ControlDesk

Calibration and Data Set Management

For ControlDesk 7.4

Release 2021-A – May 2021

dSPACE

Contents

Reference Information   69

Automation   119

Troubleshooting   127

Limitations   131

Glossary   133

Index   171

About This Document

Content

This document introduces you to calibration and data set management with ControlDesk.

Symbols

dSPACE user documentation uses the following symbols:

Symbol	Description
⚠ DANGER	Indicates a hazardous situation that, if not avoided, will result in death or serious injury.
⚠ WARNING	Indicates a hazardous situation that, if not avoided, could result in death or serious injury.
⚠ CAUTION	Indicates a hazardous situation that, if not avoided, could result in minor or moderate injury.
NOTICE	Indicates a hazard that, if not avoided, could result in property damage.
Note	Indicates important information that you should take into account to avoid malfunctions.
Tip	Indicates tips that can make your work easier.
⁉	Indicates a link that refers to a definition in the glossary, which you can find at the end of the document unless stated otherwise.
📖	Precedes the document title in a link that refers to another document.

Naming conventions

dSPACE user documentation uses the following naming conventions:

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

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

Special folders

Some software products use the following special folders:

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

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

or

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

Documents folder   A standard folder for user-specific documents.

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

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

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

Accessing dSPACE Help and PDF Files

After you install and decrypt dSPACE software, the documentation for the installed products is available in dSPACE Help and as PDF files.

dSPACE Help (local)   You can open your local installation of dSPACE Help:
- On its home page via Windows Start Menu
- On specific content using context-sensitive help via **F1**

dSPACE Help (Web)   You can access the Web version of dSPACE Help at www.dspace.com/go/help.

To access the Web version, you must have a *mydSPACE* account.

PDF files   You can access PDF files via the icon in dSPACE Help. The PDF opens on the first page.

Tutorial Videos

Tutorial Videos

Introduction	The dSPACE website provides calibration tutorial videos.

Changing values in the Table Editor	You can use the Table Editor in ControlDesk to display and edit values of variables with data types that can be represented in tabular form, such as maps and curves.

The Table Editor lets you change values in various ways.

The tutorial videos show you:
- How to change function values.
- How to change axis points and export table data.

Refer to https://www.dspace.com/go/tutorial_cd_te_change_val.

Public product videos	For public product videos, refer to ControlDesk product videos.

Related topics	**Basics**

> Tutorial Videos for ControlDesk (ControlDesk Introduction and Overview 📖)

Basics and Instructions

Where to go from here

Information in this section

Calibrating Parameters

Introduction   ControlDesk allows you to perform the calibration of parameter values on dSPACE real-time hardware and on ECUs.

Where to go from here   Information in this section

Introduction to Offline and Online Calibration

Introduction   ControlDesk uses a specific memory concept for calibrating parameters. This allows you to perform parameter calibration either offline on the host PC, or online on the hardware.

Where to go from here   Information in this section

Basics on Memory Pages

Introduction

Calibration memory usually consists of one or more areas called *memory pages*. Memory pages are used for parameter calibration. Each page contains a complete set of parameters, but only one of the pages is visible to the the ECU microcontroller at a time.

ControlDesk supports memory page concepts with one or two memory pages.

> **Note**
>
> The memory of dSPACE real-time hardware does not consist of memory pages.

Implementation of memory pages

ECU with CCP or XCP and ECU with DCI-GSI2   The ECU memory is divided into one or more areas representing the ECU's memory pages. For example, the ECU's flash memory may contain the reference page, whereas the ECU's RAM may contain the working page. CCP or XCP is used as the serial communication protocol to access the pages.

ECUs with two memory pages

Many ECUs (or the connected ECU interface) and V-ECUs ⧉ generated with VEOS 4.3 or later provide two memory pages. One is the *working page*, the other is the *reference page*:

Working page   The working page contains a complete data set with ECU parameters. You can change the values of these parameters with ControlDesk. The working page allows you to perform parameter calibration.

Reference page   The reference page also contains a complete data set with ECU parameters, but you cannot change the values of these parameters. The page serves as a read-only reference, which allows you to run the ECU with a proven set of ECU parameters.

ECUs with one memory page	Some ECUs provide only one memory page. This page is the *working page*.

Working page   The working page contains a complete data set with ECU parameters. The working page allows you to perform parameter calibration. You can change the values of these parameters with ControlDesk.

If the working page is located in a read-only memory area of the ECU, you can calibrate the parameters only offline. To store the modified parameter values on your ECU permanently, you have to create an ECU Image file containing the offline calibrated parameter values, and then flash the ECU Image file to the ECU memory.

Reference page   ECUs with one memory page do not provide a reference page.

Calibration memory segments	The memory of an ECU is usually divided into calibration memory segments containing the calibratable parameters. The segments must be defined as `MEMORY_SEGMENT` in the A2L file. Calibration memory segments can be used to evaluate the memory pages of an ECU when online calibration is started. Refer to Basics on Starting Online Calibration on page 17.

Related topics	**Basics**

HowTos

References

Basics on Offline Calibration

Introduction	*Offline calibration* is a state in which the parameter values cannot be changed directly on the platform/device hardware. This applies regardless of whether or not the host PC is physically connected to that hardware. However, ControlDesk uses *mirrored memory* on the host PC to change parameter values *offline*.

Mirrored memory for offline calibration	Using mirrored memory on the host PC, you can change parameter values even if online calibration is not started yet, that is, even if there is no communication

between ControlDesk and the dSPACE real-time hardware or ECU. In this case, changes to parameter values affect only the *mirrored memory*. It mirrors the contents of the available memory pages of measurement and calibration devices and platforms.

The mirrored memory allows you to perform offline calibration.

Initial filling of the mirrored memory

Initial filling for calibration devices   ControlDesk initially fills the mirrored memory of a calibration device with the contents of the device's ECU Image file.

> ⚠ **WARNING**
>
> If the ECU Image file of a device does not provide initial parameter values for the entire memory region described by the memory segments of the device, ControlDesk sets the remaining parameter values to 0 in the mirrored memory and in the initially created data sets.
>
> Do not download such data sets to the connected device hardware when you start online calibration, since this may cause property damage or even personal injury in connection with the connected system. Upload the parameter values from the connected device hardware instead.

Initial filling for platforms   ControlDesk initially fills the mirrored memory of a platform with the contents of the platform's variable description (SDF) file, which is possible only if the SDF file contains initial parameter values

- You can configure the RTI build process to include initial parameter values in the SDF file to be generated. Refer to the *RTI variable description file options category* on the Code Generation Dialog (Model Configuration Parameters Dialogs) (RTI and RTI-MP Implementation Reference 📖).
- You can make the initial parameter values available in the variable description file generated by ConfigurationDesk. Refer to Adapting the Generation of the Variable Description File (Model Interface Package for Simulink - Modeling Guide 📖).

Initial data set contents   ControlDesk initially fills the working and reference data sets with the parameter values of the mirrored memory.
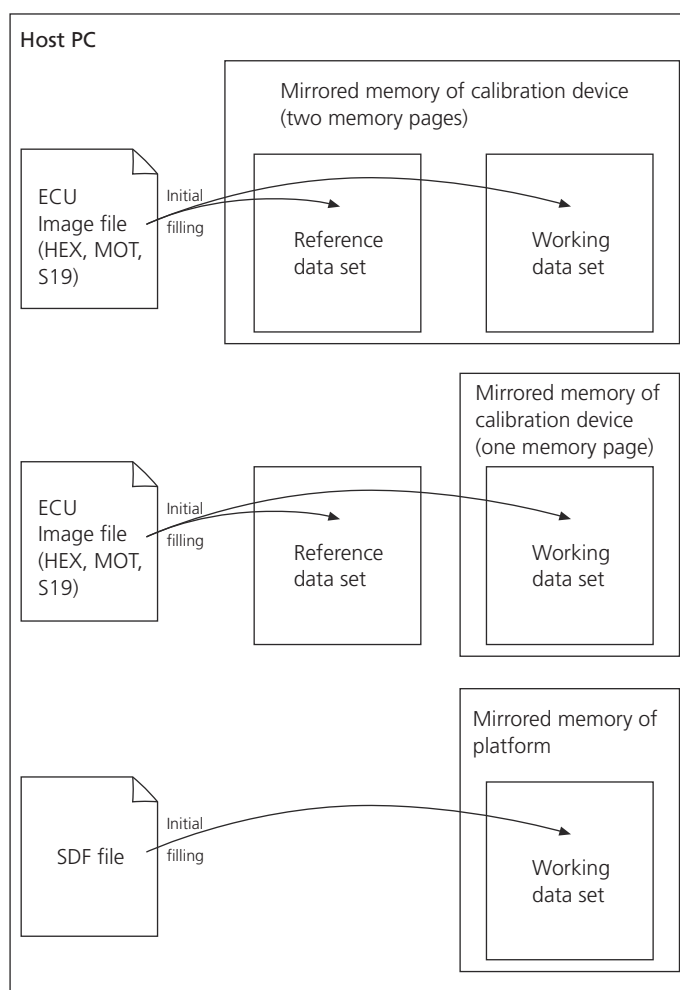
Tip

- Generating an initial data set for platforms must be enabled on the Data Set Management page of the ControlDesk Options dialog.
- In some cases, ControlDesk cannot create an initial data set automatically. In these cases, you can manually create an initial data set via the Create Data Set command.

Related topics

Basics

Basics on Starting Online Calibration

Introduction

To change parameter values on dSPACE real-time hardware or ECUs with ControlDesk, you must start online calibration.

Comparing memory contents before starting online calibration

When you start online calibration, ControlDesk lets you compare the contents of the mirrored memory of each platform/device with the memory pages of the connected dSPACE real-time hardware or ECUs.

If calibratable ECU parameters are located in calibration memory segments defined in the A2L file, ControlDesk compares the content of each segment via CRC checks.

Uploading/downloading calibration memory contents

If the contents of the mirored memory and the memory on the ECU or real-time hardware differ, you should equalize them by uploading the memory contents from the hardware to the host PC, or downloading them from the host PC to the hardware.

```
┌─────────────────────────────────────────────┐
│ Host PC                                       │
│                                               │
│        ┌───────────────────────────┐          │
│        │                           │          │
│        │  Mirrored memory content  │          │
│        │                           │          │
│        └───────────────────────────┘          │
│                                               │
└─────────────────────────────────────────────┘
             Download          Upload

┌─────────────────────────────────────────────┐
│ ECU or dSPACE                                 │
│ real-time hardware                            │
│        ┌───────────────────────────┐          │
│        │                           │          │
│        │     Memory content        │          │
│        │                           │          │
│        └───────────────────────────┘          │
│                                               │
└─────────────────────────────────────────────┘
```
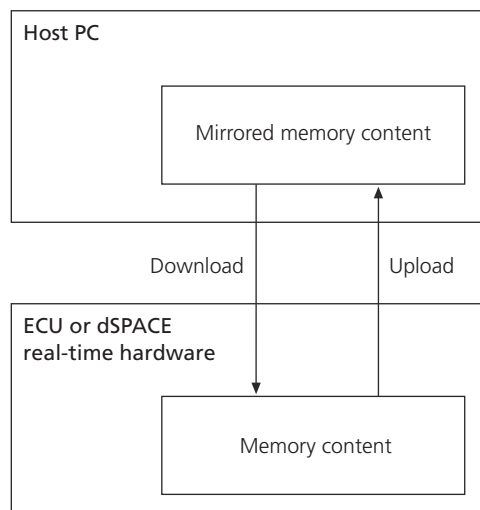
For ECUs, up/downloading actually means:

- Uploading/downloading the contents of the memory segments. Only segments with different contents are up/downloaded.
- Uploading/downloading the values of the parameters which are located outside the memory segments and connected to an instrument on a layout.

Refer to ECU Parameter Values After Online Calibration Start on page 22.

> ⚠ **WARNING**
>
> If the ECU Image file of a device does not provide initial parameter values for the entire memory region described by the memory segments of the device, ControlDesk sets the remaining parameter values to 0 in the mirrored memory and in the initially created data sets.
> Do not download such data sets to the connected device hardware when you start online calibration, since this may cause property damage or even personal injury in connection with the connected system. Upload the parameter values from the connected device hardware instead.

