detailed information on hardware topologies, refer to Basics on Specifying a Hardware Topology (SystemDesk Guide).

**Changed modeling of network communications**

With SystemDesk 3.0, modeling and structuring a system's network communication is now consistent with AUTOSAR. SystemDesk supports elements such as system signals, iSignals, signal IPDUs, and frames for specifying a system's network communication. The illustration below shows a network communication example with three system signals.

For detailed information on network communications, refer to Basics on Specifying Network Communication (SystemDesk Guide).
## Modeling Systems

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiple instantiation of software components</td>
<td>SystemDesk supports multiple instantiation of software components.</td>
</tr>
<tr>
<td>Mapping software components to ECUs</td>
<td>This version of SystemDesk helps you map software components (SWCs) to ECUs with the new SWC to ECU Mapping Editor.</td>
</tr>
</tbody>
</table>

![SystemDesk SWC to ECU Mapping Editor](image)

The SWC to ECU Mapping Editor lets you map and unmap compositions and atomic SWCs to/from ECUs without navigating in SystemDesk’s Project Manager. All the compositions and SWCs are clearly arranged in a tree view in the editor. You map/unmap compositions and SWCs via drag & drop to/from ECUs listed in the editor. You can press Ctrl to select multiple SWCs to be mapped at once. You can also define filter conditions to filter the view.

For more details, refer to [SWC to ECU Mapping Editor](SystemDesk Reference).

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapping complex data types to system signals</td>
<td>The new Complex Signal Mapping Editor is part of SystemDesk’s Signal Mapping Editor. It lets you map the single primitive data types of a complex data type to the single system signals of a signal group.</td>
</tr>
</tbody>
</table>

When you select a signal group, SystemDesk analyzes the related number of signals and their bitlength. SystemDesk performs the mapping automatically and displays possible conflicts. The editor lets you change the mapping manually.

For more details, refer to [Complex Signal Mapping Editor](SystemDesk Reference).
Improved filter mechanism in editors and control bars

The new Filter Editor lets you specify a filter rule to filter the grid of a SystemDesk editor/control bar for user-defined expressions.

![Filter Editor](image-url)
## Configuring ECUs

### Adding OS and RTE module configurations
You can now add and configure module configurations for all the basic software modules to an ECU configuration as defined by AUTOSAR. This includes RTE and OS configurations. SystemDesk’s RTE generation and simulation features support module configurations according to AUTOSAR ECU configuration parameter files (Versions 2.1, 3.0, 3.1) and RTE configurations according to a special dSPACE RTE configuration based on AUTOSAR 3.1.

### Importing and exporting SWC-internal variables
You can now import A2L variables to a software component’s internal behavior. The parameters are imported directly from an A2L file and can include parameters such as maps and curves. After RTE code generation, SystemDesk lets you export combined A2L files that contain both imported and generated calibration parameters and measurements.

SystemDesk now also lets you export A2L files after the simulation build. This allows you to export an A2L file with complete address information.
Automating ECU configuration

With this version of SystemDesk, you can automate the configuration of basic software modules. For example, you can add and configure module configuration elements such as OS tasks, NVM block descriptors, and COM signals using SystemDesk’s automation interface. SystemDesk can generate automation interfaces for basic software modules when you add a module configuration to an ECU configuration. Automation is also supported for configuring module configurations with vendor-specific parameters.
Simulating Systems

**Simulation Module**

**Direct connection of VPUs** For simulation purposes, SystemDesk 3.0 allows you to directly connect VPUs via new VPU ports without specifying and using a network communication. For example, you can quickly connect plant models to VPUs to perform closed-loop simulations.

**VPU-internal C-API** With this version of SystemDesk you can control SystemDesk’s simulation runs by the simulated code for debugging purposes. You can use the new VPU-internal C-API for this. The VPU-internal C-API lets you

- Read the elapsed simulation time
- Send messages to SystemDesk’s Message Browser
- Perform read/write access to VPU ports
- Stop or pause a simulation run

**PIL simulation** SystemDesk now lets you perform processor-in-the-loop (PIL) simulation. This allows you to verify simulation results from software-in-the-loop (SIL) simulation. You can simulate under realistic operating conditions and investigate errors caused by the target processor.

SystemDesk 3.0 supports the Freescale MPC5607 target simulation board together with the Diab 5.5.1 compiler. Further target simulation boards can be provided as an engineering service.

The XCP service is also supported for VPUs running in PIL mode. You can control PIL experiments in SystemDesk or the new Offline Simulation Player from ControlDesk Next Generation or other tools that can use the XCP service.

**VPU build with MSVC 10 compiler** The build process for VPUs now supports also the Microsoft Visual Studio 2010 compiler (MSVC 10).

**Improved export of A2L (ASAP2) files for VPUs** You can specify to export automatically an A2L (ASAP2) file - also known as an ASAM MCD-2 MC file - after the system build. The exported A2L (ASAP2) file includes the address information taken from the built VPU. The file also includes the IF_DATA section for all the measurement rasters in the simulation application.

You can use the A2L (ASAP2) file to perform experiments with a tool that uses the XCP service, for example, ControlDesk Next Generation.
Offline Simulation Player

Running simulation applications without SystemDesk

With SystemDesk 3.0 comes the new Offline Simulation Player, which allows you to run simulation applications without starting SystemDesk.

The Offline Simulation Player can run independently of SystemDesk. It lets you configure and run offline simulations of simulation projects that were originally built with SystemDesk (or dSPACE Target for Offline Simulation). Unlike the Plotter in SystemDesk, it does not visualize simulation runs.

The Offline Simulation Player’s user interface offers commands to start, pause, continue, and stop simulations runs. You can also perform single step simulation. The integrated Message Browser provides a history of all error and warning messages that occur when you work with the Offline Simulation Player. The Configuration Grid lets you configure the simulation run and provides read access to further properties.
Accessing VPUs from ControlDesk Next Generation

The Offline Simulation Player lets you use the XCP service of VPUs to calibrate and visualize an application’s parameters with ControlDesk Next Generation, see example below.
Importing and Exporting AUTOSAR Files

Improved AUTOSAR Import/Export  

**Improved (re)import of AUTOSAR elements**  
With SystemDesk 3.0 you can now (re)import multiple elements that are assigned to package group(s) to their locations in a SystemDesk project all at once.

In the improved AUTOSAR Import dialog, you can select the elements of the AUTOSAR XML data to be imported individually.

When you import AUTOSAR XML files, SystemDesk analyzes the contained AUTOSAR XML data and rebuilds the AUTOSAR structures and elements in the XML data with corresponding structures and elements in SystemDesk. Elements that are selected to be (re)imported from AUTOSAR XML file(s) automatically replace their existing counterparts in SystemDesk.

If you import a new AUTOSAR element that neither exists in the SystemDesk project nor references other elements, you can change the target for rebuilding the corresponding structure/element in the project.
Improved handling of unsupported AUTOSAR elements

AUTOSAR provides additional data that is not interpreted by SystemDesk. With this version of SystemDesk, the reexport of unsupported AUTOSAR elements is improved.

Generic AUTOSAR elements  AUTOSAR record layout elements are imported as generic AUTOSAR elements. You cannot modify their properties, but you can:

- Inspect the names and the XML tags of generic AUTOSAR elements
- Assign generic AUTOSAR elements to package element groups in SystemDesk
- Export generic AUTOSAR elements to AUTOSAR

When AUTOSAR export is performed, SystemDesk exports AUTOSAR record layout elements without changes. For more details, refer to Generic AUTOSAR Element (refer to General Page (Generic AUTOSAR Element) (SystemDesk Reference)).
Exchanging Data with TargetLink

This version of SystemDesk lets you easily exchange data with the AUTOSAR behavior modeling tool TargetLink. The illustration below shows exchanging data between TargetLink and SystemDesk schematically.

Exchanging data using software component containers has the following benefits:

- Exchanging data safely, easily, and with a defined workflow.
- Integrating software component code into SystemDesk’s simulation feature.
- Managing additional related files such as feature specifications, test specifications, etc.
SystemDesk exports a container, i.e., a bundle of files that completely represent the software component you have selected. The files are described in a catalog file that is part of the container. You are now ready to import the software component to TargetLink to create and update the software component’s implementation.
Importing data to SystemDesk from TargetLink

You can import containers that have been exported from TargetLink to SystemDesk to refine the software components' internal behaviors.

During import the bundle of files from TargetLink (external container) is synchronized with the files in SystemDesk (local container). Updated AUTOSAR files and A2L files can be imported to SystemDesk. The operations which are applied for synchronizing containers depend on configurable file categories. The default file categories are assigned automatically for a best-practice workflow.

After importing the container, you are ready to refine the imported software components' internal behaviors.

Managing containers

The Container Manager lets you manage containers exported from TargetLink and SystemDesk and is accessible from both tools. It provides the following features:

- Open container catalog files that are part of the containers, and review and synchronize the files in two containers.
- Add additional files, such as feature specifications for the software component, to the containers.
Directly compare and synchronize a container with elements in SystemDesk’s Package Manager while SystemDesk is running.

