dSPACE Release

New Features and Migration

Release 2016-A – May 2016

Embedded Success dSPACE

How to Contact dSPACE

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

How to Contact dSPACE Support

To contact dSPACE if you have problems and questions, fill out the support request form provided on the website at http://www.dspace.com/go/supportrequest.

The request form helps the support team handle your difficulties quickly and efficiently.

In urgent cases contact dSPACE via phone: +49 5251 1638-941 (General Technical 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/go/support for software updates and patches.

Important Notice

Contents

About This Document

Contents

This document informs you about the new features of all the dSPACE software products in Release 2016-A. It also gives you an overview of software products with no or minor changes. There are instructions on migrating from earlier dSPACE releases, especially from earlier product versions, if required.

Where to go from here

Information in this section

Conventions Used in the Documentation

Admonitions

The following admonitions may be used in this document.

Admonition	Description
	Indicates a hazardous situation that, if not avoided, will result in death or serious injury.
	Indicates a hazardous situation that, if not avoided, could result in death or serious injury.
	Indicates a hazardous situation that, if not avoided, could result in minor or moderate injury.

Admonition	Description
NOTICE	Indicates a hazard that may cause property damage if you do not avoid it by following the instructions given.
Note	Indicates important information that should be kept in mind, for example, to avoid malfunctions.
Tip	Indicates tips containing useful information to make your work easier.

Naming conventions

The following abbreviations and formats are used in this document:

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

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

📖 Precedes the document title in a link that refers to another document.

Indicates that a link refers to another document, which is available in dSPACE HelpDesk.

Special folders

Some software products, for example, ControlDesk Next Generation and AutomationDesk, use the following special folders:

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

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

Documents folder A standard folder for user-specific documents.

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

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

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

Accessing Online Help and PDF Files

Objective

After you install your dSPACE software, the documentation for the installed products is available as online help and Adobe® PDF files.

Online help

You can access the online help, dSPACE HelpDesk, as follows:

Windows Start menu Select Start – (All) Programs – <ProductName> – dSPACE HelpDesk (<ProductName>) to open dSPACE HelpDesk with the start page of the selected product displayed. You can also navigate and search in the user documentation of any other installed software product and its supported hardware.

Context-sensitive Press the **F1** key or click the Help button in the dSPACE software to get help on the currently active context.

Note

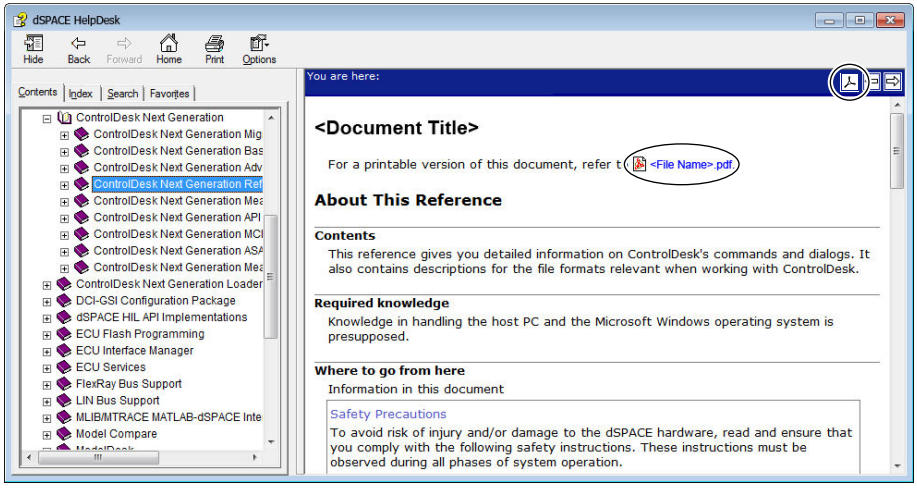
In some software products, context-sensitive help is not available.

Help menu in the dSPACE software On the menu bar, select Help – Contents or Help – Search (not available in all software products) to open dSPACE HelpDesk. It opens at the start page of the currently active product. You can also navigate and search in the user documentation of any other installed software product and its supported hardware.

PDF files

You can access the PDF files as follows:

dSPACE HelpDesk Click the PDF link at the beginning of a document or on a topic pane's header:



Overview of dSPACE Release 2016-A

Objective	Gives you an overview of the new key features in Release 2016-A and information about unchanged products.
Where to go from here	Information in this section

General Enhancements and Changes

Objective	The following new features and changes concern several dSPACE products.
Support of new dSPACE hardware	With dSPACE Release 2016-A, new dSPACE hardware is introduced: ■ MicroAutoBox: ■ DS1554 Engine Control I/O Module The I/O module can be mounted on a DS1514 I/O Board. The specific engine control I/O features, such as crankshaft, camshaft and knock sensors, are supported by the RTI FPGA

Programming Blockset. Refer to *New Features of the RTI FPGA Programming Blockset 3.1* on page 165.

- SCALEXIO:
  - SCALEXIO LabBox

    Providing 19 slots for up to 18 standard SCALEXIO I/O boards plus one DS6051 IOCNET Router.
  - DS6051 IOCNET Router

    Required for connecting SCALEXIO LabBox to a SCALEXIO Processing Unit as the computation node.
  - DS6301 CAN/LIN Board

    Providing 4 CAN/CAN FD channels and 4 LIN channels.

Contents of DVDs

The dSPACE software is provided on two disks. The disks contain the following dSPACE software packages and main products:

- Disk 1:
  - AutomationDesk 5.2
  - ControlDesk Next Generation (ControlDesk 5.6)
  - TargetLink 4.1
  - Model Compare 2.6

    Note

    Product use prohibited in United States

    You are not licensed to use Model Compare in the United States. You are not allowed to use or permit others to use this product in the United States or in any way that violates the laws of the United States.
  - SystemDesk 4.6 (supports AUTOSAR 4.x)
  - VEOS 3.6
  - Various other dSPACE software tools
- Disk 2:
  - RCP and HIL software

    RCP and HIL software is a generic term for a software package containing several dSPACE software products, such as RTI, ConfigurationDesk, MotionDesk, and ModelDesk.

Tip

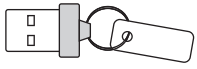
Disk 2 does not contain any other dSPACE software products.

New hardware dongles for dongle licenses

As of dSPACE Release 2014-B, the hardware dongle for dongle licenses is now a CmDongle instead of a WibuKey dongle. Both are products of WIBU-SYSTEMS and are shown below.

WibuKey dongle

CmDongle



With dSPACE Release 2014-B, the new CmDongles are shipped with new dSPACE systems for the first time.

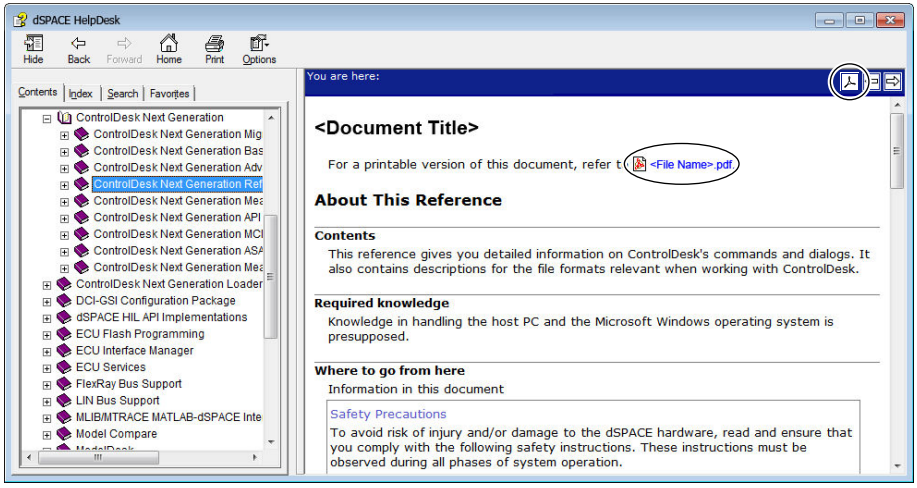
Keep the following compatibility information in mind:

- In general, you can use dSPACE Release 2016-A with an already delivered WibuKey dongle. As of dSPACE Release 2014-B, the drivers for both dongle versions are installed on your host PC. The driver software automatically detects which dongle is used. No further user action is necessary.
- If you want to use dSPACE Release 2014-A and earlier with the new CmDongle, you have to install dSPACE Installation Manager 3.8 (or later) on your host PC. This version contains the driver for the new dongle. You can download the latest version of dSPACE Installation Manager from http://www.dspace.com/go/imupdate.
- dSPACE Release 6.3 and earlier versions have not been tested for the new CmDongle. If necessary, contact dSPACE Support.

Improved user documentation

The documents in dSPACE HelpDesk are also available in PDF format. With dSPACE Release 2016-A you have now a faster access to these PDF files. If there is a PDF file for the currently opened topic, just click in the topic pane's header to open it.

For topics that are not newly published, you have still to navigate to the front page of the online book. Here you will find the hyperlink to the related PDF file.



Restrictions when working with dSPACE HelpDesk

dSPACE HelpDesk is installed in release-specific folders in `C:\Program Files(x86)\Common Files\dSPACE`. For example, if you have installed products from dSPACE Release 2015-B and products from dSPACE Release 2016-A, two dSPACE HelpDesks are available.

Note the following restrictions:

Links to documents might not work and might return the following error message: *Selection is not associated with any topics*. The possible reasons are:

- The documents for the product are not installed, because the product is not included in your license key.
- The documents for the product are installed in another dSPACE HelpDesk. For example, if a product in the current dSPACE Release has not been changed, its user documentation is installed in the dSPACE HelpDesk version that the product setup was created for.

  After you install dSPACE Release 2016-A, you can find the user documentation in dSPACE HelpDesk 2015-B for the following products:

  - Container Manager 4.4
  - Model Compare 2.6
  - TargetLink 4.1

  If you are not sure where to find the user documentation for your product, use the product-specific dSPACE HelpDesk shortcut in the Windows Start menu to open the online help.

Printed user documentation

With dSPACE Release 2016-A, the printed user documentation is not delivered automatically. You can now decide which of the available printed documents you want to have. To order printed documentation, refer to http://www.dspace.com/go/requestreleasematerial.

Note

If you do not order printed documentation, use dSPACE HelpDesk or PDF files to obtain information about new features, enhancements, and the safety precautions regarding your products.

Software support discontinuation

Discontinuation of 32-bit software support With dSPACE Release 2016-A, dSPACE software supports only 64-bit operating systems and only 64-bit MATLAB variants.

Discontinuation of MicroAutoBox software support dSPACE Release 2015-B was the last release supporting MicroAutoBox with its variants 1401/1501, 1401/1504, 1401/1505/1506, 1401/1505/1507, and 1401/1507.

As of dSPACE Release 2016-A, dSPACE software supports only MicroAutoBox II with its variants 1401/1507, 1401/1511, 1401/1513, 1401/1511/1514, and 1401/1513/1514.

It is no longer possible to register a MicroAutoBox. If you register a MicroAutoBox using dSPACE software from dSPACE Release 2015-B or earlier, and you try to access it via software from dSPACE Release 2016-A, this does not work.

For further product-specific migration instructions, refer to:

- *Migrating to ControlDesk Next Generation (ControlDesk 5.6)* on page 114
- *Migrating to dSPACE HIL API .NET 2.1* on page 129
- *Migrating to dSPACE Python Extensions 2.1* on page 131
- *Migrating to dSPACE XIL API 2016-A* on page 137
- *Migration Aspects of RTI/RTI-MP and RTLib* on page 156

Planned discontinuation of dSPACE software

Discontinued test automation software for platform access
The following test automation software for platform access is being distributed for the last time with dSPACE Release 2016-A:

- rtplib2

  With dSPACE Release 2016-B, Test Automation Python Modules will no longer provide the `rtplib2` Python module. The `matlablib2` and `rs232lib2` Python modules are still available.

- HIL API MAPort

  With dSPACE Release 2016-B, the dSPACE Python implementation and the dSPACE .NET implementation of the HIL API MAPort will no longer be available.

You can migrate your test automation projects to ASAM XIL API as the HIL API successor. The migration from HIL API .NET to XIL API .NET requires only a few modifications in your application. Refer to *Migrating HIL API Applications to XIL API Applications* (🕮 *dSPACE XIL API Implementation Guide*). For the migration of HIL API Python and rtplib2 to XIL API .NET, you can use your new .NET application from within Python via PythonNet. For more information on migration, refer to the Test Automation Tools Support Center: http://www.dspace.com/go/pscta.

DTS V7 support in AutomationDesk The Remote Diagnostics (COM) library is supporting DTS V7 for the last time with dSPACE Release 2016-A. You should migrate your projects to ControlDesk as the diagnostic tool.

For help on migration, refer to http://www.dspace.com/go/pscta.

Discontinued methods for handling messages As of dSPACE Release 2016-B, all dSPACE products use improved methods for handling messages such as errors and warnings.

As a consequence, messages will then no longer be written to the `dSPACE.log` file. This means that they will no longer be available as ASCII text.

To collect diagnostics information and send it to dSPACE Support, you have to use the dSPACE Installation Manager.

Planned discontinuation of dSPACE hardware

DS1005 PPC Board You can still buy the product up to December 2016. New releases of dSPACE software are guaranteed to continue supporting the DS1005 until at least the end of 2019.

However, for new projects we recommend that you use the successor, the dSPACE DS1007 PPC Processor Board.

DS1103 PPC Controller Board You can still buy the product up to December 2016. New releases of dSPACE software are guaranteed to continue supporting the DS1103 until at least the end of 2018.

However, for new projects we recommend that you use the successor, dSPACE MicroLabBox.

MicroAutoBox II 1401/1511/1512 and MicroAutoBox II 1401/1512/1513 You can still buy the MicroAutoBox variants with DS1512 I/O Board up to December 2016. New releases of dSPACE software are guaranteed to continue supporting these MicroAutoBox variants until at least the end of 2019. However, for new projects we recommend that you use the successors MicroAutoBox II 1401/1511/1514 or MicroAutoBox II 1401/1513/1514.

Product Version Overview

Objective

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

Product	dSPACE Release			
	2014-B	2015-A	2015-B	2016-A
AutomationDesk	4.1	5.0	5.1	5.2 Refer to *AutomationDesk* on page 43.
Automotive Simulation Models	7.0	8.0	8.1	8.2 Refer to *Automotive Simulation Models (ASM)* on page 49.
NEW: Bus Manager (stand-alone)	–	–	–	5.5 Refer to *Bus Manager (Stand-Alone)* on page 81.
ConfigurationDesk	5.2	5.3	5.4	5.5 Refer to *ConfigurationDesk* on page 83.
Container Manager	4.3	4.3	4.4	4.4

Product	dSPACE Release			
	2014-B	2015-A	2015-B	2016-A
ControlDesk Next Generation	5.3	5.4	5.5	5.6 Refer to *ControlDesk Next Generation* on page 93.
DCI Configuration Tool	3.3	3.4	3.5	3.6 Refer to *DCI Configuration Tool* on page 121.
dSPACE CAN API	2.7.1	2.7.1	2.7.4	2.7.5
dSPACE ECU Flash Programming Tool	2.2.5	2.2.6	2.2.6	2.3 Refer to *dSPACE ECU Flash Programming Tool* on page 123.
dSPACE FlexRay Configuration Package	3.4	3.5	3.6	3.7 Refer to *dSPACE FlexRay Configuration Package* on page 125.
dSPACE HIL API .NET	1.6	1.8	2.0	2.1 Refer to *dSPACE HIL API .NET* on page 129.
dSPACE Python Extensions	1.7	1.8	2.0	2.1 Refer to *dSPACE Python Extensions* on page 131.
dSPACE XIL API	2.0	2015-A	2015-B	2016-A Refer to *dSPACE XIL API* on page 135.
ECU Interface Manager	1.5	1.6	1.7	1.8
Firmware Manager	1.2	1.3	2.0	2.1 Refer to *Firmware Manager* on page 141.
Model Compare	2.5	2.5	2.6	2.6
ModelDesk	4.0	4.1	4.2	4.3 Refer to *ModelDesk* on page 143.
Model Interface Package for Simulink	–	3.0	3.1	3.2 Refer to *Model Interface Package for Simulink* on page 147.

Product	dSPACE Release			
	2014-B	2015-A	2015-B	2016-A
MotionDesk	3.5	3.6	3.7	3.8 Refer to *MotionDesk* on page 149.
MotionDesk Blockset	2.3.1	2.3.2	2.4	2.4.1 Refer to *MotionDesk* on page 149.
Real-Time Testing	2.4	2.5	2.6	3.0 Refer to *Real-Time Testing* on page 153.
RTI[1)]	7.3	7.4	7.5	7.6 Refer to *RTI/RTI-MP and RTLib* on page 155.
RTI-MP[2)]	7.3	7.4	7.5	7.6 Refer to *RTI/RTI-MP and RTLib* on page 155.
RTI Bypass Blockset	3.3	3.4	3.5	3.6 Refer to *RTI Bypass Blockset* on page 159.
RTI CAN Blockset	3.3	3.4	3.4.1	3.4.2
RTI CAN MultiMessage Blockset	4.0	4.1	4.2	4.3 Refer to *RTI CAN MultiMessage Blockset* on page 161.
RTI Electric Motor Control Blockset	1.0	1.1	1.2	1.3 Refer to *RTI Electric Motor Control Blockset* on page 163.
RTI Ethernet Blockset	1.1	1.2	1.2	1.2
RTI Ethernet (UDP) Blockset	1.3	1.4	1.4	1.4
RTI FPGA Programming Blockset	2.8	2.9	3.0	3.1 Refer to *RTI FPGA Programming Blockset* on page 165.
RTI LIN MultiMessage Blockset	2.4	2.5	2.5.1	2.6 Refer to *RTI LIN MultiMessage Blockset* on page 169.
RTI RapidPro Control Unit Blockset	2.2	2.2.1	2.2.1	2.2.1
RTI USB Flight Recorder Blockset	1.2	1.2	1.2	1.2

Product	dSPACE Release			
	2014-B	2015-A	2015-B	2016-A
RTI Watchdog Blockset	1.0	1.0	1.0	1.0
SCALEXIO firmware	3.1	3.2	3.3	3.4 Refer to *SCALEXIO Firmware* on page 171.
SYNECT server	1.4	1.4.1	1.4.1	1.4.1
SystemDesk	4.3	4.4	4.5	4.6 Refer to *SystemDesk* on page 173.
TargetLink/TargetLink Data Dictionary	4.0	4.0	4.1	4.1
Variable Editor	1.8	2.1	2.2	2.3
VEOS	3.3	3.4	3.5	3.6 Refer to *VEOS* on page 181.

1) Including the standard I/O blocksets.
2) Including the RTI Gigalink Blockset.

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

New Product Key Features

Objective

This is an overview of each product's new key features. For more information, refer to the product-specific sections.

Information in this topic

AutomationDesk

The new key features of AutomationDesk are:

- Enhancements to the user interface, such as new options for the initial collapse state of projects and libraries, and the display of Python modules and Python packages in the Library Browser.
- New IsNotEqual automation block in the Evaluation library.
- The interfaces of the Main Library blocks can be enhanced with any type of data objects.
- Enhancements to the COM API, such as element highlighting and demo projects in C#.
- Enhancements to the user documentation.

For more information on the new features, refer to *New Features of AutomationDesk 5.2* on page 43.

NEW: Bus Manager (stand-alone)	The Bus Manager now also is available as a stand-alone tool. The *Bus Manager (stand-alone)* lets you configure CAN and LIN bus communication and generate bus simulation container (BSC) files for bus simulation on VEOS. For more information, refer to *Features of the Bus Manager (Stand-Alone) 5.5* on page 81.
ConfigurationDesk (Implementation Version)	The new key features of ConfigurationDesk are: ■ Generation of bus simulation container (BSC) files for bus simulation on VEOS ■ Generation of working views based on project topologies (e.g., model topology, hardware topology) ■ Enhanced Properties Browser ■ Extended function blocks: Current Signal Capture, CAN ■ Support of new hardware: DS6301 CAN/LIN Board, SCALEXIO LabBox For more information , refer to *ConfigurationDesk – Implementation* on page 84.

ControlDesk Next Generation	The new key features of ControlDesk 5.6 are:
	■ Platform/device enhancements:
	  ■ CAN/LIN channels of SCALEXIO and VEOS as bus interfaces
	  ■ LIN Bus Monitoring device: Support for FIBEX and AUTOSAR system description files
	  ■ LIN Bus Monitoring device: Variable observer functionality
	  ■ SCALEXIO and DS1007 platforms: Naming processing unit/processor boards during registration
	  For more information on the new features, refer to *New Features of Platform Management and Platforms/Devices (ControlDesk 5.6)* on page 94.
	■ Variable management enhancements:
	  ■ Support of A2L 1.7 files
	  For more information on the new features, refer to *New Variable Management Features (ControlDesk 5.6)* on page 96.
	■ Layouting and instrument enhancements:
	  ■ Visualizing entire struct arrays
	  ■ Custom instrument properties

- Instrument-specific enhancements to the following instruments:
  - Browser
  - Index Plotter
  - Time Plotter
  - XY Plotter

  For more information on the new features, refer to *New Layouting Features (ControlDesk 5.6)* on page 97 and *New Instrument Features (ControlDesk 5.6)* on page 97.

- Measurement and recording enhancement:
  - Bookmark when invoking an XIL API EESPort manual trigger

  For more information on the new features, refer to *New Measurement and Recording Features (ControlDesk 5.6)* on page 99.

- Data set management enhancement:
  - Filtering the Data Set Manager's parameter list *directly*

  For more information on the new features, refer to *New Data Set Management Features (ControlDesk 5.6)* on page 99.

- Bus Navigator enhancements:
  - CAN/CAN FD/LIN bus monitoring support for SCALEXIO and VEOS
  - LIN Bus Monitoring device: Support for FIBEX and AUTOSAR system description files
  - Support of AUTOSAR system template version 4.2.2
  - Specifying a default format for IDs and data
  - Bus instrument generation (for CAN and LIN) for VEOS
  - Bus Instrument (RX Type for CAN): Enhancement for RTI CAN MultiMessage Blockset applications
  - Bus Instrument (TX Status for CAN and LIN) for Bus Manager applications
  - Bus Instrument (TX Type for CAN and LIN): Enhancement for Bus Manager applications
  - Bus Instrument (RX Type for CAN and LIN): Enhancement for Bus Manager applications
  - Bus Navigator controlbar enhancements

  For more information on the new features, refer to *New Bus Navigator Features (ControlDesk 5.6)* on page 100.

- ECU Diagnostics enhancements:
  - Automating the execution of ECU flash sessions
  - Configuring the job execution for security access via the GUI
  - Executing the protocol-specific TesterPresent service
  - Evaluation of the suppress positive response bit during ECU connection checks
  - Diagnostics variables: Descriptions for some block group types

  For more information on the new features, refer to *New ECU Diagnostics Features (ControlDesk 5.6)* on page 103.
- Electrical error simulation (failure simulation) enhancements:
  - Monitoring the switching behavior of failure simulation hardware
  - Automating EESPort configurations
  - Specifying whether ControlDesk disconnects a concurrent client
  - Enhanced EESPort Configurations controlbar
  - Bookmark when invoking an XIL API EESPort manual trigger
  - Replacing a signal in all error sets in a single step

  For more information on the new features, refer to *New Electrical Error Simulation Features (ControlDesk 5.6)* on page 105.
- Automation enhancements:
  - Automating the execution of ECU flash sessions
  - Automating EESPort configurations
  - Accessing open and closed documents via the component-specific interface

  For more information on the new features, refer to *New Automation Features (ControlDesk 5.6)* on page 106.
- Further enhancements:
  - Improved user documentation

  For more information on the new features, refer to *Further Enhancements with ControlDesk Next Generation (ControlDesk 5.6)* on page 107.

DCI Configuration Tool

The new key feature of the DCI Configuration Tool is:

- Saving support report to text file with the command line interface

	For more information on the new features, refer to *New Features of the DCI Configuration Tool 3.6* on page 121.
dSPACE ECU Flash Programming Tool	The new key features of the dSPACE ECU Flash Programming Tool are: ■ CAN FD support ■ Support of the DCI-CAN2 interface For more information on the new feature, refer to *New Features of the dSPACE ECU Flash Programming Tool 2.3* on page 123.
dSPACE FlexRay Configuration Package	The new key features of the dSPACE FlexRay Configuration Package are: ■ New transmission mode for cyclic IPDU transmission on the basis of LPDU timing ■ Support of minimum delay time The new key features of the dSPACE FlexRay Configuration Tool are: ■ Support of AUTOSAR System Template 4.2.2 ■ New filter conditions in the property filter ■ Changed parameterization of unused bits of multiplexed PDUs ■ Reduced number of required hardware resources for FlexRay configurations based on dual-channel communication cluster files (SCALEXIO) For more information on the new features, refer to *New Features of dSPACE FlexRay Configuration Package 3.7* on page 125.
dSPACE XIL API	The new key features of dSPACE XIL API are: ■ Enhanced functionalities to the MAPort configuration ■ Enhanced functionalities to the EESPort for monitoring the switching behavior of your electrical error simulation hardware For more information on the new features, refer to *New Features of dSPACE XIL API 2016-A* on page 135.
Firmware Manager	The new key feature of the Firmware Manager is: ■ User interface changed to ribbon style ■ Support of the new SCALEXIO hardware For more information on the new features, refer to *New Features of Firmware Manager 2.1* on page 141.

ModelDesk	The new key features of ModelDesk are: ■ Road Generator creates construction sites ■ Road Generator can import OpenDRIVE 1.4 files ■ Maneuver Editor supports "sine with dwell" steering. ■ Projects and experiments can be renamed ■ Processing supports calculation of measurement data variables ■ Importing elements to the Pool via tool automation For more information on the new features, refer to *New Features of ModelDesk 4.3* on page 143.
MotionDesk	The new key features of MotionDesk are: ■ Highly improved duration for generating road networks ■ Visualizing construction sites ■ New observer that can be controlled in ModelDesk ■ Comfort features, such as copy and paste of observers For more information on the new features, refer to *New Features of MotionDesk 3.8* on page 149.
Real-Time Testing	The new key features of Real-Time Testing are: ■ The Python interpreter is based on Python 2.7.10 ■ New rttlib.dscanapilib module for accessing CAN or CAN FD buses of a SCALEXIO system or VEOS ■ Support of the new DS6301 CAN/LIN Board For more information on the new features, refer to *New Features of Real-Time Testing 3.0* on page 153.
RTI, RTI-MP and RTLib	The new key feature of RTI, RTI-MP and RTLib is: ■ Support of MATLAB R2016a For more information on the new feature, refer to *New Features of RTI/RTI-MP and RTLib* on page 155.
RTI CAN MultiMessage Blockset	The new key features of the RTI CAN MultiMessage Blockset are: ■ Support of SCALEXIO systems with a DS6301 CAN/LIN Board ■ Support of AUTOSAR System Template 4.2.2 as database files For more information on the new features, refer to *New Features of the RTI CAN MultiMessage Blockset 4.3* on page 161.