Platform/device-specific aspects of the up/download

The way you can up/download calibration memory contents depends on the platform/device type and on whether the reference page is read-only or not. The table below shows the up/download possibilities:

|	Upload from …		Download to …	
	… Working Page of the Hardware	… Reference Page of the Hardware	… Working Page of the Hardware	… Reference Page of the Hardware
ECUs with two memory pages				
ECU with DCI-GSI2	✓	✓	✓	– [1] / ✓ [2]
ECU with CCP	✓	✓	✓	– [1] / ✓ [2]
ECU with XCP[3]	✓	✓	✓	– [1] / ✓ [2]
ECUs with one memory page				
ECU with CCP	✓	–	– [1] / ✓ [2]	–
ECU with XCP	✓	–	– [1] / ✓ [2]	–
dSPACE real-time hardware				
DS1006 Processor Board	✓ [4], [5]		– [6] / ✓ [5]	
DS1007 PPC Processor Board	✓ [4], [5]		– [6] / ✓ [5]	
DS1104 R&D Controller Board	✓ [4], [5]		– [6] / ✓ [5]	
MicroAutoBox II/III	✓ [4], [5]		– [6] / ✓ [5]	
MicroLabBox	✓ [4], [5]		– [6] / ✓ [5]	
SCALEXIO	✓ [4], [5]		– [6] / ✓ [5]	
VEOS	✓ [4], [5]		– [6] / ✓ [5]	

[1] If the page is located in a read-only memory area (e.g., flash memory), downloading is impossible.
[2] If the page is located in a read/write memory area (e.g., RAM), downloading is possible.
[3] Including V-ECUs ⏎

	Upload from …		Download to …	
	… Working Page of the Hardware	… Reference Page of the Hardware	… Working Page of the Hardware	… Reference Page of the Hardware

[4] If you selected one or more data sets for downloading on calibration start, the parameter values of these data sets are downloaded in any case. They are downloaded even if you specified, for example, Upload as the Start online calibration behavior for the related platform. Refer to How to Select Additional Data Sets for Downloading on Online Calibration Start on page 58.

[5] The memory of dSPACE real-time hardware does not consist of memory pages. You therefore cannot select between a working and a reference page.

[6] If the SDF file for dSPACE real-time hardware contains no initial parameter values, downloading is not possible until you have uploaded parameter values from the hardware to the host PC. You can configure the RTI build process to include initial parameter values in the SDF file to be generated. Refer to the *RTI variable description file options page* in the Code Generation Dialog (Model Configuration Parameters Dialogs) (RTI and RTI-MP Implementation Reference 🕮).

Equalizing the memory contents

To get identical parameter values on the host PC and on the connected hardware, ControlDesk opens the Calibration Memory dialog for you to upload or download the parameter values.

Configuring the start online calibration behavior   To prevent ControlDesk from always opening the Calibration Memory dialog, you can specify a default start online calibration behavior individually for each platform/device. It defines the way ControlDesk is to up/download the parameter values when online calibration is started, without prompting you.

You can specify the start online calibration behavior for a platform/device via its Properties controlbar (refer to General Settings Properties (ControlDesk Platform Management 🕮)).

> **Note**
>
> When specifying the start online calibration behavior of a platform/device, you must decide between safe and fast online transition.

The following table shows the default start online calibration behaviors provided by ControlDesk. The available default behaviors depend on the selected platform/device type and the ECU's memory page concept, on whether calibration memory segments are defined in the A2L file of the ECU, and on whether parameters outside the memory segments are selected for calibration.

Start Online Calibration Behavior	Description
Prompt user	ControlDesk opens the Calibration Memory Dialog, prompting you to upload or download the parameter values.
Upload[1]	ControlDesk uploads the parameter values from the hardware to ControlDesk. If no initial data set is available for the platform/device, ControlDesk creates a working data set during the upload, and also a reference data set if the hardware supports this.
Download [2]	ControlDesk downloads the parameter values from ControlDesk to the hardware.
Upload WP, Upload RP	For the memory pages whose contents differ from the contents of the mirrored memory, ControlDesk uploads the parameter values from the connected hardware to ControlDesk. [3]

Start Online Calibration Behavior	Description
Download WP, Upload RP [2]	For the memory pages whose contents differ from the contents of the mirrored memory, ControlDesk equalizes the parameter values as follows: [3] ▪ For the working page, the parameter values are downloaded from ControlDesk to the hardware. ▪ For the reference page, ControlDesk uploads the parameter values from the hardware to ControlDesk.
Upload WP, Download RP [2]	For the memory pages whose contents differ from the contents of the mirrored memory, ControlDesk equalizes the parameter values as follows: [3] ▪ For the working page, ControlDesk uploads the parameter values from the hardware to ControlDesk. ▪ For the reference page, the parameter values are downloaded from ControlDesk to the hardware.
Download WP, Download RP [2]	For the memory pages whose contents differ from the contents of the mirrored memory, ControlDesk downloads the parameter values from ControlDesk to the connected hardware. [3]
Upload connected variables [1]	ControlDesk uploads the values of all the parameters and writable measurements that are currently selected for calibration from the hardware to the host PC. These are the parameters and writable measurements that are currently connected to an instrument and visualized on a layout. They are shown in the **Variables** controlbar with a red chain symbol. **Note** For a platform/device without a working data set, selecting the 'upload connected variables' option the first time online calibration is started prevents ControlDesk from creating a working and a reference data set. However, ControlDesk needs a working data set to save calibration data in ControlDesk. Use the 'upload' option for the first online transition instead.
Download connected variables [2]	ControlDesk downloads the values of all the parameters and writable measurements that are currently selected for calibration from ControlDesk to the hardware. These are the parameters and writable measurements that are currently connected to an instrument and visualized on a layout. They are shown in the **Variables** controlbar with a red chain symbol.
Ignore differences [2]	(Available only if memory segments are defined in the ECU's A2L file) Online calibration is started without a data consistency check and without equalizing the contents of the memory pages and the mirrored memory. ControlDesk lets you calibrate parameters even if the parameter values on the connected hardware and in ControlDesk are different. **Note** You must ensure data integrity yourself.

[1] If you selected one or more data sets for downloading on calibration start, the parameter values of these data sets are downloaded in any case. They are downloaded even if you specified, for example, **Upload** as the **Start online calibration behavior** for the related platform. Refer to How to Select Additional Data Sets for Downloading on Online Calibration Start on page 58.

[2] If no initial data set is available for the platform/device, ControlDesk nevertheless opens the **Calibration Memory** dialog the first time you start online calibration.

[3] ControlDesk uploads/downloads only memory segments that differ. Memory segments that are equal on the ECU memory and on the mirrored memory are not up/downloaded. This speeds up the up/download.

Online calibration after up/download

For each platform/device, ControlDesk lets you specify which memory page is to be active in ControlDesk and on the ECU after online calibration is started initially or after an automatic reconnect. After online calibration is started, the memory page is automatically activated according to this **Initial Page** setting. You can specify the initial page setting via the **Properties** controlbar (refer to General Settings Properties (ControlDesk Platform Management 📖)).

While online calibration is running, parameter value changes take effect synchronously on the memory pages of the hardware and in the mirrored memory. In other words, parameter values on the dSPACE real-time hardware or ECU and on the host PC are always the same as long as you are online.

Skipping the up/download (only measurement possible)

You can skip the up/download. The parameter values on the host PC and on the hardware are still inconsistent. Online calibration is not started, but you can measure variables.

Calibrating parameters is not possible.

Checking the application

When you start online calibration, ControlDesk checks whether the application currently running on the real-time hardware matches the variable description currently active in the ControlDesk experiment.

When you start online calibration, ControlDesk automatically detects whether an application has been terminated. For details, refer to Detecting terminated applications (ControlDesk Platform Management 📖).

Observing variables when online calibration is started

If the variable observer functionality is supported for a platform/device, ControlDesk starts observing variables on the hardware as soon as online calibration is started, that is, the observed values are displayed in ControlDesk even if no measurement is currently running.

For further information on the variable observer functionality, refer to Observing Variables (ControlDesk Measurement and Recording 📖).

Initial data synchronization when online calibration is started

ControlDesk supports synchronized data acquisition. To make signals from different sources comparable in a measurement, ControlDesk provides *initial synchronization* of the time stamps of different platforms/devices when online calibration is started and *periodic resynchronization* during measurement. Refer to Basics on Synchronized Data Acquisition (ControlDesk Measurement and Recording 📖).

You can increase the synchronization accuracy during the first minute of a measurement. If a high synchronization accuracy on measurement start is required, wait for about 20 s after going online globally before you start measuring.

Related topics

Basics

HowTos

References

ECU Parameter Values After Online Calibration Start

Introduction

The values of ECU parameters and writable measurements after online calibration start depend on:

- The memory location of a parameter or writable measurement
- The connection of a parameter or writable measurement to an instrument on the layout
- The selected upload/download behavior

Overview

The following table shows how ECU parameter values are uploaded or downloaded depending on their memory location, their connection to instruments, and the selected upload/download behavior.

Upload/Download Behavior [1]	Parameters In Memory Segment [2]		Parameters Outside Memory Segment [2], [3]		
	Parameter Connected [4]	Parameter Unconnected [4]	Parameter Connected [4]	Parameter Unconnected [4]	
				Ignore Unconnected = ☑ [5]	Ignore Unconnected = ☐ [6]
Upload	PC ⬆ ECU	PC ⬆ ECU	PC ⬆ ECU	–	PC ⬆ ECU
Download [7]	PC ⬇ ECU	PC ⬇ ECU	PC ⬇ ECU	–	PC ⬇ ECU

Upload/Download Behavior [1]	Parameters In Memory Segment [2]		Parameters Outside Memory Segment [2], [3]		
	Parameter Connected [4]	Parameter Unconnected [4]	Parameter Connected [4]	Parameter Unconnected [4]	
				Ignore Unconnected = ☑ [5]	Ignore Unconnected = ☐ [6]
Upload Connected	PC ⬆ ECU	–	PC ⬆ ECU	–	–
Download Connected [7]	PC ⬇ ECU	–	PC ⬇ ECU	–	–

[1] For more information on specifxying the upload and download behavior, refer to Uploading/downloading calibration memory contents on page 17.

[2] The Memory Segments dialog displays the memory segments of a specific device. Refer to Memory Segments (ControlDesk Platform Management 📖).

[3] Writable measurements are always located outside of memory segments.

[4] Connected parameters are marked by a red chain in the Variables controlbar.

[5] If Ignores unconnected parameters outside of memory segments is selected on the Data Acquisition page of the ControlDesk Options dialog. Refer to Data Acquisition Page (ControlDesk Measurement and Recording 📖).

[6] If Ignores unconnected parameters outside of memory segments is cleared on the Data Acquisition page of the ControlDesk Options dialog. Refer to Data Acquisition Page (ControlDesk Measurement and Recording 📖).

[7] Downloaded only if the parameter is completely available in the data set. Otherwise it is uploaded during the first online transition. Refer to Problem When Downloading ECU Parameters with RESERVED Entries Outside of Memory Segments on page 128.

Related topics

Basics

HowTos

References

Data Acquisition Page (ControlDesk Measurement and Recording 📖)
Memory Segments (ControlDesk Platform Management 📖)

How to Start and Stop Online Calibration

Objective

To experiment on the connected hardware or VEOS, you must start online calibration.

Basics

If online calibration is running when you modify parameters, they are immediately downloaded to the connected platform/device hardware.

If the experiment contains platforms, their applications are automatically loaded to the hardware or VEOS if no applications are loaded yet. You can let the loaded applications be started automatically when you start online calibration.

The **Project** 🔖 controlbar and the **Platforms/Devices** 🔖 controlbar indicate whether each platform/device is online or not. Refer to Basics of Platform/Device States (ControlDesk Platform Management 📖).

Preconditions

To start online calibration, at least one platform/device contained in the experiment must be available.

Method

To start and stop online calibration

1 On the Home ribbon, click Status Control – Go Online to start online calibration, or press `Ctrl+F7`.

   Before you can start online calibration, ControlDesk compares the parameter values on the connected hardware or VEOS with the corresponding parameter values in ControlDesk.

   As a consequence, ControlDesk opens the Calibration Memory dialog if one of the following conditions is true:

   ▪ The parameter values on the hardware or VEOS and in ControlDesk differ.

   ▪ There is at least one platform in the experiment whose *start online calibration behavior* is set to 'Prompt user'. See General Settings Properties (ControlDesk Platform Management 📖).

   ▪ Parameters outside the defined memory segments are selected for calibration.

   ▪ No memory segments are defined in the A2L file.
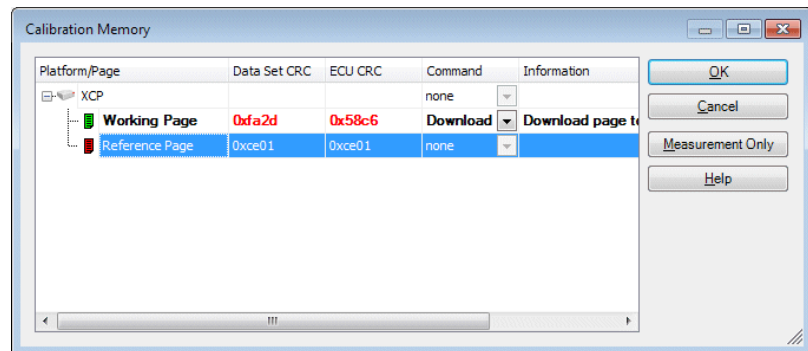
> **Tip**
>
> To prevent ControlDesk from always opening the Calibration Memory dialog, you can specify a default start online calibration behavior for each platform/device individually during platform/device configuration. ControlDesk then equalizes the parameter values on the hardware or VEOS and in ControlDesk according to the specified settings without opening the Calibration Memory dialog. However, if parameter upload/download according to the specified default behavior is not possible (for example, parameter values are to be downloaded, but no initial data is available), the Calibration Memory dialog nevertheless opens.