For details, refer to Basics on Exchanging Containers (SystemDesk Guide).

dSPACE Target for Offline Simulation 1.2

New version of dSPACE Target for Offline Simulation

dSPACE Target for Offline Simulation 1.2 for preparing a Simulink® model for integration into SystemDesk has the following new features:

- Generation of an A2L file
- Automatic generation of a VPU DLL
- Support of Simulink models with multitasking
Support of MATLAB® Releases R2007b+, R2008a+, R2008b, R2009a, R2009b, and R2010a

Features Discontinued in SystemDesk 3.0

| Discontinued support of the SystemDesk (SDXML) format | With SystemDesk 3.0, the SystemDesk SDXML format is no longer supported. You can still import/export AUTOSAR files such as software component description files or ECU parameter configuration files. Instead of the SystemDesk (SDXML) format you should either use AUTOSAR files or, for single-user projects, SystemDesk's project (SDP) files. |
| Discontinued support of version control systems | In connection with the discontinuation of the SystemDesk format, placing elements under version control is also discontinued in SystemDesk 3.0. However, you can still use SystemDesk with several users by placing exported AUTOSAR files such as software component description files or ECU parameter configuration files under your preferred version control system and structuring your projects accordingly. |
Migrating to SystemDesk 3.0

To reuse SystemDesk 2.1 projects and automation scripts with SystemDesk 3.0, you have to migrate them. For detailed information on migrating projects and automation scripts, refer to the Migrating SystemDesk 2.1 to SystemDesk 3.0 document which you can download from www.dspace.de/goto?MigratingSD21.

To reuse a project last saved in SystemDesk 2.0 or earlier with SystemDesk 3.0, you have to migrate the project to SystemDesk 2.1 first. For detailed information on migrating projects to SystemDesk 2.1, refer to the related New Features and Migration document which is available via http://www.dspace.com/goto?migration.
# TargetLink

<table>
<thead>
<tr>
<th>Where to go from here</th>
<th>Information in this section</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>New Features of TargetLink 3.2 and dSPACE Data Dictionary 3.2</strong></td>
<td>142</td>
</tr>
<tr>
<td><strong>Migrating to TargetLink 3.2 and dSPACE Data Dictionary 3.2</strong></td>
<td>167</td>
</tr>
</tbody>
</table>
New Features of TargetLink 3.2 and dSPACE Data Dictionary 3.2

For last-minute information on TargetLink 3.2 and on potential difficulties with model upgrades, it is recommended to visit the TargetLink 3.2 website at http://www.dspace.com/goto?TargetLinkDocumentationUpdate.

Where to go from here

Information in this section

| New Production Code Generation Features | 142 |
| New AUTOSAR-Related Features           | 154 |
| New Features of the dSPACE Data Dictionary | 160 |

New Production Code Generation Features

Where to go from here

Information in this section

| Online Parameter Modification | 143 |
| Debugging in SIL Simulation Mode  | 145 |
| New TargetLink Sqrt Block       | 145 |
| Enhancements to the Target Simulation Module | 146 |
| New TargetLink API commands     | 148 |
| Code Generator Options          | 149 |
| General Enhancements and Changes | 151 |
Online Parameter Modification

Modifying parameter values online

TargetLink allows you to modify the parameter values of a simulation application in SIL and PIL simulation modes.

You can modify parameter values without having to regenerate code. This lets you perform the simulation with parameter values different from those specified in the model, which reduces the time needed for testing and lets you perform multiple tests of identical code with different parameter sets.

For further information, refer to Basics of Modifying Parameter Values of the Simulation Application Online (TargetLink Production Code Generation Guide).

Online parameter modification does not support modifying the parameter values for the simulation in MIL mode. To modify the parameter values for the simulation in MIL mode, you have to set them in your model.

M-interface to the simulation application

With the tl_sim_interface command, TargetLink provides the M-interface to the simulation application. You can use it to access the simulation application downloaded to the evaluation board or loaded to the memory of the host PC (if it is a SIL application). The command has a number of parameters for performing various tasks, for example, connecting the simulation application to and disconnecting it from the simulation platform, resetting the simulation application, obtaining address information of variables and functions, reading and writing variables. You can use this command in the MATLAB command window or in M files to access the simulation application and modify parameter values online.
TargetLink 3.2 provides a new demo model that helps you understand how to modify parameter values online. The illustration below shows the root level of the online_parameter_modification demo model.

The different parts of the example show you the following aspects of modifying parameter values online:

- Modifying parameter values online and starting the simulation
- Toggling data variants and starting the simulation
- Modifying parameter values in a hook function

For further information, refer to Example of Modifying Parameter Values Online ([TargetLink Production Code Generation Guide](#)).
Debugging in SIL Simulation Mode

Debugging production code

TargetLink supports debugging with the Microsoft® Developer Studio in SIL simulation mode, i.e., you can set breakpoints and debug the generated code while a SIL simulation is running. Debugging is possible with both the Microsoft® Visual Studio Professional Edition and the Microsoft® Visual Studio Express Edition (except Microsoft Visual Studio Express Edition 2010).

Debugging in SIL simulation mode is useful if you want to check whether and in which order code is executed, why the generated production code does not behave as expected or whether a task of a multi-rate system is calculated at all.

For details, refer to How to Debug in SIL Simulation Mode (TargetLink Production Code Generation Guide).

New TargetLink Sqrt Block

Sqrt block instead of function

As of TargetLink 3.2 and MATLAB R2010a, the sqrt function no longer appears in the Math block. If TargetLink 3.2 is used in combination with MATLAB R2010a (or newer), you can use a separate Sqrt block to perform square-root calculations. This block includes the sqrt, signedsqrt and rsqrt functions. If the rsqrt function is to be calculated,
only a floating-point data type can be selected, otherwise integer data types are also allowed. Up to MATLAB R2009b, the \texttt{sqrt} function was provided by the Math block only. It depends on the TargetLink and MATLAB versions installed whether the Math block provides the \texttt{sqrt} function, or if the Sqrt block is available instead.

If you open a model built with TargetLink < 3.1 in combination with MATLAB R2010a for the first time and an update is performed by the system, a Sqrt block is inserted for each Math block that computes the square root function. In all other cases, you have to perform a manual block upgrade by running the \texttt{tl\_fix\_sqrt} function on your model. \texttt{tl\_fix\_sqrt} replaces any Math block that uses \texttt{sqrt} with an equivalent Sqrt block that ensures the same behavior. This function does not affect libraries. To replace blocks in a library, open and unlock the library and apply this function to it.

Example

\begin{verbatim}
    tl_fix_sqrt ('System',gcb);
\end{verbatim}

Enhancements to the Target Simulation Module

(New) evaluation boards, microcontrollers, and compilers

The following table shows the combinations of evaluation boards, microcontrollers, and compilers supported by TargetLink 3.2 (TargetLink abbreviations). New evaluation boards, microcontrollers, and compiler versions are underscored. For details, refer to TargetLink Target Reference.

<table>
<thead>
<tr>
<th>Evaluation Board</th>
<th>Microcontroller Type</th>
<th>Compiler 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCT HCS12 T-Board (DP256)</td>
<td>Freescale MC9S12DP256</td>
<td>Cosmic 4.5, 4.6, 4.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Metrowerks CodeWarrior 3.1</td>
</tr>
<tr>
<td>MCT HCS12 T-Board (DP512)</td>
<td>Freescale MC9S12DP512</td>
<td>Cosmic 4.7, 4.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Cosmic 4.5, 4.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Metrowerks CodeWarrior 3.1, 5.0</td>
</tr>
<tr>
<td>Freescale 56F8367 Evaluation Module</td>
<td>Freescale MC56F8367</td>
<td>Metrowerks CodeWarrior 8.1</td>
</tr>
<tr>
<td>Axiom CMD-0565</td>
<td>Freescale MPC565</td>
<td>Wind River Diab 5.5, 5.6, 5.7</td>
</tr>
<tr>
<td>Axiom CML-0555</td>
<td>Freescale MPC555</td>
<td>No longer supported: Wind River Diab 5.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Green Hills 5.0, 5.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Green Hills 4.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Metrowerks CodeWarrior 8.1, 8.5, 8.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Wind River Diab 5.5, 5.6, 5.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Wind River Diab 5.3</td>
</tr>
</tbody>
</table>
For detailed information on the evaluation boards supported by TargetLink, refer to Combinations of Evaluation Boards, Microcontrollers, and Compilers (Link to TargetLink Target Reference).