RTI Electric Motor Control Blockset	The new key feature of the RTI Electric Motor Control Blockset is: ■ Support of SSI interface-based encoders for position measurement For more information on the new features, refer to *New Features of RTI Electric Motor Control Blockset 1.3* on page 163.
RTI FPGA Programming Blockset	The new key features of the RTI FPGA Programming Blockset are: ■ Extended Xilinx® software support ■ New FPGA frameworks for the DS1554 Engine Control I/O Module. ■ Enhancements to the FPGA framework for a DS2655 FPGA Base Board ■ Enhancement to the FPGA frameworks for the DS2655M1 Multi-I/O Module. For more information on the new features, refer to *New Features of the RTI FPGA Programming Blockset 3.1* on page 165.
RTI LIN MultiMessage Blockset	The new key features of the RTI LIN MultiMessage Blockset are: ■ Support of SCALEXIO systems with a DS6301 CAN/LIN Board ■ Support of AUTOSAR System Template 4.2.2 as database files ■ Support of J2602-compliant AUTOSAR system description files For more information on the new features, refer to *New Features of the RTI LIN MultiMessage Blockset 2.6* on page 169.
SCALEXIO firmware	The new key features of the SCALEXIO firmware are: ■ Support of the new DS6301 CAN/LIN Board ■ Support of the new SCALEXIO LabBox ■ Support of SCALEXIO Real-Time PC with a Intel® Xeon® Processor E3-1275 v3 For more information on the new features, refer to *New Features of the SCALEXIO Firmware 3.4* on page 171.
SystemDesk	The new key features of SystemDesk 4.6 are: ■ Support of AUTOSAR 4.2.2, 4.2.1, 4.1.3, 4.1.2, 4.1.1, 4.0.3, and 4.0.2. ■ Creating V-ECUs from external code For more information on the new features, refer to *New General Features* on page 174.

VEOS

The new key features of VEOS are:

- Support for CAN/LIN restbus simulation and import of bus simulation container (BSC) files
- Monitoring bus communication using ControlDesk's Bus Navigator
- Manipulating bus messages
- Displaying and editing bus simulation elements
- C++ support

For more information on the new features, refer to *New Features of VEOS 3.6* on page 181.

Aspects of Migrating from Previous Releases

Objective	After you install products of the current dSPACE Release, some additional steps might be necessary. The migration steps required when you come from the last dSPACE Release are described in the product-specific migration topics in this document. If you come from an older dSPACE Release, refer to the related *New Features and Migration* document.

Migrating to dSPACE Release 2016-A

Objective	After you install Release 2016-A, some additional steps might be necessary.
Migrating from dSPACE Release 2015-B	**Product-specific migration steps** Product-specific migration steps are usually performed automatically by the products. For exceptions, refer to the product-specific migration descriptions.
Migrating from dSPACE Release 2015-A or earlier	To migrate from dSPACE Release 2015-A or earlier to Release 2016-A, you also have to perform the migration steps of the intervening dSPACE Releases. All of the required migration steps can be performed with Release 2016-A installed. For more information on the required migration steps, refer to the *New Features and Migration* documents of the intervening dSPACE Releases.

Previous release documents

The PDF files of previous releases are called `NewFeaturesAndMigrationxx.pdf`, where xx stands for the release number.

You can find the *New Features and Migration* files for previous releases at the following locations:

- In the installation folder of the current dSPACE HelpDesk. Refer to `C:\Program Files (x86)\Common Files\dSPACE\HelpDesk 2016-A\Print\PreviousReleases`
- On the dSPACE DVDs. Refer to `\Doc\Print\PreviousReleases`
- At www.dspace.com/go/migration for download. Here you can also find *New Features and Migration* documents for very early releases.

Changes to TRC File Generation

Where to go from here

Information in this section

Basics on the TRC File Changes

Objective

The enhanced code generation leads to improvements for the simulation behavior of the executable application. To profit from these improvements in dSPACE software, the TRC file generation was enhanced.

Enhancements in the generated TRC file

With the MATLAB/Simulink R2014a release, the enhanced code generation by Simulink® Coder™ was introduced to optimize the simulation behavior. It provides a simpler behavior for tuning all parameters and support for referenced models. Additional Simulink Coder functions introduced with MATLAB R2015b now allow dSPACE to fully support these new features via the enhanced TRC file generation.

The main advantages of the enhanced TRC file generation are:

- Same view of model parameters in MATLAB workspace and TRC file

  All tunable model parameters defined by MATLAB workspace variables are available in the top-level Tunable Parameters group in the TRC file. This lets you access global parameters very quickly and independently of the model hierarchy. Modifying the model hierarchy later on will not affect the variable path already specified for layout connections or test scripts.

- Working with MATLAB structures

  If a MATLAB structure is tunable according to the Simulink Coder rules, the structure levels and structure fields are generated into the code.

  This means:

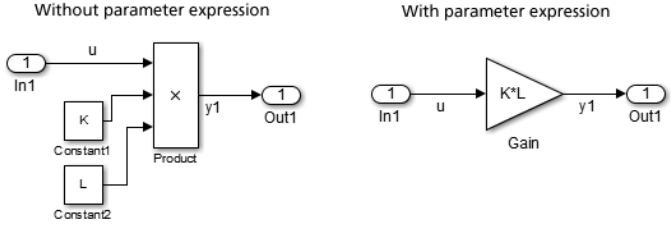
  - Structured parameters are available in the TRC file
  - Non-virtual Simulink buses are represented more efficiently in the TRC file
  - Bus arrays are available in the TRC file

- Higher performance

  For non-virtual Simulink buses, the performance of code generation and compiling will be highly increased.

- More compact models by using tunable parameter expressions

  Complex workarounds for modeling parameter expressions can be simplified, for example, as shown in the model below. The MATLAB workspace variables K and L are automatically generated as tunable parameters.



- Handling of global parameters for Default parameter behavior = Tunable or Inlined (formerly Inline Parameters option off and on)

The mapping between the configured tunable workspace variables and Simulink.Parameter objects, and variables in the generated code does not depend on the Default parameter behavior option (formerly Inline Parameters option).

- Improved model referencing support

  Simulink referenced models were restricted to using the Inline Parameters option set to On (MATLAB R2015b and later: Default parameter behavior set to Inlined). Now the dSPACE tool chain also supports the Default parameter behavior option set to Tunable for referenced models when MATLAB R2015b or later is used.

- Support of Simulink mask parameters

  Simulink mask parameters are now available in the TRC file and can be accessed by dSPACE software, such as ControlDesk Next Generation.

- Same behavior of Simulink simulation and simulations running on dSPACE platforms

  As a result of the above mentioned enhancements for consistent parameter tuning, the behavior of a Simulink simulation and a simulation on dSPACE platforms will be the same.

Support of Simulink Coder enhancements

For the support of the coder enhancements in the generated TRC file, MathWorks and dSPACE together developed additional build functionality which was released with MATLAB R2015b. The resulting additions of the TRC file syntax required complex modifications in all the TRC file-generating and TRC file-consuming dSPACE products.

The full support of these enhancements is realized as of dSPACE Release 2015-B used with MATLAB R2015b. If you use dSPACE Release 2015-B or later with an earlier MATLAB version, the code generation mainly remains the same as with earlier dSPACE releases.

No migration is required if you change the dSPACE Release but keep the MATLAB Release.

For an overview of the different behavior when using the current dSPACE Release, refer to the following table:

dSPACE Release 2016-A used with MATLAB ...			
R2014b	**R2015a**	**R2015b**	**R2016a**
Sharing the same parameter variable across multiple blocks is supported for block parameters that are defined with an *unstructured* MATLAB workspace variable and *without* expressions. All other block parameter definitions have the same behavior as with the previous MATLAB releases R2013b or R2014a.		Full support of the above mentioned Simulink Coder features.	
Internal adaptions to the coder changes are done automatically.		The standard Simulink Coder behavior is used.	
Using the Inline Parameters option set to `Off` for referenced models is not supported.		Using the Inline Parameters option set to `Off` for referenced models is supported.[1)]	

[1)] As of MATLAB R2015b this setting is similar to the Default parameter behavior option set to `Tunable`.

Details on the TRC file changes introduced with MATLAB R2015b

The following changes were made with dSPACE Release 2015-B and MATLAB R2015b.

Model Root group The entries in the Model Root group have changed as follows:

- To improve performance and usability, entries for virtual Simulink buses and muxed signals (e.g., `Out1{SubArray1}`) are no longer generated into the variable description.

  This also applies to the labels of these signals.

  This is an incompatible change that requires manual migration. Refer to *Migrating Changes in Software That Generates TRC Files* on page 39.

- Entries for non-virtual Simulink buses are now generated as one structured variable in the variable description, e.g., `Out1{MyField}` has changed to `Out1.MyField`.

  This also applies to the labels of non-virtual Simulink buses.

  This is an incompatible change that requires manual migration. Refer to *Migrating Changes in Software That Generates TRC Files* on page 39.

- Simulink mask parameters are now generated into the variable description at the entries of the related masked subsystems.

- Input signals of signal sink blocks are now generated into the variable description also when you use ConfigurationDesk or VEOS for the build process.
- The Include states and Include derivatives options are now also available for ConfigurationDesk and VEOS.

Tunable Parameters group The entries in the Tunable Parameters group have changed as follows:

- MATLAB workspace variables and Simulink.Parameter objects, which are used as block parameters in the model, are now generated as global variables in the Tunable Parameters group. Internal optimizations during code generation might be the reason that a variable will not be generated into the variable description.

  If a block's parameter definition contains an expression, the local block parameter is no longer available. This is an incompatible change that requires manual migration. Refer to *Migrating Changes in Software That Generates TRC Files* on page 39.
- Structured workspace variables and Simulink.Parameter objects that are used as block parameters in the model are now generated as global structured parameters in the Tunable Parameters group.

  The structure has to fulfill the Simulink Coder conditions for a tunable structured parameter.
- Previously, each referenced model of a model referencing hierarchy had its own Tunable Parameters group. These groups are no longer generated.

  All global parameters referenced in the top-level model or in the referenced models are generated into the Tunable Parameters group of the top-level model.

  This is an incompatible change that requires manual migration. Refer to *Migrating Changes in Software That Generates TRC Files* on page 39.

Handling n-D look-up tables As of dSPACE Release 2015-B Look-Up Table blocks with a dimension higher than 2, such as a 4x3x2 matrix, are no longer automatically divided into two-dimensional slices.

This is an incompatible change that requires manual migration. Refer to *Migrating Changes in Software That Generates TRC Files* on page 39.

Data Stores group To improve performance and data consistency with other blocks, the Data Stores group is no longer generated into the variable description.

This is an incompatible change that requires manual migration. Refer to *Migrating Changes in Software That Generates TRC Files* on page 39.

Structured variables Structured variables, such as non-virtual buses or tunable structured parameters, are generated into the code and represented in the variable description as a `struct` element. The hierarchy of fields and members in a structured element is described in dot notation, for example, `myStruct.mySubstruct.myValue[0]`.

References A variable description now contains block parameters as references. The source of a reference can be a global parameter: e.g., a MATLAB workspace variable available in the Tunable Parameters group, or a mask parameter. For structured parameters, the reference can specify a field of a structure.

Note

For the support of structures and references, the following keywords have been added to the TRC file syntax:

- `array-incr`
- `offs`
- `struct`
- `endstruct`
- `refvar`
- `refgroup`
- `refelem`
- `DEPRECATED`

If you have used one of these keywords as a variable name, it is detected during the generation of the TRC file and not added to the file. There might be definitions in user code that you must check. Otherwise, there might be an error in the software that uses the TRC file.

Up-to-date information

For more information on TRC file generation and the latest migration instructions, refer to the dSPACE website: http://www.dspace.com/go/trc.

Migrating Changes in Software That Generates TRC Files

Objective	Despite the complex changes in the code generation, only a few manual migrations are required. Most of the changes based on the enhancements are automatically migrated by the dSPACE products.
Using MATLAB R2014b with dSPACE Release 2016-A	There is no manual migration required. The variable description contains further global parameters in the Tunable Parameters group for unstructured workspace variables. These global parameters are shared with the corresponding block parameters if the block parameter is not defined with an expression. Writing a new value to one of the global parameters changes the related block parameters too. Using the Inline Parameters option set to `Off` for referenced models is not supported.
Using MATLAB R2015a with dSPACE Release 2016-A	Same notes as with MATLAB R2014b.
Using MATLAB R2015b with dSPACE Release 2016-A	If you use MATLAB R2015b, the new Simulink Coder features are fully supported. The incompatible changes require migration steps that are described below in general. Detailed instructions are not given, because they depend on various conditions such as the complexity of your model, the software you are using, and the internal structure of your test scripts, for example. There are therefore only some basic examples to show a general way to migrate. For more information, refer to http://www.dspace.com/go/trc.
Using MATLAB R2016a with dSPACE Release 2016-A	Same notes as with MATLAB R2015b.
Migration steps required in TRC file generating software	dSPACE products that generate TRC files such as RTI, ConfigurationDesk and VEOS, support the new Simulink Coder enhancements as is. There are only the following changes that might require a manual migration to provide information in the variable description. **Update assertion mode (only RTI)** The rtiAssertionMode variable is no longer generated into the variable description. The Assertion mode setting on the RTI simulation options page is still available for configuring the mode before you start the build process.

Update access to Data Stores group The Data Stores group is no longer generated into the variable description. Instead of using Data Store Memory blocks, you have to use Data Store Read blocks for read access or the combination of Constant blocks with Data Store Write blocks for write access. Instead of the entries in the Data Stores group you then find entries of the Data Store Read blocks or the Constant blocks in the Model Root group.

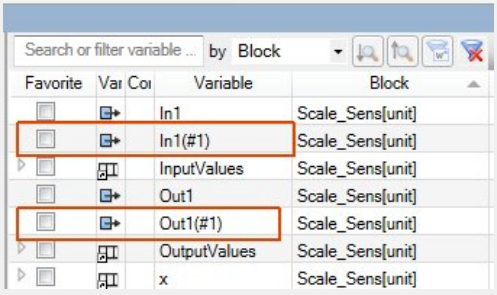
For this migration step, it is not required to use dSPACE Release 2015-B or later. You can also do it with earlier dSPACE releases.

Migrating Changes in Software That Uses TRC Files

Objective	Products that use TRC files, such as ControlDesk Next Generation, use the generated variable description to connect elements in the software with variables in the simulation application. Most of the variable path modifications caused by the TRC file changes can be automatically migrated by the dSPACE products, but for some changes you have to do manual migration in your software product.
Migration steps required in TRC file consuming software	If you have already used ControlDesk, AutomationDesk, or test scripts of any kind that are accessing variables via their variable paths and you rebuild the simulation application with MATLAB R2015b or later, you have to check whether the variable paths have been discontinued or changed in the variable description. If you are using ControlDesk you are provided support for finding inconsistent connections: for example, via specially marked instruments or the `Check Mapping` command in the Signal Editor. In AutomationDesk, variable access is realized via variable aliases. Therefore, modifications in the variable description cannot be automatically recognized. However, if you are using a variable pool in your project, it is sufficient to update this. To graphically support the new TRC file features, such as structures and references, ControlDesk and AutomationDesk provide the new Variable Browser.

For the changes that were not able to be migrated automatically in the software, you have to perform the following manual migration.

Issue	Migration Step
Update variable paths of parameters with expressions	Update any connections in ControlDesk or variables defined in test scripts that contain expressions with MATLAB workspace variables, mask parameters, or Simulink Parameter objects. Usually, it is not sufficient to change the variable path to the generated global parameter only to get the required variable access for controlling. You also have to consider any element of the expression or the resulting variable of the block.
Update variable paths of virtual Simulink buses	Update any connections in ControlDesk and test scripts accessing signals within a virtual Simulink bus to directly accessed signal source blocks. As an alternative, you can add a Bus Selector block to your model and then connect the block's output variables.
Update variable paths of non-virtual Simulink buses	Update connections in ControlDesk and test scripts that access signals within a non-virtual Simulink bus to the corresponding field of the structured variable. The formerly generated measurement arrays in the variable description are now represented by struct elements. **Note** The syntax of structured elements has changed from `Out1{myField.mySubField}` to `Out1.myField.mySubField`. This might conflict with variable names containing dots.
Update variable paths of Tunable Parameters groups for referenced models	Update any connections in ControlDesk or variables defined in test scripts that refer to tunable parameters of referenced models. The variable path of such a variable must be changed to the top-level Tunable Parameters group.
Update access to Data Stores group	Update any connections in ControlDesk and test scripts that refer to the variables of the discontinued Data Stores group to the variables of the inserted Data Store Read or Write blocks in your model.
Update connections to look-up tables	■ ControlDesk does not recognize all the look-up tables in a TRC file. As a result, these look-up tables are not available as maps or curves, for example, in ControlDesk's Variable Browser. The recognition of look-up tables does not work in the following cases: ■ The table data of the look-up table is contained in a structured parameter. ■ The table data of the look-up table references mask parameters. ■ The look-up table references table data, for example, in the Tunable Parameters group. In this case, ControlDesk recognizes only the first look-up table that references this table data. All the other look-up tables are not recognized. ■ The look-up table has three or more dimensions. To update the connection to such a look-up table, connect the individual variables of the look-up table in these cases. ■ ControlDesk no longer provides a map or curve for the `tableData` parameter of a look-up table if the `tableData` parameter was parameterized with numeric values.

Issue	Migration Step
	To update the connection to such a parameter, connect the `LookUpTableData` variable of the look-up table (instead of the `tableData` parameter). **Note** ■ ControlDesk creates a *duplicate variable* for a variable that is referenced by the table data or an axis of a look-up table. Compared to the name of the original variable, the name of the duplicate variable is extended by `(#1)`. Do not connect such a duplicate variable to a ControlDesk instrument. The following illustration shows an example of duplicate variables:  ■ If a value block in the Tunable Parameters group is referenced by the table data or an axis of a look-up table, ControlDesk uses an incorrect variable type for this value block. Instead of the *value block* variable type, it uses one of the following variable types: ■ Map ■ Curve ■ Common axis

Note

■ ControlDesk automatically migrates variable connections after you rebuild a simulation application with MATLAB R2015b or later and reload the application's variable description. However, if you then reload the variable description of the simulation application built with a MATLAB Release earlier than R2015b, the migrated variable connections are lost, and you have to update these connections manually.

■ For information on limitations in connection with the changed TRC file generation, refer to *Limitations for SDF Files* (🕮 *ControlDesk Next Generation Variable Management*).

For more information on TRC file changes, refer to *Basics on the TRC File Changes* on page 33.

AutomationDesk

Where to go from here

Information in this section

New Features of AutomationDesk 5.2

Information in this topic

General enhancements

Enhanced user interface The following user interface enhancements facilitate working with AutomationDesk:

- You can now add Python modules and Python packages that you use for your Custom Library to the Library Browser. This increases the visibility of related files and lets you access them rapidly.
- The properties of the Signal Editor have been updated.

- You can now add an entire library folder with its contained blocks and data objects to the favorites list in the Library Favorites Viewer.
- You can now open a Custom Library by dragging its related ADL file from a file explorer to the Library Browser.
- If you enter the leading round bracket of a function in the Expression Editor or the Condition Editor, a tooltip is displayed providing the function description.
- The Options dialog provides two new properties:
  - Open projects and libraries in collapsed state lets you specify whether projects and libraries are opened in collapsed state, independently of the saved collapsed state.
  - Set modified flag when collapse state changes lets you specify whether changing the collapse state of an element in a project or library results in the modified flag. The project or library is then to be saved when you close it, including the current collapse state.

Improved user documentation In the *AutomationDesk Guide*, you can now find basic information and instructions for the following libraries and features:

- MotionDesk Access library. Refer to Accessing MotionDesk.
- XIL API Convenience library - Model Access Port. Refer to Accessing Simulation Platforms via the XIL API Convenience Library.
- XIL API Convenience library - Electrical Error Simulation Port. Refer to Simulating Electrical Errors via the XIL API Convenience Library.
- Signal-Based Testing library. Refer to Implementing Signal-Based Tests.
- Signal Editor. Refer to *Handling Signals* (🕮 *AutomationDesk Guide*).

There is a tutorial video available showing you how to apply Signal-Based Testing. Refer to https://www.dspace.com/go/tutorial_ad_sbt (requires a mydSPACE login).

For public product videos, refer to AutomationDesk product videos.

Enhancements to the libraries

The following libraries have been enhanced:

Main Library You can now add data objects to the following automation blocks to enhance their interfaces:

- Parallel (for each thread)
- For
- While
- Repeat
- IfThenElse (for the main block and for each branch)
- TryFinally
- TryExcept
- Range
- RangeDict
- RangeDataContainer
- RangeBlockDataObjects

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

Evaluation library The Evaluation library provides a new automation block:

- The IsNotEqual block evaluates whether the values of the input signal are different from the specified reference values.

For more information, refer to *Evaluation* (🕮 *AutomationDesk Library Reference*).

Enhancements to the COM API

The AutomationDesk COM API provides the following enhancements:

- New method to highlight a specific element in the AutomationDesk user interface
- New property to specify whether the Result Browser is opened after the execution
- New property for the File data object to specify the file path as the relative or absolute path
- Demo project in C#

For more information, refer to 🕮 *AutomationDesk API Reference*.

Discontinuations for future versions

The following libraries, automation blocks and data objects will be discontinued in dSPACE Release 2016-B:

- Test Framework library

  You should migrate your projects based on the Test Framework library to the Test Builder library. For help on migration, refer to http://www.dspace.com/go/TestBuilderMigration.

- Platform Access library

  The Platform Access library is being delivered for the last time with dSPACE Release 2016-A. You should migrate your projects based on the Platform Access library to the XIL API library or to the XIL API Convenience library. This provides the MAPort for reading, writing and stimulating variables of a connected platform.

  For help on migration, refer to http://www.dspace.com/go/pscta.

- Failure simulation automation blocks in the ControlDeskNG Access library

  ControlDesk's Failure Simulation Module is being delivered for the last time with dSPACE Release 2016-A. To prepare electrical error simulation via automation, use the Electrical Error Simulation Port (EESPort) in the XIL API library or in the XIL API Convenience library instead of the failure simulation blocks in the ControlDeskNG Access library.

  For help on migration, refer to http://www.dspace.com/go/pscta.

- InitCaptureResultIDFReader and InitCaptureResultIDFWriter automation blocks in the XIL API library

  The InitCaptureResultIDFReader and InitCaptureResultIDFWriter automation blocks are being delivered for the last time with dSPACE Release 2016-A. Because the IDF format will be discontinued in future versions, you should replace these automation blocks with the CaptureResultReader and CaptureResultWriter data objects, which support the MDF format. For more information, refer to CaptureResultReader (Data Object) (🕮 *AutomationDesk Library Reference*) and CaptureResultWriter (Data Object) (🕮 *AutomationDesk Library Reference*).

- DTS V7 support in the Remote Diagnostics (COM) library

  The Remote Diagnostics (COM) library is supporting DTS V7 for the last time with dSPACE Release 2016-A. You should migrate your projects to ControlDesk as the diagnostic tool.

  For help on migration, refer to http://www.dspace.com/go/pscta.

- Platform management automation API version 1.0

  Platform management automation API version 1.0 is being supported for the last time with AutomationDesk 5.2 from dSPACE Release 2016-A. For more information, refer to *GetPlatformManagement* (🕮 *AutomationDesk Library Reference*).

The elements that are planned to be discontinued are specially marked in the Library Browser.

Migrating to AutomationDesk 5.2

General migration aspects

If you open an AutomationDesk project with a later AutomationDesk version, the software automatically detects whether migration is necessary. Click OK in the message dialog to start migration. 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.

Note

Before you open an older project with the new AutomationDesk version, make sure that 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 displayed in the Log Viewer.
- The built-in libraries, required custom libraries and other packages must be loaded correctly.

If you are using a version control system, there are some preconditions for successful migration, refer to *How to Migrate Projects Under Version Control* (🕮 *AutomationDesk Guide*).

For more information, refer to *Migrating AutomationDesk* (🕮 *AutomationDesk Guide*).

Libraries

Platform Access library Note that the Platform Access library does not support the new dSPACE platforms DS1007 PPC Processor Board, MicroLabBox, and VEOS.

Evaluation library If you use the IsEqual block in a sequence with a negative value specified for the Tolerance data object, a ValueError is now raised. With earlier AutomationDesk versions, the specified value is used with its absolute value.

Automotive Simulation Models (ASM)

Where to go from here

Information in this section

Information in other sections

Migrating ASM Models (🕮 *ASM User Guide*)
Provides general information on the migration of ASM models.

All ASM Blocksets

Migration of All ASM Blocksets

Scope handling

During migration, all disabled scopes within the model need to be restored to ensure that the model works properly, including the new scope handling procedure. You might have to disable these scopes again after migration. You can easily disable scopes via the Scope Handling GUI button provided by ASM. This makes direct use of the commenting feature in Simulink.

ASM Base InCylinder Blockset

Where to go from here

Information in this section

New Features of ASM Base InCylinder Blockset 2.2

MASS_FUEL_BURNED block

This block is new. It calculates the crank angle for three different levels of mass fuel burned fractions.

CYLINDERPRESSURE_MAX block

This block is new. It estimates the crank angle for the maximum cylinder pressure.

MDL_PAR Environment

New parameters and signals have been introduced to make it possible to mix stimulated and controlled signals. The new parameters describe different simulation modes. These can be used in all maneuver types for the manual or automatic control of the engine model in HIL and Simulink simulation.

For example, it is now possible to use the stimulus accelerator and brake pedals while the driver controls the selector lever or the gear as well as the clutch pedal.

Migrating to ASM Base InCylinder Blockset 2.2

CRANK_MECHANISM block

A new output port has been added: CrankAngle[aTDC]. It provides the crank angle with a different unit. It counts from -360 deg to 360 deg with 0 deg as working top dead center.

The related signal in the ASMSignalBus has been renamed from CrankAngle_Display[deg] to CrankAngle[aTDC].

ASM Diesel Engine Blockset

Where to go from here

Information in this section

New Features of ASM Diesel Engine Blockset 2.3

RAIL_CONTROL_ CRANKBASED block

The phi_FMU_Energized[deg] outport signal is sorted in ascending order and modulated to a 720-degree crank angle.

There are two new block parameters:

- With the new Const_max_num_HPPCam parameter, a maximum number of cams can be defined. This way, the maximum size of the vector can be defined. With this parameter, you can switch between different variants with different numbers of high-pressure cams.
- With the Const_i_HighPresPump parameter, the transmission ratio between the engine and pump can be set.

To initialize the model with the correct initial conditions, a memory block is applied to the Trigger[0Hold|1Pass] signal.

HPP_CRANKBASED block

The calculation of delivery length has been modified. The variable is calculated with the approaching cam instead of the past one.

There are two new block parameters:

- With the new Const_phi_Camshaft_InitOffs parameter, the offset of the camshaft against the crankshaft can be varied more easily.
- With the new Const_max_num_HPPCam parameter, a maximum number of cams can be defined. This way, the maximum size of the vector can be defined. With this parameter, you can switch between different variants with different numbers of high-pressure cams.

The sorting algorithm of the FMU control signal in ascending order has been removed. Instead of this algorithm, the ranges of the

	current compression stroke are defined. The element of the phi_FMU_Energized[deg] control signal that fits the range is used for the calculation of the delivery length. Cut event handling has been introduced. An event is cut when the control signal to energize the fuel metering unit is interrupted by the end of the capture window of the I/O.
LP_INTAKE_MANIFOLD block	The block has been revised. The behavior of the block has been changed. The LP_INTAKE_MANIFOLD block of the former release was renamed to the LP_INTAKE_MANIFOLD_4_0 block. During migration to dSPACE Release 2016-A, the former block is moved to the FormerVersions sublibrary.