2 In the dialog, select a download or upload command for the relevant memory pages from the Command drop-down list, then click OK.

This equalizes the parameter values on the hardware or VEOS and in ControlDesk.

> **Tip**
>
> If you only want to measure variables, you can also click Measurement only. This has the following effects:
> - Calibrating parameters is not possible.
> - Measuring variables is possible.

ControlDesk checks if online calibration is possible for each platform/device in the experiment, as follows:

- Is the corresponding platform/device hardware physically connected to the host PC?
- Is the platform/device configured correctly?
- Is the ControlDesk license still valid?

If a connection cannot be established to the platform/device hardware or VEOS of at least one platform/device, ControlDesk opens the Platform/Device State Overview dialog (unless you disable the display of

that dialog on the **Platform Management** page of the **ControlDesk Options** dialog). The dialog looks similar to the following:
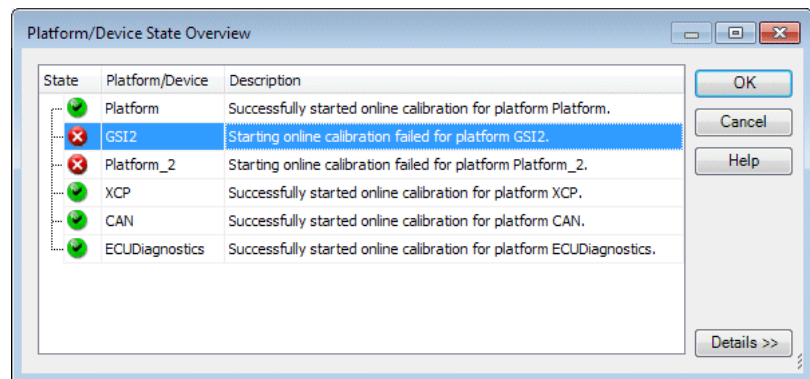
Online calibration is possible only for the platforms/devices whose state is ![icon].

3 Select **OK** (available only if online calibration is possible for at least one platform/device) or **Cancel**.

- If you click **OK**, ControlDesk starts online calibration for the platforms/devices for which it has detected no errors. Starting online calibration fails for all other platforms/devices, so that you cannot perform measurements, recordings, or online calibration on them.

- If you click **Cancel**, ControlDesk does not start online calibration. This lets you reconfigure the platforms/devices of your experiment, connect missing hardware to the host PC, for example, before you try to start online calibration again.

4 On the **Home** ribbon, click **Status Control – Go Offline** to stop online calibration, or press **Ctrl+F8**.

Result

ControlDesk either starts or stops communication with the available hardware or VEOS. The status bar displays whether ControlDesk is online or offline:

Visualization	Mode
Offline	Offline
Online	Online

After you start online calibration, the memory pages specified by the **Initial Page** properties of the platforms/devices are automatically activated in ControlDesk and on the ECU.

Tip

As long as ControlDesk is in online calibration mode, you can start and stop online calibration individually for single platforms/devices. This allows you, for example, to switch a single platform/device into the offline calibration mode while the other platforms/devices remain online, or to start online calibration later for a platform/device which was not available when online calibration was started globally for ControlDesk. For details, refer to Start Online Calibration (for Single Platform/Device) on page 83 and Stop Online Calibration (for Single Platform/Device) on page 87.

Related topics

Basics

References

Basics of Calibrating Parameters

Introduction

ControlDesk provides instruments for changing parameters. Lower and upper limits for the parameter values are defined in the variable description of the calibration device.

Where to go from here

Information in this section

Overview of the variable types that can be calibrated with ControlDesk.

If the ECU supports writable measurement variables, you can calibrate them with ControlDesk. Writable measurement variables are marked by a specific attribute in the ECU's A2L file.

Variable Types That can be Calibrated

Overview

The following variable types can be calibrated:

Common axis   A parameter ⍰ that consists of a 1-dimensional array containing axis points. A common axis can be referenced by one or more curves ⍰ and/or map ⍰s. Calibrating the data points of a common axis affects all the curves and/or maps referencing the axis.

Common axes are represented by the �🔢 symbol.

Curve   A parameter ⍰ that consists of

▪ A 1-dimensional array containing the axis points for the x-axis. This array can also be specified by a reference to a common axis ⍰.

- Another 1-dimensional array containing data points. The curve assigns one data point to each axis point.

Curves are represented by the ⊞ symbol.

Map   A parameter ⍰ that consists of

- A 1-dimensional array containing the axis points for the x-axis. This array can also be specified by a reference to a common axis ⍰ .
- A 1-dimensional array containing the axis points for the y-axis. This array can also be specified by a reference to a common axis ⍰ .
- A 2–dimensional array containing data points. The map assigns one data point of the array to each pair of x-axis and y-axis points.

Maps are represented by the ⊞ symbol.

String   A text variable in ASCII format.

Strings are represented by the ⊞ symbol.

Parameter (variable type)   A scalar parameter ⍰ , as well as the individual elements of a value block ⍰ .

Scalar parameters are represented by the **P** symbol.

Value block   A parameter ⍰ that consists of a 1- or 2-dimensional array of scalar parameters ⍰ .

In variable lists, ControlDesk displays entries for the value block itself and for each array element.

Value blocks are represented by the ⊞ symbol.

Related topics

Basics

> Instruments and the Variable Types they can Visualize (ControlDesk Instrument Handling 📖)

Calibrating Writable Measurement Variables

Introduction

Usually, measurement variables are read-only. If the ECU supports writable measurement variables, you can calibrate them with ControlDesk. Writable measurement variables are marked by a specific attribute in the ECU's A2L file.

Making a measurement variable writable

Use cases   Below are use cases for making measurement variables writable:

- Setting adaptive parameters
- Setting function outputs to specific values (required, for example, for ECU testing and ECU starting-up)
- Setting any RAM variables

READ_WRITE attribute in the A2L file   In the ECU's A2L file, a writable measurement variable can be identified by the `READ_WRITE` attribute. The A2L file definition of a writable measurement variable should look like this:

```
/begin MEASUREMENT
  Writable_Measurement     /* variable name */
  .
  .
  .
  READ_WRITE          /* measurement variable is writable */
  .
  .
/end MEASUREMENT
```

Example:

Below is an example of a writable measurement variable in an A2L file.

```
/begin MEASUREMENT
   ABSactive_ub        /* variable name */
   UBYTE               /* type */
   EQ1_LSB1            /* ref. to compu method */
   1                   /* resolution */
   0.390625            /* accuracy: resolution * 2^-8 * 100 */
   0                   /* lower limit */
   1                   /* upper limit */
   MAX_REFRESH 4 5     /* maximum refresh rate = 50 msec */
   ECU_ADDRESS 0x84A6  /* address: ABSactive_ub */
   READ_WRITE          /* measurement variable is writable */
   /begin IF_DATA ASAP1B_CCP
      KP_BLOB 0x0 0x84A6 1
   /end IF_DATA
/end MEASUREMENT
```

Identifying writable measurement variables

You can check whether a measurement variable is writable in two ways:

- Select the measurement variable in the Variables controlbar. In the Properties controlbar, check whether the variable's Accessibility property is "Read-write". Refer to Properties of Variables (Variables Controlbar) (ControlDesk Variable Management 📖).
- Visualize the measurement variable in a calibration instrument. Check whether the selected variable is writable.

Adding writable measurement variables to data sets

ControlDesk can handle writable measurement variables as parameters and add them to data sets. For detailed information, refer to Adding Writable Measurement Variables to Data Sets on page 50.

Calibrating writable measurement variables

You can calibrate writable measurement variables in two ways:

- Via calibration instruments
- Via ASAM MCD-3 MC

  To calibrate a writable measurement variable via ASAM MCD-3 MC, you must configure the corresponding Characteristic object. For instructions, refer to How to Calibrate a Parameter (ControlDesk MCD-3 Automation 📖).

> **Note**
>
> To calibrate writable measurement variables, the working data set of the platform/device must be active.

Related topics

Basics

HowTos

How to Calibrate a Parameter (ControlDesk MCD-3 Automation 📖)

Calibrating Fixed Parameters of FMUs

Introduction

Simulation applications built for the SCALEXIO, MicroAutoBox III and VEOS platform can contain Functional Mock-up Units (FMUs). During the build process of the application, TRC variable description file entries are generated for the variables of an FMU.

Fixed parameters of an FMU are generated as tunable parameters into the variable description file.

According to the Functional Mock-up Interface (FMI) standard, changing the value of a fixed parameter via the corresponding TRC variable does not immediately affect the simulation results. The effect occurs when the simulation state changes from **STOP** to **RUN**.

The simulation state is **STOP** in the following cases:

- Directly after the offline simulation application is downloaded to VEOS
- When you stop the simulation

The simulation state changes to **RUN** when you start the simulation.

Related topics

Basics

Importing Functional Mock‑Up Units (FMUs) (VEOS Manual 📖)
Working with Functional Mock-up Units (ConfigurationDesk Real-Time Implementation Guide 📖)

Handling Parameter Range Limits

Introduction

Each parameter has defined limits in which you can calibrate its values. These limits are displayed in the Properties controlbar when the variable is selected in the Variables controlbar.

Weak and hard parameter range limits

Weak limits   ControlDesk offers weak limits for each parameter. If the associated variable description contains limits for the parameter, the defined parameter range is used as initial setting for the weak limits. Otherwise, ControlDesk creates weak limits according to the data type.

Weak limits are writable. You can change them in the Properties controlbar.

Hard limits   Depending on the type of the variable description, additional parameter range limits can be defined. In this case, there is a narrow and a wide parameter range limit. An A2L variable description file, for example, can optionally contain *extended limits* for a parameter. ControlDesk displays the narrow parameter range limits as weak limits and the wide parameter range limits as hard limits.

Hard limits are read-only. They cannot be changed in the Properties controlbar.

Exceeding range limits

If you calibrate a parameter value and the intended value exceeds a weak or a hard limit, ControlDesk opens the Exceeded Limits dialog that lets you specify how to handle the value.

The following illustration shows an example:
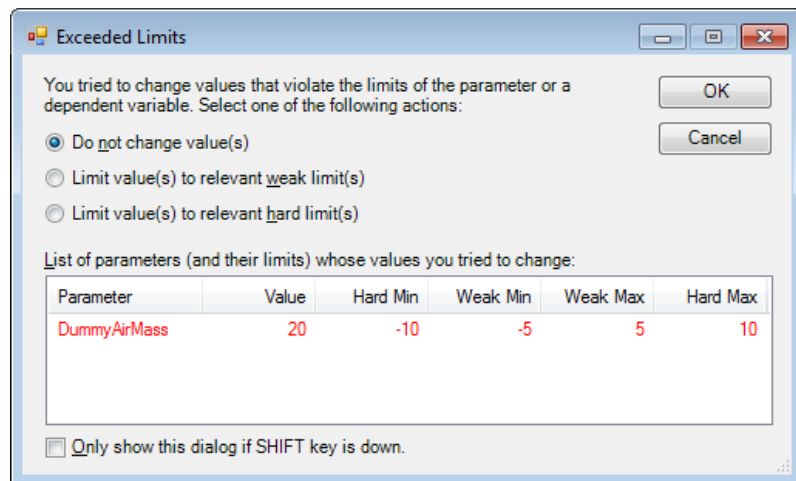
For more information on the dialog, refer to Adjust Value(s) (ControlDesk Instrument Handling 📖).

The following table describes the relations between definitions in the variable description, weak and hard limit settings in the Properties controlbar, and possible actions in the Exceeded Limits dialog when the range limits are exceeded.