<table>
<thead>
<tr>
<th>Evaluation Board</th>
<th>Microcontroller Type</th>
<th>Compiler(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axiom MPC5554DEMO</td>
<td>Freescale MPC5554</td>
<td>Green Hills 5.0, 5.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Green Hills 4.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Metrowerks CodeWarrior 2.3, 2.4, 2.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Metrowerks CodeWarrior 2.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>n/a GNU 3.4, 4.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Wind River Diab 5.5, 5.6, 5.7, 5.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Wind River Diab 5.3</td>
</tr>
<tr>
<td>dSPACE DS1603</td>
<td></td>
<td>Microtec 3.2, 3.3, 3.5, 3.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Wind River Diab 5.5, 5.6, 5.7, 5.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Wind River Diab 5.3</td>
</tr>
<tr>
<td>Freescale MPC5561EVB</td>
<td>Freescale MPC5561</td>
<td>Green Hills 5.0, 5.1</td>
</tr>
<tr>
<td>Freescale MPC5561EVB USB</td>
<td></td>
<td>Metrowerks CodeWarrior 2.3, 2.4, 2.6</td>
</tr>
<tr>
<td>Freescale MPC5604BEVB</td>
<td>Freescale MPC5600</td>
<td>Wind River Diab 5.5, 5.6, 5.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Green Hills 5.1</td>
</tr>
<tr>
<td>MCT S12X T-Board</td>
<td>Freescale MC9S12XDPS12</td>
<td>Cosmic 4.7, 4.8</td>
</tr>
<tr>
<td>MCT S12X T-Board USB</td>
<td></td>
<td>No longer supported: Cosmic 4.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Metrowerks CodeWarrior 4.7, 5.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Metrowerks CodeWarrior 4.6</td>
</tr>
<tr>
<td>T+ME Promotion Package 166</td>
<td>Infineon c167</td>
<td>Altium Tasking 8.6, 8.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Altium Tasking 7.5, 8.0, 8.5</td>
</tr>
<tr>
<td>Infineon TBTC1766</td>
<td>Infineon TC1766</td>
<td>Altium Tasking 2.5, 3.2, 3.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Altium Tasking 2.3, 3.0</td>
</tr>
<tr>
<td>Infineon TBTC1767</td>
<td>Infineon TC1767</td>
<td>Altium Tasking 2.5 (2.5r2p1 and younger), 3.2, 3.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Altium Tasking 3.0</td>
</tr>
<tr>
<td>Infineon TBTC1796</td>
<td>Infineon TC1796</td>
<td>Altium Tasking 2.5, 3.2, 3.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Altium Tasking 2.3, 3.0</td>
</tr>
<tr>
<td>Infineon SK-EB XC2287</td>
<td>Infineon XC2287</td>
<td>Altium Tasking C166 VX 2.3, 2.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Altium Tasking C166 VX 2.1, 2.2</td>
</tr>
<tr>
<td>NEC Fx3-CAN IT!</td>
<td>NEC V850ES/FG3-μPD70F3377</td>
<td>Green Hills 5.0, 5.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: Green Hills 4.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NEC 3.30, 3.40</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No longer supported: NEC 3.10, 3.20</td>
</tr>
<tr>
<td>Renesas M3A-2154</td>
<td>Renesas M32192</td>
<td>Gaio 9, 10</td>
</tr>
<tr>
<td>Renesas EVB7058</td>
<td>Renesas SH-2E/7058</td>
<td>Renesas 4.3, 5.0</td>
</tr>
<tr>
<td>Renesas SH-72513 System Development Kit (SDK72513)</td>
<td>Renesas SH-7A-FPUSHT72513</td>
<td>Renesas 9.0, 9.1, 9.3</td>
</tr>
</tbody>
</table>

(1) Compiler Suite Version Supported
Discontinued boards

No longer supported, no longer distributed

The following boards are no longer supported by TargetLink and no longer distributed by dSPACE:

- Renesas EVB7055F (Renesas SH-2E/SH7055F microcontroller)
- Renesas EVB2633F (Renesas H8S/2633F microcontroller)

New TargetLink API commands

API commands available with TargetLink 3.2

In TargetLink 3.2, there are the following new TargetLink API commands:

- As of version 3.2, TargetLink allows you to modify the parameter values of a simulation application (Refer to Online Parameter Modification on page 143). The `tl_sim_interface` command provides the M-interface to the simulation application. It has a number of parameters (known as actions) for performing various tasks, for example, connecting the simulation application to and disconnecting it from the simulation platform, resetting the simulation application, modifying parameters, running the simulation.

  For further information, refer to `tl_sim_interface` ([TargetLink API Reference](#)).
### Code Generator Options

With TargetLink 3.2 the following new Code Generator options are available.

<table>
<thead>
<tr>
<th>Description</th>
<th>Explanation</th>
<th>Default Value</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DoNotUseAssignArithmeticForAccumulation</strong></td>
<td>Controls how sums with more than two operands are implemented.</td>
<td>If the option is not set (default), an auxiliary variable is introduced to hold intermediate results that are accumulated by assign additions or subtractions. If the option is set, TargetLink will not introduce an auxiliary variable and will not use assign arithmetic ((+=, -=)) for accumulations, so all terms are handled in a common statement. Example: Option not set: <code>Int16 Aux; Aux = In1; Aux += In2; Aux -= In3; Out = Aux + In4;</code> Option set: <code>Out = In1 + In2 - In3 + In4;</code></td>
<td>off</td>
<td>-</td>
</tr>
<tr>
<td><strong>SupportSinglePrecisionLibraries</strong></td>
<td>Allows the generation of ANSI-C99 single-precision, floating-point library operations.</td>
<td>For statements with floating-point data types and math library functions, this option allows the Code Generator to use single-precision, floating-point library functions from the ANSI-C99 standard. These functions end with an <code>f</code> added to the normal math library functions, e.g. <code>sinf</code>. Possible values: • No single-precision floating-point library functions are generated • Single-precision floating-point library functions are generated if at least one operand or the result is floating-point but neither an operand nor the result is <code>Float64</code> • Single-precision floating-point library functions are always generated if at least one operand or the result is floating-point (either <code>Float32</code> or <code>Float64</code>)</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td><strong>SuppressPerformanceWarningForTableDataEvaluation</strong></td>
<td>Suppresses a performance warning popup window, if the number of table data items exceeds a defined number of values.</td>
<td>If the table data exceeds a defined number of values, system performance can run low. A popup window informs you of this. However, popup windows can hinder automatic code generation. This option suppresses popup windows.</td>
<td>off</td>
<td>-</td>
</tr>
<tr>
<td>Obsolete and changed Code Generator options</td>
<td>For information on obsolete and changed Code Generator options, refer to Obsolete and Changed Code Generator Options on page 176.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related topics</td>
<td>References</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Code Generator Options (<a href="#">TargetLink Block and Object Reference</a>)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
General Enhancements and Changes

Preferences Editor

The new TargetLink Preferences Editor replaces the former Preferences dialog, which was available up to TargetLink 3.1. The TargetLink Preferences Editor contains configuration-relevant settings and allows you to easily modify settings such as the preferred code editor, synchronization properties and compiler installation paths.

For details, refer to Customizing the TargetLink Environment Via the TargetLink Preferences Editor (TargetLink Advanced Practices Guide).
For a number of Stateflow objects, TargetLink now provides separate object dialogs. As shown in the TargetLink SF Input dialog below, these dialogs help you enter TargetLink data for Stateflow.

There are separate dialogs for the following Stateflow object types:
- SF Local
- SF Output
- SF Input
- SF Fcn Output
- SF Fcn Input
- SF Imported
- SF Exported
- SF Temporary
- SF Constant
- SF Parameter

For details, refer to Stateflow Object Dialogs of TargetLink Block and Object Reference.
### Option to switch between TargetLink and Simulink block dialogs

TargetLink provides a new option that controls whether the TargetLink or Simulink block dialog opens when you double-click a TargetLink block. You can use this option if you work with the TargetLink Blockset stand-alone and prefer to view and set block properties in Simulink dialogs.

You can set the option via TargetLink's TargetLink Preferences Editor (TargetLink Tool and Utility Reference) or by entering `tl_pref('set', 'DialogProvider', 'Simulink')` in the MATLAB command window.

To switch back to TargetLink block dialogs, enter `tl_pref('set', 'DialogProvider', 'TargetLink')`.

![Note](https://via.placeholder.com/150)

Although it is possible to switch to Simulink dialogs in TargetLink full-featured mode, it is strongly recommended to use this option only with the TargetLink Blockset stand-alone.

### Improved code generation for floating-point processors

- With TargetLink, you can specify the default base data type for all the blocks copied from the TargetLink block library (tllib). For details, refer to How to Preselect the Default Base Data Type (TargetLink Advanced Practices Guide).

  If you specify a floating-point data type (`Float32`, `Float64`), this default base data type now also applies to parameters, for example, to gain parameters `Ki` or `Kp` used in the PIPT1 demo model.

- Support of 32-bit floating-point library functions according to the C99 standard. Via the SupportSinglePrecisionLibraries Code Generator option, you can allow the Code Generator to use single-precision, floating-point library functions from the ANSI-C99 standard. For details, refer to Code Generator Options (TargetLink Block and Object Reference).

### Multiport Switch block

The Multiport Switch Block provides three new options (Zero-based contiguous, One-based contiguous and Specify indices) that let you specify the data port order. For details, refer to Multiport Switch Block (TargetLink Block and Object Reference).
Discontinued tool

Previous versions of TargetLink and the dSPACE Data Dictionary provided you with the `tl_export2dd` tool that let you import TargetLink model data into the dSPACE Data Dictionary. As of TargetLink 3.2, this tool is not part of the TargetLink Base Suite anymore.

If you want to use `tl_export2dd`, you can download it via the TargetLink Product Support Center (www.dspace.com/goto?TargetLinkProductSupportCenter).

Related topics

- Basics
  - Changes in TargetLink API Functions on page 178

New AUTOSAR-Related Features

Where to go from here

Information in this section

<table>
<thead>
<tr>
<th>Features of the TargetLink AUTOSAR Module</th>
<th>154</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exchanging Software Component Containers with SystemDesk</td>
<td>157</td>
</tr>
</tbody>
</table>

Features of the TargetLink AUTOSAR Module

Supported AUTOSAR Releases

The TargetLink AUTOSAR Module supports the following AUTOSAR Releases:

- AUTOSAR Release 3.1 with Versions 3.1.0 (continued), 3.1.2 and 3.1.4 (new)
- AUTOSAR Release 3.0 with Versions 3.0.2 (continued), 3.0.4, and 3.0.6 (new)
- AUTOSAR Release 2.1 with Version 2.1.4 (continued)
### AUTOSAR compiler abstraction

TargetLink now supports compiler abstraction as defined by AUTOSAR. To make source code platform-independent, AUTOSAR has defined a set of macro definitions for source code elements such as functions, variables, and pointers. TargetLink can use these macros to define runnables, thus generating software component code that complies with AUTOSAR’s compiler abstraction definition.