Changes in the ASM Diesel Engine Demo Model

MDL_PAR Environment	New parameters and signals have been introduced to make it possible to mix stimulated and controlled signals. The new parameters describe different simulation modes. These can be used in all maneuver types for the manual or automatic control of the engine model in HIL and Simulink simulation. For example, it is now possible to use the stimulus accelerator and brake pedals while the driver controls the selector lever or the gear as well as the clutch pedal.
Signal conditioning	The I/O interface of the MDL_In subsystem now has an algorithm to process the I/O signal for the HPP_CRANKBASED block. Besides the state information of the high-pressure pump (UpdateCounter, UpdateState, PulseState), it also considers the leading edge of the control signal. The block outport is an eight-dimensional vector. If the control signal of the pump has fewer signals, it must be extended to the expected size with dummies (e.g.: **999**).
Low-pressure EGR	The AirPath model has a new interface to easily replace the low-pressure EGR model. The new OpenLowPressureEGRModel button lets you add a prepared low-pressure EGR demo to your model. Then, use drag & drop to add the demo model from the library to the AirPath model and replace the existing low-pressure EGR.

Migrating to ASM Diesel Engine Blockset 2.3

Scope handling	During migration, all disabled scopes within the model need to be restored to ensure that the model works properly, including the new scope handling procedure. You might have to disable these scopes again after migration. You can easily disable scopes via the Scope Handling GUI button provided by ASM. This makes direct use of the commenting feature in Simulink.
ENGINE_SETUP block	The Const_num_Cyl_vector parameter has been removed.
RAIL_CONTROL_ CRANKBASED block	The phi_FMU_energized[deg] outport has been renamed phi_FMU_Energized[deg]. The block now has a new parameter in the premigrate variant (`_asmmigratepre` folder): Const_max_num_HPPCam The parameter is initialized with the value **8**, which is the maximum size of the phi_FMU_energized[deg] vector. It is recommended to reduce the value to the number of cams of the high-pressure pump. In this case, code generation is required. There are two new parameters in the postmigrate variant (`_asmmigratepost` folder): Const_i_HighPresPump and Const_num_Cam. The new parameters are mapped with the parameters of the HPP_CRANKBASED block.
HPP_CRANKBASED block	Two new parameters are introduced to the premigrate variant of the block (`_asmmigratepre` folder): Const_phi_Camshaft_InitOffs and Const_max_num_HPPCam. The Const_phi_Camshaft_InitOffs parameter is initialized with the value **8**, which is the maximum size of the phi_FMU_energized[deg] vector. It is recommended to reduce the value to the number of cams of the high-pressure pump. In this case, code generation is required. Several signals in the ASMSignalBus have been renamed. The Sw_Mode_FMU_Control parameter has been removed.
LP_INTAKE_MANIFOLD block	During migration, the LP_INTAKE_MANIFOLD block is redirected to the former version LP_INTAKE_MANIFOLD_4_0 block in the ASM Diesel Engine Library.
AIRFILTER block	A memory block has been removed from the library block and placed before the mdot_AirFilter[kg|h] inport of the library block.

ASM Diesel Exhaust Blockset

Changes in the ASM Diesel Exhaust Demo Model

Exhaust system demo model

The exhaust system demo model has two new outports, T_Out_DPF[K] and p_Out_ExhThrottle[Pa], and a new inport, p_Out_DPF[Pa]. The exhaust throttle is part of the low-pressure EGR and is located downstream of the DPF. If there is no exhaust throttle in the demo model, the p_Out_ExhThrottle[Pa] signal is the same as of p_Out_DPF[Pa].

ASM Diesel InCylinder Blockset

Migrating to ASM Diesel InCylinder Blockset 2.2

Scope handling	During migration, all disabled scopes within the model need to be restored to ensure that the model works properly, including the new scope handling procedure. You might have to disable these scopes again after migration. You can easily disable scopes via the Scope Handling GUI button provided by ASM. This makes direct use of the commenting feature in Simulink.

ASM Drivetrain Basic Blockset

Where to go from here

Information in this section

New Features of ASM Drivetrain Basic Blockset 4.2

Engine simulation

New blocks have been introduced to simulate a simplified engine. It is now possible to build up a simplified virtual vehicle using only the ASM Drivetrain Basic Blockset.

The new blocks are:

- ENGINE: simulates simplified engine dynamics
- FUEL_CONSUMPTION: used with the ENGINE block to calculate the fuel consumption and the carbon dioxide emissions
- ENGINE_OPERATION_BASIC: part of the soft ECU model of the engine. It detects the engine state and activates the starter.
- IDLE_SPEED_CONTROL_ENGINE_BASIC: part of the soft ECU model of the engine. It simulates the idle speed controller.
- TORQUE_INTERVENTION_ENGINE_BASIC: part of the soft ECU model of the engine. It realizes an external engine torque request.

Rearranged library blocks

A new subsystem named Engine has been added on the top level. It contains the ENGINE and FUEL_CONSUMPTION blocks.

Inside the Soft ECU subsystem, two new subsystems have been added:

- Transmission: contains the soft ECU blocks of the transmission.
- Engine: contains the new soft ECU blocks of the engine.

SOFT_ECU_TRANSMISSION block

It is now possible to stimulate the SOFT_ECU_TRANSMISSION block with stimulus or reference gear and clutch pedal position signals.

	The stimulus gear and clutch pedal are forwarded to the output without further actions. Depending on whether the AMT switch is active or not, the clutch pedal signal is used for the AMT clutch or the lockup clutch, respectively. The reference gear is used as a set value that affects the other outputs. These features are deactivated per default and can be activated using a switch in the demo model.
GEAR_SHIFTER block	The block has been restructured to offer new features and make it more compatible with other ASM blocks. For example, it is now possible to use the stimulus accelerator and brake pedal signals while the GEAR_SHIFTER block controls the selector lever or the gear as well as the clutch position.

Migrating to ASM Drivetrain Basic Blockset 4.2

SOFT_ECU_TRANSMISSION block	During migration, the new inports are connected to dummy values. However, you can still connect the inports with the related signals from the model to use the block functionality.
GEAR_SHIFTER block	Due to the considerable changes in this block, it cannot be migrated automatically. Therefore, during the migration, the link to the GEAR_SHIFTER block is changed to the former implementation version: FormerVersions/GEAR_SHIFTER_13_0. The former version of the block still contains some new features. For more information, refer to the documentation. To use the new GEAR_SHIFTER block, add it to your model from the ASM Environment Library. In this case, you must manually adapt the inports and outports.
CYCLES block	The integrator inside the block is changed from continuous to discrete. A bug related to the key signal definition when the start time is not zero has been fixed.
LONGITUDINAL_ CONTROLLER_HYBRID block	The actual vehicle acceleration is used instead of the approximated value. The vehicle acceleration is fed to the block via a new inport that is connected with the corresponding signal during migration.

ASM Electric Components Blockset

Where to go from here

Information in this section

New Features of ASM Electric Components Blockset 3.2

KEY_SIGNALS_ICE block

A new parameter has been added to the block to set a maximum starter request on time.

STARTER_ICE block

To check for the minimal transmission speed, hysteresis has been implemented for the block. It therefore hides a toggling starter on the signal.

TRQ_REQUEST_ COORDINATION block

A new inport has been added to select a right torque table for the different drive modes. The following torques are used:

- For the ICE drive mode: the torque of the ICE table
- For the EM drive mode: the torque of the EM table
- For the Hybrid drive mode: the sum torque of the ICE and EM table

Migrating to ASM Electric Components Blockset 3.2

Scope handling

During migration, all disabled scopes within the model need to be restored to ensure that the model works properly, including the new scope handling procedure. You might have to disable these scopes again after migration. You can easily disable scopes via the Scope Handling GUI button provided by ASM. This makes direct use of the commenting feature in Simulink.

BRAKE_CONTROL block

The calculation of the crank shaft braking force F_Brake_CrankShaft has been corrected. The force must be calculated by a multiplication

of the electric machine torque with the transmission and differential ratio and a division with the tire radius. Therefore, the incorrect division of the transmission and differential ratio was changed to a multiplication. This correction can result in a higher brake force of the electric machine during recuperation.

Signal modification

For several blocks, all inport and outport bus signals have been changed to vector signals. The automatic migration adds a subsystem to the affected outports to reverse the vector back to the former bus signal.

This applies to the following blocks:

- BLOCK_MODULATOR
- BLDC_CONTROLLER
- BLDC_CONTROLLER_BASIC
- THREE_PHASE_INVERTER
- BRUSHLESS_DC_MACHINE_ALPHA_BETA
- PMSM_CONTROLLER_BASIC
- PMSM_CONTROLLER
- PMSM_D_Q_NONLINEAR
- PERMANENT_MAGNET_SYNCHRONOUS_MACHINE_D_Q
- THREE_LEVEL_THREE_PHASE_INVERTER
- THREE_LEVEL_SPACE_VECTOR_MODULATOR
- SCIM_CONTROLLER_BASIC
- SQUIRREL_CAGE_ASYNCHRONOUS_MACHINE_D_Q
- THREE_PHASE_DCM_INVERTER
- THREE_LEVEL_HALF_BRIDGE_INVERTER
- HALF_BRIDGE_INVERTER
- HALL_ENCODER

ASM Environment Blockset

Where to go from here

Information in this section

New Features of ASM Environment Blockset 4.4

GEAR_SHIFTER block

The block has been restructured to offer new features and make it more compatible with other ASM blocks. For example, it is now possible to use the stimulus accelerator and brake pedal signals while the GEAR_SHIFTER block controls the selector lever or the gear as well as the clutch position.

LATERAL_CONTROL2 block

The LATERAL_CONTROL2 block has been enhanced with an additional yaw rate controller. It has no preview functionality and is intended for use during steady-state cornering maneuvers. It can be activated via the maneuver definition in ModelDesk only during a circle maneuver. In all other cases, the position controller with preview is used.

ROAD block

The ROAD block has been adapted for the new line definition and shapes features. The modifications are implemented within the S-function. There are only minor changes to the block interface.

The signal bus at the Info outport has been extended by an additional signal: d_Veh_Ref_Line[m].

The signal bus at the TrafficFellows outport has been extended by an additional signal: d_Fellows_RefLine[m].

Migrating to ASM Environment Blockset 4.4

GEAR_SHIFTER block

Due to the considerable changes in this block, it cannot be migrated automatically. Therefore, during the migration, the link to the

	GEAR_SHIFTER block is changed to the former implementation version: FormerVersions/GEAR_SHIFTER_13_0. The former version of the block still contains some new features. For more information, refer to the documentation. To use the new GEAR_SHIFTER block, add it to your model from the ASM Environment Library. In this case, you must manually adapt the inports and outports.
LONGITUDINAL_ CONTROLLER_HYBRID block	The actual vehicle acceleration is used instead of the approximated value. The vehicle acceleration is fed to the block via a new inport which is connected with the corresponding signal during the migration.
SIGNAL_SELECTION block	The SIGNAL_SELECTION block is not used in new ASM demos anymore. Therefore, it has been moved to the Former versions sublibrary. During migration, the link to the block is changed to the former implementation version: FormerVersions/SIGNAL_SELECTION_1_0.
LATERAL_CONTROL1 block	The block now has new inports to switch between the different angle steering modes internally: Steer_Mode[1Stim|2Driver|3Fix] and Angle_SteeringWheel_Maneuver[deg]. The new inports are connected to the related signals during the migration process.
LATERAL_CONTROL2 block	The block now has new inports to switch internally between the different angle steering modes: Steer_Mode[1Stim|2Driver|3Fix] and Angle_SteeringWheel_Maneuver[deg]). The new inports are connected to the related signals during the migration process. There are also new inports for the new yaw rate control functionality: YawRate_Vehicle[rad|s], LatCtrl_Mode[1Pos|2Yaw] and Curv_Road_Circle[1|m][-Right|_+Left]. During migration, the new inports are connected to dummy values and only the position controller is used. Moreover, the yaw rate controller parameters (Const_Kp_YawRate_Ctrl and Const_Ki_YawRate_Ctrl) are promoted to mask parameters, which are added and initialized during the migration process.
ROAD block	According to the new road functionalities, the format of the road MAT files has been changed. The road MAT files are automatically migrated during standard model migration. The road MAT files can also be migrated separately by using the `asm_migrate_road` function.

ASM Gasoline Engine Basic Blockset

Where to go from here

Information in this section

New Features of ASM Gasoline Engine Basic Blockset 2.0.3

MDL_PAR Environment

New parameters and signals have been introduced to make it possible to mix stimulated and controlled signals. The new parameters describe different simulation modes. These can be used in all maneuver types for the manual or automatic control of the engine model in HIL and Simulink simulation.

For example, it is now possible to use the stimulus accelerator and brake pedals while the driver controls the selector lever or the gear as well as the clutch pedal.

REL_AIRMASS_MAPBASED block

The REL_AIRMASS_MAPBASED block has been introduced to improve the transient behavior of ASM Gasoline Engine. It is based on using a look-up table to negate the effects of changing pressure in the intake manifold. These would lead to the calculation of incorrect injection quantities in the SoftECU model.

Migrating to ASM Gasoline Engine Basic Blockset 2.0.3

Scope handling

During migration, all disabled scopes within the model need to be restored to ensure that the model works properly, including the new scope handling procedure. You might have to disable these scopes again after migration. You can easily disable scopes via the Scope Handling GUI button provided by ASM. This makes direct use of the commenting feature in Simulink.

ENGINE_SETUP block	The Const_num_Cyl_vector parameter has been removed.

ASM Gasoline Engine Blockset

Where to go from here

Information in this section

New Features of ASM Gasoline Engine Blockset 3.3

RAIL_CONTROL_ CRANKBASED block

The phi_FMU_Energized[deg] outport signal is sorted in ascending order and modulated to a 720-degree crank angle.

There are two new block parameters:

- With the new Const_max_num_HPPCam parameter, a maximum number of cams can be defined. This way, the maximum size of the vector can be defined. With this parameter, you can switch between different variants with different numbers of high-pressure cams.
- With the Const_i_HighPresPump parameter, the transmission ratio between the engine and pump can be set.

To initialize the model with the correct initial conditions, a memory block is applied to the Trigger[0Hold|1Pass] signal.

HPP_CRANKBASED block

The calculation of delivery length has been modified. The variable is calculated with the approaching cam instead of the past one.

There are two new block parameters:

- With the new Const_phi_Camshaft_InitOffs parameter, the offset of the camshaft against the crankshaft can be varied more easily.
- With the new Const_max_num_HPPCam parameter, a maximum number of cams can be defined. This way, the maximum size of the vector can be defined. With this parameter, you can switch between different variants with different numbers of high-pressure cams.

The sorting algorithm of the FMU control signal in ascending order has been removed. Instead of this algorithm, the ranges of the

	current compression stroke are defined. The element of the phi_FMU_Energized[deg] control signal that fits the range is used for the calculation of the delivery length. Cut event handling has been introduced. An event is cut when the control signal to energize the fuel metering unit is interrupted by the end of the capture window of the I/O.
PORTINJECTOR block	Internal calculation of q_Mean_Inj[mm3|cyc] has been introduced to the block and added to the ASMSignalCollector.
REL_AIRMASS_MAPBASED block	The REL_AIRMASS_MAPBASED block has been introduced to improve the transient behavior of ASM Gasoline Engine. It is based on using a look-up table to negate the effects of changing pressure in the intake manifold. These would lead to the calculation of incorrect injection quantities in the SoftECU model.

Changes in the ASM Engine Gasoline Demo Model

MDL_PAR Environment	New parameters and signals have been introduced to make it possible to mix stimulated and controlled signals. The new parameters describe different simulation modes. These can be used in all maneuver types for the manual or automatic control of the engine model in HIL and Simulink simulation. For example, it is now possible to use the stimulus accelerator and brake pedals while the driver controls the selector lever or the gear as well as the clutch pedal.
Signal conditioning	The I/O interface of the MDL_In subsystem now has an algorithm to process the I/O signal for the HPP_CRANKBASED block. Besides the state information of the high-pressure pump (UpdateCounter, UpdateState, PulseState), it also considers the leading edge of the control signal. The block outport is an eight-dimensional vector. If the control signal of the pump has fewer signals, it must be extended to the expected size with dummies (e.g.: **999**).
Demo parameterization	Demo parameterization is now also possible for a small 1.5 l, four-cylinder turbocharged gasoline engine.

Migrating to ASM Gasoline EngineBlockset 3.3

Scope handling	During migration, all disabled scopes within the model need to be restored to ensure that the model works properly, including the new scope handling procedure. You might have to disable these scopes again after migration. You can easily disable scopes via the Scope Handling GUI button provided by ASM. This makes direct use of the commenting feature in Simulink.
ENGINE_SETUP block	The Const_num_Cyl_vector parameter has been removed.
RAIL_CONTROL_ CRANKBASED block	The phi_FMU_energized[deg] outport has been renamed phi_FMU_Energized[deg]. The block now has a new parameter in the premigrate variant (`_asmmigratepre` folder): Const_max_num_HPPCam The parameter is initialized with the value **8**, which is the maximum size of the phi_FMU_energized[deg] vector. It is recommended to reduce the value to the number of cams of the high-pressure pump. In this case, code generation is required. There are two new parameters in the postmigrate variant (`_asmmigratepost` folder): Const_i_HighPresPump and Const_num_Cam. The new parameters are mapped with the parameters of the HPP_CRANKBASED block.
HPP_CRANKBASED block	Two new parameters are introduced to the premigrate variant of the block (`_asmmigratepre` folder): Const_phi_Camshaft_InitOffs and Const_max_num_HPPCam. The Const_phi_Camshaft_InitOffs parameter is initialized with the value **8**, which is the maximum size of the phi_FMU_energized[deg] vector. It is recommended to reduce the value to the number of cams of the high-pressure pump. In this case, code generation is required. Several signals in the ASMSignalBus have been renamed. The Sw_Mode_FMU_Control parameter has been removed.
AIRFILTER block	A memory block has been removed from the library block and placed before the mdot_AirFilter[kg|h] inport of the library block.
Related topics	Basics • *Migrating ASM Models* (📖 *ASM User Guide*)

ASM Gasoline InCylinder Blockset

Migrating to ASM Gasoline InCylinder Blockset 2.2

Scope handling	During migration, all disabled scopes within the model need to be restored to ensure that the model works properly, including the new scope handling procedure. You might have to disable these scopes again after migration. You can easily disable scopes via the Scope Handling GUI button provided by ASM. This makes direct use of the commenting feature in Simulink.
HEAT_RELEASE_VIBE block	In the HEAT_RELEASE_VIBE block, the TagVisibility parameter of internally used Goto blocks has been changed to scoped. Now you can use multiple instances of the block in a model.

ASM Traffic Blockset

Where to go from here

Information in this section

New Features of ASM Traffic Blockset 3.4

SOFT_ECU_ACC block

AEB can be used without active ACC. For this, new outports have been added to the SOFT_ECU_ACC block to enable the engine and brake torque request to activate the AEB functionality independently of the ACC signal.

LINE_SENSOR block

The LINE_SENSOR block is new. It detects lane markings, junction borders, and road shapes (of the Line and Continuous object types). Each shape is described by a list of discrete points.

Object_Sensor_3D block

There is a new 3-D sensor model. The new model includes the functionality of the existing RadarSensor_3D model and contains additional features for sensor scheduling, object sorting, static object detection, and a new geometric scope form (elliptical cone with ellipsoid head). The existing RadarSensor_3D block has been moved to the Former Version section. In the ASM Traffic demo model, the RadarSensor_3D model has been replaced by the new Object_Sensor_3D model.

OBJECT_SENSOR_3D_ CALCULATION block

The block is new. It contains a new sensor algorithm for collision detection, nearest point calculation, and object sorting.

The new algorithm features two sensor scope forms:

- 1. Rectangular pyramid
- 2. Elliptical cone with ellipsoid head

In addition to fellow vehicles, the algorithm can also detect static road objects. The output of the sensor can be sorted by ID or distance.

OBJECT_SENSOR_3D_ PARAMETERS block	This block is new for parameterizing one Object_Sensor_3D instance. With this block, the geometric parameters (position and orientation of the apex and geometry of the scope form) and scheduling parameters are set.
OBJECT_SENSOR_3D_ GEOMETRY_PARAMETERS block	This block is new for parameterizing the geometric properties (position and orientation of the apex and geometry of the scope form) of one Object_Sensor_3D instance.
OBJECT_SENSOR_3D_ PARAM_VECTORIZATION block	This block is new for separating the geometry parameters from the scheduling parameters of multiple Object_Sensor_3D blocks within one sensor chain.
OBJECT_SENSOR_3D_ PARAM_SELECTION block	This block is new for selecting the geometry parameters of the sensor instance to be calculated in the current simulation step.
OBJECT_SENSOR_3D_ MAPPING block	This block is new for mapping the current sensor output signal to a specific sensor instance. Additionally, this block calculates the relative velocity and acceleration.
SENSOR_SCHEDULER block	This block is new for calculating the sensor instance during the current simulation step, according to the scheduling parameters of the sensor instances. The conflict flag and conflict information can help solve potential scheduling conflicts.
SENSOR_SCHEDULING_ PARAMETERS block	This block is new block for parameterizing the scheduling properties of one Object_Sensor_3D block. It can be combined with the OBJECT_SENSOR_3D_GEOMETRY_PARAMETERS block.
SAMPLE_PULSE block	This block is new for sampling sensor input signals. With this block, efficient quasi-parallel sensor calculations are possible.
OBJECT_PROPERTIES_BOX block	This block is new for outputting the bounding box properties for static objects, according to a list of object types.
OBJECT_POSITION block	This block is new for outputting the positions of objects (static objects and fellow vehicles), according to on a list of object IDs.
FELLOW_POSITIONS block	The fellow vehicle reset has been modified and the block has a new inport: Flag_FellowUsed.

Changes in the ASM Traffic Demo Model

Object_Sensor_3D model	In the ASM Traffic demo model, the RadarSensor_3D block has been replaced by the new Object_Sensor_3D block. The demo model parameterization has been modified so the standard geometric sensor scope form has changed from the rectangular pyramid to the elliptical cone with ellipsoid head. The detection of static objects is enabled, the output sort mode is set to *static*.
LINE_SENSOR block	The LINE_SENSOR block has been added to the ASM Traffic demo model.
MDL_DISP subsystem	Due to the changes in the sensor model, the scopes in MDL_DISP have been modified.
Drivetrain demo	In the Drivetrain subsystem, the CRANKSHAFT, Transmission and Final_Drive_Assembly models can now easily be copied to the model via the Open Demos buttons and drag & drop. During the copy process, you are asked to confirm the replacement of the existing demo. If you select OK, the model is replaced and the inports and outports are automatically connected. If you select Cancel, a normal copy process is performed. This feature is only available in new ASM demos.
Brake Hydraulics demo	The BRAKE_HYDRAULICS model can now easily be copied to the model via the Open ASM_Brakehydraulics_lib button and drag & drop. During the copy process, you are asked to confirm the replacement of the existing demo. If you select OK, the model is replaced and the inports and outports are automatically connected. If you select Cancel, a normal copy process is performed. This feature is only available in new ASM demos.

Migrating to ASM Traffic Blockset 3.4

Scope handling	During migration, all disabled scopes within the model need to be restored to ensure that the model works properly, including the new scope handling procedure. You might have to disable these scopes again after migration. You can easily disable scopes via the Scope

	Handling GUI button provided by ASM. This makes direct use of the commenting feature in Simulink.
FELLOW_POSITIONS block	Due to the modified fellow vehicle reset, the block gets a new Flag_FellowUsed inport. During automatic migration, this new inport is connected to a dummy Constant block.

ASM Trailer Blockset

Where to go from here

Information in this section

Changes in the ASM Trailer Demo Model

Drivetrain demo

In the Drivetrain subsystem, the CRANKSHAFT, Transmission and Final_Drive_Assembly models can now easily be copied to the model via the Open Demos buttons and drag & drop.

During the copy process, you are asked to confirm the replacement of the existing demo. If you select OK, the model is replaced and the inports and outports are automatically connected. If you select Cancel, a normal copy process is performed. This feature is only available in new ASM demos.

Brake Hydraulics demo

The BRAKE_HYDRAULICS model can now easily be copied to the model via the Open ASM_Brakehydraulics_lib button and drag & drop.

During the copy process, you are asked to confirm the replacement of the existing demo. If you select OK, the model is replaced and the inports and outports are automatically connected. If you select Cancel, a normal copy process is performed. This feature is only available in new ASM demos.

Wheel camber angle

The wheel camber angle is now also animated in MotionDesk.

Migrating to ASM Trailer Blockset 2.5

Scope handling

During migration, all disabled scopes within the model need to be restored to ensure that the model works properly, including the new scope handling procedure. You might have to disable these scopes

	again after migration. You can easily disable scopes via the Scope Handling GUI button provided by ASM. This makes direct use of the commenting feature in Simulink.
TIRE_MODEL_TMEASY_xy block	There has been a bug fix for the external tire radius, which was connected to the wrong signal. The Simulink Signal Specification block has been added to support the simplified initialization mode in Simulink.
TIRE_MODEL_MAGIC_FORMULA_xy block	There has been a bug fix for the external tire radius, which was connected to the wrong signal. The Simulink Signal Specification block has been added to support the simplified initialization mode in Simulink.
MODULAR_MASS_MATRIX_WHEEL_xy block	The Simulink Signal Specification block has been added to support the simplified initialization mode in Simulink.

ASM Truck Blockset

Where to go from here

Information in this section

Changes in the ASM Truck Demo Model

Brake Hydraulics demo

The BRAKE_HYDRAULICS model can now easily be copied to the model via the Open ASM_Brakehydraulics_lib button and drag & drop.

During the copy process, you are asked to confirm the replacement of the existing demo. If you select OK, the model is replaced and the inports and outports are automatically connected. If you select Cancel, a normal copy process is performed. This feature is only available in new ASM demos.

Migrating to ASM Truck Blockset 2.4

Scope handling

During migration, all disabled scopes within the model need to be restored to ensure that the model works properly, including the new scope handling procedure. You might have to disable these scopes again after migration. You can easily disable scopes via the Scope Handling GUI button provided by ASM. This makes direct use of the commenting feature in Simulink.

TIRE_MODEL_TMEASY_xy block

There has been a bug fix for the external tire radius, which was connected to the wrong signal. The Simulink Signal Specification block has been added to support the simplified initialization mode in Simulink.

TIRE_MODEL_MAGIC_FORMULA_xy block	There has been a bug fix for the external tire radius, which was connected to the wrong signal. The Simulink Signal Specification block has been added to support the simplified initialization mode in Simulink.
MODULAR_MASS_MATRIX_WHEEL_xy block	The Simulink Signal Specification block has been added to support the simplified initialization mode in Simulink.

ASM Vehicle Dynamics Blockset

Where to go from here

Information in this section

New Features of ASM Vehicle Dynamics Blockset 3.3

SOFT_ECU_TRANSMISSION block

It is now possible to stimulate the SOFT_ECU_TRANSMISSION block with stimulus or reference gear and clutch pedal position signals.