Variable Description	Weak and Hard Limits	Exceeding Range Limits
Contains no parameter range limits	▪ Initial setting: Weak limits according to the data type of the parameter. No hard limits. ▪ In the **Properties** controlbar, you can change the weak limits within the bounds of the initial setting.	▪ Values are displayed in red if the intended value change exceeds a weak limit. ▪ You cannot exceed the limit values. The following illustration shows an example: | Parameter	Value	Hard Min	Weak Min	Weak Max	Hard Max
Soll_Lambda	5		0	1.9921875		
Contains parameter range limits	▪ Initial setting: Weak limits according to the limits. No hard limits. ▪ In the **Properties** controlbar, you can change the weak limits within the bounds of the initial setting.	▪ Values are displayed in red if the intended value change exceeds a weak limit. ▪ You cannot exceed the limit values. The following illustration shows an example: | Parameter	Value	Hard Min	Weak Min	Weak Max	Hard Max
Soll_Lambda	1.9		1	1.5		
Contains additional (e.g., extended) parameter range limits	▪ Initial setting: Weak limits for the narrow range and hard limits for the wide range. ▪ In the **Properties** controlbar, you can change the weak limits within the bounds of the hard limits. ▪ You cannot change the hard limits.	▪ Values are displayed in blue if the intended value change exceeds a weak limit. ▪ Values are displayed in red if the intended value change exceeds a hard limit. ▪ You can exceed the weak limits but not the hard limits. The following illustration shows an example: | Parameter	Value	Hard Min	Weak Min	Weak Max	Hard Max
DummyAirMass	7	-10	-5	5	10
DAirMass	20	-15	-2	2	15	

Options when exceeding limits

The following illustration shows in which cases the Exceeded Limits dialog opens as well as the result of the dialog options.
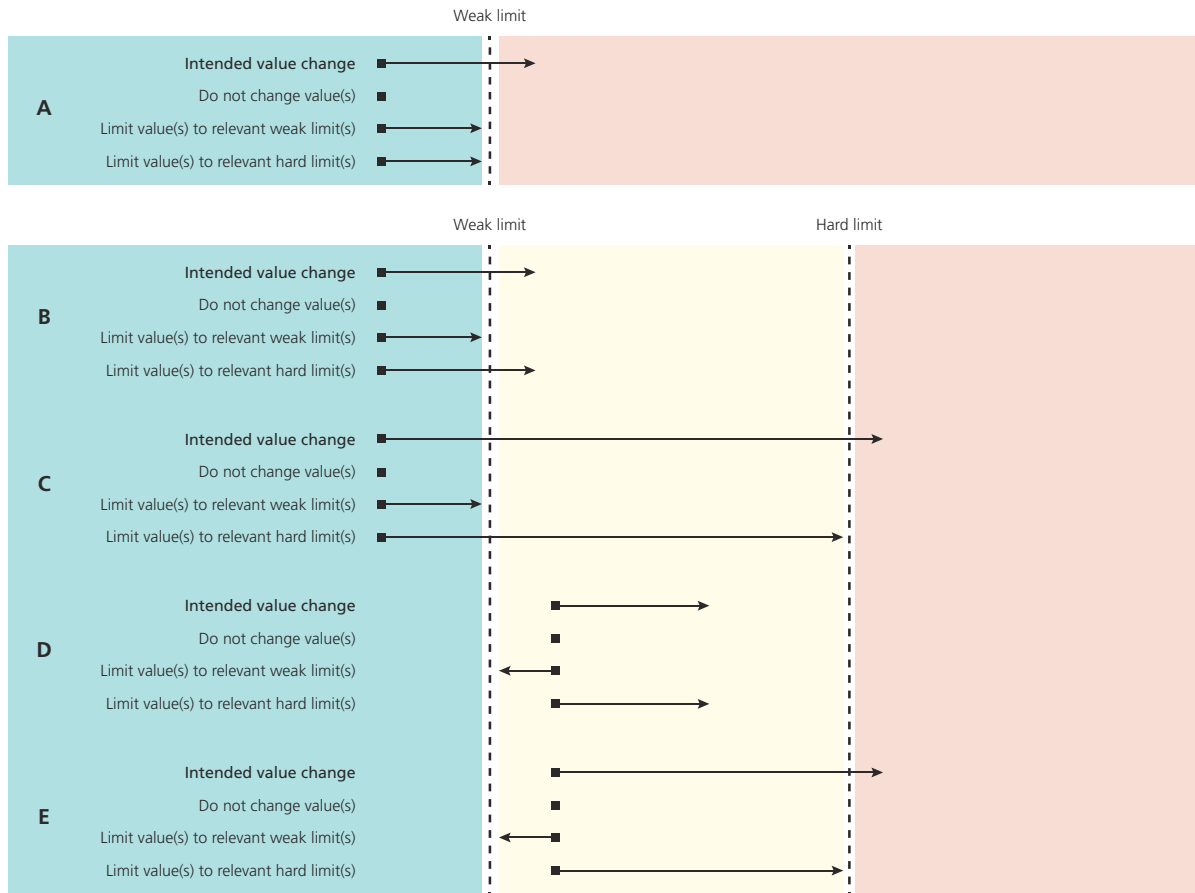
Related topics

References

Adjust Value(s) (ControlDesk Instrument Handling 📖)
Properties of Variables (Variables Controlbar) (ControlDesk Variable Management 📖)

Storing Calibration Data to the Flash Memory of dSPACE Real-Time Hardware

Store Calibration Parameter to Flash solution

You can store calibrated parameter values to the flash memory of the MicroAutoBox II. Using the *Store Calibration Parameter to Flash* solution, you can let a flash application store calibrated parameter values during the regular

shutdown process or when you switch the SimState from RUN to STOP in ControlDesk.

When the application restarts from the flash memory of the MicroAutoBox II, the application starts with the parameter values recently calibrated. Without the solution, the application would start with the original parameter values as defined in the Simulink® model.

For detailed information on the solution, install it from the dSPACE Solutions DVD and read the user guide. For information on installing it, refer to the `ReadMe.txt` file.

How to Activate the Working or Reference Page

Objective

For devices with two memory pages, you must activate the working page to calibrate parameters, since the reference page usually is not writable. You can activate the reference page to switch to the reference data set as a reliable data set in problem situations.

Data sets

For basic information on data sets, refer to Variable Types That can be Calibrated on page 28.

Identifying the active memory page

For devices with two memory pages, the reference data set has a [Reference Data Set] declaration after its name, the working data set a [Working Data Set] declaration. The data set on the active memory page (the active data set) is written in bold letters.
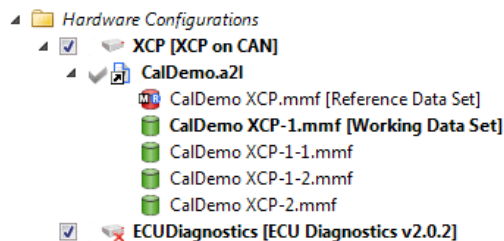
> **Note**
>
> If you activate another memory page after you start proposed calibration (see How to Perform Proposed Calibration on page 41), ControlDesk automatically stops proposed calibration. All parameter modifications you made since you started proposed calibration are lost.

Restrictions
For a device with one memory page, the working page is always active. You cannot activate the reference page.

Possible methods
There are two ways to activate the working or reference page.

- You can activate the working or the reference data set on all supporting devices. Refer to Method 1 on page 36.
- You can activate the working or the reference data set on a specific device. Refer to Method 2 on page 36.

Method 1
To activate the working or reference page on all supporting devices

1 On the Home ribbon, click Status Control - , or press **F7** to activate the reference page.
- Or -

On the Home ribbon, click Status Control - , or press **F8** to activate the working page.

Method 2
To activate the working or reference page on a specific device

1 On the Home ribbon, click Status Control - Reference Page - Activate Reference Page On <Device_name> to activate the reference page on the selected device.
- Or -

On the Home ribbon, click Status Control - Working Page - Activate Working Page On <Device_name> to activate the working page on the selected device.

> **Tip**
>
> As an alternative, you can activate a working or reference data set on a device via its context menu in the Project controlbar. Choose the data set on the inactive memory page and select Activate.

Result
The data set on the activated memory page now is visible to the ECU.

If you have selected the reference page, you cannot calibrate parameter values. Edit fields in instruments are grayed out.

Next steps
You can now calibrate parameter values and watch the effects in the measurement instruments.

- If you want to calibrate scalar parameters, refer to:
  - How to Change Values with the Variable Array (Alphanumeric Input Value Cell Type) (ControlDesk Instrument Handling ▢)

- How to Change Values with the Slider (ControlDesk Instrument Handling 📖)
- How to Change Values with the Knob (ControlDesk Instrument Handling 📖)

- If you want to calibrate scalar parameters in bit format, refer to How to Change Values with the Variable Array (Bitfield Editor Value Cell Type) (ControlDesk Instrument Handling 📖).
- If you want to calibrate nonscalar parameters, refer to How to Change Function Values in the Table Editor (ControlDesk Instrument Handling 📖).

Related topics

Basics

References

How to Display Reference Values When Calibrating

Objective

If the working page is active, you can display the reference value of a parameter to compare it with the current value.

Preconditions

The working page must be active.

Method

To display reference values when calibrating

1 Right-click an instrument row (Table Editor: table cell) and select Display value(s) – Show Reference (Table Editor: Show values), or press `Ctrl+F`.

2 Select a display option that contains the reference value.

The specific option you can select depends on the instrument. For example, the Table Editor lets you choose to display current values and reference values simultaneously, but other instruments switch between them.

Result

The variable's reference value is shown in the instrument. Instruments with rows display an **R** next to the parameter name.

> **Tip**
>
> If you want to display the reference value and the current value simultaneously in a Variable Array, create two instrument-parameter connections by duplicating the parameter within the instrument. Then use one instrument row to display the current value and the other to display the reference value.
>
>

Related topics

References

Display Values (ControlDesk Instrument Handling 🕮)
Show Current/Reference (ControlDesk Instrument Handling 🕮)

How to Write the Same Value to Several Parameters

Objective

You can connect a *main variable* and *additional write variables* to a calibration instrument (except for the Table Editor). When you write a value to the main variable, this value is also written to all the related additional write variables.

Basics

Main variable A scalar variable that is visualized in an instrument that can be used to change parameter values. In addition to the main variable, additional write variable 🗗 s can also be connected to (but not visualized in) the same instrument. When you change the value of the main variable in an instrument, the changed value is also applied to all the additional write variables connected to that instrument.

Additional write variable A scalar parameter or writable measurement variable that can be connected to an instrument in addition to the main variable 🗗 . When the value of the main variable changes, the changed value is also applied to all the additional write variables connected to the instrument.

Restrictions

It is not guaranteed that all the value changes are processed in the same sampling step. If this is required, use the Signal Editor.

Preconditions

The working page must be active. Refer to How to Activate the Working or Reference Page on page 35.

Method

To write the same value to several parameters

1   From the Instrument Selector, drag an instrument for calibration to the layout, for example, a slider.

2   From the Variables controlbar, drag a parameter to the instrument via the *right mouse button*.

    ControlDesk opens the context menu.

3   From the context menu, select Connect as Main Variable.

    The selected parameter is connected to the instrument as the main variable.

4   From the Variables controlbar, drag another parameter to the instrument via the *right mouse button*.

    ControlDesk opens the context menu.

5   From the context menu, select Connect as Additional Write Variable(s).

    The selected parameter is connected to the instrument as an additional write variable.

6   Repeat step 4 and step 5 to add more additional write variables to the instrument if required.

    In the Properties controlbar, ControlDesk displays the number of connected additional write variables.

    For example, the Properties controlbar of a slider connected to a parameter as the main variable and two parameters as additional write variables looks like this:
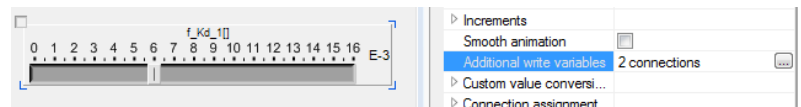
    If you click the [...] button, the Additional Write Variables dialog opens and displays the connected additional write variables and their properties.
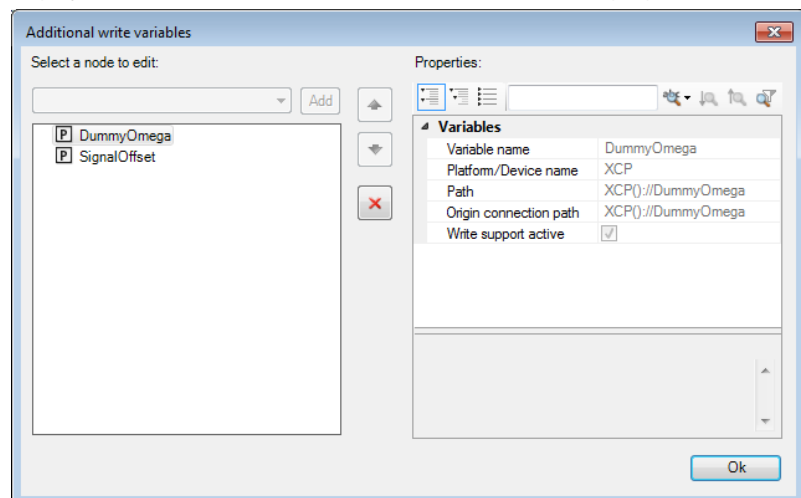
7 Change the value of the parameter in the instrument.

The specified value is set to the main variable and the additional write variables.

Result

You set a new value to the main variable and the additional write variables. The main variable is visible in the instrument, the additional write variables are invisible, but can be displayed via the Properties controlbar.

Related topics

References

Connect as Additional Write Variable(s) (ControlDesk Instrument Handling 📖)
Connect as Main Variable / Connect Variable(s) (ControlDesk Instrument Handling 📖)

Performing Proposed Calibration

Where to go from here

Information in this section

Basics of Proposed Calibration

Introduction

By default, ControlDesk downloads each modified parameter directly after you modified it. To prepare several parameter values and download them together, you can run a proposed calibration.