Whenever you define a runnable, you can choose to generate runnable code with AUTOSAR’s compiler abstraction macros. To generate code with compiler abstraction, you either do not change a runnable’s `FunctionClass`, which is unset by default, or select `AUTOSAR/RUNNABLE`. If you select another `FunctionClass` for the runnable, such as `GLOBAL_FCN`, TargetLink generates code without compiler abstraction macros.

For details of how runnable code has changed in this TargetLink version, refer to "AUTOSAR-Related Migration Aspects" on page 184.

### Improved support of RTE API functions

**Rte_IRead**

TargetLink now supports generating code for implicit reading of composite data elements, i.e. arrays and structures.

### Array passing

Versions 3.0.6 and 3.1.4 of AUTOSAR define the array passing scheme in detail. Whenever the RTE accesses an array there are two ways to implement corresponding RTE API functions:

- By pointing to the array type.
- By pointing to the array element type

TargetLink implements software components with RTE API functions that use the latter array passing scheme, i.e., they use pointers to array element types.

Make sure that your RTE generator uses the same array passing scheme as TargetLink as described above.

If you use SystemDesk as an RTE code generator, you have to set SystemDesk’s `RteUsePtr2ArrayBaseTypeForArgs` and the `RteUsePtr2ArrayBaseTypeForRetVal` RTE code generation options to `True` when working with TargetLink.
Improved generation of TargetLink subsystems

With the `tl_generate_swc_model` MATLAB API command, you can generate a TargetLink subsystem with imports/exports and a predefined substructure from AUTOSAR data. The following improvements are provided by this version of TargetLink:

- TargetLink now generates buses with initial values for structured data that is exchanged with a TargetLink subsystem. This allows you to generate AUTOSAR-compliant code after TargetLink subsystem generation immediately.
- You can now directly generate one TargetLink subsystem for each software component that is defined in the AUTOSAR data.
- TargetLink now provides hook functions for customizing generation of TargetLink subsystems.

For details on generating TargetLink subsystems, refer to Basics on Generating TargetLink Subsystems from AUTOSAR Data ([TargetLink AUTOSAR Modeling Guide](#)).

Improved AUTOSAR import/export

- With this version of TargetLink the AUTOSAR, revision of an imported file is obtained directly from it. If you want to import/export AUTOSAR data from/to a file according to a revision that is not explicitly supported, you have to supply the corresponding AUTOSAR schema during import/export via the import/export configuration. However, your AUTOSAR data to be imported/exported must comply with one of the supported AUTOSAR Releases.
- TargetLink now lets you specify the overwrite/merge behavior of AUTOSAR elements during import. By configuring the AUTOSAR import you can select whether AUTOSAR elements such as type definitions, data access points, or runnables are overwritten during import, or whether additional/changed attributes are merged with the elements.

For details on importing and exporting AUTOSAR files, refer to Importing and Exporting AUTOSAR Files ([TargetLink AUTOSAR Modeling Guide](#)).
Exchanging Software Component Containers with SystemDesk

This version of TargetLink comes with a new approach for AUTOSAR-compliant development in combination with the architecture tool SystemDesk. It is based on the exchange of so-called SWC containers. The illustration below shows exchanging SWC containers between TargetLink and SystemDesk schematically.

Exchanging data using software component containers has the following benefits:

- Exchanging data safely, easily, and with a defined workflow.
- Integrating software component code into SystemDesk’s simulation feature.
- Managing additional related files such as feature specifications, test specifications, etc.
Exporting data from TargetLink to SystemDesk

TargetLink lets you export a container, i.e., a bundle of files that completely represent the implementation of the software component you have generated code for. The files are described in a catalog file that is part of the container. You are now ready to import software component implementations to SystemDesk for further refinement.

Importing data to TargetLink from SystemDesk

You can import containers that have been exported from SystemDesk to TargetLink to create or update software components that reside in the /Pool/Autosar area of the Data Dictionary.

During import the bundle of files from SystemDesk (external container) is synchronized with the files in TargetLink (local container). Updated AUTOSAR files can be imported to TargetLink. The operations which are applied for synchronizing containers depend on configurable file categories. The default file categories are assigned automatically for a best-practice workflow.

After importing the container, you are ready to refine the imported software components’ behavior and generate software component implementations.
Managing containers

The Container Manager lets you manage containers exported from TargetLink and SystemDesk and is accessible from both tools. It provides the following features:

- Open container catalog files that are part of the containers, and review and synchronize the files in two containers.
- Add additional files, such as feature specifications for the software component, to the containers.
- Directly compare and synchronize a container with elements in SystemDesk’s Package Manager while SystemDesk is running.

For details, refer to Basics on Exchanging Containers (TargetLink AUTOSAR Modeling Guide).
New Features of the dSPACE Data Dictionary

Objective
The dSPACE Data Dictionary 3.2 (DD) has the following new features, enhancements and changes:

Where to go from here
Information in this section

| New Key Features | 160 |
| New and Modified DD MATLAB API Commands | 165 |

New Key Features

Objective
Information on the new key features of dSPACE Data Dictionary 3.2 is provided below.

Improved Search
The search functionality in the Data Dictionary Manager was improved.

The Find Object dialog has been modified.

In addition to an object's name, you can now specify a property name and/or property value. You can even find objects with properties that are not set. Apart from regular expressions, you can now also use wildcards in the search strings to search for object names, property names and/or property values that you know only a part of.
If you want to find all the objects that match the specified search criteria, you can click Find All. The new Find Object Results pane opens and displays the search results.

The hyperlinks in the Object column let you navigate to the DD objects in the Data Dictionary Navigator.

For further information, refer to How to Find Data Dictionary Objects (dSPACE Data Dictionary Basic Concepts Guide).
Finding references to DD objects

The Data Dictionary Manager lets you find other objects that a DD object is referenced by, for example, other DD objects, Simulink blocks and Stateflow objects. The functionality is available via context menus of Data Dictionary objects.

The new Find References Results pane displays the search results. The hyperlinks in the Object column let you navigate to the DD objects in the Data Dictionary Navigator.

For further information, refer to How to Find an Object Reference. (dSPACE Data Dictionary Basic Concepts Guide).

162
New Features and Migration November 2010
Adding custom functionality to the Data Dictionary Manager

The dSPACE Data Dictionary lets you add custom functionality, i.e., user-defined MATLAB functions (M files) to the Data Dictionary Manager. You can add user-defined menu commands that invoke user-defined MATLAB functions (M files).

You can specify menu commands in

- The menus of the menu bar
- The context menus of objects in the Data Dictionary Navigator
The context menus of properties in the property value list

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
<th>Open Editor</th>
<th>Rename</th>
<th>Delete/Unset Property</th>
<th>GoTo Target</th>
<th>Edit Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>full link attached to database</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ValueType</td>
<td>on</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Width</td>
<td>100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CreateTyped</td>
<td>on</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BaseProperty</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BasePropertyClass</td>
<td>&quot;SB&quot;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The menu extensions are specified in the `DDManagerMenuExtension.xml` XML file in `%USERPROFILE%\Application\Data\dSPACE\<GUID>\TargetLink\DDMenuExtension` where `<GUID>` stands for a globally unique identifier that is provided by the TargetLink installation or in separate XML files that have to be stored in the current working folder.

For further information, refer to Adding Custom Functionality to the Data Dictionary Manager (dSPACE Data Dictionary Basic Concepts Guide).

Custom messages and custom output views

To display feedback from your own tools, you can issue custom messages either in the Message Browser or in a custom output view, i.e., a separate pane that you can create in the Data Dictionary Manager.

For further information, refer to How to Create Custom Output Views and How to Issue Custom Messages (dSPACE Data Dictionary Basic Concepts Guide).
Edit Target

There is a new Edit Target menu command in the context menu that is displayed when you right-click a reference property in the property value list. The command lets you open the object-specific dialog to edit the referenced object.

The command is available for references to variable or typedef objects.

For further information, refer to Edit Target (dSPACE Data Dictionary Manager Reference).

New and Modified DD MATLAB API Commands

Modified DD MATLAB API command: dsddman

The dsddman API command is the command-line interface to the Data Dictionary Manager. It has been extended to let you display feedback from your own tools, either in the Message Browser or a custom output view, i.e., a separate message pane in the Data Dictionary Manager.

dsddman('AddMessage', ...) lets you issue a custom message in the Message Browser. For further information, refer to How to Issue Custom Messages (dSPACE Data Dictionary Basic Concepts Guide).
- `dsddman('DemandCustomOutputView', ...)` lets you create a custom output view. For further information, refer to How to Create Custom Output Views (dSPACE Data Dictionary Basic Concepts Guide).
- `dsddman('AddCustomMessage', ...)` lets you issue a custom message in a custom output view. For further information, refer to How to Issue Custom Messages (dSPACE Data Dictionary Basic Concepts Guide).
- `dsddman('ClearOutputView', ...)` lets you clear a custom output view.
- `dsddman('CloseOutputView', ...)` lets you close a custom output view.