The stimulus gear and clutch pedal are forwarded to the output without further actions. Depending on whether the AMT switch is active or not, the clutch pedal signal is used for the AMT clutch or the lockup clutch, respectively.

The reference gear is used as a set value that affects the other outputs.

These features are deactivated per default and can be activated using a switch in the demo model.

IDLE_SPEED_CONTROL_ENGINE_BASIC block

A new functionality to offer an external engine idle speed request has been introduced.

VEHICLE_MOVEMENT_INFO_CAR block

A vehicle acceleration signal including gravitation has been added to ASMSignalBus.

Changes in the ASM Vehicle Dynamics Demo Model

Drivetrain demo

In the Drivetrain subsystem, the CRANKSHAFT, Transmission and Final_Drive_Assembly models can now easily be copied to the model via the Open Demos buttons and drag & drop.

	During the copy process, you are asked to confirm the replacement of the existing demo. If you select OK, the model is replaced and the inports and outports are automatically connected. If you select Cancel, a normal copy process is performed. This feature is only available in new ASM demos.
Brake Hydraulics demo	The BRAKE_HYDRAULICS model can now easily be copied to the model via the Open ASM_Brakehydraulics_lib button and drag & drop. During the copy process, you are asked to confirm the replacement of the existing demo. If you select OK, the model is replaced and the inports and outports are automatically connected. If you select Cancel, a normal copy process is performed. This feature is only available in new ASM demos.
Wheel camber angle	The wheel camber angle is now also animated in MotionDesk.

Migrating to ASM Vehicle Dynamics Blockset 3.3

Scope handling	During migration, all disabled scopes within the model need to be restored to ensure that the model works properly, including the new scope handling procedure. You might have to disable these scopes again after migration. You can easily disable scopes via the Scope Handling GUI button provided by ASM. This makes direct use of the commenting feature in Simulink.
SOFT_ECU_TRANSMISSION block	During migration, the new inports are connected to dummy values. However, you can still connect the inports with the related signals from the model to use the block functionality.
ENGINE block	Some port and parameter names have been revised. There are no functional changes.
FUEDL_CONSUMPTION block	Some port and parameter names have been revised. There are no functional changes.
IDLE_SPEED_CONTROL_ ENGINE_BASIC block	The block has new inports for the external idle speed request. These inports are connected to dummy values during migration. Moreover, the idle speed controller parameters are promoted to mask parameters, which are added and initialized during migration.

Steering block	The Simulink Unit Delay block has been replaced with a memory block to avoid some warnings in MATLAB.
STEERING_VARIABLE_RATIO block	The Simulink Unit Delay block has been replaced with a memory block to avoid some warnings in MATLAB.
STEERING_3DOF_ VARIABLE_RATIO block	There has been a bug fix for the steering system end-limits simulation. The Simulink Unit Delay block has been replaced with a memory block to avoid some warnings in MATLAB.
TIRE_MODEL_TMEASY_xy block	There has been a bug fix for the external tire radius, which was connected to the wrong signal. The Simulink Signal Specification block has been added to support the simplified initialization mode in Simulink.
TIRE_MODEL_MAGIC_ FORMULA_xy block	There has been a bug fix for the external tire radius, which was connected to the wrong signal. The Simulink Signal Specification block has been added to support the simplified initialization mode in Simulink.
Rigid Axle	There has been a bug fix for the suspension kinematics calculation of the rigid axle.
Subframe	There has been a bug fix for the additional lateral and longitudinal displacement due to the angle gamma-compliance based on the lateral force.

Bus Manager (Stand-Alone)

Features of the Bus Manager (Stand-Alone) 5.5

Bus Manager as stand-alone application

As of dSPACE Release 2016-A, the Bus Manager is available in two versions: as a component in ConfigurationDesk and now also as a stand-alone tool.

The *Bus Manager (stand-alone)* lets you configure CAN and LIN bus communication for offline simulation on VEOS. After you configure the bus communication, the Bus Manager lets you generate bus simulation container (BSC) files. You can use the BSC files to implement the configured bus communication in offline simulation applications and perform restbus simulation on VEOS.

To implement bus communication in real-time applications for dSPACE SCALEXIO systems, you must use the *Bus Manager in ConfigurationDesk* instead. You can open projects you used with the Bus Manager (stand-alone) in ConfigurationDesk and continue working with them.

For more information on the Bus Manager (stand-alone), refer to 🕮 *Bus Manager Implementation Guide*.

ConfigurationDesk

Objective

ConfigurationDesk is provided in two variants useful for different scenarios. You can use ConfigurationDesk - Implementation Version to implement real-time applications. You can use ConfigurationDesk - Configuration Version to configure dSPACE RapidPro hardware.

ConfigurationDesk – Implementation

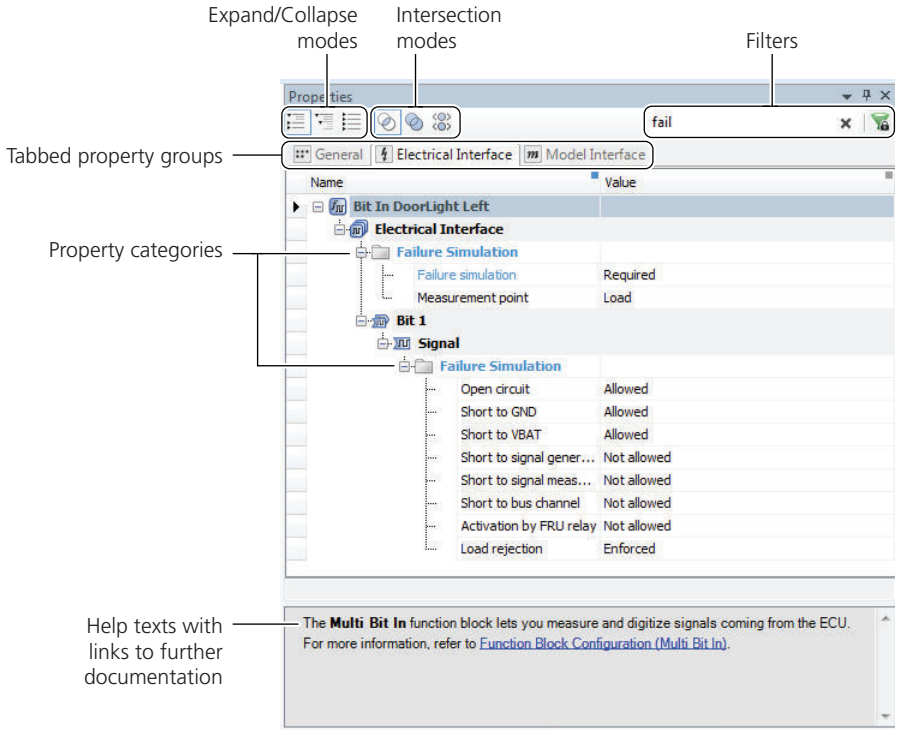
Where to go from here

Information in this section

New Features of ConfigurationDesk 5.5 (Implementation Version)

Enhanced Properties Browser

The Properties Browser has been redesigned to enable structured and targeted access to properties. Different display modes and filter options let you adjust the displayed element hierarchy and properties to your needs.



For example, you can easily apply property settings for a large number of elements of the same type.

Tip

The display modes and filter settings let you structure and reduce the display of properties according to your needs. Available properties are also matched to assigned hardware resources.

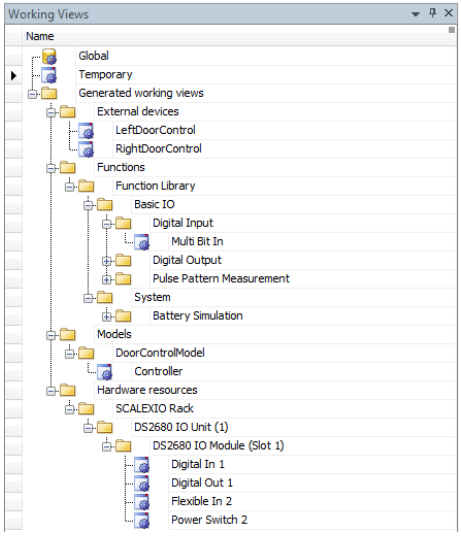
For more information on the Properties Browser's features, refer to *Configuring Signal Chain Elements with the Properties Browser* (🕮 *ConfigurationDesk Real-Time Implementation Guide*).

Improved signal chain handling

Several commands have been added to improve the handling of signal chain elements:

- Select Elements by Type: Lets you select all elements of a specific type from a browser, window, or selection. Works well in conjunction with the redesigned Properties Browser.

- Generate Working Views: From a browser, you can create new working views and working view groups according to the structure of used signal chain elements:



- Show in Browser: Shows selected signal chain elements from a graphical window or table window in the appropriate browser.

Simplified generation of communication interfaces for behavior models

To simplify your work, ConfigurationDesk provides methods for you to easily create communication interfaces for behavior models. For this purpose, ConfigurationDesk lets you create model port blocks that have the same configuration but the inverse data direction as model port blocks that already exist in the model topology. Refer to *Simplified Preparation of Model Interfaces for Model Communication* (🕮 *ConfigurationDesk Real-Time Implementation Guide*).

Note

The Model Interface Package for Simulink also provides this feature, so you can create inverse model port blocks in ConfigurationDesk as well as in your Simulink model.

New features of Simulink implementation container files

Additional properties in the Properties Browser

ConfigurationDesk now displays the following additional SIC file properties in its Properties Browser:

- Format version

  This property displays the format version of the SIC file.

- Exporting tool

  This property displays the tool and its version that the SIC file was exported from.

- Precompiled for

  Displays the following information:

  - The platform that the SIC file was precompiled for
  - The ConfigurationDesk version that the SIC file was precompiled with

New features of the FMU support

Support of precompiled FMUs containing source files
ConfigurationDesk lets you create precompiled FMUs that still contain the original source files. Additionally, your precompiled FMUs can contain precompiled libraries for different dSPACE Releases. Refer to *Creating Precompiled FMUs* (📖 *ConfigurationDesk Real-Time Implementation Guide*).

Additional properties in the Properties Browser
ConfigurationDesk now displays the following additional FMU properties in its Properties Browser:

- Format version

  This property displays the format version of the FMI standard that the FMU complies with.

- Exporting tool

  This property displays the tool and its version that the FMU was exported from.

- Precompiled for

  Displays the following information:

  - The platform that the FMU was precompiled for
  - The ConfigurationDesk version that the FMU was precompiled with

New features of the V-ECU support

Supported V-ECU implementation container versions The following table shows the tool versions that export supported V-ECU implementation containers, and the related container versions:

V-ECU Implementations Created with Products of...	V-ECU Implementation Version
dSPACE Release 2013-B and earlier:	
SystemDesk 3.2	1.0
TargetLink 3.5	1.0
dSPACE Release 2014-A:	
SystemDesk 4.2	2.0
TargetLink 3.5	1.0
dSPACE Release 2014-B:	
SystemDesk 4.3	2.1
TargetLink 4.0	2.1
dSPACE Release 2015-A:	
SystemDesk 4.4	2.2
TargetLink 4.0	2.1
dSPACE Release 2015-B:	
SystemDesk 4.5	2.3
TargetLink 4.1	2.3
dSPACE Release 2016-A:	
SystemDesk 4.6	2.4
TargetLink 4.1	2.3

V-ECU implementations with company-specific operating systems The operating system of a V-ECU implementation can now be a company-specific operating system. V-ECU implementations with company-specific operating systems are created with SystemDesk's V-ECU Manager.

Enhanced function block types

Current Signal Capture The Current Signal Capture function block provides the following new features:

- You can use the input voltage as the trigger source (digital threshold-based triggering) that is used to start a sequence. You have to specify a threshold value and the edge direction for the trigger. A sequence is started when the measured voltage input signal matches the specified trigger condition.

- Capturing the digital states (0 or 1) of the input voltage for each sample in a sequence. The states can be written to the behavior model via a specific function port. You have to specify a threshold value to differentiate between the two digital states.

For more information, refer to Configuring the Basic Functionality (Current Signal Capture).

CAN The CAN function block now supports the low power mode according to ISO 11898-5 for CAN communication. When enabled, low power mode is supported for the channels of the assigned hardware. Via a specifc function port you can provide the required low power mode (stand-by, sleep, silent, normal) from within the behavior model to control the CAN transceiver. In addition, if the transceiver detects a wake-up in the bus communication, a status flag can be written to the behavior model.

For more information, refer to *CAN* (🕮 *ConfigurationDesk I/O Function Implementation Guide*).

Function block types providing digital output signals For function blocks that provide digital output signals (such as the Multi Bit Out function block) the digital output configuration has been improved. This configuration defines how the digital outputs are operated to get the required binary output signal (0 or 1). Possible configurations are high-side switch, low-side switch or push-pull configuration.

Up to ConfigurationDesk 5.4 (on dSPACE Release 2015-B) the setting for the Digital output mode property is determined from the settings of the High reference potential and Reference potential properties and the channel type. As of ConfigurationDesk 5.5 (on dSPACE Release 2016-A) you can select the digital output configuration directly via the Interface type property.

If you migrate former applications to dSPACE Release 2016-A, no additional work is required, because the Digital output mode property setting is transferred to the Interface type property setting by the software.

New features of the Bus Manager

Configuring bus communication for offline simulation applications The Bus Manager now lets you configure CAN and LIN bus communication to implement it in offline simulation applications. After you configure the bus communication, the Bus Manager lets you generate bus simulation container (BSC) files. You can use the BSC files to implement the configured bus communication in offline simulation applications and perform restbus simulation on VEOS.

For more information, refer to *Working with Bus Simulation Containers* (🕮 *ConfigurationDesk Bus Manager Implementation Guide*).

Configuring bus communication via bus configuration features The Bus Manager now provides various bus configuration features that let you configure the bus communication of bus configurations. Each bus configuration feature provides feature-specific settings that you can configure for run time. The following bus configuration features are available:

- ISignal Value
- IPDU Raw Data
- IPDU Trigger
- Communication Controller Enable
- LIN Schedule Table

For more information, refer to *Working with Bus Configuration Features* (🕮 *ConfigurationDesk Bus Manager Implementation Guide*).

New features concerning hardware support

ConfigurationDesk supports the following new SCALEXIO hardware:

- SCALEXIO LabBox

  SCALEXIO LabBox provides 19 slots for up to 18 standard SCALEXIO I/O boards plus one DS6051 IOCNET Router.

- DS6301 CAN/LIN Board

  The DS6301 provides four CAN and four LIN channels for bus communication.

Migrating to ConfigurationDesk 5.5

Possible M script adjustments for automation

Value returned for ICaApplicationMain: SetCustomInformation and ICaComponent: Configure `ICaApplicationMain: SetCustomInformation` and `ICaComponent: Configure` now return a value. In most cases, this value is `None` (e.g., in Python).

Note

Even if the returned value is not used, M script clients should be aware that a printout follows after calling one of the methods if no semicolon ends the statement. This can cause unexpected output from existing scripts.

	For more information on automation changes, refer to *Changes to the Automation Interface for Release 2016-A* (🕮 *ConfigurationDesk Automating Tool Handling*)
Hardware topology containing a DS2671 Bus Board	If you migrate to dSPACE Release 2016-A and your hardware topology contains a DS2671 Bus Board, note the following: After migration, the DS2671 does not support the features and configuration settings introduced with former dSPACE Releases (e.g., CAN FD mode). Workaround: Replace the hardware topology after migration.
Discontinuation of platform management automation API version 1.0 as of dSPACE Release 2016-B	Platform management automation API version 1.0 is being supported for the last time with ConfigurationDesk 5.5 from dSPACE Release 2016-A. Refer to *Automating Platform Management* (🕮 *ConfigurationDesk Automating Tool Handling*).

ControlDesk Next Generation

Where to go from here

Information in this section

Information in other sections

🕮 *ControlDesk Next Generation Introduction and Overview* Introduces ControlDesk Next Generation.

New Features of ControlDesk Next Generation (ControlDesk 5.6)

Where to go from here

Information in this section

New Features of Platform Management and Platforms/Devices (ControlDesk 5.6)

Information in this topic

CAN/LIN channels of SCALEXIO and VEOS as bus interfaces

CAN/LIN channels of SCALEXIO and VEOS can now be selected as bus interfaces for the following devices:

- Bus monitoring devices
  - CAN Bus Montoring device

    Refer to *How to Configure a CAN Bus Monitoring Device* (🕮 *ControlDesk Next Generation Platform Management*).
  - LIN Bus Montoring device

    Refer to *How to Configure a LIN Bus Monitoring Device* (🕮 *ControlDesk Next Generation Platform Management*).
- Measurement and calibration devices
  - CCP device

    Refer to *How to Configure a CCP Device* (🕮 *ControlDesk Next Generation Platform Management*).
  - XCP on CAN device

    Refer to *How to Configure an XCP on CAN Device* (🕮 *ControlDesk Next Generation Platform Management*).

LIN Bus Monitoring device: Support for FIBEX and AUTOSAR system description files

The LIN Bus Monitoring device now also supports the following variable description file formats in addition to LDF:

- FIBEX:
  - Version 4.1.0, 4.1.1
  - Version 3.1.0, 3.1.1
  - Version 3.0.0
- AUTOSAR system description files according to the AUTOSAR system template:
  - Version 4.2.2
  - Version 4.2.1
  - Version 4.1.1 ... 4.1.3
  - Version 4.0.3
  - Version 3.2.1 ... 3.2.3
  - Version 3.1.4

Note

- Keep the migration aspects in mind when you reuse an experiment that was originally created with ControlDesk 5.5 or earlier and contains a LIN Bus Monitoring device. Refer to *Migrating from ControlDesk 5.5 to 5.6* (🕮 *ControlDesk Next Generation Migration*).
- LDF files (format version 1.2 and earlier) were supported by the LIN Bus Monitoring device for the last time with ControlDesk 5.5 (dSPACE Release 2015-B).

LIN Bus Monitoring device: Variable observer functionality

ControlDesk's variable observer functionality is now also supported for the LIN Bus Monitoring device.

Refer to *Observing Variables* (🕮 *ControlDesk Next Generation Measurement and Recording*).

SCALEXIO and DS1007 platforms: Naming processing unit/processor board during registration

You can specify a name for each processing unit of a SCALEXIO system and processor board of a DS1007 system during system registration.

Refer to *Register Platforms* (🕮 *ControlDesk Next Generation Platform Management*).

New Variable Management Features (ControlDesk 5.6)

Support of A2L 1.7 files

The ASAM e.V. association (Association for Standardisation of Automation and Measuring Systems) recently released version 1.7 of the ASAM MCD-2 MC standard. ControlDesk now supports A2L files of this version.

For more information on ControlDesk's support for A2L 1.7 files, refer to *Basics on Importing A2L Files of Version 1.7* (🕮 *ControlDesk Next Generation Variable Management*)

Structured data types in A2L files With version 1.7 of the ASAM MCD-2 MC standard, the ASAM e.V. association introduced the *definition of structured data types* to the standard.

ControlDesk's support for struct variables ControlDesk supports struct variables, i.e., variables with structured data types. A struct variable contains a structured list of variables that can have various data types. In ControlDesk, a struct variable can contain either

parameters and value blocks or measurement variables and measurement arrays. ControlDesk supports nested structs, i.e., structs that contain further structs and/or struct arrays as elements.

In ControlDesk's Variable Browser, a struct variable can be identified by the  icon.

For information on variable types in ControlDesk, refer to *Basics on Variable Types* (🕮 *ControlDesk Next Generation Variable Management*).

Visualizing struct variables in ControlDesk You can connect the variables contained in a struct variable to different instruments in one step by customizing the connection assignment.

For instructions, refer to *How to Customize the Connection Assignment of Variables to Instruments* (🕮 *ControlDesk Next Generation Layouting*).

New Layouting Features (ControlDesk 5.6)

Visualizing entire struct arrays

ControlDesk now lets you visualize entire struct arrays - arrays of homogeneous structs. You can connect the contained variables to different instruments in one step by customizing the connection assignment. For instructions, refer to *How to Customize the Connection Assignment of Variables to Instruments* (🕮 *ControlDesk Next Generation Layouting*).

New Instrument Features (ControlDesk 5.6)

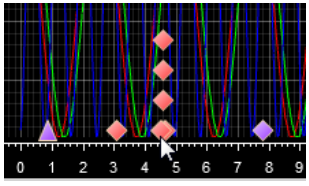
Custom properties of instruments

ControlDesk now lets you add and configure custom properties of instruments. You can let an instrument script be executed when a custom property of an instrument changes. This allows you to extend the instrument functionality.

Refer to *Adding Custom Properties to an Instrument* (🕮 *ControlDesk Next Generation Customization*).

Time Plotter enhancements

Grouped bookmarks If bookmarks overlap other bookmarks in the display of the Time Plotter, they are represented by a grouped bookmark symbol. Move the pointer onto a grouped bookmark symbol to access the single bookmark symbols it contains.



Refer to *Basics on Bookmarks* (🕮 *ControlDesk Next Generation Measurement and Recording*).

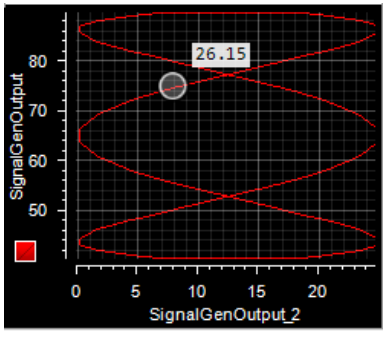
Enabling/Disabling the display of bookmarks The Time Plotter's Cursor toolbar now lets you enable/disable the display of bookmarks. Refer to *Show Toolbar* (🕮 *ControlDesk Next Generation Instrument Handling*).

Hours tic format You can display the values on the x-axis of a Time Plotter in the Hours tic format (`hh:mm:ss`).

Refer to *Axes and Signal Properties (Time Plotter/Index Plotter)* (🕮 *ControlDesk Next Generation Instrument Handling*).

XY Plotter enhancements

Time cursor available The XY Plotter now lets you enable a *time cursor* to specify a time position in the chart.



Refer to *Time Cursor Properties (Time Plotter / XY Plotter)* (🕮 *ControlDesk Next Generation Instrument Handling*).

Background picture available The XY Plotter now lets you specify a background picture.

Refer to *Picture Properties* (🕮 *ControlDesk Next Generation Instrument Handling*).

Browser instrument enhancements

Connection of variables You can connect variables to a Browser by specifying one or more connection nodes and then dragging variables on the instrument. If you combine the connected variables with code stored in the file archive (HTML, Java Script, JSON, ...), you can create custom Browser instruments with web controls.

File archive You can add various files, such as HTML files including Java Script, to the Browser's file archive. The files are saved with the Browser and are also available if you add the Browser to the Custom Instruments category in the Instruments Selector.

Refer to *Basics on Handling the Browser* (🕮 *ControlDesk Next Generation Instrument Handling*).

New Measurement and Recording Features (ControlDesk 5.6)

Bookmark when invoking an XIL API EESPort manual trigger

ControlDesk now lets you specify whether to automatically set a bookmark during a measurement or recording when you invoke an XIL API EESPort manual trigger via the *Trigger (Error Configuration)* (🕮 *ControlDesk Next Generation Electrical Error Simulation via XIL API EESPort*) command. The bookmark type is *XIL API EESPort Manual Trigger*.

Refer to *Edit Bookmark Settings* (🕮 *ControlDesk Next Generation Measurement and Recording*).

New Data Set Management Features (ControlDesk 5.6)

Filtering the Data Set Manager's parameter list directly

The Data Set Manager now lets you filter the parameter list *directly*.

For reference information on the Data Set Manager, refer to *Data Set Manager* (🕮 *ControlDesk Next Generation Calibration and Data Set Management*).

New Bus Navigator Features (ControlDesk 5.6)

Information in this topic

CAN/CAN FD/LIN bus monitoring support for SCALEXIO and VEOS

ControlDesk's Bus Navigator now supports:

- CAN/CAN FD/LIN bus monitoring on SCALEXIO
- CAN/LIN bus monitoring on VEOS

The Monitoring List now also displays the bus load when you perform bus monitoring on SCALEXIO or on VEOS.

To perform bus monitoring on SCALEXIO or VEOS:

1. Add a CAN or LIN bus monitoring device to a ControlDesk experiment.
2. Configure the device to use a controller of a registered SCALEXIO or VEOS platform.

For instructions, refer to *How to Configure a CAN Bus Monitoring Device* (🕮 *ControlDesk Next Generation Platform Management*) / *How to Configure a LIN Bus Monitoring Device* (🕮 *ControlDesk Next Generation Platform Management*).

LIN Bus Monitoring device: Support for FIBEX and AUTOSAR system description files

The Bus Navigator now also supports FIBEX and AUTOSAR system description files as variable description file formats in connection with the LIN Bus Monitoring device. Refer to *LIN Bus Monitoring device: Support for FIBEX and AUTOSAR system description files* on page 95.

Note

- Keep the migration aspects in mind when you reuse an experiment that was originally created with ControlDesk 5.5 or earlier and contains a LIN Bus Monitoring device. Refer to *Migrating from ControlDesk 5.5 to 5.6* (🕮 *ControlDesk Next Generation Migration*).
- LDF files (format version 1.2 and earlier) were supported by the LIN Bus Monitoring device for the last time with ControlDesk 5.5 (dSPACE Release 2015-B).

Support of AUTOSAR system template version 4.2.2

ControlDesk now also supports AUTOSAR system template version 4.2.2 in connection with the following devices:

- CAN Bus Monitoring device
- FlexRay Bus Monitoring device
- LIN Bus Monitoring device

Specifying a default format for IDs and data

You can specify whether the decimal or hexadecimal format is the default for displaying IDs and bus data. The default format is used in the Monitoring List and in Bus instruments, for example. You can change the format later on.

Refer to *Bus Navigator Page* (🕮 *ControlDesk Next Generation Bus Navigator*).

Bus instrument generation for VEOS

ControlDesk's Bus Navigator now lets you generate Bus instruments on the basis of EXPSWCFG configuration data files created by ConfigurationDesk's Bus Manager for use with VEOS.

Instrument generation is supported for the following communication protocols:

- CAN
- LIN

Refer to *Working with Instruments of the Bus Navigator* (🕮 *ControlDesk Next Generation Bus Navigator*).

Bus Instrument (RX Type for CAN): Enhancement for RTI CAN MultiMessage Blockset applications

Display of CRC status and error value The Bus instrument (RX Type for CAN) can display the CRC status and error value of a CAN message if the bus configuration is based on a Simulink model containing blocks from the RTI CAN MultiMessage Blockset.

Refer to *Bus Instrument (RX Type for CAN)* (🕮 *ControlDesk Next Generation Bus Navigator*).

Bus Instrument (TX Status for CAN and LIN) for Bus Manager applications

The Bus Navigator now also provides a Bus Instrument (TX Status for CAN and LIN) for Bus Manager applications.

Enabling/disabling communication controllers The instrument lets you enable or disable communication controllers.

Triggering IPDU transmission The instrument lets you specify trigger access options for IPDUs.

For reference information on the Bus Instrument (TX Status), refer to:

- *Bus Instrument (TX Status Type for CAN)* (🕮 *ControlDesk Next Generation Bus Navigator*)
- *Bus Instrument (TX Status Type for LIN)* (🕮 *ControlDesk Next Generation Bus Navigator*)

Bus Instrument (TX Type for CAN and LIN): Enhancement for Bus Manager applications

Displaying raw data of an IPDU The instrument lets you display the raw data for each byte of an IPDU. You can also specify a substitute value to be transmitted.