Proposed calibration   A calibration mode in which the parameter value changes that the user makes do not become effective on the hardware until they are applied. This allows several parameter changes to be written to the hardware

together. Being in proposed calibration mode is like being in the offline calibration mode temporarily.
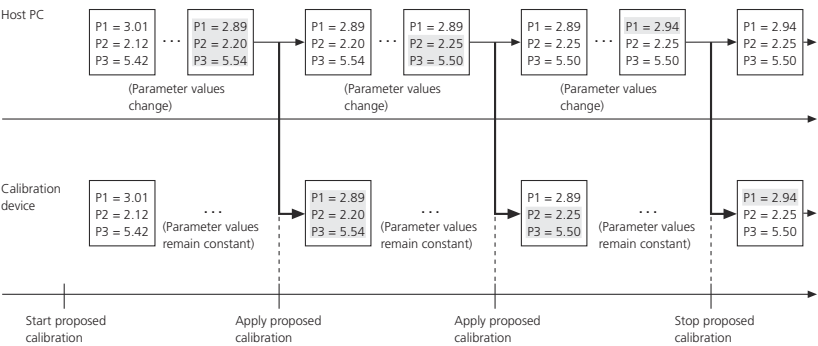
A star is displayed next to the name of the parameter visualized in a calibration instrument if proposed calibration is active.

Symbol	Description
★	Proposed calibration is active, value has been changed.
★	Proposed calibration is active, value has not been changed.

> **Note**
>
> If you activate another memory page (see How to Activate the Working or Reference Page on page 35) after you start proposed calibration, ControlDesk automatically stops proposed calibration. All parameter modifications you made since you started proposed calibration will get lost.

Related topics

HowTos

References

How to Perform Proposed Calibration

Objective

In certain situations you may want to delay the downloading of parameter changes. You can use ControlDesk's *proposed calibration* feature to change several parameters and download them together.

Restrictions
It is not guaranteed that all the value changes are processed in the same sampling step. If this is required, use the Signal Editor.

Preconditions
The working page must be active. Refer to How to Activate the Working or Reference Page on page 35.

Method
To perform proposed calibration

1 On the Home ribbon, click Calibration – Proposed Calibration – Start Proposed Calibration.

2 Change the desired parameter values.

   If you want to undo the modifications since the last parameter download (see step 3), click Cancel Proposed Calibration.

3 On the Home ribbon, click Calibration – Proposed Calibration – Apply Proposed Calibration to download the modified parameter values.

   Now ControlDesk downloads all the parameters that you modified since the last download.

4 Repeat steps 2 and 3 as often as required.

5 On the Home ribbon, click Calibration – Proposed Calibration – Stop Proposed Calibration.

   Now ControlDesk downloads modified parameters directly after you modified them.

Result
The changes of parameter values during proposed calibration mode were downloaded together.

Related topics

Basics

References

Managing Data Sets

Introduction

ControlDesk lets you work with complete sets of the parameters of the application running on dSPACE real-time hardware, or an ECU.

Where to go from here

Information in this section

Introduction to Data Sets

Introduction

Basics on data sets and how you can filter them.

Where to go from here

Information in this section

Basics of Data Sets

Introduction

A data set is a complete set of the parameters of dSPACE real-time hardware, or an ECU. When you calibrate parameter values, you actually change the values of the data set of the currently active memory page.

The following illustration shows a device and a platform with their data sets as an example:
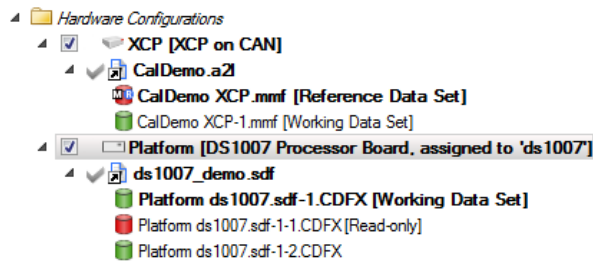
Initial data set and offline calibration

For measurement and calibration devices, ControlDesk generates an initial data set. For platforms, you can manually create an initial data set or change the default settings.

Initial data set of measurement and calibration devices   For measurement and calibration devices ⓘ, the initial data set is generated from the Variable description ⓘ (A2L) file and the ECU Image file ⓘ. The variable description file contains the addresses and data type information of calibration parameters. This ensures that ControlDesk can access the memory part of calibration devices correctly. The ECU Image file contains either code or data, or both.

> **Note**
>
> Devices without a memory segment definition do not have initial data sets. You can create an initial working data set for them manually (see Create Data Set).

Initial data set of platforms   For platforms ⓘ, the initial data set is generated from the variable description (SDF) file. This is possible only if the SDF file contains initial parameter values.

For real-time applications implemented with RTI, you can configure the RTI build process to include initial parameter values in the SDF file to be generated. Refer to the *RTI variable description file options category* on the Code Generation Dialog (Model Configuration Parameters Dialogs) (RTI and RTI-MP Implementation Reference 📖).

> **Note**
>
> If a platform has no data set, you can create an initial data set for it manually (see Create Data Set). You can also specify to generate initial data sets for platforms by default on the Data Set Management page of the ControlDesk Options dialog.

Initial data set as the basis for offline calibration   The initial data set is the basis for the filling of the working page and, if available, the reference page of ControlDesk's mirrored memory. The mirrored memory allows you to perform offline calibration. For details, refer to Basics on Offline Calibration on page 14.

Working data set and reference data set

Working data set   There is exactly one data set that you can define as the *working data set*. The working data set is written to the working page of the platform/device.

The working data set is writable and lets you calibrate parameter values.

Reference data set   For devices, you can also define one data set as the *reference data set*. The reference data set is not writable. It functions as a reliable data set with proven parameter values, i.e., it can be used as a backup in problem situations.

ControlDesk uses the reference data set for comparisons with the working data set in instruments. For example, the differences between the two data sets are displayed with tendency symbols.

Relation to memory pages   The data sets of dSPACE real-time hardware or an ECU are related to the available memory pages. Refer to Data Sets and their Relation to Memory Pages on page 47.

Online calibration without working data set

In most cases, you will use a working data set to calibrate parameter values on a platform or device. If no working data set is available and you have no need for offline calibration, you can write parameter values directly to a platform or device and read them from there. Refer to **Write to Hardware** on page 117.

Types of data sets

The following table shows the different types of data sets that are displayed in the **Project** controlbar.

Icon	Type	Description
Red (plus M)	Master data set[1), 2)]	The initial data set which contains the parameter values from the ECU Image file. The Master data set is read-only.
Red (plus R)	Reference data set[1)]	The data set which contains the parameter values for comparisons in instruments (e.g., tendency symbols). If the device has a reference page, the reference data set is connected to it. Each time you define a reference data set, this data set is set to read-only.

Icon	Type	Description
Red	Read-only data set[2]	Each time you define a reference data set, this data set is set to read-only. For devices with one memory page, you can make a data set read-only via its context menu.
Green	Writable data set[1]	Each time you copy a data set, the initial status of the new data set is writable.
Gray	Closed data set	A closed data set is not loaded to the active experiment.
Gray/green	Sub data set	A sub data set does not contain all the necessary parameters to use it as a reference or working data set by itself. However, it can be merged with the current working data set.
Gray/red	Read-only sub data set[2]	You can make a sub data set read-only via its context menu. Each time you copy a sub data set, the initial status of the new sub data set is writable.
	Data set for download on online calibration start	If more than one data set/sub data set is specified for being downloaded on calibration start, the data sets/sub data sets are downloaded one after the other according to their order in the Project controlbar.

[1] If a data set is connected to a memory page, it has a declaration ([Working Data Set] or [Reference Data Set]) after its name. The data set on the active memory page is written in bold letters.

[2] If a data set is read-only, it has a [Read-only] declaration after its name.

Data set grouping

In the Project controlbar, you can enable data set grouping. ControlDesk displays a Data Sets folder, and you can add groups as well as subgroups to it and rearrange the data sets via drag & drop.
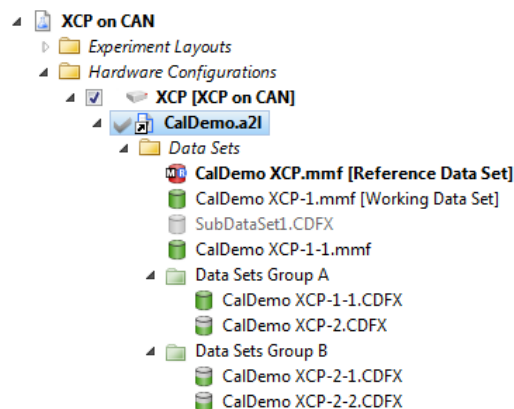
You can activate data set grouping via the context menu of a variable description. Refer to Data Set Grouping on page 101.

Related topics

HowTos

References

Data Sets and their Relation to Memory Pages

Introduction

The data sets of dSPACE real-time hardware or an ECU are related to the available memory pages.

Data sets of ECUs with two memory pages

Many ECUs (or the connected ECU interface) provide two memory pages.

Working data set   The data set containing the parameter values on the ECU's working page is the read/write *working data set*.

Reference data set   The data set containing the parameter values on the ECU's reference page is the read-only *reference data set*.

ControlDesk lets you activate the working page or the reference page, so you can easily switch between the working data set and the reference data set.

Active memory page   Either the working or reference data set – the data set on the currently active memory page – is visible to the microcontroller of the ECU at a time.
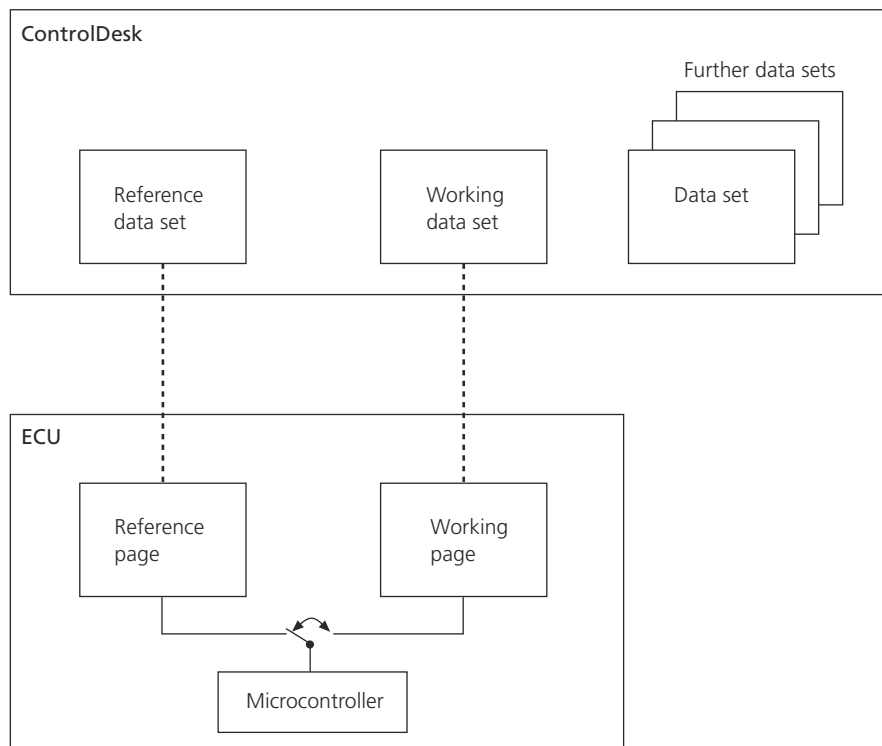
Data sets of ECUs with one memory page

Some ECUs provide only one memory page.

Working data set   The data set containing the parameter values on the ECU's memory page is the read/write *working data set*. The working data set is always active. However, if the memory page is located in a read-only memory area, parameter values can only be calibrated offline.