New option for DD MATLAB API command `dsdd_compare`

The new `IgnoreAttributes` option with the `dsdd_compare` API command lets you exclude certain attributes from the comparison of two DD project files or selected DD objects.

The following attributes can be excluded:
- `access`
- `numOfChildren`
- `objectKind`
- `temporary`

For further information, refer to `dsdd_compare` (dSPACE Data Dictionary MATLAB API Reference).

Related topics

References

- `dsddman` (dSPACE Data Dictionary MATLAB API Reference)
Migrating to TargetLink 3.2 and dSPACE Data Dictionary 3.2

Upgrade process

To migrate from TargetLink 3.1 to TargetLink 3.2, only the dSPACE Data Dictionary needs to be upgraded. This is done by the DD update process. Upgrading TargetLink models and libraries is not necessary. To migrate models/libraries from older TargetLink Releases, you also have to perform the migration steps of the intervening TargetLink Releases. Refer to the previous TargetLink Migration Guides available on your DVD and to the documentation of the tl_upgrade API command.
For last-minute information on TargetLink 3.2 and on potential difficulties, you are recommended to visit the TargetLink 3.2 website at http://www.dspace.com/goto?TargetLinkDocumentationUpdate.

- The upgrade process for dSPACE Data Dictionary 3.2 upgrades only Data Dictionary files created under TargetLink 2.x or 3.x (and the associated Data Dictionary versions). Projects that were created under TargetLink 1.3 or even older versions cannot be upgraded directly to TargetLink 3.2 and dSPACE Data Dictionary 3.2. You must first perform an upgrade to a TargetLink 2.x version (including the associated Data Dictionary version) before you can upgrade to TargetLink 3.2 and dSPACE Data Dictionary 3.2.

Previous New Features and Migration documents are available via Internet and on the dSPACE DVD. You can download them from http://www.dspace.com/goto?migration or read them from the dSPACE DVD (see the \Doc folder). The PDF files are called TLNewFeaturesAndMigrationVer.x.y.pdf, where x.y stands for the release number.

- When upgrading models and libraries, first upgrade models or libraries that themselves do not reference any other libraries, i.e., the blocks they contain have no links to other libraries. Start with the bottom library and then upgrade the libraries above it in ascending order. You must not open the model or a referencing library until this is done. For related information on upgrading libraries, refer to How to Prepare TargetLink User Libraries for Upgrade (TargetLink Production Code Generation Guide) and tl_upgrade (TargetLink API Reference).

Where to go from here

Information in this section

| Migrating to dSPACE Data Dictionary 3.2 | 169 |
| How to Upgrade a Data Dictionary Without Include Files | 173 |
| How to Upgrade a Data Dictionary With Include Files | 174 |
| Obsolete and Changed Code Generator Options | 176 |
Migrating to dSPACE Data Dictionary 3.2

Upgrading Data Dictionary files

To migrate from TargetLink 3.1 to TargetLink 3.2, only the dSPACE Data Dictionary needs to be upgraded. This is done by the DD update process. Upgrading TargetLink models and libraries is not necessary.

For last-minute information on TargetLink 3.2 and on potential difficulties, you are recommended to visit the TargetLink 3.2 website at http://www.dspace.com/goto?TargetLinkDocumentationUpdate.

- The upgrade process for dSPACE Data Dictionary 3.2 upgrades only Data Dictionary files created under TargetLink 2.x or 3.x (and the associated Data Dictionary versions). Projects that were created under TargetLink 1.3 or even older versions cannot be upgraded directly to TargetLink 3.2 and dSPACE Data Dictionary 3.2. You must first perform an upgrade to a TargetLink 2.x version (including the associated Data Dictionary version) before you can upgrade to TargetLink 3.2 and dSPACE Data Dictionary 3.2.

Previous New Features and Migration documents are available via Internet and on the dSPACE DVD. You can download them from http://www.dspace.com/goto?migration or read them from the dSPACE DVD (see the \Doc folder). The PDF files are called TLLNewFeaturesAndMigrationVer.x.y.pdf, where x.y stands for the release number.

- For information on upgrading Data Dictionary files containing AUTOSAR objects, refer to AUTOSAR-Related Migration Aspects on page 184.
Deletion of Subsystem and <Application> areas

Due to the changes in the data model, the data in the Subsystem and <Application> areas, which resulted from code generation and build processes performed before the upgrade, is not consistent with new data models. Upgrading a DD project file therefore deletes the Subsystem and <Application> areas from the Data Dictionary. You are asked to confirm the deletion.

In batch mode, the subsystem node is deleted without query, i.e., only a message is issued. To create correct data in the Subsystem and <Application> areas after the upgrade, for example, data needed for generating ASAP2 files or AUTOSAR XML files, generate code for all the code generation units again.

Method to upgrade Data Dictionary files

dSPACE Data Dictionary 3.2 provides an upgrade process that automatically upgrades older DD files to Version 3.2.

The upgrade process can be called in three ways:

- Automatically opening an old TargetLink model.

  When you open a TargetLink model with an old (not upgraded) DD file, TargetLink first runs the dSPACE Data Dictionary's upgrade process.
Via the Tools menu in the Data Dictionary Manager
Manually via Tools -- Upgrade Current DD in the DD Manager.

- Via the Tools menu in the Data Dictionary Manager
- Manually via Tools -- Upgrade Current DD in the DD Manager.

The menu command is available only if the DD project file needs upgrading. Otherwise, it is disabled (grayed).

**Data Dictionary API command**
To call the upgrade process via the Data Dictionary’s API, type `dsdd('Upgrade')` in the MATLAB Command Window.

### Preconditions when upgrading Data Dictionary files

To ensure the DD file is upgraded correctly, the following preconditions must be met:

- There must be write permission for the DD file and the file must not be write-protected. If Data Dictionary Include files are used, there must be write access to all of them.
If Data Dictionary Include files are used, they must be saved after updating as well as the main DD file. This ensures that the Data Dictionary partitions into several files. To save the Include files correctly, you must first make the appropriate settings for them.

Related topics

- How to Upgrade a Data Dictionary With Include Files on page 174
- How to Upgrade a Data Dictionary Without Include Files on page 173
How to Upgrade a Data Dictionary Without Include Files

Objective
If you open a TargetLink model with an old, nonupgraded Data Dictionary file, you have to upgrade the Data Dictionary file.

Method
To upgrade a Data Dictionary without Include files
1 Open the model and the referenced dSPACE Data Dictionary, or type `dsdd(‘open’,<DDFile>)` in the MATLAB Command Window. The Data Dictionary needs upgrading dialog automatically opens if an older DD version is involved.

2 Click Yes if no Include files are used in the Data Dictionary. If Include files are used, refer to How to Upgrade a Data Dictionary With Include Files on page 174.

3 Click Yes.

4 Save the Data Dictionary.

When you have saved the Data Dictionary (assuming you have write permission to the relevant DD file), the upgrade of the DD file is completed.

Result
The next time you open the DD file, the upgrade dialog will not open because the DD file is up-to-date.
How to Upgrade a Data Dictionary With Include Files

Objective
If you open a TargetLink model with an old, nonupgraded Data Dictionary file, you have to upgrade the Data Dictionary file.

Method
To upgrade a Data Dictionary with include files
1. Open the model and the referenced dSPACE Data Dictionary, or type `dsdd('Open',<DDFile>)` in the MATLAB Command Window.
   The Data Dictionary needs upgrading dialog automatically opens if an older DD version is involved.

2. Select No in the upgrade dialog.
3. Under /Config/DDIncludeFiles, set the AutoLoad and AutoSave properties for each Include file as shown in the illustration below.
This ensures that after the Data Dictionary and the Include files have been upgraded, the upgraded Include files are saved when the Data Dictionary is saved. You can set these properties for a large number of Include files via the Object Explorer.

4 Start the DD upgrade (including the included files) via Tools - Upgrade Current DD in the DD Manager, or enter `dsdd('Upgrade')` in the MATLAB Command Window.

5 Save the Data Dictionary (assuming you have write permission to the relevant DD file). This completes the upgrade of the DD file itself as well as the included DD files.

**Result**

The next time you open the DD file, the upgrade dialog will not open because the DD file is up-to-date and so are the included Data Dictionary files. After the files have been properly upgraded, you might want to restore the old settings for the Data Dictionary Include files.
Obsolete and Changed Code Generator Options

Obsolete Code Generator options

The following Code Generator options are no longer used in TargetLink 3.2:
- InvalidateCodeOnError
- TreatSpecificErrorsAsWarnings

If you generate code while the obsolete options are still in the model, a warning occurs.

Changed Code Generator options

The Code Generator options listed below were changed.

Because of the changed RequirementInfoAsCodeComment option, comments containing requirement information are no longer included in the generated code automatically. If you want to have requirement information as comments in the generated code, set the RequirementInfoAsCodeComment to on.