Triggering IPDU transmission The instrument lets you specify trigger access options for the selected IPDU.

For reference information on the Bus Instrument (TX Type), refer to:

- *Bus Instrument (TX Type for CAN)* (🕮 *ControlDesk Next Generation Bus Navigator*)
- *Bus Instrument (TX Type for LIN)* (🕮 *ControlDesk Next Generation Bus Navigator*)

Bus Instrument (RX Type for CAN and LIN): Enhancement for Bus Manager applications

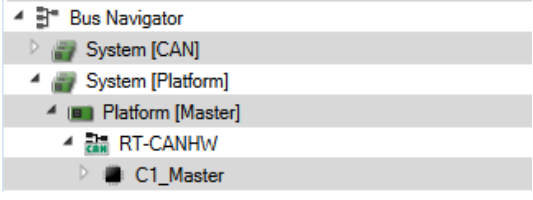
Displaying raw data of an IPDU The instrument lets you display the raw data for each byte of an IPDU.

For reference information on the Bus Instrument (RX Type), refer to:

- *Bus Instrument (RX Type for CAN)* (🕮 *ControlDesk Next Generation Bus Navigator*)
- *Bus Instrument (RX Type for LIN)* (🕮 *ControlDesk Next Generation Bus Navigator*)

Bus Navigator controlbar enhancements

Alternating row colors You can specify alternating row colors for the Bus Navigator controlbar.



Refer to *Variables Page* (🕮 *ControlDesk Next Generation Variable Management*).

Specifying the column to be searched The Bus Navigator controlbar now lets you select the column that is used for searching or filtering.

Refer to *Bus Navigator* (🕮 *ControlDesk Next Generation Bus Navigator*)

Incremental search The Bus Navigator controlbar now supports incremental search, including wildcards.

Refer to *Bus Navigator* (🕮 *ControlDesk Next Generation Bus Navigator*).

Case-sensitive search You can now specify whether the search in the Bus Navigator controlbar is case-sensitive.

Refer to *Case-Sensitive Search (Bus Navigator)* (🕮 *ControlDesk Next Generation Bus Navigator*).

Monitoring list: Count column	To displays the count of a message/frame, the Monitoring list now lets you display the Count column. Refer to *Customize Columns (Monitoring List)* (🕮 *ControlDesk Next Generation Bus Navigator*).

New ECU Diagnostics Features (ControlDesk 5.6)

Automating the execution of ECU flash sessions	ControlDesk lets you automate the execution of ECU flash programming sessions. For more information on programming, refer to *Automating ECU Diagnostics Tasks* (🕮 *ControlDesk Next Generation ECU Diagnostics*).
Configuring the job execution for security access	When you configure an ECU Diagnostics device, you can now specify whether ControlDesk executes a diagnostic job for security access when online calibration is started for that device via the Security access execution behavior option. For instructions, refer to *How to Configure an ECU Diagnostics Device* (🕮 *ControlDesk Next Generation Platform Management*).
Executing the protocol-specific TesterPresent service	When you specify the behavior of the TesterPresent service used to check the connection to the ECU, you can now let ControlDesk execute the TesterPresent service that matches the diagnostic

protocol via the Send the protocol-specific TesterPresent service setting of the TesterPresent behavior option. With this setting, ControlDesk sends TesterPresent messages with request and response PDU values that depend on the diagnostic protocol.

For more information, refer to *Configure Platform/Device* (🕮 *ControlDesk Next Generation Platform Management*).

Evaluation of the suppress positive response bit during ECU connection checks

If you use the UDS diagnostic protocol, ControlDesk now evaluates the `SuppressPositiveResponseBit` flag when executing the services used by `startCommunication` and `stopCommunication`.

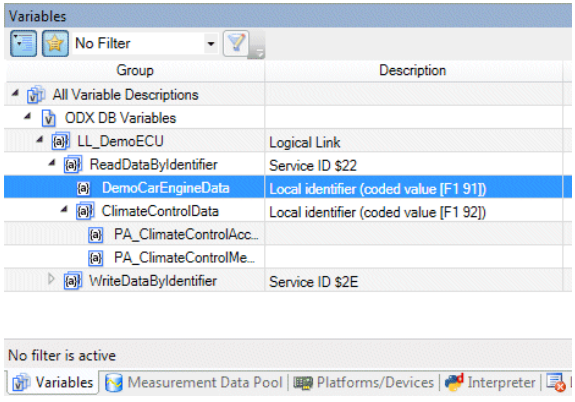
For more information, refer to *Conventions in Connection with ODX Databases* (🕮 *ControlDesk Next Generation ECU Diagnostics*).

Diagnostics variables: Descriptions for some block group types

The tree view of ControlDesk's Variable Browser displays the variables according to the structure in the related variable description. Descriptions for some block group types now help you identify the variables.

These are some examples:

- For a logical link block group, the description text is 'Logical link'.
- The description of a service block group displays the associated service ID.
- The description of the block group of a local identifier in a diagnostic service shows the coded value of the associated service parameter.



Refer to *Measuring and Calibrating Variables via the ECU Diagnostics Device* (🕮 *ControlDesk Next Generation ECU Diagnostics*).

New Electrical Error Simulation Features (ControlDesk 5.6)

Monitoring the switching behavior of failure simulation hardware

You can monitor the switching behavior and transition states of the failure simulation hardware via specific measurement variables. These variables are independent from other model variables.

For more information, refer to *Basics on Monitoring the Switching Behavior of the Failure Simulation Hardware* (🕮 *ControlDesk Next Generation Electrical Error Simulation via XIL API EESPort*).

Automating EESPort configurations

ControlDesk lets you add and configure XIL API EESPort configurations.

For reference information on the API, refer to *Automating Electrical Error Simulation via XIL API EESPort* (🕮 *ControlDesk Next Generation Electrical Error Simulation via XIL API EESPort*).

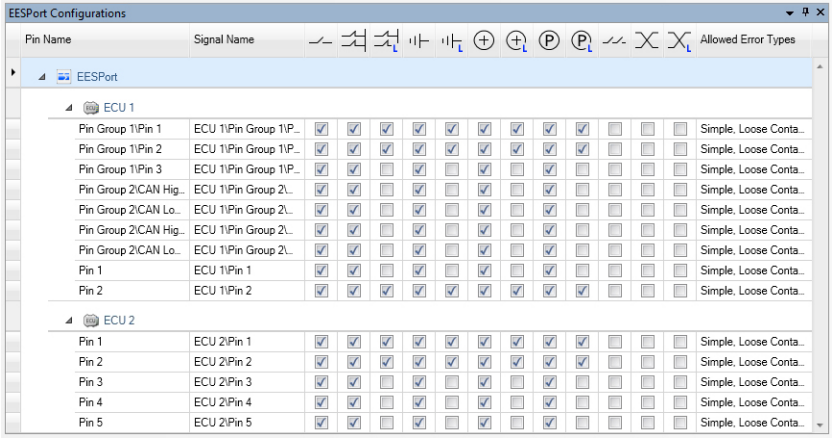
Specifying whether ControlDesk disconnects a concurrent client

You can now specify whether ControlDesk automatically disconnects another client that might be connected to the SCALEXIO failure simulation hardware when you configure the XIL API EESPort. You can specify this behavior via the Override access EESPort property.

For reference information, refer to *EESPort - Configurations Properties* (🕮 *ControlDesk Next Generation Electrical Error Simulation via XIL API EESPort*).

Enhanced EESPort Configurations controlbar

The EESPort Configurations controlbar now displays icons for the allowed error categories of each ECU pin of a hardware-in-the-loop (HIL) simulator. The following illustration shows an example configuration:



	For reference information on the controlbar, refer to *EESPort Configurations* (🕮 *ControlDesk Next Generation Electrical Error Simulation via XIL API EESPort*).
Bookmark when invoking an XIL API EESPort manual trigger	ControlDesk now lets you specify whether to automatically set a bookmark during a measurement or recording when you invoke an XIL API EESPort manual trigger via the *Trigger (Error Configuration)* (🕮 *ControlDesk Next Generation Electrical Error Simulation via XIL API EESPort*) command. The bookmark type is *XIL API EESPort Manual Trigger*. Refer to *Edit Bookmark Settings* (🕮 *ControlDesk Next Generation Measurement and Recording*).
Replacing a signal in all error sets in a single step	You can replace a signal in all the error sets of an error configuration *in a single step*. For instructions, refer to *Tips and Tricks for Configuring Electrical Errors* (🕮 *ControlDesk Next Generation Electrical Error Simulation via XIL API EESPort*).

New Automation Features (ControlDesk 5.6)

Automating the execution of ECU flash sessions	ControlDesk lets you automate the execution of ECU flash programming sessions. For more information on programming, refer to *Automating ECU Diagnostics Tasks* (🕮 *ControlDesk Next Generation ECU Diagnostics*).
Automating EESPort configurations	ControlDesk lets you add and configure XIL API EESPort configurations. For reference information on the API, refer to *Automating Electrical Error Simulation via XIL API EESPort* (🕮 *ControlDesk Next Generation Electrical Error Simulation via XIL API EESPort*).
Accessing open and closed documents via the component-specific interface	**Accessing open and closed layouts via `LayoutManagement / IXaLayoutManagement <<Interface>>`** Via the `LayoutManagement / IXaLayoutManagement <<Interface>>` interface, you can access ■ Open layouts via the interface's `Layouts / IXaLayouts <<Collection>>` collection property ■ Closed layouts via the interface's `Files / IXaFiles <<Collection>>` collection property

For more information, refer to *Automating Visualization of Variables on Layouts* (🕮 *ControlDesk Next Generation Automation*).

Accessing open and closed measurement data files via `MeasurementDataManagement / IXaMeasurementDataManagement <<Interface>>` Via the `MeasurementDataManagement / IXaMeasurementDataManagement <<Interface>>` interface, you can access

- Open measurement data files via the interface's `Measurements / IXaMeasurements <<Collection>>` collection property
- Closed measurement data files via the interface's `Files / IXaFiles <<Collection>>` collection property

For more information, refer to *Automating Measurement and Recording* (🕮 *ControlDesk Next Generation Automation*).

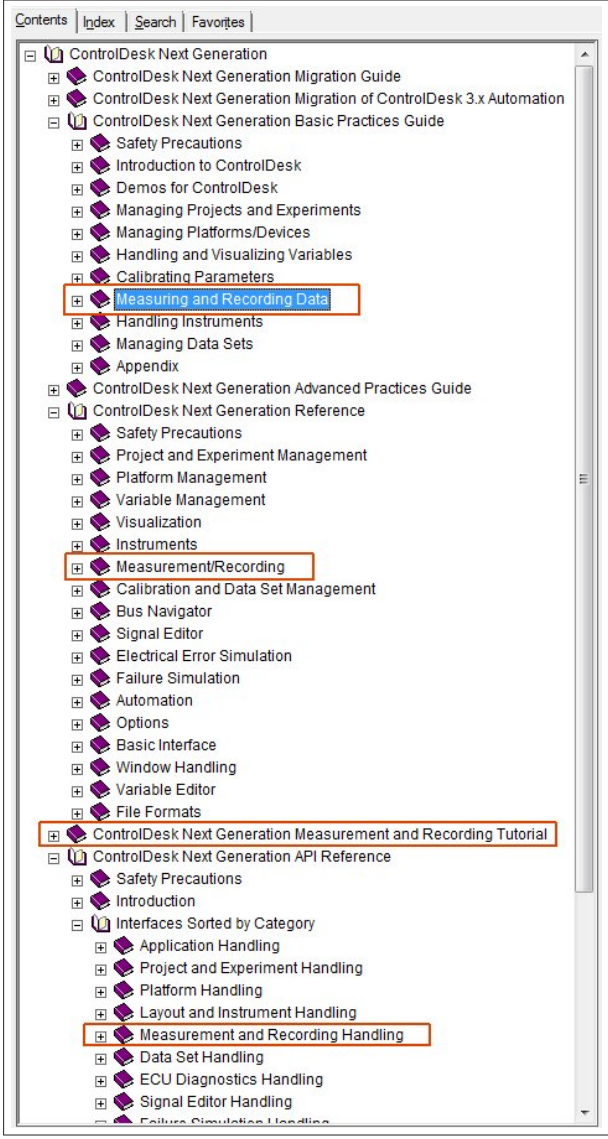
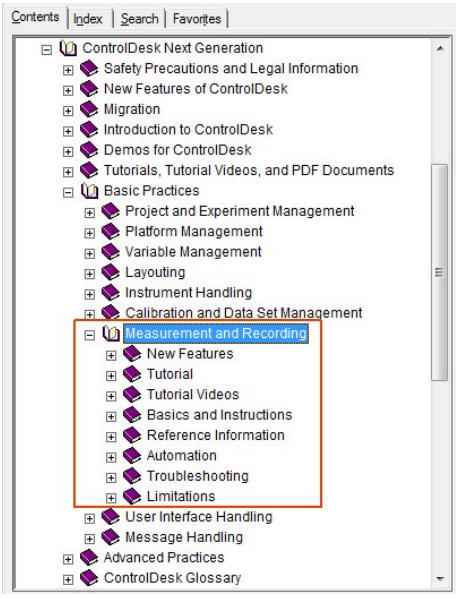
Further Enhancements with ControlDesk Next Generation (ControlDesk 5.6)

Improved user documentation

Subject-oriented user documentation As of ControlDesk 5.6, the structure of the user documentation is *subject-oriented*, i.e., you can find the entire documentation for a specific subject such as *Measurement and Recording* or *Signal Editor* under a single node in dSPACE HelpDesk.

Up to and including ControlDesk 5.5, the structure of the user documentation was *document-oriented*, i.e., information on a specific subject was spread over different documents such as the *ControlDesk Next Generation Basic Practices Guide* and the *ControlDesk Next Generation Reference*.

The following table shows a comparison of the structures as an example:

Document-Oriented Structure (up to and Including ControlDesk 5.5)	Subject-Oriented Structure (as of ControlDesk 5.6)
	

Subject-orientation and PDF files Following the new subject-orientation of the ControlDesk user documentation, there is one PDF file containing the entire user documentation for each subject of ControlDesk.

For a list of the PDF files available for ControlDesk, refer to *PDF Documents for ControlDesk* (🕮 *ControlDesk Next Generation Introduction and Overview*).

Tutorial videos There are tutorial videos available for the following subjects:

- Measuring on dSPACE platforms
- Electrical error simulation
- Customizing ControlDesk instruments

Refer to https://www.dspace.com/go/controldesktutorial (requires a mydSPACE login).

For public product videos, refer to ControlDesk product videos.

Migrating to ControlDesk Next Generation (ControlDesk 5.6)

Where to go from here

Information in this section

Information in other sections

📖 *ControlDesk Next Generation Introduction and Overview*
Introduces ControlDesk Next Generation.

Discontinuations in ControlDesk

Information in this topic

Discontinuations as of ControlDesk 5.6

MicroAutoBox software support dSPACE Release 2015-B was the last release supporting MicroAutoBox with its variants 1401/1501, 1401/1504, 1401/1505/1506, 1401/1505/1507, and 1401/1507.

As of dSPACE Release 2016-A, dSPACE software supports only MicroAutoBox II with its variants 1401/1507, 1401/1511, 1401/1513, 1401/1511/1514, and 1401/1513/1514.

For migration aspects, refer to *Migrating to ControlDesk Next Generation (ControlDesk 5.6)* on page 114.

MicroAutoBox Embedded PC support dSPACE Release 2015-B was the last release supporting 32-bit operating systems.

As a consequence, MicroAutoBox Embedded PC with Intel® Atom™ Processor N270 with the Windows 7 (32 bit) operating system was supported for the last time with ControlDesk 5.5 (dSPACE Release 2015-B).

ControlDesk's ASAP3 interface ControlDesk's ASAM ASAP3-compatible interface was delivered for the last time with ControlDesk 5.5 (dSPACE Release 2015-B).

To automate calibration and measurement tasks, you can alternatively use:

- ControlDesk's automation interface. Refer to *Introduction to the ControlDesk Automation API* (🕮 *ControlDesk Next Generation Automation*).
- ControlDesk's ASAM MCD-3-compatible interface. Refer to 🕮 *ControlDesk Next Generation MCD-3 Automation*.

CDF import/export The Calibration Data File (CDF) format used to import/export data sets was supported for the last time with ControlDesk 5.5 (dSPACE Release 2015-B).

To exchange calibration data, use one of the other file formats supported by ControlDesk such as CDFX (ASAM Calibration Data File 2.0), DCM, or DSV. The CDFX format is ControlDesk's default exchange format for data sets.

Refer to *Exporting and Converting Data Sets* (🕮 *ControlDesk Next Generation Calibration and Data Set Management*).

User-defined databases (UDDBs) User-defined databases (UDDBs) were supported for the last time with ControlDesk 5.5 (dSPACE Release 2015-B).

As a consequence, to replace the UDDB-based manipulation of CAN communication on dSPACE real-time hardware, you have to change the real-time model.

For information on migration aspects, refer to *Migrating to ControlDesk Next Generation (ControlDesk 5.6)* on page 114.

LDF (format version 1.2 and earlier) LDF files (format version 1.2 and earlier) were supported by the LIN Bus Monitoring device for the last time with ControlDesk 5.5 (dSPACE Release 2015-B).

MAT file (version 6) export ControlDesk 5.5 and earlier created version 6 MAT files that can be loaded in MATLAB Versions 5 (R8) or later.

As of version 5.6, ControlDesk creates version 7.3 MAT files that can be loaded in MATLAB Versions 7.3 (R2006b) or later.

Discontinuations for ControlDesk as of dSPACE Release 2016-B

ControlDesk Failure Simulation Module ControlDesk's Failure Simulation Module is being delivered for the last time with ControlDesk 5.6 from dSPACE Release 2016-A.

- To prepare electrical error simulation via the graphical user interface (GUI), use the ControlDesk XIL API EESPort GUI, which is introduced with ControlDesk 5.5 (dSPACE Release 2015-B).
- To prepare electrical error simulation via automation, use the dSPACE XIL API .NET implementation supporting the Electrical Error Simulation Port (EESPort).

For information on migration aspects, refer to *Migrating to ControlDesk Next Generation (ControlDesk 5.6)* on page 114.

Platform management automation API version 1.0 Platform management automation API version 1.0 is being supported for the last time with ControlDesk 5.6 from dSPACE Release 2016-A.

For information on migrating to API version 2.0, which was introduced with ControlDesk 5.2 from dSPACE Release 2014-A, refer to *Migrating from ControlDesk 5.1 to 5.2* (🕮 *ControlDesk Next Generation Migration*).

Plotter The Plotter is being delivered for the last time with ControlDesk 5.6 from dSPACE Release 2016-A.

Use one of the following instruments instead:

- Index Plotter
- Time Plotter
- XY Plotter

For information on the differences between the different plotter types, refer to *Differences Between Plotter, Time Plotter, Index Plotter, and XY Plotter* (🕮 *ControlDesk Next Generation Instrument Handling*).

Table Editor The Table Editor is being delivered for the last time with ControlDesk 5.6 from dSPACE Release 2016-A.

It will be replaced by an enhanced Table Editor.

MDF (format versions 2.0 and 3.0) export The export of MDF measurement data files (MDF file format versions 2.0 and 3.0) is being supported for the last time with ControlDesk 5.6 from dSPACE Release 2016-A.

Support for *importing* MDF files (format versions 2.0 and 3.0) will continue.

To export measurement data, use one of the other file formats supported by ControlDesk. Refer to *How to Configure the Storage Settings for Recording* (🕮 *ControlDesk Next Generation Measurement and Recording*).

Methods for handling messages As of dSPACE Release 2016-B, all dSPACE products use improved methods for handling messages such as errors and warnings.

As a consequence:

- Messages are no longer written to the `dSPACE.log` file, i.e., they are no longer available as plain text.

  To collect diagnostics information including log messages and send it to dSPACE Support, *use the dSPACE Installation Manager*.
- The dSPACE Message Monitor, which allows you to monitor log messages that are recorded by the dSPACE Message Service, is being supported for the last time with ControlDesk 5.6 from dSPACE Release 2016-A.
- The `LogFilePath` property of the `Log / ILoLog <<Interface>>`, which gets the full path name of the message log file, is being supported for the last time with ControlDesk 5.6 from dSPACE Release 2016-A.

Migration of ControlDesk 3.x experiments The migration of ControlDesk 3.x experiments for reuse in ControlDesk Next Generation is being supported for the last time with ControlDesk 5.6 from dSPACE Release 2016-A.

Tip

To reuse a ControlDesk 3.x experiment in ControlDesk from dSPACE Release 2016-B or later:

1. Migrate the ControlDesk 3.x experiment using ControlDesk from dSPACE Release 2016-A or earlier. Refer to *Migrating from ControlDesk 3.x Experiments* (🕮 *ControlDesk Next Generation Migration*).
2. Migrate the project from ControlDesk from dSPACE Release 2016-A or earlier to ControlDesk from dSPACE Release 2016-B or later. Refer to *Migrating from Prior Versions of ControlDesk Next Generation* (🕮 *ControlDesk Next Generation Migration*).

Migration of CalDesk projects The migration of CalDesk projects for reuse in ControlDesk Next Generation is being supported for the last time with ControlDesk 5.6 from dSPACE Release 2016-A.

Tip

To reuse a CalDesk project in ControlDesk from dSPACE Release 2016-B or later:

1. Migrate the CalDesk project using ControlDesk from dSPACE Release 2016-A or earlier. Refer to *Migration from CalDesk* (🕮 *ControlDesk Next Generation Migration*).
2. Migrate the project from ControlDesk from dSPACE Release 2016-A or earlier to ControlDesk from dSPACE Release 2016-B or later. Refer to *Migrating from Prior Versions of ControlDesk Next Generation* (🕮 *ControlDesk Next Generation Migration*).

Migrating to ControlDesk Next Generation (ControlDesk 5.6)

To migrate from ControlDesk 5.5 to ControlDesk 5.6 and reuse existing experiments, you might have to carry out the following migration steps.

Note

To migrate to ControlDesk 5.6 from versions earlier than 5.5, you might also have to perform the migration steps of the intervening ControlDesk versions.

Information in this topic

MicroAutoBox platform: Migrating experiments

dSPACE Release 2015-B was the last release supporting MicroAutoBox with its variants 1401/1501, 1401/1504, 1401/1505/1506, 1401/1505/1507, and 1401/1507.

As of dSPACE Release 2016-A, dSPACE software supports only MicroAutoBox II with its variants 1401/1507, 1401/1511, 1401/1513, 1401/1511/1514, and 1401/1513/1514.

ControlDesk 5.6 and later lets you open an experiment with a MicroAutoBox platform configured for an unsupported MicroAutoBox variant, and reconfigure the platform so it can be reused with MicroAutoBox II.

To do this, perform the following steps:

1. Open the experiment to be reused in ControlDesk 5.6 or later.

   ControlDesk changes the MicroAutoBox platform's connection type automatically from BUS to NET, and prompts you to specify the IP address of MicroAutoBox II.

2. Specify the IP address of MicroAutoBox II.

The experiment can now be reused with MicroAutoBox II.

Failure Simulation Module: Discontinuation and migration

ControlDesk's Failure Simulation Module is being delivered for the last time with ControlDesk 5.6 from dSPACE Release 2016-A.

- To prepare electrical error simulation via the graphical user interface (GUI), use the ControlDesk XIL API EESPort GUI, which is introduced with ControlDesk 5.5 (dSPACE Release 2015-B).

  To use the ControlDesk XIL API EESPort GUI, the Failure Simulation Package is required, which is based on XIL API's EESPort. The implementation is based on dSPACE XIL API .NET.

  Keep in mind that electrical error configurations of ControlDesk's Failure Simulation Module are not compatible with XIL API EESPort configurations.

  For migration, you can use the `FailureSimulationExportTool` to export information from an existing ControlDesk failure simulation system (FSN) file to the following files:

  - A hardware-dependent port configuration (PORTCONFIG) file

    You can use the file to create a new EESPort. For instructions, refer to *How to Create a New EESPort* (🕮 *ControlDesk Next Generation Electrical Error Simulation via XIL API EESPort*).

- One error configuration XML file for each failure pattern

  You can use the files to create and configure electrical errors, refer to *How to Create and Configure an Electrical Error* (🕮 *ControlDesk Next Generation Electrical Error Simulation via XIL API EESPort*).

The `FailureSimulationExportTool` version to use depends on the installed version of ControlDesk and dSPACE XIL API .NET as shown in the following table:

Installed ControlDesk Version	Installed dSPACE XIL API .NET Version	Required FailureSimulationExportTool Version
5.3	2.0	2014-B
5.4	2015-A	2015-A
5.5	2015-B	2015-B
5.6	2016-A	2016-A

You can download the `FailureSimulationExportTool`, including a ReadMe file containing user documentation, from the *ControlDesk Next Generation Product Support Center* at http://www.dspace.com/cdngpsc.

- To prepare electrical error simulation via automation, use the dSPACE XIL API .NET implementation supporting the Electrical Error Simulation Port (EESPort).

LIN Bus Monitoring device: Repairing layout connections due to changed LDF import

As of version 5.6, ControlDesk's LDF file import in connection with the LIN Bus Monitoring device supports FIBEX and AUTOSAR system description files. As a consequence, paths to variables in an LDF file are different depending on whether you import the LDF file in ControlDesk 5.5 (or earlier) or ControlDesk 5.6 (or later).

When you reuse an experiment originally created with ControlDesk 5.5 or earlier, you can continue working with the device and layouts/instruments on the basis of the originally imported LDF file as usual.

The following limitations apply:

- Replacing and reloading the originally imported LDF file is blocked.
- When you add a new LDF file to the LIN Bus Monitoring device, ControlDesk activates this LDF variable description and tries to restore the original variable connections. Due to the changed LDF file import, however, the paths to the variables in the newly added LDF file are different, so ControlDesk cannot restore any variable connection even if you added the same LDF file.

You can use the `MigrateBusMonitoringDevices` Python script to repair the variable connections. You can download the script from the *ControlDesk Next Generation Product Support Center* at: http://www.dspace.com/cdngpsc.

To repair the variable connections, perform the following steps:

1. Start ControlDesk and activate the experiment that contains affected variable connections.
2. In a Python editor, such as PythonWin, start the `MigrateMonitoringDevice` Python script.

   In ControlDesk's Project Manager, the script adds the context menu entry Migrate Bus Monitoring Devices to the node of the currently active experiment.
3. From the context menu of the active experiment, select Migrate Bus Monitoring Devices and confirm the start dialog.

The affected variable connections in each layout of the experiment are repaired, no matter whether they are opened or closed.

The following limitations remain after executing the script:

- Bus Navigator: User-defined objects in the Bus Navigator tree, such as monitor, logger, and replay nodes, are lost.
- Measurement Configuration: Affected variables in the measurement and recording signal lists are lost.
- Measurement Configuration: Trigger rules that contain affected variables remain corrupted and must be removed or repaired manually.

Tip

If you want to add the context menu entry permanently, you can use the script as an extension script. Copy it together with the file `MigrateMonitoringDevice.extscript` to the appropriat place in the file system.