Reference data set   The read-only *reference data set* in ControlDesk is not connected to a memory page on the ECU. It is used only for comparisons, for example, displaying tendency symbols in instruments. You cannot activate a reference page.

```
┌─────────────────────────────────────────────────────────────────────┐
│ ControlDesk                                                           │
│                                                    Further data sets   │
│                                                  ┌──────────────────┐  │
│                                                ┌─┤                  │  │
│   ┌────────────────┐   ┌────────────────┐    ┌─┤ │                  │  │
│   │                │   │                │    │  │ │                  │  │
│   │   Reference    │   │    Working     │    │ Data set            │  │
│   │   data set     │   │    data set    │    │  │ │                  │  │
│   │                │   │                │    │  │─┘                  │  │
│   └────────────────┘   └────────┬───────┘    └──────────────────┘  │
│                                 ┊                                     │
└─────────────────────────────────┊─────────────────────────────────────┘
                                  ┊
┌─────────────────────────────────┊─────────────────────────────────────┐
│ ECU                             ┊                                       │
│                          ┌──────┴─────────┐                            │
│                          │                │                            │
│                          │    Working     │                            │
│                          │    page        │                            │
│                          │                │                            │
│                          └──────┬─────────┘                            │
│                   ┌─────────────┘                                      │
│           ┌───────┴────────┐                                           │
│           │ Microcontroller │                                          │
│           └─────────────────┘                                          │
└────────────────────────────────────────────────────────────────────────┘
```
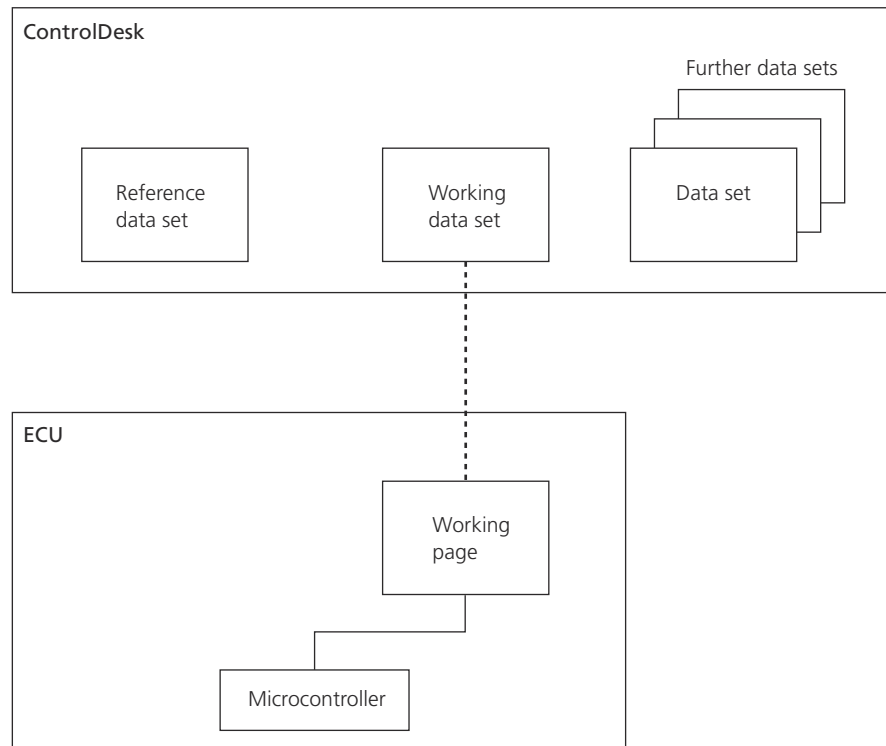
Data sets of platforms
The memory of platforms is not divided into memory segments. Nevertheless, you can also create data sets for dSPACE real-time hardware and work with them in the same way as with data sets of ECUs with one memory page with the exception that there is no reference data set for platforms.

Related topics

Adding Writable Measurement Variables to Data Sets

Introduction

You can create data sets that do not only contain parameters but also writable measurement variables.

Writable measurements

Usually, measurement variables are read-only and cannot be part of a data set. Some ECUs support writable measurements. Writable measurements can be calibrated in ControlDesk.

For more information, such as on use cases and how to identify writable measurements, refer to Calibrating Writable Measurement Variables on page 29.

Writable measurements in data sets

On the Data Set Management page of ControlDesk's Options dialog, you can activate handling writable measurements as parameters. ControlDesk then adds writable measurements to the reference data set and working data set, and all new data sets you create.

> **Note**
>
> If you activate or deactivate the handling of writable measurements as parameters, follow the rules below. Otherwise data sets might become incomplete or inconsistent.
> - Stop online calibration before you change the handling of writable measurements.
> - With exception to data sets with the MMF file name extension, remove all the data sets that are already assigned to the variable description.

Related topics

Basics

Creating Data Sets

Where to go from here

Information in this section

You can create a new data set using the copy functionality of the Project controlbar.

You can create a data set that contains all the parameters of a variable description that are connected to instruments on open layouts.

You can store a template for the default description of data sets.

How to Create a New Data Set by Copying

Objective   You can create a new data set by using the copy functionality of the **Project** ⬈ controlbar.

Method   **To create a new data set by copying**

1 Open the **Project** controlbar and right-click the data set you want to copy.

2 Select **Copy As New** from the context menu.

A copy of the data set is created and displayed in the tree view.

3 Right-click the new data set and select **Rename** to enter a new name.

4 The new data set is listed with its new name in the tree view.

Result   You created a new data set and entered a new name for it based on an existing data set.

> **Tip**
>
> You can change the order of the data sets via drag & drop.

Next steps   You can now use the new data set for calibration or reference purposes. Refer to How to Define a New Working or Reference Data Set on page 56.

Related topics   Basics

HowTos

References

How to Create a Data Set from Open Layouts

Objective   You can create a data set that contains all the parameters of a variable description that are connected to instruments on open layouts.

Method   **To create a data set from open layouts**

1   In the Project ⓘ controlbar, right-click the variable description that contains the connected parameters.

2   From the context menu, select **Create Data Set from Open Layouts**.

Because the created data set does not contain a complete set of parameters, it is listed as a sub data set in the tree view.
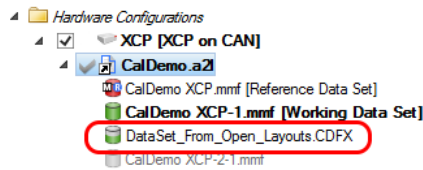
Result   You created a sub data set that contains all the parameters of a variable description that are connected to instruments on currently opened layouts.

Next steps   A sub data set does not contain the complete set of parameters of a platform/device. It can therefore not be used as a working data set or reference data set by itself. However, you can use its parameter values in the working data set:

- You can write the parameter values of a sub data set to the working data set. Refer to Write to Working Data Set on page 118.

- You can make the sub data set the new working data set. A new complete data set is created and ControlDesk fills the missing parts with the content of the current working data set. Refer to How to Define a New Working or Reference Data Set on page 56.

Related topics

Basics

HowTos

References

How to Create a Description Template

Objective

You can store a template for the default description of data sets.

Data set description

You can enter a description text for each data set in the **Project** ⓘ controlbar.
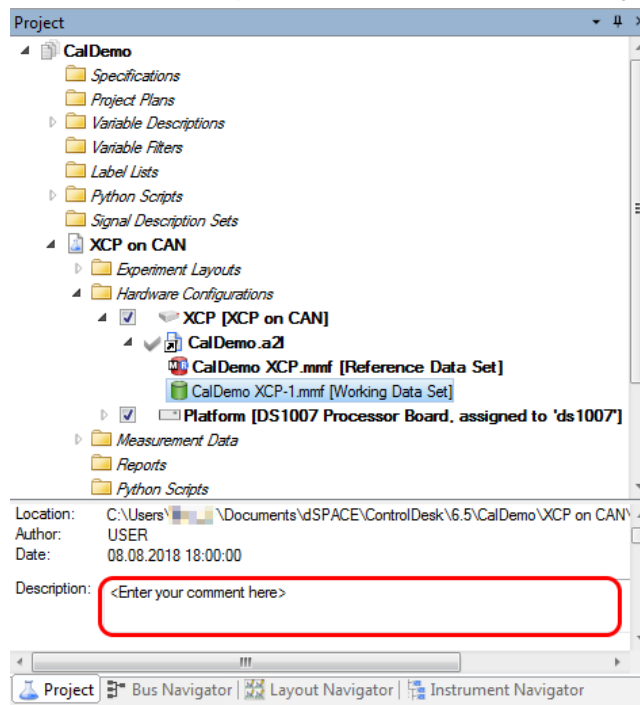
By using a description template, you can change the default text of the description. The following illustration shows an example:

Method

To create a description template

1 Open a text editor and enter the text you want to use as the default description.

2 Save the text as `DataSet_Default_Description.txt` in the common Program Data folder ⧉ .

Result

You have created a description template for data sets. ControlDesk will use the new default description for data sets when you add a new variable description.

Related topics

Basics

Handling Working and Reference Data Sets

Introduction

You can defining working and reference data sets and activate them.

Where to go from here

Information in this section

How to Define a New Working or Reference Data Set

Objective

For platforms and devices, you can define a data set as the working data set. For devices, you can also define a reference data set.

Complete data sets

The working or reference data set must be a complete data set.

> **Tip**
>
> A sub data set is an incomplete data set. However, you can define it as the new working data set. In this case, ControlDesk fills the missing parts with the content of the current working data set.

Restrictions

You cannot define a reference data set for platforms.

Method

To define a new working or reference data set

1 In the Project ⧉ controlbar, right-click the data set that you want to define as the new working or reference data set.

2 To define a new working data set, select Make Working or Make Working/Download (available if online calibration is started).

3 To define a new reference data set, select Make Reference.

Result

You have defined a new working or reference data set.

The selected data set is written to the appropriate memory page, if the page is available on the platform/device. A declaration after the data set's name ([Working Data Set] or [Reference Data Set]) shows whether it resides on a memory page.

Next steps

If the device provides a working and a reference page, you can switch between them. Refer to

Related topics

Basics

References

How to Activate the Working or Reference Page

Objective

For devices with two memory pages, you must activate the working or reference page.

Restrictions

For platforms or devices with only one memory page, the working page is always activated. You cannot activate a reference page. The corresponding buttons are disabled.

Possible methods

- You can activate the working or reference page via ribbon or shortcut key. Refer to
- You can activate the working or reference page via the **Project** ⬀ controlbar. Refer to

Method 1

To activate the working or reference page via ribbon or shortcut key

1   Open the Project controlbar.

2   On the Home ribbon, click Status Control - Reference Page - Activate Reference Page On <Device_name> to activate the reference page on the selected device.

– Or –

Click 🔲 or press **F7** to activate the reference page on all supporting devices.

3   On the Home ribbon, click Status Control - Working Page - Activate Working Page On <Device_name> to activate the working page on the selected device.

– Or –

Click 🗗 or press **F8** to activate the working page on all supporting devices.

Method 2	**To activate the working or reference page via the Project Controlbar**

1 In the Project controlbar, right-click the required working or reference data set.

2 Select Activate from the context menu.

The corresponding page is activated and the layout view refreshed.

Result	You activated the working or reference page.

> **Tip**
>
> - ControlDesk can automatically set a bookmark during a measurement or recording when a data set is activated or when a data set is downloaded. These "data set activation" and "data set download" bookmarks can then be visualized in a Time Plotter. Refer to Edit Bookmark Settings (ControlDesk Measurement and Recording 📖).
> - You can use "data set activation"/"data set download" bookmarks as predefined trigger conditions, for example, to start a triggered recording with a specified pretriggering time when the bookmark is set. Refer to How to Define a Trigger Rule for Triggered Recording (ControlDesk Measurement and Recording 📖).

Related topics

Basics

References

How to Select Additional Data Sets for Downloading on Online Calibration Start

Objective	Not only the values of the working data set can be downloaded to the memory page of a platform/device when online calibration is started. You can also specify additional data sets or sub data sets for downloading.

Description

You can select data sets or sub data sets for downloading to a platform when online calibration is started.

If more than one data set/sub data set is specified for downloading on calibration start, the data sets/sub data sets are downloaded one after the other according to their order in the Project ⍰ controlbar.

Each selected data set is marked by a down arrow.
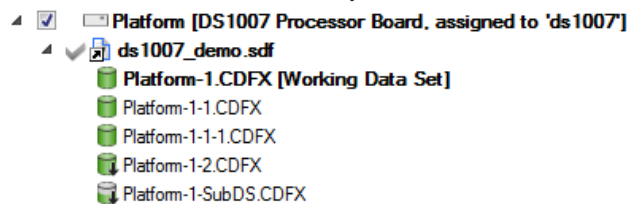
When online calibration is started, the data set list is processed from top to bottom according to the "last one wins" rule. Complete data sets overwrite all the values of the previous data set in the list, and sub data sets overwrite only those parameters they contain themselves.

> **Note**
>
> Selecting a data set for downloading on calibration start overrules any upload setting. Parameter values of a selected data set or sub data set are always downloaded, even if you select Upload in the Calibration Memory dialog or Upload at Start online calibration behavior in the platform/device configuration.

Method

To select additional data sets for downloading on online calibration start

1   In the Project controlbar, right-click the data set or sub data set to be downloaded when online calibration is started and select Download on Online Calibration Start.
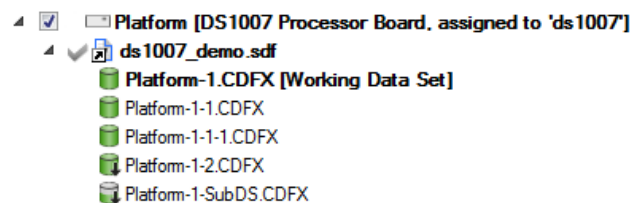
In the example, the following data sets are processed, when online calibration is started:

1. The parameter values of the working data set (`Platform-1`) are replaced by the parameter values of the data set `Platform-1-2`.

2. The parameter values of the sub data set `Platform_1_SubDS` replace their predecessors from `Platform-1-2`.

In this case the actually downloaded parameter values are a combination of the two marked data sets.

2 To change the parameter values that are downloaded, change the data set order by dragging a marked data set to another position.