<table>
<thead>
<tr>
<th>Description</th>
<th>Explanation</th>
<th>Default Value</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConsiderStateflowAuxiliariesForVariableSharing</td>
<td>For variable sharing, try to include internal auxiliary variables created for use in Stateflow charts.</td>
<td>on</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>HandleUnscaledStateflowExpressionsWithTlType</td>
<td>Variables and macros used in Stateflow have a Stateflow data type and a TargetLink data type. If the Stateflow type of a variable/macro is different from its TargetLink type and this variable is used in a Stateflow expression, the usual TargetLink code generation rules regarding scalings and types will be applied for this expression if the option is set to 'on'. Otherwise if no $LSB \neq 1.0$ and $Offsets\neq 0.0$ are involved in the expression, the expression will be copied to the generated code unchanged.</td>
<td>on</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
TargetLink

New Features and Migration November 2010

New Code Generator options

For information on new Code Generator options, refer to Code Generator Options on page 149.

Related topics

References

• Code Generator Options (TargetLink Block and Object Reference)
Changes in TargetLink API Functions

Changed API commands

In TargetLink 3.2, the behavior of the following TargetLink API command has slightly changed.

- **tl_get_blocks**
  This command now enables you to make case-sensitive searches for port blocks. In addition, it now returns the block type which is useful if the search criteria involves several block types, for example, 'TLSim'.

  ```
  [hBlocks, blockTypes] = tl_get_blocks('pipt1', 'TLSim')
  ```

  You can search for TargetLink blocks and supported (enhanced) Simulink blocks. The search criteria can be the block name (or a list of block names) used for the block in the TargetLink block library (e.g. Gain), or the block type (e.g. TL_Gain).

  The command returns a cell array containing associated block type(s), a vector with handles of found blocks, or an empty matrix if no blocks were found. If you are interested in TargetLink blocks only, use the block type as the search criteria. Inport returns unenhanced inports, and InPort or preferably TL_Inport returns the TargetLink port blocks.

  Example:
  ```
  tl_get_blocks('pipt1', 'TL_Inport')
  ```

  ```
  ans =
  15.0032
  16.0032
  ```

- **tl_get_sobjects**
  As of TargetLink 3.2, you cannot call the tl_get_sobjects command with the statechart parameter. Statecharts are now regarded as supported Simulink blocks. If you want to search for statecharts, use the tl_get_blocks command instead.

  ```
  [y,x] = tl_get_blocks('sf_demo', 'Statechart')
  ```

  ```
  y =
  14.0011
  x =
  'Stateflow'
  ```
- `tl_pref`
  As of TargetLink 3.2, you cannot call the `tl_pref` command with the `list` parameter. If you want to list all preferences, use the `get` parameter instead.

  ```matlab
  tl_pref('get')
  ans =
  CodeCovProgressBar: 'on'
  SyncSLScaling: 'off'
  SyncOutputScalingData: 'on'
  SyncSignalScalingData: 'on'
  SyncSaturationFlags: 'on'
  SyncConstrainedLimits: 'on'
  SyncParameterScalingData: 'on'
  SyncSFObjectScalingData: 'on'
  SyncSFObjectCompiledScalingData: 'on'
  SyncRTWData: 'on'
  ProjectFile: 'default.dd'
  ProjectFileAutosave: 'off'
  Editor: 'MATLAB Editor'
  BlockLibMode: 1
  DialogProvider: 'TargetLink'
  ```

- `tl_generate_code`
  The `GenerateAll` property is obsolete. For backward compatibility it is still supported but you should use `IncludeSubItems` instead.

  ```matlab
  tl_generate_code('IncludeSubItems','on')
  ```

- `tl_generate_swc_model`
  The `AutosarVersion` and `ModelClientServerPorts` properties are obsolete. The `AutosarVersion` property is obsolete because the version is recognized by the AUTOSAR import automatically. The `ModelClientServerPorts` property is obsolete because no SWC port blocks are added to the model for the client and the server ports.
## Various Migration Aspects

### Sqrt block instead of function
As of TargetLink 3.2 and MATLAB R2010a, the `sqrt` function no longer appears in the `Math` block. If you open a model built with TargetLink < 3.1 in combination with MATLAB R2010a for the first time and an update is performed by the system, a `Sqrt` block is inserted for each `Math` block that computes the square root function. For details, refer to New TargetLink Sqrt Block on page 145.

### Loop variables in Stateflow
If you model a loop variable in Stateflow (always do .. while, not based on the mechanisms of `LoopsForVectorSignals`), other variable names are now used for the loop variable (TargetLink <= 3.1: `idx, dx`; TargetLink 3.2: `idx1, idx2`).

### Optimization improvements regarding constants
TargetLink 3.2 provides various optimization improvements and simplifications regarding constants. These improvements influence the generated code, for example, with regard to constant folding (a + b; if both operands are constant, they are replaced by their result), elimination of cast operations and utilization of constant initialization values. Depending on the type of optimization, cast operations are eliminated, additional comments are inserted in the generated code or fewer computations are performed on constants.
Comments for constants

Comments for constants that describe physical values can now be switched off via the `cconfig.xml` file. To do so, set the default shown below to false:

```xml
<TL:scalar-const-comment show="false"/>
```

Once you have set `show="false"/`, constant comments no longer appear in the generated code.

Parenthesizing of variables used as operands (address and dereference operators)

Simple variables used as operands of address operators and dereference operators are no longer parenthesized.

<table>
<thead>
<tr>
<th>TargetLink 3.2</th>
<th>TargetLink &lt; 3.2</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>pVar = &amp;var;</code></td>
<td><code>pVar = &amp;{var};</code></td>
</tr>
<tr>
<td><code>var = *pVar;</code></td>
<td><code>var = *(pVar);</code></td>
</tr>
<tr>
<td>For struct components or array elements parentheses are still used.</td>
<td><code>pVar = &amp;{Vector[1]};</code></td>
</tr>
<tr>
<td><code>var = *(Struct.pPointer);</code></td>
<td><code>var = *(Struct.pPointer);</code></td>
</tr>
</tbody>
</table>

Discontinued boards

There is a new evaluation board and evaluation boards that are no longer supported and/or distributed. For further information, refer to Enhancements to the Target Simulation Module on page 146.
### Discontinued tool

Previous versions of TargetLink and the dSPACE Data Dictionary provided you with the `tl_export2dd` tool that let you import TargetLink model data into the dSPACE Data Dictionary. As of TargetLink 3.2, this tool is not part of the TargetLink Base Suite anymore.

If you want to use `tl_export2dd`, you can download it via the TargetLink Product Support Center (www.dspace.com/goto?TargetLinkProductSupportCenter).

### Rate Limiter block

As of TargetLink 3.2, the code generated for the Rate Limiter block reflects the initial value specified at the Simulink block (under the block mask) if the block is run with a discrete sample time. There are two scenarios where the code differs from previous TargetLink versions:

**Migrating from TargetLink 2.x** In TargetLink 2.x, the block library contained the Rate Limiter block with the sample time set to `continuous`. The initial value could not be specified.

After the upgrade to TargetLink 3.2, the sample time is set to `inherited` and the initial value is set to 0. Thus, the Rate Limiter block’s simulation behavior is different from TargetLink 2.x in all simulation modes.

**Migrating from TargetLink 3.x** As of TargetLink 3.2, the block library contains the Rate Limiter block with the sample time set to `inherited` and the initial value is set to 0.

As the code generated for the Rate Limiter block now reflects the initial value if the block is run with a discrete sample time, the Rate Limiter block’s simulation behavior is different from TargetLink 3.x in SIL and PIL simulation modes.

### Cast operations with unary minus and bitwise operations

As of TargetLink 3.2, additional cast operations are applied to unary minus and bitwise operations.

**Unary minus** The operand of a unary minus operation is always cast to the result type.

Additionally, if the operand is unsigned, it is cast to a signed type before the minus operator is applied. This is done to comply with MISRA rule 12.9, which does not allow the unary minus operator to be applied to an expression whose underlying type is unsigned.

If the width of the result type is larger than the width of the operand, the operand is cast to the output width before the minus operator is applied.
TargetLink < 3.2
b = -a;

TargetLink 3.2
b = (Int16)-a;

**Bitwise operations**  A bitwise operation is always cast to the result type.

If the original width is larger than the width of the operand, the operand is cast to the original width before the ~ operator is applied.

With binary bitwise operations, the operands are cast to the result type if both are less wide than the output and are signed differently.

TargetLink < 3.2
d = ~c&4;

TargetLink 3.2
d = (UInt16)((UInt16)(~c)&((UInt16)4));

**Comparisons with Float32 constants**  To ensure that comparisons (<, <=, >, >=) are calculated correctly, Float32 constants that cannot be represented in Float32 are rounded to the next higher or lower number that can be represented in Float32. Whether the constant is rounded up or down depends on the operator and the position of the constant (left- or right-hand side for the comparison expression).

TargetLink < 3.2
F32Var > 2097151.95F

TargetLink 3.2
F32Var > 2097151.875F

**Abs operation**  As of TargetLink 3.2, the code generated for Abs operations is different, if all of the following conditions are fulfilled:

- There are only integer operands in the Abs operation.
- The Abs operation is saturated.
- The Abs argument has a signed data type.
- The Abs argument is scaled with an offset = 0.
The sequence of variable definitions and declarations may change if the model that the code is generated for references variables that are specified in the Data Dictionary and assigned to a DD module object with the property CodeGenerationBasis set to ModelAndDDBased.