For more information on using extension scripts, refer to *Executing Extension Scripts When ControlDesk Starts Up* (🕮 *ControlDesk Next Generation Customization*).

User-defined databases (UDDBs)

User-defined databases (UDDBs) were supported for the last time with ControlDesk 5.5 (dSPACE Release 2015-B).

As a consequence, to replace the UDDB-based manipulation of CAN communication on dSPACE real-time hardware, you have to change the real-time model.

Keep the following migration aspects in mind:

- When you open an experiment with UDDB-specific items in ControlDesk 5.6 or higher, these items are not loaded to ControlDesk.
- When you open an experiment with Bus Instruments for experimental messages based on a UDDB, these instruments are loaded, but they are not functional.

DSSIGCONV tool, Measurement Data API: Changed behavior when converting files

You can use the DSSIGCONV tool to:

- Extract data from a measurement data file via the `/t:time1,time2` option.
- Split a measurement data file into several files via the `/p:parts` option.

In ControlDesk 5.6, the values of the following `Description` properties in measurement data files created by using these options have changed:

- `StartTimestamp`
- `StopTimestamp`
- `Length`

Behavior up to and Including ControlDesk 5.5	Behavior as of ControlDesk 5.6
Up to and including ControlDesk 5.5, these values were taken from the original measurement data file *without modification*.	As of ControlDesk 5.6, these values are *adapted in relation to the values of the original measurement data file*.

This change also applies to measurement data files created by saving a part of a measurement by using ControlDesk's Measurement Data API.

- For more information on using the DSSIGCONV tool, refer to *How to Extract Data from a Measurement Data File* (🕮 *ControlDesk Next Generation Measurement and Recording*).
- For more information on using the Measurement Data API, refer to 🕮 *ControlDesk Next Generation Measurement Data API*.

Tool automation changes

Change to the IPmVEOSGeneralSettings interface In ControlDesk 5.6, the `ModelIsExecutedInRealTime` property of the `VEOSGeneralSettings / IPmVEOSGeneralSettings <<Interface>>` interface has been removed.

Refer to *VEOSGeneralSettings / IPmVEOSGeneralSettings <<Interface>>* (📖 *ControlDesk Next Generation Automation*).

Moved definitions of XIL API MAPort platform-specific interfaces In ControlDesk 5.6, the definitions of the following interfaces have been moved:

- `XILAPIMAPortAssignment / IPmXILAPIMAPortAssignment <<Interface>>`
- `XILAPIMAPortGeneralSettings / IPmXILAPIMAPortGeneralSettings <<Interface>>`
- `XILAPIMAPortPlatform / IPmXILAPIMAPortPlatform <<Interface>>`

Up to and including ControlDesk 5.5, the definitions of these interfaces were located in the following library:

`dSPACE.InterfaceDefinitions.PlatformManagement.Automation<Version>.dll`

As of ControlDesk 5.6, the definitions of these interfaces are located in the `AutomationDevicesInterfaces<Version>.dll` library.

As a consequence, if you did not add all the assemblies from the `./Main/bin/AutomationAssemblies` folder of your ControlDesk installation to your C# project, you have to add the `AutomationDevicesInterfaces.dll` library to your project.

Migrating from prior ControlDesk Next Generation versions

To migrate from prior ControlDesk Next Generation versions and reuse existing experiments, you might have to carry out additional migration steps. For more information on the migration steps, refer to 📖 *ControlDesk Next Generation Migration*.

DCI Configuration Tool

New Features of the DCI Configuration Tool 3.6

Saving support report to text file with the command line interface

The command line interface of the DCI Configuration Tool provides a new parameter that saves the support report generated by the DCI Configuration Tool to a text file. You can attach the generated file to an e-mail later on.

Refer to *How to Use the DCI Configuration Tool via Command Line* (📖 *DCI Configuration*).

dSPACE ECU Flash Programming Tool

New Features of the dSPACE ECU Flash Programming Tool 2.3

CAN FD support	Besides the classic CAN protocol, the dSPACE ECU Flash Programming Tool now also supports CAN FD (CAN with Flexible Data Rate) as the XCP transport layer. Currently, there are two CAN FD protocols on the market, which are not compatible with each other: the non-ISO CAN FD protocol (representing the original CAN FD protocol from Bosch) and the ISO CAN FD protocol (representing the CAN FD protocol according to the ISO 11898-1:2015 standard). The dSPACE ECU Flash Programming Tool supports both CAN FD protocols.
Support of the DCI-CAN2 interface	The dSPACE ECU Flash Programming Tool supports dSPACE's DCI-CAN2 interface for CAN and CAN FD. Refer to *Supported ECU Interface Types* (🕮 *ECU Flash Programming*).

dSPACE FlexRay Configuration Package

Where to go from here

Information in this section

New Features of dSPACE FlexRay Configuration Package 3.7

FlexRay Configuration Package

New transmission mode for cyclic IPDU transmission based on LPDU timing dSPACE FlexRay Configuration Package 3.7 provides the new *LPDU timing triggered* transmission mode (represented by the integer value 98). This transmission mode can be assigned to each PDU, for which no timing information is specified in the underlying FIBEX or AUTOSAR system description file. It allows you to create a cyclic timing on the basis of the corresponding LPDU timing for the PDU.

For more information, refer to *How to Configure PDU Transmission Modes* (🕮 *FlexRay Configuration Tool Guide*).

Support of minimum delay time dSPACE FlexRay Configuration Package 3.7 supports the AUTOSAR *minimum delay time* feature.

The minimum delay time of an IPDU specifies the minimum delay time (in seconds) between successive transmissions of this IPDU. It

determines the time span that must elapse before new IPDU data can be packed into the LPDU. This means that if minimum delay time support is enabled for an IPDU, the current delay time is calculated when the IPDU is committed. A new transmission of this IPDU within the not yet expired minimum delay time is not possible.

Creating FlexRay configurations with minimum delay time support is possible in connection with AUTOSAR communication cluster files based on a FlexRay Configuration Tool supported AUTOSAR version ≥ 4.0.3 The minimum delay time feature is supported for PDUs with 'Application' frame content type, for which a minimum delay time > 0.0 s and at least one IPDU timing (cyclic, event-controlled) is defined in the underlying communication cluster file. The minimum delay time feature is not applied to PDUs for which the transmission mode 99 (User-Defined) or 98 (LPDU timing triggered) is currently selected.

You can disable and enable the minimum delay time support for an IPDU during run time.

Refer to *How to Configure PDUs for Minimum Delay Time Support* (🕮 *FlexRay Configuration Tool Guide*).

FlexRay Configuration Tool

Support of AUTOSAR System Template 4.2.2 The FlexRay Configuration Tool now supports the AUTOSAR System Template based on AUTOSAR Release 4.2.2 for describing FlexRay networks.

New filter conditions in the property filter The property filter, which is available in the Configuration view and Monitoring view, now also provides the following filter conditions based on frame properties for you to specify the filter criterion:

- Separate channel handling
- HW enable static frame after model start
- SW enable static frame after model start
- Enable minimum delay time
- Override global default transmission mode
- PDU default transmission mode

Refer to *Filter* (🕮 *FlexRay Configuration Tool Reference*).

Changed parameterization of unused bits of multiplexed PDUs The parameterization of unused bits of multiplexed PDUs has been changed for dSPACE FlexRay Configuration Package 3.7. Now the bits of a subframe, which are not used to transmit signal values, are parameterized with the initialization value of unused bits in frames as specified on the Generators page of the General Properties

dialog. Up to dSPACE FlexRay Configuration Package 3.6, the unused bits of a subframe contained the old data of another subframe, which was previously sent.

Reduced number of required hardware resources on SCALEXIO systems for FlexRay configurations based on dual-channel communication cluster files For SCALEXIO systems, the number of required hardware resources for FlexRay configurations on the basis of a dual-channel communication cluster file has been reduced for some cases. With older versions of the FlexRay Configuration Tool, two real controllers were always required to work with a FlexRay configuration on the basis of a dual-channel communication cluster file, regardless of whether one or two FlexRay channels actually were within the FlexRay configuration. (In the generated M file, the `usedChannels` parameter was always set to 'AB'.) As of FlexRay Configuration Tool 3.7, this behavior changes. When you work with a SCALEXIO system and create a FlexRay configuration based on a dual-channel communication cluster file, the required number of real controllers now depends on the number of channels that are actually used in the FlexRay configuration. If there is only one channel used within the FlexRay configuration, only one real controller is required. If there are two FlexRay channels within the FlexRay configuration, two real controllers are required. (In the generated M file, the `usedChannels` parameter has the value 'A', 'B', or 'AB'.)

Refer to *Dual Channel Configurations* (🕮 *FlexRay Configuration Tool Guide*).

RTI FlexRay Configuration Blockset

Planned discontinuation of signal-based modeling as of dSPACE Release 2017-A Signal-based modeling with the RTI FlexRay Configuration Blockset is being supported for the last time with dSPACE Release 2016-B. As of dSPACE Release 2017-A, the RTI FlexRay Configuration Blockset supports only PDU-based modeling.

Migrating to dSPACE FlexRay Configuration Package 3.7

Modified bit position information of CRC signals in multiplexed PDUs

To perform checksum algorithms, the FlexRay Configuration Tool uses the `dsftcom_crc(...)` function. One of the parameters that this function has is the `pCsBitPos` parameter, which specifies an array containing the bit positions of the signals where you can save the CRC values to or read the CRC values from.

As of FlexRay Configuration Tool 3.7, the bit position information of CRC signals in multiplexed PDUs is modified. The start bit position of a CRC signal included in a sub-PDU is now specified relative to the PDU. With older versions of the FlexRay Configuration Tool, the start bit position of a CRC signal in a sub-PDU was specified relative to the sub-PDU.

If you work with CRC signals in multiplexed PDUs and want to reuse existing checksum algorithms, you must adapt the checksum algorithms.

Discontinuation of the FlexRay Replay Script Generator

The FlexRay Replay Script Generator was delivered for the last time with dSPACE Release 2015-B. As of dSPACE Release 2016-A, the FlexRay Replay Script Generator is no longer available.

As of dSPACE Release 2016-A, you can still integrate the Python interpreter into a FlexRay timetable task. This lets you still replay user-created Python scripts time-synchronously to the FlexRay bus.

dSPACE HIL API .NET

Where to go from here

Information in this section

New Features of dSPACE HIL API .NET 2.1

dSPACE HIL API .NET 2.1 does not provide new features.

Migrating to dSPACE HIL API .NET 2.1

Discontinuation of MicroAutoBox software support

dSPACE Release 2015-B was the last release supporting MicroAutoBox with its variants 1401/1501, 1401/1504, 1401/1505/1506, 1401/1505/1507, and 1401/1507.

As of dSPACE Release 2016-A, dSPACE software supports only MicroAutoBox II with its variants 1401/1507, 1401/1511, 1401/1513, 1401/1511/1514, and 1401/1513/1514.

To use MicroAutoBox II with dSPACE HIL API .NET by using the MAPort implementation, you only have to modify the platform registration. You do not need to modify your HIL API application, because the MAPort configuration only contains the platform name *ds1401* that is similar to MicroAutoBox and MicroAutoBox II.

If you use dSPACE HIL API .NET 2.1 and you have registered a MicroAutoBox with a bus connection, the execution stops with the following MAPortException.

```
Code: 35
CodeDescription: Could not create port instance
VendorCode: 2147746533
VendorCodeDescription: Could not append the following platform: 'ds1401'.
                       Support of platform MABX I has been discontinued.
```

For more information on the dSPACE HIL API .NET implementation, refer to 🕮 *dSPACE HIL API .NET Implementation Document*.

Planned discontinuation of dSPACE HIL API .NET

With dSPACE Release 2016-B, dSPACE HIL API .NET implementation will not longer be available.

You can migrate your test automation projects to ASAM XIL API as the HIL API successor. The migration from HIL API .NET to XIL API .NET requires only a few modifications in your application. Refer to *Migrating HIL API Applications to XIL API Applications* (🕮 *dSPACE XIL API Implementation Guide*).

For more information, refer to the Test Automation Tools Support Center: http://www.dspace.com/go/pscta.

dSPACE Python Extensions

Where to go from here

Information in this section

New Features of dSPACE Python Extensions 2.1

Test Automation Python modules

The `matlablib2` Python module has a new property:

- `ConnectedMATLABInstallations`

  To get a list of the connected MATLAB installations containing the installation paths and whether they are configured as the preferred MATLAB instance.

Migrating to dSPACE Python Extensions 2.1

Discontinuation of MicroAutoBox software support

dSPACE Release 2015-B was the last release supporting MicroAutoBox with its variants 1401/1501, 1401/1504, 1401/1505/1506, 1401/1505/1507, and 1401/1507.

As of dSPACE Release 2016-A, dSPACE software supports only MicroAutoBox II with its variants 1401/1507, 1401/1511, 1401/1513, 1401/1511/1514, and 1401/1513/1514.

dSPACE Platform Management API If you have accessed MicroAutoBox via the Platform Management API and you want to migrate to MicroAutoBox II, you have to change the registration information in your test application.

The example shows you how to migrate from MicroAutoBox connected via *Bus* to MicroAutoBox II connected via *Net* with 192.168.10.1 as the IP address, for example.

MicroAutoBox	MicroAutoBox II

`import dspace.com`
`PlatformManagement = \`
`win32com.client.Dispatch("DSPlatformManagementAPI2")`
`# Define Enums object`
`PMEnums = dspace.com.Enums(PlatformManagement)`
`# Create RegistrationInfo object`
`RegInfo = \`
`PlatformManagement.CreatePlatformRegistrationInfo\`
`(PMEnums.PlatformType.MABX)`
`# Set ConnectionType`
`RegInfo.ConnectionType = \`
`PMEnums.InterfaceConnectionType.`**`Bus`**
`#`
`#`
`# Register platform`
`myPlatform = PlatformManagement.RegisterPlatform(RegInfo)`
	`import dspace.com`
	`PlatformManagement = \`
	`win32com.client.Dispatch("DSPlatformManagementAPI2")`
	`# Define Enums object`
	`PMEnums = dspace.com.Enums(PlatformManagement)`
	`# Create RegistrationInfo object`
	`RegInfo = \`
	`PlatformManagement.CreatePlatformRegistrationInfo\`
	`(PMEnums.PlatformType.MABX)`
	`# Set ConnectionType`
	`RegInfo.ConnectionType = \`
	`PMEnums.InterfaceConnectionType.`**`Net`**
	`# Set IP address`
	`RegInfo.NetClient = "192.168.10.1"`
	`# Register platform`
	`myPlatform = PlatformManagement.RegisterPlatform(RegInfo)`

For information on the Platform Management API, refer to 🕮 *dSPACE Platform Management API Reference*.

dSPACE HIL API Python Implementation (MAPort) If you have accessed MicroAutoBox via the MAPort in HIL API Python, and you want to migrate to MicroAutoBox II, you do not need to change your test application, because the platform identifier *ds1401* used in the MAPort configuration is the same for MicroAutoBox and MicroAutoBox II.

If you use dSPACE HIL API Python 2.1 and you have registered a MicroAutoBox with a bus connection, the execution stops with the following MAPortException.

```
Code: 35
CodeDescription: Could not create port instance
VendorCode: 2147746533
VendorCodeDescription: Could not append the following platform: 'ds1401'.
                       Support of platform MABX I has been discontinued.
```

For information on the dSPACE HIL API Python Implementation, refer to 🕮 *dSPACE HIL API Python Implementation Document*.

rtplib2 If you have accessed MicroAutoBox via the `rtplib2` Python module, and you want to migrate to MicroAutoBox II, you do not need to change your test application, because the platform identifier *ds1401* used as the `PlatformName` parameter when instantiating the `Appl` object is the same for MicroAutoBox and MicroAutoBox II.

If you use rtplib2 from Python Extensions 2.1 and you have registered a MicroAutoBox with a bus connection, the execution stops with the following rtpliberror.

```
rtplibError: Error occurred during execution of function 'RtpAppl_inf':
   Could not append the following platform: 'ds1401'.
   Support of platform MABX I has been discontinued.
```

For information on the `rtplib2` Python module, refer to *Accessing Simulator Variables (rtplib2)* (🕮 *Test Automation Python Modules Reference*).

Planned discontinuation of software contained in dSPACE Python Extensions

With dSPACE Release 2016-B, dSPACE Python Extensions will no longer provide:

- dSPACE HIL API Python Implementation
- `rtplib2`

You can migrate your test automation projects to ASAM XIL API as the HIL API successor.

For information on migrating from HIL API Python or `rtplib2` to XIL API .NET, refer to the Test Automation Tools Support Center: http://www.dspace.com/go/pscta.

Discontinuation of platform management automation API version 1.0 as of dSPACE Release 2016-B Platform management automation API version 1.0 is being supported for the last time with Python Extensions 2.1 from dSPACE Release 2016-A. For more information, refer to *Platform Management Automation API Versions* (🕮 *dSPACE Platform Management API Reference*).

dSPACE XIL API

Where to go from here

Information in this section

New Features of dSPACE XIL API 2016-A

Enhanced MAPort functionality

The MAPort configuration file provides two new optional attributes:

- IncompatibilityBehavior

  Lets you specify how to react if I/O components included in the generated real-time application are not available at the connected platform. By default, the download of the real-time application is then stopped. With this attribute you can specify to simulate the missing I/O components or to ignore them.

  This attribute is only relevant to the SCALEXIO system, DS1007 and MicroLabBox.

- StimulusDataStreamingBufferSize

  Lets you specify the buffer size used by Real-Time Testing for data streaming. The buffer size affects the execution time. This attribute is relevant when you stimulate SignalValueSegment or DataFileSegment signals.

Enhanced EESPort functionality

The switching behavior of errors simulated on the electrical error simulation hardware can now be monitored during the execution of the executable application. This feature leads to the following enhancements.

- New real-time variables generated into the variable description file

  If you build an executable application, the generated variable description file will always provide the following five variables in the XIL API/EESPort subgroup:

  - Active ErrorSet

    To display the number of the currently active error set.

  - Error Activated

    To indicate whether one or more errors are activated.

  - Error Switching

    To indicate the undefined transition state when switching the failure simulation hardware.

  - Flags

    Reserved for future use.

  - Trigger

    Reserved for future use.

  These variables are not generated into the variable description file of DS1103 and DS1104 real-time applications.

- New RealTimeConfiguration section in the EESPort configuration file

  To enable or disable the monitoring of these variables and to customize the variable names, the EESPort configuration file now provides the following new elements:

```
<RealTimeConfiguration PlatformName="<PlatformName>" SystemDescriptionFilePath="<Xyz.sdf>"
  <Tracing Enabled="true|false">
    <Variable Value="<VariableGroupAndName>" Type="ErrorActivated" />
    <Variable Value="<VariableGroupAndName>" Type="ActiveErrorSet" />
    <Variable Value="<VariableGroupAndName>" Type="ErrorSwitching" />
    <Variable Value="<VariableGroupAndName>" Type="Flags" />
    <Variable Value="<VariableGroupAndName>" Type="Trigger" />
  </Tracing>
</RealTimeConfiguration>
```

- Enhancement of the EESPortConfiguration API

  The EESPortConfiguration API provides new properties to specify the RealTimeConfiguration section in an EESPort configuration file.

For more information, refer to *Monitoring the Switching Behavior of Electrical Error Simulation Hardware* (🕮 *dSPACE XIL API Implementation Guide*) and *Creating dSPACE EESPort Configuration Files* (🕮 *dSPACE XIL API Implementation Guide*).

Migrating to dSPACE XIL API 2016-A

Migrating applications from dSPACE HIL API .NET to dSPACE XIL API .NET

For information on the required migration steps, refer to *Migrating HIL API Applications to XIL API Applications* (🕮 *dSPACE XIL API Implementation Guide*).

Discontinuation of MicroAutoBox software support

dSPACE Release 2015-B was the last release supporting MicroAutoBox with its variants 1401/1501, 1401/1504, 1401/1505/1506, 1401/1505/1507, and 1401/1507.

As of dSPACE Release 2016-A, dSPACE software supports only MicroAutoBox II with its variants 1401/1507, 1401/1511, 1401/1513, 1401/1511/1514, and 1401/1513/1514.

To use MicroAutoBox II with dSPACE XIL API .NET by using the MAPort implementation, you only have to modify the platform registration. You do not need to modify your XIL API application, because the MAPort configuration only contains the platform name *ds1401* that is the same for MicroAutoBox and MicroAutoBox II.

If you use dSPACE XIL API .NET 2016-A and you have registered a MicroAutoBox with bus connection, the execution stops with the following TestbenchPortException.

```
Code: 1047
CodeDescription: Port configuration could not be completed successfully
VendorCode: 1
VendorCodeDescription: Platform 'ds1401' is not registered
```

For more information on the dSPACE XIL API implementation, refer to 🕮 *dSPACE XIL API Implementation Guide*.

New version of EESPortConfiguration API

The version of the EESPortConfiguration API is to be changed from 2.1.0.0 to 2.2.0.0.

The version is required when you reference the related assembly in your application:

- Using dSPACE XIL API .NET 2015-B:

  ```
  dSPACE.XIL.Testbench.EESPort.Interfaces.Extended.dll
  (Version 2.1.0.0, PublicKeyToken=f9604847d8afbfbb)
  ```

- Using dSPACE XIL API .NET 2016-A:

  ```
  dSPACE.XIL.Testbench.EESPort.Interfaces.Extended.dll
  (Version 2.2.0.0, PublicKeyToken=f9604847d8afbfbb)
  ```

ECU Interface Manager

Migrating to ECU Interface Manager 1.8

Migrating projects last saved with a former version of ECU Interface Manager

In ECU Interface Manager 1.8, you can reuse projects that were last saved with a former version of the ECU Interface Manager.

When you open such a project for the first time, you are asked whether to update it:

- When you start the update, you can continue working with the project with ECU Interface Manager 1.8.
- When you postpone the update, actions are blocked, except for exporting the application. You can update the project later.
- When you save the project, you are asked whether to overwrite the old project file:
  - When you overwrite the old project, you can no longer use it with a former version of the ECU Interface Manager.
  - When you do not overwrite the old project, you have to specify another location and/or name for the project file. This lets you keep a version of the project that you can work with in the former version of ECU Interface Manager.

New software module description file schema

As of ECU Interface Manager 1.6, ECU suppliers can now use a generic schema to create a software module description file (→ *Software module description file* (🕮 *ECU Interface Manager Guide*)).

You can also import software module description files based on the dSPACE-specific schema, which was originally introduced with ECU Interface Manager 1.0.

Note

- The dSPACE-specific schema is supported for downward compatibility reasons only. It will be replaced by the generic schema in the next dSPACE releases.
- Multicore support and further developments are not available with the dSPACE-specific schema.

  Use the *generic schema* instead.

For more information on the generic schema, refer to *Generic Schema of Software Module Description Files* (🕮 *ECU Interface Manager Reference*).

Firmware Manager

Where to go from here

Information in this section

New Features of Firmware Manager 2.1

Enhanced platform support

The Firmware Manager supports the following new boards of a SCALEXIO system:

- DS2656 FPGA Board (only for dSPACE Solutions)
- DS6051 IOCNET Router for SCALEXIO LabBox
- DS6301 CAN/LIN Board

Usability improvements

The user interface of the Firmware Manager has been changed to ribbon menus, already commonly used in ControlDesk Next Generation and other dSPACE tools.

For more information, refer to *Basics on the Firmware Manager* (🕮 *Firmware Manager Document*).

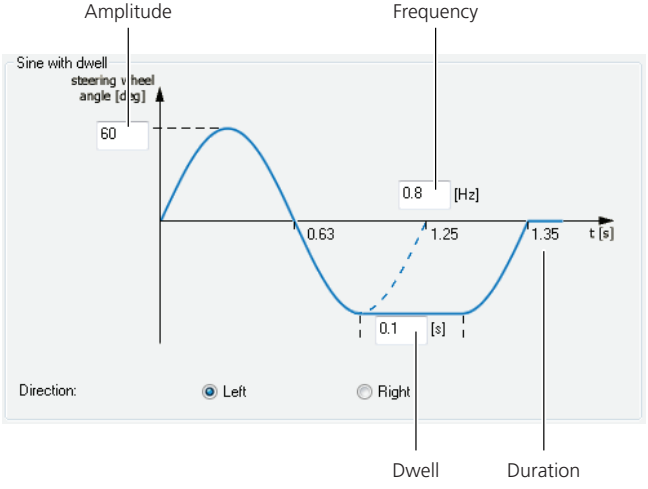
Migrating to Firmware Manager 2.1

Discontinuation of MicroAutoBox software support	dSPACE Release 2015-B was the last release supporting MicroAutoBox with its variants 1401/1501, 1401/1504, 1401/1505/1506, 1401/1505/1507, and 1401/1507. As of dSPACE Release 2016-A, dSPACE software supports only MicroAutoBox II with its variants 1401/1507, 1401/1511, 1401/1513, 1401/1511/1514, and 1401/1513/1514.
Changed archive format	The archive format for SCALEXIO has been changed with Firmware Archives 2.1 contained in dSPACE Release 2016-A. This archive cannot be loaded with the Firmware Manager 2.0 or earlier.

ModelDesk

New Features of ModelDesk 4.3

Project management	**Project or experiment names** You can rename projects or experiments or save them under a different name.
Maneuver Editor	**Sine with dwell steering** The Maneuver Editor provides a new predefined steering profile: Sine with dwell steering. In a sine with dwell steering, the steering angle is specified by a sinusoidal waveform with a dwell in the negative half-wave. See the following illustration.



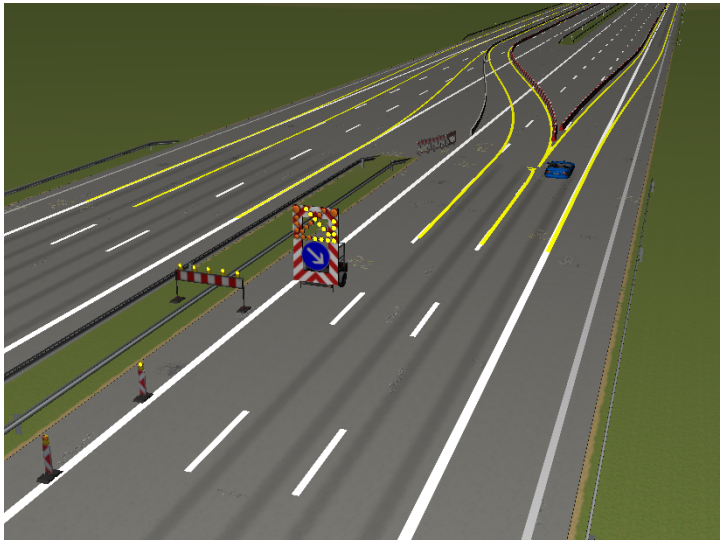
A sine with dwell steering maneuver is necessary when a electronic stability control system is tested according to the FMVSS 126 and ECE R13-H.

Control mode for circle maneuver segment When using the circle maneuver segment, you can select whether the position or yaw rate control mode is used in the model. This effects the lateral controller 2. Refer to *Lateral Controller 2* (🕮 *ASM Environment Reference*).

Road Generator

Construction site support You can now model construction sites with objects such as interim lanes, barriers, yellow traffic lines and delineator posts. These objects can be specified with the new shape objects in the Road Generator. Using shapes you can add additional lines (independent from specified lanes), rows of traffic objects, or barriers to a road. The course of a shape can be specified by two or more points related to a road or junction element.