Result   You have selected data sets for downloading on calibration start.

Related topics   Basics

References

Exporting and Converting Data Sets

Introduction   You can export single data sets or groups of data sets and create ECU images or Application images.

Where to go from here   Information in this section

How to Export a Data Set

Objective   You can export a data set into different file formats. This allows you to use a data set generated with ControlDesk in other programs.

Export file formats

You can select one of the following formats:

- ASAM Calibration Data Format (CDFX)
- ASAM Calibration Data Format (XML)

The CDFX format is the standard format for a ControlDesk data set.

You can specify the ASAM Calibration Data Format version (2.0 or 2.1) on the **Data Set Management page** of the **ControlDesk Options** dialog.

> **Note**
>
> When you export a data set to a format other than CDFX, parameters without initial data are removed from the exported data set.

> **Note**
>
> The exported data set file is not displayed in the tree view of the Project⧉ controlbar.

Method

To export a data set

1 In the **Project** controlbar, right-click the data set to be exported to open its context menu.

2 From the context menu, select **Export**.

The **Save As dialog** opens.
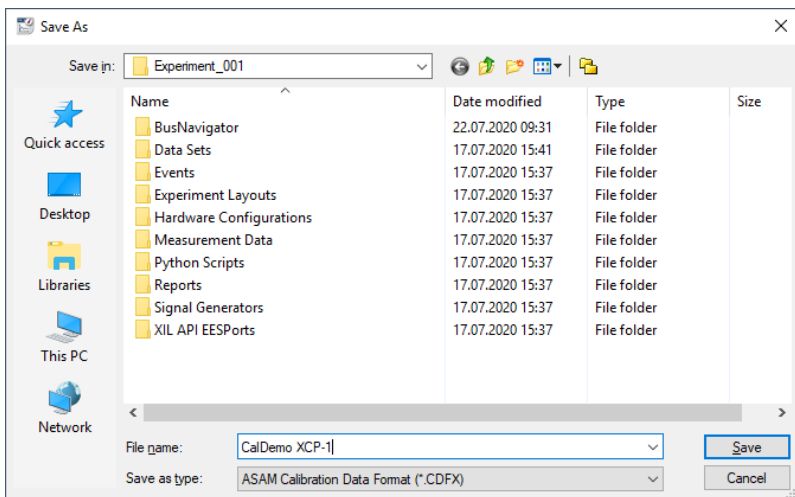
3 Select the desired file format, and enter a unique name. If you select a file from the file list, it is overwritten by the new data.

4 Click **Save** to export the data set.

Result

ControlDesk exports the data set to the specified file.

How to Create an ECU Image File

Objective

You can use a data set to create a new ECU Image file.

ECU Image file

Calibration devices usually come with an ECU Image file. It contains the code of the ECU application, or the data of the parameters in it, or both. It may be stored as a HEX, MOT, or S19 file.

Content of the created ECU Image file

The content of the created ECU Image file depends on the content of the initially loaded ECU Image file and the way the values of the initial data set are generated.

Content of the Initially Loaded ECU Image File	Content of the Created ECU Image File
– (No ECU Image file loaded) [1]	Data only
Data only [2]	Data only
Code and data [2]	Code and data

[1] The values of the initial data set(s) are uploaded from the platform/device hardware the first time you start online calibration.
[2] The values of the initial data set(s) are read from the initially loaded ECU Image file.

The created ECU Image file can be used for ECU flash programming.

Method

To create an ECU Image file

1 In the Project 🗗 controlbar, right-click the desired data set.

2 From the context menu, select **Create ECU Image** and select the desired file format.

Result	ControlDesk creates a new ECU Image file with the name of the selected data set. In the **Project** controlbar, the created ECU Image file is displayed as sub node of the selected data set.
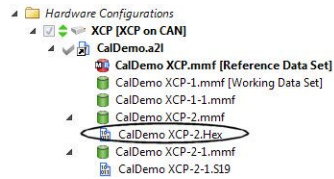
> **Tip**
>
> If you want to use the created ECU Image file with the dSPACE ECU Flash Programming Tool, double-click the ECU Image file in the **Project** controlbar.

Related topics	**Basics**

Basics of Data Sets..44
Programming the ECU Flash Memory (ControlDesk Platform Management 📖)

References

Create ECU Image... 100

How to Create an Application Image

Objective	You can use a data set to create a new application image for a dSPACE real-time platform.
Extensible real-time application	After the build process, a real-time application contains the parameter values specified in the Simulink model. To change these 'original' parameter values, you have to edit them in the model and then recompile the real-time application.

Depending on the real-time platform, the real-time application can be extensible. An extensible real-time application allows you to add one additional set of parameter values later on without having to recompile the application. This is especially useful during fleet tests since no MATLAB®/ Simulink®/ Stateflow® installation is required for rebuilding the application. |

Application image

An application image contains all the files that are created when you build a real-time application. It particularly includes the variable description (SDF) file.

For an extensible real-time application, ControlDesk lets you create a new application image from a data set (or sub data set). The new application image then contains the additional set of parameter values.

The following illustration shows the workflow to get an application image with additional parameter values.

```
          ┌─────────────────────────────┐
          │   MATLAB / Simulink model    │
          └─────────────────────────────┘
                        ⇩         ①
          ┌─────────────────────────────┐
          │    RTI / ConfigurationDesk    │
          └─────────────────────────────┘
                        ⇩
          ╭─────────────────────────────╮  ②
          │  Application image           │
          │  (• Parameter values from    │
          │     Simulink model)          │
          ╰─────────────────────────────╯
                        ⇩
          ┌─────────────────────────────┐  ③
          │        ControlDesk           │
          │                              │
          │        Experiment            │
          │            ↓                 │
          │         Data set             │
          │            ↓                 │
   ⑤     │   Create Application Image    │
          └─────────────────────────────┘
                        ⇩
          ╭─────────────────────────────╮  ④
          │  Application image           │
          │  (• Parameter values from    │
          │     Simulink model           │
          │   • Parameter values from    │
          │     ControlDesk              │
          │     data set)                │
          ╰─────────────────────────────╯
```
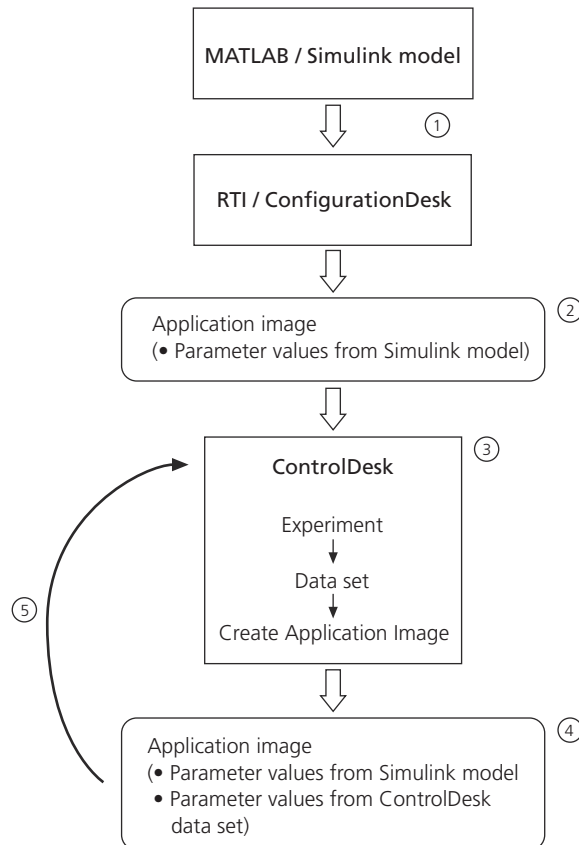
1. Starting with a Simulink model, build an initial real-time application via RTI or ConfigurationDesk. For the MicroAutoBox II, you must activate the **Enable data set storage in application** option.

2. The application image contains the parameter values from the Simulink model. In ControlDesk, add the contained SDF file to the platform.

3. In ControlDesk, select a data set to create an application image with one additional set of parameter values.

4. The created application image can be used in ControlDesk with the new parameter values.

Original and additional parameter values

When you create a new application image via ControlDesk, you select a data set to specify a second set of parameter values in the real-time application. The original set of parameter values, specified via the Simulink model, remains unchanged.

```
┌─────────────────────────────────────────────────────┐
│              Extensible real-time application         │
│                                                       │
│   ┌───────────────────────────────────────────────┐  │
│   │                     Code                       │  │
│   └───────────────────────────────────────────────┘  │
│                                                       │
│   ┌───────────────────────────────────────────────┐  │
│   │        Parameter values from Simulink model    │  │
│   └───────────────────────────────────────────────┘  │
│                                                       │
│   ┌───────────────────────────────────────────────┐  │
│   │     Parameter values from ControlDesk data set │  │
│   └───────────────────────────────────────────────┘  │
└─────────────────────────────────────────────────────┘
```
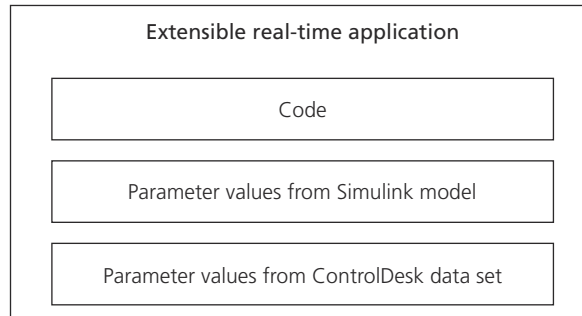
You can switch between the two sets of parameter values via the context menu of the working data set.

Restrictions

Creating an application image is supported only for the following platforms:

- DS1007 PPC Processor Board
- MicroAutoBox II/III
- MicroLabBox
- SCALEXIO

Preconditions (relevant only for MicroAutoBox II)

When the real-time application is built, the Enable data set storage in application option in the Simulink® Coder™ must be enabled. For details, refer to RTI general build options page (RTI and RTI-MP Implementation Reference 📖).

Method 1

To create an application image and load the real-time application to the hardware

1 If online calibration is running, you must stop it.

2 In the Project 🗗 controlbar, right-click the data set.

3 From the context menu, select Create and Load Application Image.

   The values of the selected data set are written to the application image and activated in the real-time application.

4 On the Home ribbon, click Status Control – Go Online to start online calibration, or press `Ctrl` + `F7`.

   ControlDesk loads the changed real-time application to the RAM of the real-time hardware. You are prompted to stop the running application and load the new one. If you work with a DS1007 platform, you must stop the old application and unload it from the RAM of the platform via the platform's context menu in the Platforms/Devices 🗗 controlbar to let ControlDesk load the changed real-time application to the hardware.

> **Tip**
>
> You can also load the changed application to the flash memory of the real-time hardware. The next time the hardware boots, it runs with the additional data set. Refer to Create and Flash Application Image on page 94.

Method 2

To create an application image without loading the real-time application to the hardware

1   In the Project controlbar, right-click the data set.

2   From the context menu, select Create Application Image.

    A standard Save As dialog opens.

3   Specify the folder and file name for the new application image.

    If you select an SDF file from the file list, it will be replaced. If you save the application image under a new file name, the associated MAP, TRC and PPC files are named after the new SDF file. The references to the PPC and TRC files contained in the SDF file are automatically adapted to the new file names.

Result

You have created a new application image, containing a new real-time application with an additional set of parameter values.

- If you used method 1, the new values are already active on the real-time hardware and ready to use in the experiment.
- If you used method 2, the new or changed application image is only stored in a folder. To use this application image in an experiment, you have to select the corresponding variable description (SDF file). Refer to How to Add a Variable Description to a Platform/Device (ControlDesk Variable Management 📖).

> **Tip**
>
> - You can also select a sub data set to create an application image. In this case the gaps are filled by parameter values from the Simulink model.
> - You can switch between the parameter values from the data set and the original ones from the Simulink model by activating and deactivating the application image. Refer to Activate Application Image or Deactivate Application Image.

Related topics

Basics

References

Reference Information

Where to go from here

Information in this section

Calibration

Where to go from here

Information in this section

Activate Reference Page/Working Page on <Device Name>

Access	This command is available only for devices and only for data sets that are connected to a memory page and currently inactive. You can access the command via:

Ribbon	Home – Status Control – Reference Page/Working Page
Context menu of	Project ⍰ controlbar – inactive reference data set or inactive working data set
Shortcut key	None
Icon	Activate Reference Page Activate Working Page

Purpose	To activate either the reference page or the working page on a specific device.

Result	Only the parameter values of the active memory page are 'visible' to the ECU which is calibrated.

Description	Of all the data sets for a device, there is exactly one that you can define as the working data set and one that you can define as the reference data set. If the device provides a working and a reference page, both data sets are written to the appropriate memory page. Only the parameter values of the active data set are visible to the ECU.

> **Note**
>
> If you activate another memory page after you start proposed calibration (see How to Perform Proposed Calibration on page 41), ControlDesk automatically stops proposed calibration. All parameter modifications you made since you started proposed calibration are lost.