Sequence generated with TargetLink 3.1
A2L_None Int16 Axis[3] =
{
/*[0..2]*/ 1, 2, 3
/* 1., 2., 3. */
};
A2L_None UInt16 NumAxisPoints = 3 /* LSB: 2^0 OFF: 0 MIN/MAX: 0
.. 65535 */;

Sequence generated with TargetLink 3.2
A2L_None UInt16 NumAxisPoints = 3 /* LSB: 2^0 OFF: 0 MIN/MAX: 0
.. 65535 */;
A2L_None Int16 Axis[3] =
{
/*[0..2]*/ 1, 2, 3
/* 1., 2., 3. */
};

**AUTOSAR-Related Migration Aspects**

**Changed runnable signature**

Due to the AUTOSAR compiler abstraction, runnable code generated with this version of TargetLink can differ from the runnable code of prior TargetLink versions.

TargetLink now supports the **FUNC**, **P2CONST**, and **P2VAR** macros for functions, variables, and pointers.

**Runnable code of prior TargetLink versions**

```c
void Run(sint16 ScalarIn,
    sint16 ArrayIn[10],
    StructType* StructIn,
    sint16* ScalarOut,
    sint16 ArrayOut[10],
    StructType* StructOut)
```

```c
void Run(sint16 ScalarIn,
    sint16 ArrayIn[10],
    StructType* StructIn,
    sint16* ScalarOut,
    sint16 ArrayOut[10],
    StructType* StructOut)
```
Possible runnable code of this TargetLink version

```c
FUNC(void, RTE_APPL_CODE) Run(sint16 ScalarIn,
P2CONST(sint16, AUTOMATIC, RTE_APPL_DATA) ArrayIn,
P2CONST(StructType, AUTOMATIC, RTE_APPL_DATA) StructIn,
P2VAR(sint16, AUTOMATIC, RTE_APPL_DATA) ScalarOut,
P2VAR(sint16, AUTOMATIC, RTE_APPL_DATA) ArrayOut,
P2VAR(StructType, AUTOMATIC, RTE_APPL_DATA) StructOut)
```

The Rte_Type.h header file contains the compiler abstraction macro definitions. It is included in the software component header file (<SWC/Runnable>.h) if you generate code supporting compiler abstraction. Otherwise the Rte_Type.h header file is included in the software component C-file (<SWC/Runnable>.c).

**Suggested migration** TargetLink generates runnable code including the AUTOSAR compiler abstraction macros only if the FunctionClass of a runnable is unset or AUTOSAR/RUNNABLE.

If you want to generate unchanged runnable code for TargetLink models of versions prior to 3.2, you have to specify a runnable FunctionClass different from AUTOSAR/RUNNABLE such as GLOBAL_FCN. However, the default FunctionClass for runnables of TargetLink versions prior to 3.2 is GLOBAL_FCN.

**Support of MATLAB versions after 2009b**

Previously unsupported MATLAB versions, i.e., MATLAB versions after 2009b might result in an unrecoverable MATLAB crash if you want to simulate models that have been built with TargetLink versions prior to 3.2.

**ReceiverComSpec and SenderComSpec blocks** Since version 3.1 the TargetLink AUTOSAR Block Library contains the ReceiverComSpec and SenderComSpec blocks. S-functions for models with these blocks cannot be run with new MATLAB versions unless built with TargetLink 3.2.

**Suggested migration** Make a new build if you want to work with new MATLAB versions such as 2010b and unchanged TargetLink 3.1 models.

**Improved exchange of data types with an architecture modeling tool**

**Changed Data Dictionary Master template for AUTOSAR**

For improved exchange of software components with an architecture modeling tool the following changes have been made to the dsdd_master_autosar.dd template:

- The TLDataTypes and the DataTypes typedef groups have been added.
- The optimization options of the AUTOSAR variable classes have been changed.
The CodeGenerationBasis property of the AUTOSAR/Rte_Type module object has been changed to ModelAndDDBased.

Do not use or import packages and subpackages named DataTypes such as /DataTypes because the /AUTOSAR/DataTypes package is predefined by AUTOSAR.

During import the Data Dictionary would import elements of the standard AUTOSAR data types (/AUTOSAR/DataTypes) and the imported package or subpackage to the same DD group object named /Pool/Typedefs/DataTypes.

**Suggested migration** To use the improved exchange of software components with an architecture modeling tool with models prior to TargetLink 3.2, you have to perform the following steps:

1. Type `ddman` in MATLAB’s Command Window to open the DD Manager.
2. Open the Data Dictionary of your model in the DD Manager.
3. From the menu of the DD Manager, select View – Show DD Merge Explorer.

   The DD Manager opens a file selection dialog.
4. In the file selection dialog, select the `dsdd_master_autosar.dd` file that is located in the ./Dsdd/Config folder of your installation.

   The DD Manager opens the template file in the DD Merge Explorer pane for you to copy specific items to the Data Dictionary of your model.
5. In the DD Merge Explorer, right-click the Pool/Typedefs node to open its context menu.
6. From the context menu, select Copy left (merge-overwrite).

   The DD Manager merges the typedefs node to your Data Dictionary.
7. In the DD Merge Explorer, right-click the /Pool/VariableClasses/AUTOSAR node to open its context menu.
8. From the context menu, select Copy left (merge-overwrite).

   The DD Manager merges the variable classes to your Data Dictionary.
9. In the DD Merge Explorer, right-click the Modules/TLPredefinedModules/AUTOSAR/Rte_Type node to open its context menu.
10. From the context menu, select **Copy left (merge-overwrite)**. The DD Manager merges the module to your Data Dictionary.

For details on types for AUTOSAR models, refer to *Basics on AUTOSAR-Compliant Types and Scalings* (TargetLink AUTOSAR Modeling Guide).
Compatibility Information

Where to go from here

<table>
<thead>
<tr>
<th>Information in this section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supported MATLAB Releases</td>
<td>190</td>
</tr>
<tr>
<td>Operating System</td>
<td>191</td>
</tr>
<tr>
<td>Limitations for Windows Vista/Windows 7</td>
<td>192</td>
</tr>
<tr>
<td>Limitations for 64-bit Windows Operating Systems</td>
<td>194</td>
</tr>
</tbody>
</table>
Supported MATLAB Releases

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

<table>
<thead>
<tr>
<th>MATLAB Release...</th>
<th>RCP and HIL Software</th>
<th>TargetLink 3.2</th>
<th>Model Compare 2.1</th>
<th>SystemDesk 3.0&lt;sup&gt;1)&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>R2010b</td>
<td>Yes&lt;sup&gt;2)&lt;/sup&gt;</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>R2010a</td>
<td>Yes&lt;sup&gt;2)&lt;/sup&gt;</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>R2009bSP1</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>R2009a</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>R2008b</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>R2008a+</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>R2007b+</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>R2007a+</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>R2006b</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>R2006a+</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

<sup>1</sup> DSOffSim (dSPACE target for offline simulation) installed with SystemDesk requires MATLAB.

<sup>2</sup> Not supported by the RTI FPGA Programming Blockset - FPGA Interface

For up-to-date information on additional MATLAB releases which can be used in combination with dSPACE software, refer to http://www.dspace.com/goto?sw3rdparty.

dSPACE software supports only 32-bit versions of MATLAB. 64-bit MATLAB versions are not supported.

### Operating System

The following table shows which software items in Release 7.0 support which operating system:

<table>
<thead>
<tr>
<th>Operating System...</th>
<th>RCP &amp; HIL Software</th>
<th>TargetLink 3.2</th>
<th>ControlDesk 4.0</th>
<th>Model Compare 2.1</th>
<th>SystemDesk 3.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows XP Professional (32-bit version) with Service Pack 3</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Windows Vista Business, Ultimate, and Enterprise (32-bit version) with the latest Service Pack</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Windows Vista Business, Ultimate, and Enterprise (64-bit version) with the latest Service Pack</td>
<td>Yes(^{4})</td>
<td>Yes</td>
<td>Yes(^{5})</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Windows 7 Professional, Ultimate, and Enterprise (32-bit version)(^{3})</td>
<td>Yes(^{4})</td>
<td>Yes</td>
<td>Yes(^{5})</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Windows 7 Professional, Ultimate, and Enterprise (64-bit version)(^{16})</td>
<td>Yes(^{4})</td>
<td>Yes</td>
<td>Yes(^{5})</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

\(^{1}\) ControlDesk Next Generation (= ControlDesk 4.0) is the successor to CalDesk and ControlDesk. However, ControlDesk versions lower than ControlDesk 4.0 are still included in the RCP and HIL software item.

\(^{2}\) Only the editions Windows Vista Business, Ultimate, and Enterprise are supported. The editions Windows Vista Home and Starter are not supported.

\(^{3}\) The 64-bit operating systems are supported by 32-bit software running in WoW64 (Windows-On-Windows64). 64-bit MATLAB versions are not supported.

\(^{4}\) Not supported by the RTI FPGA Programming Blockset - FPGA Interface

\(^{5}\) ControlDesk Next Generation’s ECU Diagnostics Module is not supported.

\(^{6}\) Only the editions Windows 7 Professional, Ultimate, and Enterprise are supported. The editions Windows 7 Home and Starter are not supported.

- As of dSPACE Release 6.5 Windows 2000 is no longer supported.
- Windows XP Professional x64 Edition is not supported.