The following illustration shows an example of a construction site on a highway.



Show in MotionDesk ModelDesk can set the position of a specific observer in MotionDesk. This is useful when you create a road network and want to examine it at a special position after scene generation. Refer to *Show in MotionDesk* (🕮 *ModelDesk Reference*).

	Importing OpenDRIVE® files The OpenDRIVE® import now supports OpenDRIVE version 1.4 and converts road objects and road signals to ModelDesk traffic objects. Refer to *Importing Roads in OpenDRIVE Format to ModelDesk* (🕮 *ModelDesk Guide*). **Advanced line configuration** You can completely configure the lines of lane sections and shapes. The line types of previous ModelDesk versions can be used as presets for the configuration. **Advanced specifying spline road segments** There are two ways to specify "spline" road segments: You can specify the end vector or alternatively the coefficients of the mathematical expression. **Trimming roads** You can trim road elements in the Road 2-D Preview. Refer to *Trim Road - Trim from <End/Start>* (🕮 *ModelDesk Reference*)
Model parameterization	**Parameter properties** The Properties panes of parameters display the extrapolation type.
Processing	Values of measurement type variables can now be calculated by MATLAB functions. Refer to *How to Calculate Measurement Data* (🕮 *ModelDesk Guide*).
Simulink simulation	ModelDesk supports MATLAB R2016a. When you use MATLAB R2016a for Simulink simulation, keep the following limitations for running simulations in mind: ■ Starting and stopping plotting requires up to one minute. ■ Downloading road, maneuver and traffic scenarios requires up to several minutes. ■ Downloading parameter sets and processing executions is not recommended. There are no limitations when the simulation is stopped or paused.
Tool automation	The tool automation is extended for several components. **Pool handling** You can now import elements to the Pool. **Road Generator** You can specify lane sections and lanes of road elements. You can specify lines. You can find a valid reference point or move the reference point of junctions.

Maneuver Editor You can use the sine with dwell steering in maneuvers.

Processing You can specify and execute measurement functions for processing.

Model Interface Package for Simulink

New Features of the Model Interface Package for Simulink 3.2

Simplified generation of communication interfaces for behavior models

To simplify your work, the Model Interface Package for Simulink provides methods for you to easily create communication interfaces for behavior models. For this purpose, the Model Interface Package lets you create model port blocks that have the same configuration but the inverse data direction as model port blocks in your Simulink model. Creating inverse model port blocks is especially useful if you need a counterpart for a structured model port block. Refer to *Simplified Preparation of Model Interfaces for Model Communication* (🕮 *Model Interface Package for Simulink - Modeling Guide*).

Note

ConfigurationDesk also provides this feature so you can create inverse model port blocks in your Simulink model as well as in your ConfigurationDesk application.

Support of Goto/From blocks during model separation

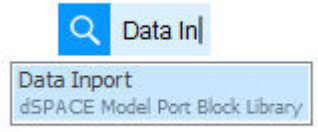
The Model Separation Block Library supports model communication that is specified indirectly via Goto/From blocks between models to be separated from an overall model. Refer to *Separation of Models Containing Goto and From Block Connections* (🕮 *Model Interface Package for Simulink - Modeling Guide*).

Model cleanup

The Model Interface Package for Simulink now provides an API command that lets you remove all data resulting from the Model Port Block Library from your Simulink model. This can be useful, for example, if you want to use a Simulink model in an environment without a Model Interface Package for Simulink installation. Refer to *dsmpb_mdlcleanup* (🕮 *Model Interface Package for Simulink Reference*).

Support of Quick Insert

The Model Interface Package for Simulink supports the Quick Insert feature introduced with MATLAB R2014b. To add a block, you can click an open space in the Simulink model. Start typing the block name when the blue magnifying glass appears.



The Quick Insert feature is available for the following blocks:

- Data Inport block
- Data Outport block
- Runnable Function block
- Model Separation Setup block

Unsupported new features of MATLAB R2016a

The following new feature introduced with MATLAB R2016a is not supported by the Model Interface Package for Simulink:

- New Variant Sink and Variant Source Simulink blocks

  Model port blocks are not allowed on the signal path of these blocks.

- Defines option in the Custom Code page of the Code Generation Dialog
- Physical units specified on Simulink signals

  Suppose physical units are specified on Simulink signals connected to model ports for which units are specified as well. In this case, the units of the signals and the model ports are not checked for consistency, because the Units property of model ports is used only for documentation purposes.

MotionDesk

Where to go from here

Information in this section

New Features of MotionDesk 3.8

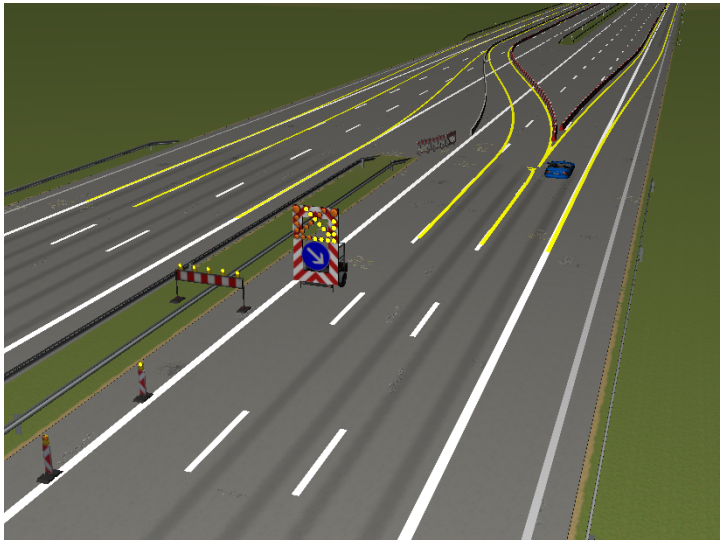
Observer

You can use the new ModelDesk Observer when you model a road network and you want to examine the road network in MotionDesk at specific locations. In ModelDesk, you can set the position and orientation via a command or by a mouse operation. ModelDesk moves the ModelDesk Observer in MotionDesk to the desired position.

You can copy the settings of an observer to the Clipboard and paste it to create a new observer that has the same settings.

Visualizing construction sites

MotionDesk supports the new features provided with ModelDesk's Road Generator. The lines can be specified more precisely and the dSPACE 3-D object library has been expanded with further 3-D objects for construction sites. So you can visualize construction sites with objects such as interim lanes, barriers, yellow traffic lines and delineator posts.



3-D object library

The dSPACE 3-D object library has new 3-D objects:

- Road accessories: Direction curve right/left, traffic light post, traffic light signal
- Construction site: Delineators, curbs, studs, warning lights
- Scenery: Boulder, bushes, shrubs, grass patches, leaves, power pole (utility pole)
- Cars: Police car, emergency ambulance

The following 3-D objects are optimized:

- Domes: Horizontal land
- Traffic: Radar 3-D
- Road accessories: Upright, road divider, street lamp, traffic light, guardrail
- Scenery: Trees

	The 3-D objects require less memory of the graphics card by using smaller textures.
State objects	The dSPACE library has new state objects: arrows and sensors. When you use these state objects, less data has to be streamed for animated force vectors and sensor points.

Migrating to MotionDesk 3.8

MicroAutoBox support discontinued	dSPACE Release 2015-B was the last release supporting MicroAutoBox with its variants 1401/1501, 1401/1504, 1401/1505/1506, 1401/1505/1507, and 1401/1507. As of dSPACE Release 2016-A, dSPACE software supports only MicroAutoBox II with its variants 1401/1507, 1401/1511, 1401/1513, 1401/1511/1514, and 1401/1513/1514.
Migrating from MotionDesk 2.2.1 and earlier	In MotionDesk 2.2.1 and earlier, MotionDesk uses 3-D objects in VRML format. To use the scenes and custom 3-D objects used in these MotionDesk versions, they must be migrated so that they can be used in MotionDesk 3.8. For more information, refer to *Migrating from MotionDesk 2.2.1 and Lower* (🕮 *MotionDesk Guide*).
Migrating from MotionDesk experiments in MDX file format	MotionDesk 3.8 cannot read old MotionDesk experiments in the MDX file format any longer. It is therefore not possible to migrate from a MotionDesk experiment with a version earlier than 2.2. If you want to migrate such old experiments, you can migrate using MotionDesk 3.0 up to MotionDesk 3.6.

Real-Time Testing

Where to go from here

Information in this section

New Features of Real-Time Testing 3.0

New Python version

The Python interpreter running on the simulation platform is based on Python 2.7.10.

New module

The rttlib library has been extended with the `rttlib.dscanapilib` module. Using this module, you can send or receive CAN or CAN FD messages. The module supports only SCALEXIO systems or VEOS. Refer to *Handling CAN Messages Using the rttlib.dscanapilib Module* (🕮 *Real-Time Testing Guide*).

New supported hardware

Real-Time Testing supports the DS6301 CAN/LIN Board.

Migrating to Real-Time Testing 3.0

Incompatible BCG files

The BCG files that are generated with Real-Time Testing 2.6 and lower cannot be used for Real-Time Testing 3.0. You must create the

BCG file of the RTT sequence again. Refer to *Creating and Starting RTT Sequences in Python Scripts* (📖 *Real-Time Testing Guide*).

RTI/RTI-MP and RTLib

Where to go from here

Information in this section

New Features of RTI/RTI-MP and RTLib

New startup script

In addition to the `dsstartup` script, there is a new optional script to configure the MATLAB startup behavior. With the `dspoststartup` script, you can run initialization functions *after* the initialization phase of the dSPACE software. For more information, refer to *dspoststartup.m* (🕮 *RTI and RTI-MP Implementation Reference*).

MicroLabBox

Enhanced software support The following features are now supported:

- Enhanced support of electric motor control

  For more information, refer to *New Features of RTI Electric Motor Control Blockset 1.3* on page 163.

For more information on the board's features, refer to 🕮 *MicroLabBox Features*.

MicroAutoBox

DS1554 Engine Control I/O Module The DS1554 Engine Control I/O Module can be mounted on MicroAutoBox II 1401/1511/1514 and 1401/1513/1514 as an I/O module to extend the I/O capabilities

of the MicroAutoBox system. The module can be used only with the RTI FPGA Programming Blockset.

For information on the I/O features provided by the RTI FPGA Programming Blockset, refer to *New Features of the RTI FPGA Programming Blockset 3.1* on page 165.

Unsupported new features of MATLAB R2016a

The following new features introduced with MATLAB R2016a are not supported by RTI/RTI-MP:

- New Variant Sink and Variant Source Simulink blocks

  RTI blocks are not allowed within the signal path of these blocks.
- Defines option in the Custom Code page of the Code Generation Dialog

  Use the Compiler options setting in the RTI general build options page of the Code Generation Dialog.

Migration Aspects of RTI/RTI-MP and RTLib

Changes in TRC file generation

You have to note some modifications on TRC file generation in RTI and RTI-MP. Refer to *Changes to TRC File Generation* on page 33.

Modified features in MATLAB R2016a

The following change has been made:

- If you open a new Simulink model via the Simulink Start page, the factory default settings are used instead of RTI-specific settings, as before. To use the RTI-specific settings, you have to create a new model by entering `new_system` in the MATLAB Command Window.

If you install a new MATLAB version, some settings are adopted from already installed MATLAB versions.To prevent unexpected behavior of your Simulink models when switching to a newer MATLAB version or dSPACE Release, always reset the MATLAB and Simulink preferences to their defaults before you start working.

MicroAutoBox

Discontinuation of MicroAutoBox software support dSPACE Release 2015-B was the last release supporting MicroAutoBox with its variants 1401/1501, 1401/1504, 1401/1505/1506, 1401/1505/1507, and 1401/1507.

As of dSPACE Release 2016-A, dSPACE software supports only MicroAutoBox II with its variants 1401/1507, 1401/1511, 1401/1513, 1401/1511/1514, and 1401/1513/1514.

Your real-time applications can be run on its successor MicroAutoBox II, if it provides the same I/O boards.

MicroAutoBox II provides numerous additional benefits, such as the Ethernet interface and the ability to use a freely programmable FPGA.

Previous dSPACE Releases still support the older revisions of MicroAutoBox.

For more information on MicroAutoBox II's features, refer to 🕮 *MicroAutoBox Features*.

RTI Bypass Blockset

Migrating to RTI Bypass Blockset 3.6

Working with models from earlier RTI Bypass Blockset versions 3.x and 2.x

The current release contains RTI Bypass Blockset 3.6, which is compatible with earlier blockset versions 3.x and 2.x. However, there are some points to note:

- *Working with models from RTI Bypass Blockset 2.5 or earlier*

  Data management was changed in comparison to the prior RTI Bypass Blockset versions. If you have a Simulink model built with RTI Bypass Blockset 2.5 or earlier and open it with RTI Bypass Blockset 3.6, the old data dictionary file (with file name extension .dd) is replaced by a new data dictionary file (.vdb) using the information stored in the Setup block. This happens as soon as you open and close the Setup block dialog via 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 3.6 and want to use it with RTI Bypass Blockset 2.5 or earlier, the model's data dictionary file required for blockset version 2.5 or earlier (file name extension .dd) is created. This happens 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 3.6 (*.vdb) remains on disk.

  To make the RTI Bypass Blockset able to recreate the data dictionary, the database files specified in the Setup block must be accessible at the specified location and must be unchanged.

- *Working with models from RTI Bypass Blockset 2.6 up to and including RTI Bypass Blockset 3.5*

  If you have a Simulink model built with RTI Bypass Blockset 2.6 up to RTI Bypass Blockset 3.5 and open it with RTI Bypass Blockset 3.6, the old data dictionary file is replaced by a new data dictionary file. However, the new data dictionary file cannot be used in earlier RTI Bypass Blockset versions. If you want to reuse the model with RTI Bypass Blockset 2.6 up to RTI Bypass Blockset 3.5, you have to create a suitable database in the earlier RTI Bypass Blockset version by reimporting the database files (A2L files) specified in the Setup block.

RTI CAN MultiMessage Blockset

Where to go from here

Information in this section

New Features of the RTI CAN MultiMessage Blockset 4.3

Support of SCALEXIO systems with a DS6301 CAN/LIN Board

The RTI CAN MultiMessage Blockset supports SCALEXIO systems with a DS6301 CAN/LIN Board. The DS6301 CAN/LIN Board provides four CAN/CAN FD channels.

Support of AUTOSAR System Template 4.2.2

The RTI CAN MultiMessage Blockset now supports the AUTOSAR System Template based on AUTOSAR Release 4.2.2 for describing CAN networks.

Refer to *General Settings Page (RTICANMM MainBlock)* (🕮 *RTI CAN MultiMessage Blockset Reference*).

Migrating to RTI CAN MultiMessage Blockset 4.3

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 RTICANMM blocks and save the model before modifying the CAN configuration.

To create new S-functions for all the RTICANMM blocks in your model in one step, you can perform one of the following actions after opening the model:

- In the MATLAB Command Window, enter `rtimmsu_update('System', gcs)`.

  For more information on the command and its options, enter `help rtimmsu_update` in the MATLAB Command Window.
- Select the Create S-Function for all CAN Blocks command from the Options menu of the RTICANMM GeneralSetup block.

For more information, refer to *Limitations with RTICANMM* (🕮 *RTI CAN MultiMessage Blockset Reference*).

Compiler messages when using code generated by an RTI CAN MultiMessage Blockset version < 4.0

If you use code that was generated by an RTI CAN MultiMessage Blockset version < 4.0, several compiler warning messages containing the phrase `<<argument of type "can_tp1_canChannel *" is incompatible with parameter of type "DsTCanCh">>` will appear during the build process of your simulation model. This is due to a modified data type. These warnings can be ignored and will disappear after you regenerate the RTICANMM code by using the current blockset version.

Using existing checksum algorithms

Checksum algorithms originally developed for an application containing CAN messages cannot be reused for applications containing CAN FD messages, because CAN FD includes new message types and longer data fields. Existing checksum algorithms can still be used for applications that just contain classic CAN messages. For CAN FD applications, you must adapt the checksum algorithms.

RTI Electric Motor Control Blockset

New Features of RTI Electric Motor Control Blockset 1.3

New block

The RTI Electric Motor Control Blockset provides a new block:

- EMC_SSI_BLx to use an absolute encoder connected to a synchronous serial interface (SSI) as an input sensor for motor control.

For more information, refer to 🕮 *RTI Electric Motor Control Blockset Reference*.

RTI FPGA Programming Blockset

Where to go from here

Information in this section

New Features of the RTI FPGA Programming Blockset 3.1

Extended Xilinx® support

The RTI FPGA Programming Blockset now supports the following products and versions of the Xilinx design tools.

Xilinx Design Tools Version	Operating System	MATLAB Version
Vivado 2015.4 (64-bit version)	Windows 7 Business, Ultimate, and Enterprise SP1 (64-bit version)	64-bit versions of: ■ MATLAB R2014b ■ MATLAB R2015a SP1 ■ MATLAB R2015b

Supported dSPACE platforms

The following dSPACE platforms are supported by the RTI FPGA Programming Blockset 3.1

- MicroAutoBox (MicroAutoBox II 1401/1511/1514 and MicroAutoBox II 1401/1513/1514)
- MicroLabBox

- Modular system (DS5203 (7K325) and DS5203 (7K410))
- SCALEXIO (DS2655)

The following hardware is not supported by Xilinx Vivado. The RTI FPGA Programming Blockset 3.1 therefore supports only the building of the processor interface for existing FPGA model INI files:

- DS5203 FPGA Board (SX95)
- DS5203 FPGA Board (LX50)
- MicroAutoBox II 1401/1511/1512
- MicroAutoBox II 1401/1512/1513

Only the RTI FPGA Programming Blockset up to version 2.9 supports Xilinx ISE and the FPGA modeling for the DS5203 (SX95) and DS5203 (LX50) boards, and the DS1512 I/O Board (MicroAutoBox II 1401/1511/1512 and MicroAutoBox II 1401/1512/1513). Due to the introduction of Vivado, Xilinx no longer supports the Xilinx System Generator for DSP in combination with the ISE Design Suite after MATLAB Release R2013b.

Support of new DS1554 Engine Control I/O Module

The DS1554 Engine Control I/O Module can be mounted on MicroAutoBox II 1401/1511/1514 and 1401/1513/1514 as an I/O module to extend the I/O capabilities of the MicroAutoBox system.

The main features of the DS1554 Engine Control I/O Module framework are I/O functions to access engine-specific sensor signals:

- Bit-wise access to 5 digital camshaft and crankshaft sensors
- Access to 1 inductive zero voltage detector
- Access to 4 analog knock sensors
- Access to 14 analog input signals
- Access to 8 bidirectional digital signals
- Access to 40 digital output signals

Enhancements to the DS2655 FPGA Base Board framework

The framework for the DS2655 FPGA Base Board provides the following enhancement.

Tracing FPGA signals You can make FPGA signals traceable to access FPGA signals with your experiment software, such as ControlDesk Next Generation. FPGA signals can be traced via FPGA variables that are added to the FPGA application during the build process. These variables provide the read access for the experiment software. A system description file (SDF file) containing additional information on tracing signals of the FPGA application and variables

of the processor model is generated during the build process of the processor application.

The FPGA_SETUP_BL block lets you make FPGA signals traceable.

Simulating the FPGA model and the processor model in Simulink To simulate the entire model in Simulink, you must implement a processor interface to exchange the data between the processor model and the FPGA model. You implement this processor interface via generated interface blocks. You have to generate the interface blocks with the dialogs of the FPGA_XDATA_READ_BL, FPGA_XDATA_WRITE_BL, and FPGA_INT_BL blocks.

Exporting the FPGA build results and the processor model to ConfigurationDesk The FPGA_SETUP_BL block lets you export the FPGA build results to a ConfigurationDesk project.

The export performs the following steps if the processor interface is implemented:

- Exports the FPGA application to ConfigurationDesk
- Separates the processor model
- In ConfigurationDesk, adds the model interface of the processor model to the signal chain
- Adds the FPGA application as a custom function to the signal chain
- Maps the function ports of the FPGA custom function to the model ports

Enhancements to the DS2655M1 Multi-I/O Module framework

Outputting signals without jitter The analog and digital output channels of the DS2655M1 Multi-I/O Module send new signal values with a minimum update period of 64 ns. Because the FPGA clock period differs from the update period, the output signal might jitter if the output of the FPGA application is not synchronized with the update period of the hardware channel.

Therefore, the Digital InOut and Analog Out functions of the FPGA_IO_WRITE_BLx block provide a new optional Tx Ready port. The port outputs a flag that indicates that the module's output channel is ready to be updated.

When you update data values only within the time slot for updating the output signal, the output signal has no jitter. The time slot begins two FPGA clock cycles before the flag is set to high and ends after three clock cycles.

Enhancements to the DS5203 frameworks

The frameworks for the DS5203 FPGA Board (7K325) and DS5203 FPGA Board (7K410) now provide new access types to access data values of the PHS bus.

64-bit data format The DS5203 frameworks provide 64-bit registers and buffers to exchange data values with the PHS bus.

Related topics

Basics

- *Migrating to RTI FPGA Programming Blockset 3.1* on page 168

Migrating to RTI FPGA Programming Blockset 3.1

Objective

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

Migrating from RTI FPGA Programming Blockset 1.1 and higher to 3.1

If you have implemented your FPGA application using RTI FPGA Programming Blockset Version 1.1 and higher, and want to use it with RTI FPGA Programming Blockset 3.1, the framework will automatically update itself to the current framework version.

The update handles all the subsystems in the model/subsystem. The parameters of the blocks stay the same after updating to the current framework version.

ConfigurationDesk custom functions incompatible with dSPACE Release 2016-A

Note

Relevant for SCALEXIO systems with a DS2655 FPGA Base Board and a DS2655M1 Multi-I/O Module

A custom function generated by using RTI FPGA Programming Blockset 2.5 from dSPACE Release 2013-A and the real-time applications (*.rta) containing the custom function are incompatible with dSPACE Release 2016-A. To produce a usable custom function you have to rebuild the FPGA model by using RTI FPGA Blockset 3.1 from dSPACE Release 2016-A.

Using different dSPACE hardware

Using an FPGA model on different dSPACE hardware requires some model modifications. Refer to *Migrating to Another dSPACE Hardware* (🕮 *RTI FPGA Programming Blockset Guide*).

RTI LIN MultiMessage Blockset

Where to go from here

Information in this section

New Features of the RTI LIN MultiMessage Blockset 2.6

Support of SCALEXIO systems with a DS6301 CAN/LIN Board

The RTI LIN MultiMessage Blockset supports SCALEXIO systems with a DS6301 CAN/LIN Board. The DS6301 CAN/LIN Board provides four LIN channels.

Support of AUTOSAR System Template 4.2.2

The RTI LIN MultiMessage Blockset now also supports the AUTOSAR System Template based on AUTOSAR Release 4.2.2 for describing LIN networks.

Refer to *General Settings Page (RTILINMM MainSetup)* (🕮 *RTI LIN MultiMessage Blockset Reference*).

Support of J2602-compliant AUTOSAR system description files

The RTI LIN MultiMessage Blockset supports the SAE J2602 standard. In addition to J2602-compliant LDF files, the RTI LIN MultiMessage Blockset now also supports AUTOSAR system description files containing information in accordance with the SAE J2602 standard for describing LIN networks.

When working with a J2602-compliant LIN or AUTOSAR system description file, the RTI LIN MultiMessage Blockset supports the same LIN attributes as when you work with other database file types. J2602-specific attributes are not supported.

Migrating to RTI LIN MultiMessage Blockset 2.6

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 RTILINMM blocks and save the model before modifying the LIN configuration.

To create new S-functions for all the RTILINMM blocks in your model in one step, you can perform one of the following actions after opening the model:

- In the MATLAB Command Window, enter `rtimmsu_update('System', gcs)`.

  For more information on the command and its options, enter `help rtimmsu_update` in the MATLAB Command Window.
- Select the Create S-Function for all LIN Blocks command from the Options menu of the RTILINMM GeneralSetup block.

For more information, refer to *Limitations of RTI LIN MultiMessage Blockset* (🕮 *RTI LIN MultiMessage Blockset Reference*).

SCALEXIO Firmware

New Features of the SCALEXIO Firmware 3.4

New supported hardware

The SCALEXIO firmware supports the new SCALEXIO hardware:

- SCALEXIO Real-Time PC with a Intel® Xeon® Processor E3-1275 v3
- DS6301 CAN/LIN Board, a new standard I/O board with 4 CAN channels and 4 LIN channels
- SCALEXIO LabBox, connected with a DS6051 IOCNET Router

SystemDesk

Where to go from here

Information in this section

New Features of SystemDesk 4.6

Where to go from here

Information in this section

New General Features

Objective

SystemDesk 4.6 has the following new general features.

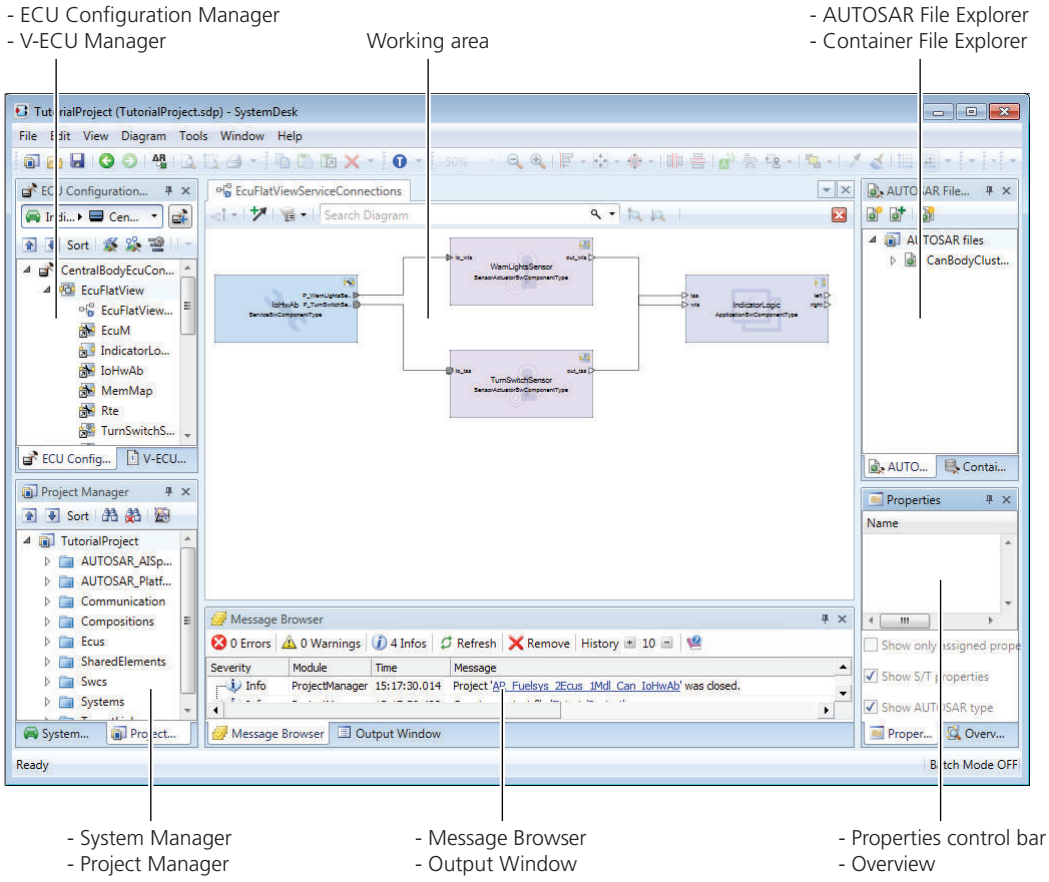
AUTOSAR Releases supported by SystemDesk 4.6

Modeling support for the AUTOSAR 4.2.2 Release
SystemDesk 4.6 supports the modeling of software and system architectures according the AUTOSAR 4.2.2 Release.