#### Limitations for Windows Vista/Windows 7

Some limitations apply when you use Windows Vista or Windows 7 in combination with dSPACE software. Refer to *Limitations for Windows Vista/Windows 7* on page 192.
Notes on 64-bit Windows operating systems  

dSPACE software runs as a 32-bit application under 64-bit Windows operating systems in a WoW64 (Windows-on-Windows 64-bit) subsystem. WoW64 is the x86 emulator that allows 32-bit Windows-based applications to run seamlessly on 64-bit versions of Windows. This allows you to use up to 4 GB virtual memory for each 32-bit process. 32-bit versions of Windows can only address up to 3.2 GB of memory in total for all running processes including the operating system itself.

Some additional limitations apply when you use a 64-bit Windows operating system with dSPACE software. Refer to Limitations for 64-bit Windows Operating Systems on page 194.

Operating system on dSPACE License Server

If you purchased floating network licenses, you have to install and configure one of the networked PCs as the dSPACE License Server.

The operating system of the dSPACE License Server must be one of the following:

- Windows XP Professional (32-bit version) with Service Pack 3
- Windows Vista Business, Ultimate, or Enterprise with the latest Service Pack
- Windows 7 Professional, Ultimate, or Enterprise
- Windows Server 2003

The dSPACE License Server does not support non-Windows operating systems.

Limitations for Windows Vista/Windows 7

Objective

Some limitations apply when you use Windows Vista/Windows 7 in combination with dSPACE software.

MATLAB support

Under Windows 7, the dSPACE software has been released for MATLAB versions since MATLAB R2009a.

For further system requirements of The MathWorks software, refer to http://www.mathworks.com/support/sysreq/current_release.
### Compatibility Information

#### S-function generation

If you use Windows 7 and MATLAB R2010b, the compilation of S-functions stops with error messages. These are caused by a Simulink bug that occurs if the MATLAB installation path contains a space (e.g., C:\Program Files\MATLAB). This bug applies to RTI and TargetLink and can be corrected by installing bugfix 661855 (refer to http://www.mathworks.com/support/bugreports/661855).

Whenever an S-function is compiled using the MEX compiler, the following error might occur:

**Microsoft compilers:**

fatal error C1083: Cannot open include file: 'simstruc.h': No such file or directory

**LCC:**

<filename:lineNumber> Could not find include file "simstruc.h"

**RTI**

The error might occur when S-functions are generated, e.g., by RTICANMM or RTILINMM.

**TargetLink**

In TargetLink, error messages might occur when compiling

- Simulation S-functions for SIL/PIL simulation
- Stand-alone S-functions
- Custom code S-functions

Depending on the selected MEX compiler, LCC or MSVC, there are different compiler messages:

**MSVC:** "fatal error C1083: Cannot open include file: <simstruc.h>: No such file or directory"

**LCC:** lcc preprocessor error: <filename:lineNumber> Could not find include file <simstruc.h>

#### Fast user switching not supported

The dSPACE software does not support the fast user switching feature of Windows Vista and Windows 7.

#### Closing dSPACE software before PC shutdown

The modified shutdown procedure of Windows operating systems causes some required processes to be aborted although they are still being used by dSPACE software. To avoid data loss, it is recommended to terminate the dSPACE software manually before a PC shutdown is performed.
USB devices under Windows 7

The first time dSPACE USB devices using cables with opto-isolation are connected to the PC, there might be a message that the device driver software was not successfully installed. The dSPACE device will nevertheless work properly later on.

Using boards with ISA interface installed in the host PC

Using ISA boards, such as the DS1103 or DS813, installed directly in the host PC, with Windows Vista/Windows 7 is not possible with the standard installation routines. If necessary, contact dSPACE Support.

Allowing communication via additional firewall rules

During installation of the dSPACE software, two additional Windows firewall rules are 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."
```

Limitations for 64-bit Windows Operating Systems

Objective

Some additional limitations apply when you use Windows Vista/Windows 7 (64-bit versions) in combination with dSPACE software.

Limitations of device drivers

Third-party BUS interfaces (CAN, LIN, or FlexRay) are supported only if they are provided with 64-bit drivers from the manufacturers.

Limitations of target compilers

For information on support for a specific target compiler, contact the respective compiler manufacturer.
The following table shows the known limitations of the dSPACE and third-party software under 64-bit Windows operating systems:

<table>
<thead>
<tr>
<th>Software</th>
<th>Limitations and Recommendations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naming of installation path</td>
<td>Under 64-bit Windows operating systems, the default installation folder for 32-bit applications is C:\Program Files (x86). Since parentheses in the path name can cause problems, make sure to choose installation paths without parentheses for the dSPACE software, MATLAB, the dSPACE DS1006 and Microtec PowerPC C compilers.</td>
</tr>
<tr>
<td>AutomationDesk 3.2</td>
<td>The winsound Python module used to create an acoustic signal in the TutorialDemo06 project is not supported.</td>
</tr>
<tr>
<td>MATLAB</td>
<td>If you install a 32-bit version of MATLAB under Windows Vista/Windows 7 (64-bit versions), the MATLAB installation program generates a message that a 64-bit version of MATLAB is available. Click OK and continue installing the 32-bit version of MATLAB. dSPACE software supports only 32-bit versions of MATLAB. 64-bit versions of MATLAB are not supported.</td>
</tr>
</tbody>
</table>
A
address operator 181
ASM Base InCylinder Blockset migrating 31
ASM Diesel Exhaust Blockset migrating 32
ASM Diesel InCylinder Blockset migrating 33
ASM Drivetrain Basic Blockset migrating 34
ASM Electric Components Blockset migrating 36
ASM Engine Diesel Blockset migrating 37
ASM Engine Gasoline Basic Blockset migrating 38
ASM Engine Gasoline Blockset migrating 39
ASM Environment Blockset migrating 41
ASM Gasoline InCylinder Blockset migrating 44
ASM Optimizer migrating 46
ASM Optimizer Blockset migrating 47
ASM Parameterization tool new features 49
ASM Traffic Blockset migrating 50
ASM Trailer Blockset migrating 51
ASM Turbocharger Blockset migrating 54
ASM Vehicle Dynamics Blockset migrating 56
AutomationDesk migration 26
code changes 23

C
changed Code Generator options 176
TargetLink API changes 178
code changes 151
constants 181

D
dereference operator brackets 181
discontinued compiler versions 146
discontinued evaluation boards 146
DiOffSim new features 137
dSPACE Data Dictionary migration 169
dSPACE FlexRay Programming Tool new features 91
dSPACE FlexRay Configuration Package migration 95
new features 93

G
general enhancements and changes 12

H
host PC software operating system 191

K
key features 17
dSPACE Data Dictionary 160

L
limitations for Windows 64-bit 194
limitations for Windows 7 192
limitations for Windows Vista 192

M
MATLAB supported releases 190
MicroAutoBox II migration 102
new features 99
migrating
ASM Base InCylinder Blockset 31
ASM Diesel Exhaust Blockset 32
ASM Diesel InCylinder Blockset 33
ASM Drivetrain Basic Blockset 34
ASM Electric Components Blockset 36
ASM Engine Diesel Blockset 37
ASM Engine Gasoline Basic Blockset 38
ASM Engine Gasoline Blockset 39
ASM Environment Blockset 41
ASM Gasoline InCylinder Blockset 44
ASM Optimizer Blockset 47
ASM Traffic Blockset 50
ASM Trailer Blockset 51
ASM Truck Blockset 53
ASM Turbocharger Blockset 54
ASM Vehicle Dynamics Blockset 56
SystemDesk migration 139

N
new API commands dSPACE Data Dictionary 165
new Code Generator options 149
new evaluation boards 146
new evaluation compiler versions 146
new features
ASM Drivetrain Basic Blockset 34
ASM Electric Components Blockset 35
ASM Environment Blockset 41
ASM Optimizer 46
ASM Parameterization tool 49
ASM Trailer Blockset 51
ASM Truck Blockset 52
ASM Vehicle Dynamics Blockset 56
AutomationDesk 23
ControlDesk 59
ControlDesk Next Generation (ControlDesk 4.0) 62
DiOffSim 137
dSPACE ECU Flash Programming Tool 91
dSPACE FlexRay Configuration Package 93
MicroAutoBox II 99
ModelDesk 97

New Features and Migration November 2010 197
Index

RTI AUTOSAR Package 103
RTI Bypass Blockset 107
RTI CAN MultiMessage Blockset 109
RTI Ethernet (UDP) Blockset 113
RTI FPGA Programming Blockset 115
RTI LIN MultiMessage Blockset 111
RTI/RTI-MP 99
RTLib 99
SystemDesk 120

O
obsolete Code Generator options 176
optimization
improvements regarding constants 180
overview
TargetLink API changes 178

P
product overview 14

R
requirements
host PC software
operating system 191
RTI AUTOSAR Package
new features 103
RTI Bypass Blockset
new features 107
RTI CAN MultiMessage Blockset
migration 110
new features 109
RTI Ethernet (UDP) Blockset
new features 113
RTI FPGA Programming Blockset
migration 116
new features 115
RTI LIN MultiMessage Blockset
migration 112
new features 111
RTI/RTI-MP
new features 99
RTLib
new features 99

S
Stateflow
loop variables 180
supported MATLAB releases 190
supported targets 146
system requirements
operating system 191
SystemDesk
migrating 139
new features 120

T
target support 146
TargetLink API changes 178

V
version history 14

W
Windows 64-bit limitations 194
Windows 7 limitations 192
Windows Vista limitations 192