Compatiblity to recent AUTOSAR Releases SystemDesk also supports AUTOSAR Releases 4.2.1, 4.1.3, 4.1.2, 4.1.1, 4.0.3 and 4.0.2 for exchanging AUTOSAR files.

Integration layout

SystemDesk now provides the Integration layout, which highlights the control bars you need when you create V-ECUs. The following illustration shows the control bar arrangement of the layout:



Configuring ECUs

Improved generation of runnable mapping

The runnable mapping defines how runnables are scheduled on the ECU you are about to configure. SystemDesk provides the Generate Mappings command that lets you generate the runnable mapping, e.g., to quickly configure an ECU for virtual validation or as a basic mapping that you can refine.

SystemDesk's generation of the runnable mapping provides the following new options:

- Use rate monotonic scheduling: To order OS tasks by priority according to their sample rate, i.e., OS tasks with a small sample rate are given a high priority.
- Optimize mapping of server events: To not map synchronously called server runnables, which cannot be invoked concurrently, if the callers cannot preempt each other.

For reference information, refer to *Generate Mappings* (🕮 *SystemDesk Reference*).

Creating V-ECUs for Virtual Validation

Creating V-ECUs from external code

With this version, SystemDesk lets you create V-ECUs from external files. You can use SystemDesk's features such as building or exporting the V-ECU implementation. This lets you simulate external AUTOSAR or non-AUTOSAR code in the context of a V-ECU for virtual validation in offline or real-time scenarios.

Creating V-ECUs SystemDesk is the dSPACE software for creating V-ECUs.

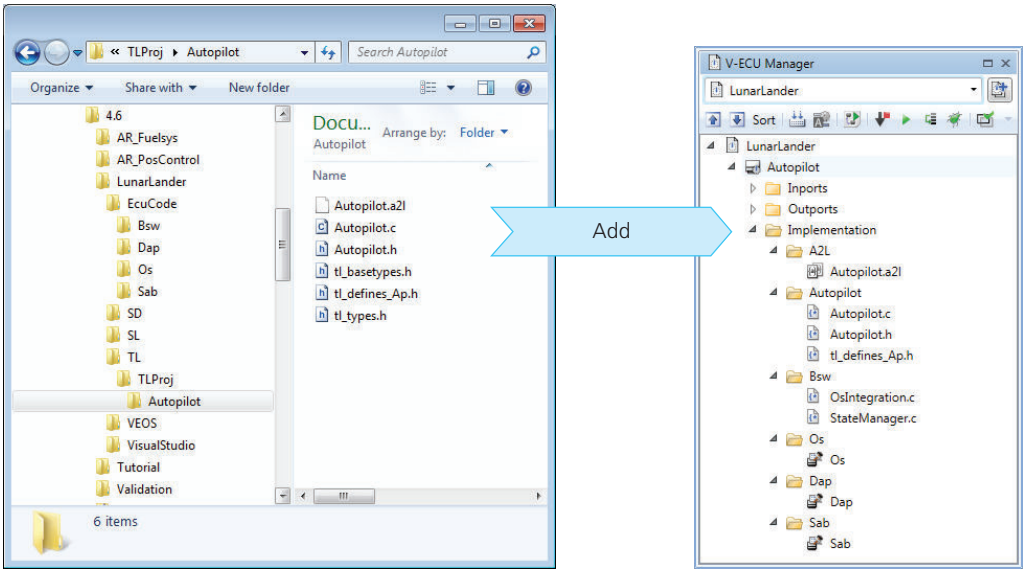
You can create and configure the following kinds of V-ECUs:

- Model-based V-ECUs: i.e., V-ECUs based on an ECU in an AUTOSAR system that is available in SystemDesk.
- Code-based V-ECUs: i.e., V-ECUs based on external files. SystemDesk lets you create a V-ECU implementation from external files.

Creating code-based V-ECU implementations To create a code-based V-ECU implementation, you have to perform the following:

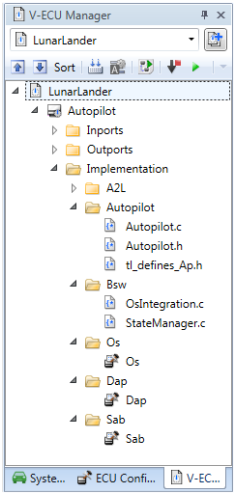
- Add platform-independent code files for application and basic SWCs
- Import or create and configure module configurations for the platform-dependent BSW
- Import A2L files with variable descriptions

The following illustration shows how to add external files to a code-based V-ECU.



For information on creating simulation systems and adding V-ECUs, refer to *Basics on Creating Simulation Systems* (🕮 *SystemDesk Guide*).

Lunar Lander demo model SystemDesk provides a new demo that shows how to create V-ECUs from external code. The V-ECU models a simplified Autopilot for landing a lunar module. The parameters of the Lunar Landing Module environment model are taken from the Apollo Lunar Module.



For information on SystemDesk's demos, refer to *Basics on Demos for SystemDesk* (🕮 *SystemDesk Guide*).

Migrating to SystemDesk 4.6

Migrating to SystemDesk 4.6

SystemDesk 4.6 automatically migrates SystemDesk 4.4, and 4.5 SDP project files upon loading.

Note

You are recommended to install the most recent patch for SystemDesk 4.4 or 4.5. Then, save the SDP project files you want to migrate before opening them in SystemDesk 4.6.

VEOS

Where to go from here

Information in this section

New Features of VEOS 3.6

Information in this topic

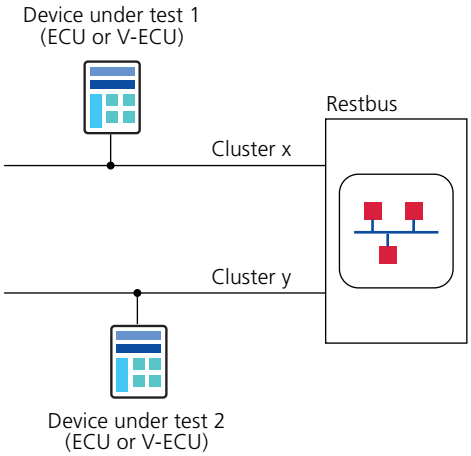
Support for CAN/LIN restbus simulation and import of bus simulation container (BSC) files

VEOS now supports *restbus simulation for CAN and LIN*.

Usually, not all of the ECUs in a simulation system are devices under test (DUTs). In such a system, the ECUs under test can be connected to a *restbus* that simulates the bus communication of other ECUs of the simulation system.

Restbus simulation lets you test one or more ECUs while simulating the other ECUs of the related communication clusters. The → *restbus* (📖 *VEOS Guide*) can simulate the bus communication of other ECUs of the simulation system. These restbus ECUs are combined and executed in a single application process, which simplifies the simulation system. The bus communication of the restbus can be manipulated, and the effects on the ECU(s) to be tested can be observed.

The following illustration shows a schematic of restbus simulation:



For more information, refer to *Restbus Simulation in VEOS* (📖 *VEOS Guide*).

Importing bus simulation containers To support restbus simulation, VEOS now lets you import bus simulation container (BSC) files to an offline simulation application. A BSC file contains the implementation of a bus configuration and is generated by the Bus Manager.

For information on generating BSC files with the Bus Manager, refer to *Workflows for Using the Bus Manager and Configuring Bus Communication for Offline Simulation* in the Bus Manager documentation.

For information on importing BSC files to VEOS Player, refer to *Importing Bus Simulation Containers* (📖 *VEOS Guide*).

Monitoring bus communication using ControlDesk's Bus Navigator

During offline simulation, you can monitor the entire bus communication by using ControlDesk's Bus Navigator. To do so, you use a bus monitoring device for the specific communication cluster type.

For instructions on configuring bus monitoring devices, refer to:

- *How to Configure a CAN Bus Monitoring Device* (🕮 *ControlDesk Next Generation Platform Management*)
- *How to Configure a LIN Bus Monitoring Device* (🕮 *ControlDesk Next Generation Platform Management*)

For more information on bus monitoring by using ControlDesk's Bus Navigator, refer to *Monitoring, Logging and Replaying Bus Communication* (🕮 *ControlDesk Next Generation Bus Navigator*).

Manipulating bus messages

During offline simulation, you can handle CAN messages and LIN frames of a bus VPU to manipulate restbus communication, for example. Restbus communication is the bus communication of the bus VPU. ControlDesk's Bus Navigator provides specific Bus Instruments for this.

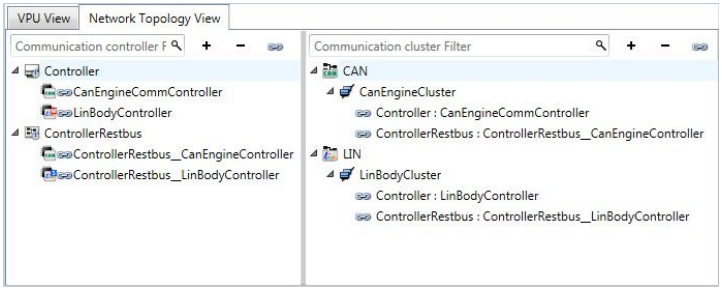
For more information, refer to *Working with Instruments of the Bus Navigator* (🕮 *ControlDesk Next Generation Bus Navigator*).

Displaying and editing bus simulation elements

VEOS Player now displays bus simulation elements in the new Network Topology View.

Example The following illustration shows an example of the Network Topology View of an offline simulation application consisting of a V-ECU named Controller and a bus VPU named ControllerRestbus:

- The CAN controllers of the V-ECU and bus VPU are connected via the CAN communication cluster named CanEngineCluster.
- The LIN controllers of the V-ECU and bus VPU are connected via the LIN communication cluster named LinBodyCluster.



Refer to *Network Topology View* (🕮 *VEOS Player Reference*).

Connecting controllers to clusters VEOS Player lets you connect communication controllers to communication clusters.

For instructions, refer to *How to Connect Communication Controllers to Communication Clusters* (🕮 *VEOS Guide*).

C++ support

The VEOS Player lets you import and build model implementations containing C and C++ source code for the host PC target.

However, there is a limitation with the C++ support in connection with the MSVC compiler. Refer to *Limitations for VEOS* (🕮 *VEOS Guide*).

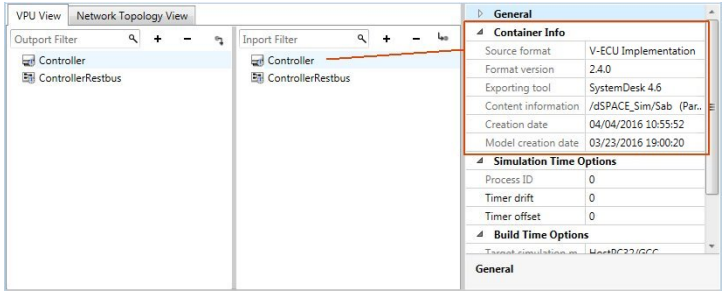
Display of container information

For each VPU in an offline simulation application, VEOS Player now displays information on the container that the VPU was built from.

Information such as the following is displayed:

- Source format, e.g., V-ECU Implementation
- Format version
- Exporting tool, e.g., SystemDesk 4.6

The following illustration shows an example:



For more information, refer to *V-ECU / Environment VPU / Controller VPU / Bus VPU* (🕮 *VEOS Player Reference*).

Compatibility of VEOS 3.6

Information in this topic

Compatibility overview

Compatibility in general dSPACE recommends using only software products from the same dSPACE Release. This provides maximum run-time compatibility.

OSA compatibility VEOS 3.6 is compatible with offline simulation application (OSA) files created with products of dSPACE Release 2016-A (OSA version 3.6).

Note

- OSA files created or modified with VEOS 3.6 cannot be loaded in earlier VEOS versions.
- The following applies for OSA files created or modified with VEOS 3.5 or earlier:
  - They can be loaded and simulated in VEOS 3.6 only if it does not contain bus communication elements.
  - They can neither be modified in VEOS 3.6 nor imported into an OSA file created with VEOS 3.6.

CTLGZ compatibility The following table shows the compatibility between VEOS 3.6 and CTLGZ files (V-ECU implementations):

V-ECU Implementations Created with Products of...	V-ECU Implementation Version
dSPACE Release 2013-B and earlier: ■ SystemDesk 3.2 ■ TargetLink 3.5	1.0
dSPACE Release 2014-A: ■ SystemDesk 4.2	2.0
dSPACE Release 2014-B: ■ SystemDesk 4.3 ■ TargetLink 4.0	2.1
dSPACE Release 2015-A: ■ SystemDesk 4.4	2.2
dSPACE Release 2015-B: ■ SystemDesk 4.5 ■ TargetLink 4.1	2.3
dSPACE Release 2016-A: ■ SystemDesk 4.6	2.4

SIC compatibility VEOS 3.6 is compatible with Simulink implementation container (SIC) files created with Model Interface Package for Simulink 3.2 from dSPACE Release 2016-A (SIC version 1.1).

BSC compatibility VEOS 3.6 is compatible with bus simulation container (BSC) files created with the Bus Manager of dSPACE Release 2016-A (BSC version 1.0).

FMU compatibility VEOS supports only the FMI for Co-Simulation interface, but not the FMI for Model Exchange interface.

For detailed and up-to-date compatibility information on FMI support in VEOS, refer to:

http://www.dspace.com/go/FMI-Compatibility.

Real-Time Testing compatibility To use RTT in connection with VEOS and ControlDesk, the RTT version used by the VEOS Simulator running the simulation system and the RTT version active on the host PC must be identical.

The table below shows the VEOS Simulator version and the corresponding RTT version:

VEOS Simulator	Real-Time Testing Version
... from VEOS 3.0	Real-Time Testing Version 2.0
... from VEOS 3.1	Real-Time Testing Version 2.2
... from VEOS 3.2	Real-Time Testing Version 2.3
... from VEOS 3.3	Real-Time Testing Version 2.4
... from VEOS 3.4	Real-Time Testing Version 2.5
... from VEOS 3.5	Real-Time Testing Version 2.6
... from VEOS 3.6	Real-Time Testing Version 3.0

ControlDesk 5.6 automatically uses the VEOS Simulator from VEOS 3.6. You can therefore use RTT in connection with VEOS and ControlDesk if RTT 2.6 is active on the host PC.

Migrating to VEOS 3.6

Changed behavior when importing an FMU to VEOS Player

As of VEOS 3.6, the import behavior for Functional Mock-up Units (FMUs) is as follows:

If the FMU to be imported contains C source code files and a DLL compiled for 32-bit Windows, VEOS Player uses the latter for the build process.

Migrating ASM models

You cannot simulate an ASM model on VEOS 3.6 (dSPACE Release 2016-A) if the model is contained in an OSA or SIC file created with a previous dSPACE Release.

To simulate an ASM model that was last saved with a dSPACE Release earlier than Release 2016-A on VEOS 3.6, perform the following steps:

1. Migrate the ASM model to dSPACE Release 2016-A.

   For information on migrating ASM models, refer to *Migrating ASM Models* (🕮 *ASM User Guide*).

2. Generate a Simulink implementation container (SIC) file on the basis of the ASM model by using the *Model Interface Package for Simulink*.

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

3. Import the SIC file to the VEOS Player of VEOS 3.6.

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

Compatibility Information

Where to go from here

Information in this section

Supported MATLAB Releases

MATLAB®

Working with various dSPACE products requires that you have installed MATLAB.

Tip

For system requirements of MathWorks® software, refer to http://www.mathworks.com/support/sysreq/current_release.

MATLAB Release...	...Is Supported by dSPACE Release 2016-A					
	RCP and HIL Software	AutomationDesk 5.2 [1]	TargetLink 4.1	Model Compare 2.6	dSPACE Python Extensions 2.1 [2]	XIL API .NET MAPort 2016-A
R2016a (64-bit)	✓ [3]	✓	–	–	✓	✓
R2015b (64-bit)	✓	✓	✓	✓	✓	✓
R2015a SP1 (64-bit)	✓	✓	✓	✓	✓	✓
R2014b (64-bit)	✓	✓	✓	✓	✓	✓
R2014a (64-bit)	–	–	✓	✓	–	–

[1] AutomationDesk's MATLAB Access library requires MATLAB.
[2] matlablib2 of dSPACE Python Extensions requires MATLAB.
[3] R2016a is not supported by the RTI FPGA Programming Blockset – FPGA Interface.

Note

As of dSPACE Release 2016-A, dSPACE software only supports 64-bit MATLAB variants. 32-bit MATLAB variants are not supported any longer.

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

Notes on MATLAB support There are some product-specifc notes and limitations for MATLAB support. Refer to *Overview of Bit Architecture and MATLAB Support of dSPACE Products on DVDs* on page 193

RCP and HIL software: C-compiler for building MEX files Note that the RCP and HIL software supports only Microsoft Windows SDK 7.1 to build MEX functions.

This compiler is a free download from Microsoft. The compiler additionally requires the .NET framework 4.0, which is also available at no charge from Microsoft. To download the compiler and the

framework and for further instructions, refer to http://www.mathworks.com/support/compilers/R2016a/index.html.

You need to install this compiler and configure it as a MEX compiler in MATLAB if you intend to use RCP and HIL products that require a MEX compiler, such as RTI CAN MultiMessage Blockset, RTI LIN MultiMessage Blockset, or Automotive Simulation Models.

Operating System

Operating system on host PC

The following operating system is supported by the dSPACE products on dSPACE Release 2016-A:

- Windows 7 Professional, Ultimate, and Enterprise with Service Pack 1 (64-bit version)

  Only the listed editions are supported. The Windows 7 Home and Starter editions are not supported.

  Note

  As of dSPACE Release 2016-A, dSPACE software only supports 64-bit operating systems. 32-bit operating systems are not supported any longer.

- Some limitations apply when you use Windows 7 in combination with dSPACE software. Refer to *Limitations for Using Windows 7* on page 197.

Using MicroAutobox Embedded PC as host PC:

ControlDesk Next Generation can also be installed on the MicroAutoBox Embedded PC (with Intel® Core™ i7-3517UE Processor) running under Windows 7 Professional, Ultimate, and Enterprise, 64-bit version.

Allowing communication via additional firewall rules

Additional Windows firewall rules are installed during the installation of various dSPACE software products. For example, one rule allows communication with a dSPACE expansion box such as AutoBox. Another rule allows MotionDesk to receive motion data from a network channel. These example rules are created by the following commands:

- `netsh 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."
```

- `netsh 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."`

If you are running third-party firewall software on your host PC, ensure that the TCP/IP communication of dSPACE software is not blocked.

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 Vista Business, Ultimate, or Enterprise (64-bit version) with the latest Service Pack
- Windows 7 Professional, Ultimate, or Enterprise (64-bit version) with the latest Service Pack
- Windows Server 2008 R2
- Windows Server 2012, Windows Server 2012 R2

Note

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

Run-Time Compatibility of dSPACE Software

Definition

Run-time compatibility means that:

- dSPACE products can be used in parallel after software installation, even if they are installed in different folders.
- dSPACE products without interaction can run independently of each other.

Compatibility of products in dSPACE Release 2016-A

dSPACE recommends using only software products from the same dSPACE Release. This provides maximum run-time compatibility.

Note that:

- Limitations regarding run-time compatibility in the dSPACE tool chain might occur if products from different dSPACE Releases are mixed.

  If dSPACE products interact directly (through automation interfaces) or indirectly (through common file types like A2L), limitations might apply. For minor limitations, refer to the relevant product documentation. The major limitations are described in the following.

  In rare cases, an additional patch must be installed for a product to achieve run-time compatibility. For more information on the patch and whether a patch is necessary, refer to http://www.dspace.com/go/CompPatch.

- RCP and HIL software products (on Release 2016-A) cannot be used in combination with RCP and HIL software products from earlier dSPACE releases.

Major limitation for working with a SCALEXIO system The products for working with a SCALEXIO system must be compatible. This is guaranteed only for products delivered with the same dSPACE Release. Contact dSPACE for more information if you have any questions.

Combining dSPACE products from earlier releases

For more information and notes on the combined use of different products from and with earlier releases, refer to http://www.dspace.com/go/ds_sw_combi.

Overview of Bit Architecture and MATLAB Support of dSPACE Products on DVDs

Objective

As of dSPACE Release 2016-A, dSPACE software supports only 64-bit operating systems and 64-bit MATLAB variants. However, some dSPACE products are available as a 32-bit variant on the dSPACE DVD set.

There are some product-specific notes and limitations for MATLAB support and for using the 32-bit dSPACE software products under 64-bit operating systems. Refer to the sections below the following table.

Overview

The following table shows a detailed list of all dSPACE products on the dSPACE DVDs, their MATLAB support, and their support of the bit architecture:

dSPACE Product		Product Supports 64-bit MATLAB	Product Independent of MATLAB Bit Architecture	Product Contained as 32-Bit Variant
ControlDesk Next Generation		–	✓	✓
SystemDesk		–	✓	✓
AutomationDesk		✓	–	✓
TargetLink		✓	–	–
Model Compare		✓	–	–
VEOS		–	✓	✓
Real-Time Testing		–	–	✓
Platform API Package	dSPACE Python Extensions	✓ 1)	✓ 2)	✓
	HIL API .NET MAPort	–	✓ 3)	✓
	XIL API .NET MAPort	✓ 4)	✓ 4)	✓
Failure Simulation Package	XIL API .NET EESPort	–	✓	✓

dSPACE Product		Product Supports 64-bit MATLAB	Product Independent of MATLAB Bit Architecture	Product Contained as 32-Bit Variant
RCP and HIL software package	RTI and RTI-MP	✓	–	–
	RTI Gigalink Blockset	✓	–	–
	RTI CAN Blockset	✓	–	–
	RTI CAN MultiMessage Blockset	✓	–	–
	RTI LIN MultiMessage Blockset	✓	–	–
	RTI FlexRay Configuration Blockset	✓	–	–
	RTI FPGA Programming Blockset	✓	–	–
	RTI Electric Motor Control Blockset	✓	–	–
	RTI Ethernet Blockset	✓	–	–
	RTI Ethernet UDP Blockset	✓	–	–
	RTI XCP on Ethernet Blockset	✓	–	–
	RTI Watchdog Blockset	✓	–	–
	RTI RapidPro Control Unit Blockset	✓	–	–
	RTI Bypass Blockset	✓	–	–
	RTI USB Flight Recorder Blockset	✓	–	–
	Bus Manager (stand-alone)	–	✓	✓
	ConfigurationDesk	✓	–	✓
	FlexRay Configuration Blockset	✓	–	–
	FlexRay Configuration Tool	–	✓	✓
	ModelDesk	✓	–	✓
	Automotive Simulation Models	✓	–	–
	MotionDesk	✓	–	✓
	MotionDesk Blockset	✓	–	–
	Flight Rec Data Merger	–	✓	✓

dSPACE Product		Product Supports 64-bit MATLAB	Product Independent of MATLAB Bit Architecture	Product Contained as 32-Bit Variant
	Model Interface Package for Simulink	✓	–	–
	Further products of RCP and HIL software package	–	✓	✓

[1] dSPACE Python Extensions contain the matlablib2 Python library. This library supports remote control and access of 64-bit MATLAB. matlablib2 itself is contained on the 64-bit DVD as a 32-bit variant.

[2] HIL API MAPort (Python), rtplib2 und rs232lib2 and the Platform Management API are independent of MATLAB architecture and contained on the 64-bit DVD as a 32-bit variant.

[3] HIL API .NET MAPort cannot be used from 64-bit MATLAB.

[4] XIL API .NET MAPort can be used from 64-bit MATLAB via MATLAB Interface for .NET.

For more information on the compatibility of dSPACE products with 64-bit MATLAB versions, refer to http://www.dspace.com/go/matlab64bit.

Product-specific MATLAB limitations

Restricted MAT file support The Signal Editor of ControlDesk Next Generation (ControlDesk 5.6) only supports reading and writing MAT files of file format version 5.0. MAT files of this version can be created in MATLAB by using the `save` command with the option '-v6'.

ModelDesk When you use MATLAB R2016a for Simulink simulation, there are the following limitations when the simulation runs:

- Starting and stopping plotting requires up to one minute.
- Downloading road, maneuver and traffic scenarios requires up to several minutes.
- Downloading parameter sets and processing executions is not recommended.

There are no limitations when the simulation is stopped or paused.

Using 32-bit dSPACE software under Windows 7 (64-bit)

32-bit dSPACE software runs under 64-bit Windows operating systems in a WoW64 (Windows-on-Windows 64-bit) subsystem. WoW64 is the x86 emulator of Windows that allows 32-bit Windows-based applications to run seamlessly on 64-bit versions of Windows. This lets you use up to 4 GB of virtual memory for each 32-bit process if the application is prepared for using the large memory area. Otherwise, the virtual address space of a process is limited to 2 GB.

Limitations of device drivers	Third-party bus interfaces (CAN, LIN, or FlexRay) are supported by 32-bit dSPACE software products only if they have 64-bit drivers from the manufacturers.
Limitations for TargetLink	**Importing an A2L file** The import of A2L (ASAM MCD-2 MC) files is not supported in the 64-bit version of TargetLink.

Tip

You can use a workaround to import A2L files in the 64-bit version of TargetLink: The stand-alone TargetLink Data Dictionary Manager is shipped as a 32-bit application. Open your current DD file in the stand-alone TargetLink Data Dictionary Manager, import your data via A2L Import, and save the DD file again. After reloading the DD file in the 64-bit instance, you can continue your work.

Limitations for Using Windows 7

Objective	Some limitations apply when you use Windows 7 in combination with dSPACE software.
Fast user switching not supported	dSPACE software does not support the fast user switching feature of Windows.
Closing dSPACE software before PC shutdown	The shutdown procedure of Windows operating systems might cause 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 performing a PC shutdown.
User Account Control	It is recommended to disable Windows' User Account Control (UAC) during the installation of dSPACE software. If you cannot disable UAC, note the following Windows behavior: If UAC is enabled, the setup programs run with the administrator account instead of the user account. Therefore, it is important that the administrator account has access to the required drives, particularly the required network drives.

USB devices

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

Windows' 8dot3name creation option must be enabled

Note

It is strongly recommended that Windows' 8dot3name creation option is enabled (= default setting after installation of Windows) before installing third-party software (such as MATLAB®/Simulink®) and the dSPACE software.

If the option is disabled during software installation, serious errors can occur when running the dSPACE software, for example, the build process might be aborted. To repair an installation that has been installed with the disabled 8dot3name creation option, you have to reinstall the dSPACE software and the required third-party software. Using the dSPACE Maintenance Setup does not solve this problem.

For instructions on checking the setting and enabling the option, refer to the Microsoft Windows documentation.

P

R

S

U

V

W