Related topics

Basics

HowTos

References

Activate Reference Page/Working Page on All Supporting Devices

Access

The command is available only for devices. You can access this command via:

Ribbon	Home – Status Control
Context menu of	None
Shortcut key	**F7** (activates the reference page) **F8** (activates the working page)
Icon	![icon] Reference page ![icon] Working page

Purpose

To activate either the reference page or the working page on all supporting devices.

Result

Only the parameter values of the active page are 'visible' to the ECUs.

Description

Of all the data sets for a device, there is exactly one that you can define as the working data set and one that you can define as the reference data set. For devices that provide a working and a reference page, both data sets are written to the appropriate memory page. Only the parameter values of the active page are visible to the ECUs.

> **Note**
>
> If you activate another memory page after you start proposed calibration (see How to Perform Proposed Calibration on page 41), ControlDesk automatically stops proposed calibration. All parameter modifications you made since you started proposed calibration are lost.

Related topics	**Basics**
	HowTos
	References

Apply Proposed Calibration

Access This command is available only if proposed calibration has been started. It is available for platforms and devices. You can access the command via:

Ribbon	Home – Calibration – Proposed Calibration
Context menu of	None
Shortcut key	None
Icon	

Purpose To apply proposed calibration.

Result At first, the parameters are modified only in the cache (mirrored memory on the host PC). ControlDesk downloads the modified parameters to the platforms/devices. The proposed calibration remains active.

Related topics

HowTos

References

Measurement/Recording-Related Commands (ControlDesk Measurement and Recording 📖)

Cancel Proposed Calibration

Access

This command is available only if the proposed calibration has been started. It is available for platforms and devices. You can access the command via:

Ribbon	Home – Calibration – Proposed Calibration
Context menu of	None
Shortcut key	None
Icon	

Purpose

To cancel proposed calibration.

Description

ControlDesk does not download the modified parameters to the platforms/devices of the active experiment, but resets the parameters to the values that are currently used on the platforms/devices. The proposed calibration remains active.

Related topics

HowTos

References

Measurement/Recording-Related Commands (ControlDesk Measurement and Recording 📖)

Copy Reference Page to Working Page

Access	This command is available only for the XCP device whose device hardware is connected to the host PC. You can access this command via:

Ribbon	None
Context menu of	• **Project** ⧉ controlbar - XCP device • **Platforms/Devices** ⧉ controlbar - XCP device
Shortcut key	None
Icon	None

Purpose	To copy the reference page to the working page.

Result	The parameter values from the reference page are copied to the working page.

Description	ControlDesk uses the `COPY_CAL_PAGE` XCP command, where the reference page is the source page and the working page is the destination page.

Related topics	**Basics**

References

Copy Working Page to Reference Page

Access	This command is available only for the XCP device whose device hardware is connected to the host PC. You can access this command via:

Ribbon	None
Context menu of	• **Project** ⧉ controlbar - XCP device • **Platforms/Devices** ⧉ controlbar - XCP device
Shortcut key	None
Icon	None

Purpose	To copy the working page to the reference page.

Result	The parameter values from the working page are copied to the reference page.

Description	ControlDesk uses the `COPY_CAL_PAGE` XCP command, where the working page is the source page and the reference page is the destination page.

Related topics

Basics

References

Download On Online Calibration Start

Access   You can access this command via:

Ribbon	None
Context menu of	Project ⓘ controlbar – data set, data set group
Shortcut key	None
Icon	None

Purpose	To download the parameter values of a data set to the assigned platform/device when online calibration is started.

Result	The parameter values of the selected data set are downloaded to the platform/device on online calibration start if they are not overwritten by selected data sets later in the data set sequence.

> **Note**
>
> Selecting a data set for downloading on calibration start overrules any upload setting. Parameter values of a selected data set or sub data set are always downloaded, even if you select Upload in the Calibration Memory dialog or Upload at Start online calibration behavior in the platform/device configuration.

Description

By default only the working data set is downloaded to a platform/device when online calibration is started. You can select additional data sets for downloading. A selected data set is marked by a down arrow: .

Use the context menu of a data set group to select all the data sets within the group for download on calibration start.

Whether the values of a selected data set are actually downloaded, and which ones, depends on the order of the data sets in the data set list (last one wins).

Related topics

HowTos

Go Offline

Access

This command is available only if online calibration has been started. You can access it via:

Ribbon	Home – Status Control
Context menu of	None
Shortcut key	`Ctrl+F8`
Icon	

Purpose

To stop online calibration globally for ControlDesk.

Result

ControlDesk stops communication with all the platform/device hardware of the current experiment.

After you stop online calibration, the hardware remains in the "connected" state.

For details on the possible platform/device states, refer to Basics of Platform/Device States (ControlDesk Platform Management 📖).

> **Tip**
>
> ControlDesk lets you stop online calibration for single platforms/devices. Refer to Stop Online Calibration (for Single Platform/Device) on page 87.

Related topics

Basics

Basics of Platform/Device States (ControlDesk Platform Management 📖)

HowTos

References

Go Online

Access

This command is available only if online calibration has not been started. You can access it via:

Ribbon	Home – Status Control
Context menu of	None
Shortcut key	**Ctrl+F7**
Icon	⬍

Purpose

To start online calibration globally for ControlDesk.

Result

ControlDesk connects itself to the platform/device hardware of the active experiment, so that you are now directly accessing the hardware. If you work with an XCP or CCP device for which the **Read XCP settings from ECU** or **Read CCP settings from ECU** property is selected, the XCP/CCP service configuration information is uploaded from the ECU when the device is connected to the ECU. If there are inconsistencies between the configuration information on the ECU and the configuration information that is currently used by the device, ControlDesk opens a dialog for you to specify whether to use the current value

or to upload the corresponding value from the ECU for each of the settings concerned.

- If the experiment contains platforms, their applications are automatically loaded to the hardware if no applications are loaded yet. You can let the loaded applications start automatically when you start online calibration.

  When you start online calibration, ControlDesk automatically detects whether an application has been terminated. For details, refer to Detecting terminated applications (ControlDesk Platform Management 📖).

- If the parameter values on the host PC (mirrored memory) and on the hardware are equal, you can perform online parameter calibration.

- If the parameter values on the host PC (mirrored memory) differ from those on the hardware, ControlDesk opens the Calibration Memory Dialog, which lets you choose which parameters you want to use.

> **Tip**
>
> To prevent ControlDesk from always opening the Calibration Memory dialog, you can specify a default start online calibration behavior for each platform/device individually during platform/device configuration. ControlDesk then equalizes the parameter values on the hardware and the host PC according to the specified settings without opening the Calibration Memory dialog. However, if parameter upload/download according to the specified default behavior is not possible (for example, parameter values are to be downloaded, but no initial data is available), the Calibration Memory dialog nevertheless opens.

- If online calibration is not possible for at least one platform/device, ControlDesk displays the Platform/Device State Overview dialog (unless you disable the display of that dialog on the Platform Management page of the ControlDesk Options dialog).

Immediately after you start online calibration, the memory page specified by the Initial page setting during device configuration is activated in ControlDesk.

> **Tip**
>
> After online calibration was started globally for ControlDesk, you can start or stop online calibration for single platforms/devices belonging to the current experiment. Refer to Start Online Calibration (for Single Platform/Device) on page 83 and Stop Online Calibration (for Single Platform/Device) on page 87.

Calibration Memory dialog   To get identical parameter values in ControlDesk's mirrored memory and on the platform/device.

Platform/Device/Page   Lists the platforms/devices (and their memory pages) whose parameter values on the connected dSPACE real-time hardware or ECU differ from those in ControlDesk's mirrored memory.

Data Set CRC   Lists the CRC checksum values of the memory pages in the mirrored memory of ControlDesk (if possible). For more information, refer to How to Customize Calibration Memory Segments (ControlDesk Platform Management 📖).

ECU CRC   Lists the CRC checksum values of the memory pages on the connected dSPACE real-time hardware or ECU (if possible). For more information, refer to How to Customize Calibration Memory Segments (ControlDesk Platform Management 📖).

Command   Lets you specify for each platform/device and memory page whether to upload parameter values from the connected hardware to ControlDesk, or download them from ControlDesk to the hardware. The available commands depend on the selected platform/device type and the ECU's memory page concept.

Command	Description
Upload[1]	ControlDesk uploads the parameter values from the hardware to ControlDesk.
Download	ControlDesk downloads the parameter values from ControlDesk to the hardware.
Upload connected variables[1]	ControlDesk uploads the values of all the parameters and writable measurements that are currently selected for calibration from the hardware to the host PC. These are the parameters and writable measurements that are currently connected to an instrument and visualized on a layout. They are shown in the Variables controlbar with a red chain symbol.
	Note For a platform/device without a working data set, selecting the 'upload connected variables' option the first time online calibration is started prevents ControlDesk from creating a working and a reference data set. However, ControlDesk needs a working data set to save calibration data in ControlDesk. Use the 'upload' option for the first online transition instead.
Download connected variables	ControlDesk downloads the values of all the parameters and writable measurements that are currently selected for calibration from ControlDesk to the hardware. These are the parameters and writable measurements that are currently connected to an instrument and visualized on a layout. They are shown in the Variables controlbar with a red chain symbol.

[1] If data sets are selected for downloading on calibration start, their parameter values are always downloaded. Refer to How to Select Additional Data Sets for Downloading on Online Calibration Start on page 58.

Measurement Only   Leaves the parameter values in ControlDesk's mirrored memory and on the hardware inconsistent, and closes the Calibration Memory dialog. As a result,

- ControlDesk connects to the hardware.
- Online calibration is not started.
- Measuring variables is possible.

Platform/Device State Overview dialog

To get details on the configuration and connection state for platforms/devices in the active experiment.

State   Displays a symbol representing the current configuration and connection state individually for each platform/device. The symbols have the following meanings:

Symbol	Description
✔	The platform/device hardware or VEOS is connected and configured correctly. Starting online calibration for the platform/device is possible. [1]
✖	The platform/device hardware or VEOS is either not connected or not configured correctly. Starting online calibration for the platform/device will fail. [2]

[1] For an ECU Diagnostics device containing more than one logical link selected for use this means that at least one of the logical links is connected and configured correctly, and that online calibration is possible for these logical links. The Project controlbar displays status information individually for each logical link.

[2] For an ECU Diagnostics device this means that none of the contained logical links selected for use is connected or configured correctly, and that starting online calibration will fail for each logical link.

Platform/Device   Lists all the platforms/devices in the currently active experiment.

Description   Displays the current state.

Details   Lets you get a complete list of error messages for the platforms/devices for which starting online calibration currently is not possible.

Cancel   Lets you close the dialog without starting online calibration. This lets you reconfigure the platforms/devices of your experiment or connect missing hardware to the host PC, for example, before you try to start online calibration again.

Related topics

Basics

HowTos

References

Refresh Values

Access

You can access this command via:

Ribbon	Home - Calibration
Context menu of	None
Shortcut key	**Ctrl+Alt+R**
Icon	

Purpose

To update the values in all instruments that are connected to parameters.

Description

Usually parameter values are updated when you change them in an instrument. This is defined by the *On change* raster setting in the **Measurement Configuration** controlbar. In addition, regardless of whether you have changed the parameter value or not, ControlDesk reads the value every three seconds, to get a smooth visualization in plotter instruments. In both cases, the values are not read from the platform/device hardware, but from the mirrored memory on the ControlDesk PC. No bandwidth of the ECU interface is therefore allocated to this reading operations.

In contrast, refreshing the connected parameters allows you to update parameter values on demand and to read the values directly from the hardware. This is typically useful when multiple parameters refer to the same ECU address as it is common for bit masked parameters.

If online calibration is started, ControlDesk reads the values from the connected hardware. For offline calibration they are read from the mirrored memory.

> **Note**
>
> When online calibration is started, refreshing the connected parameters can cause a lot of data traffic.

Related topics

Basics

Basics on Measurement Rasters (ControlDesk Measurement and Recording 📖)

Start Online Calibration (for Single Platform/Device)

Access

This command is available only if online calibration was started globally for ControlDesk, but online calibration was not started for the selected platform/device. You can access it via:

Ribbon	None
Context menu of	Project ⑦ controlbar – platform/device
Shortcut key	None
Icon	None

Purpose

To start online calibration for the selected platform/device.

Description

If there is a platform/device that was not available when online calibration was started globally for ControlDesk, or whose online calibration has been stopped, you can start online calibration for it later on, without having to stop and restart global online calibration.

To stop online calibration for single platforms/devices, refer to Stop Online Calibration (for Single Platform/Device) on page 87.

Result

ControlDesk connects itself to the platform/device hardware. If you work with an XCP or CCP device for which the **Read XCP settings from ECU** or **Read CCP settings from ECU** property is selected, the XCP/CCP service configuration information is uploaded from the ECU when the device is connected to the ECU. If there are inconsistencies between the configuration information on the ECU and the configuration information that is currently used by the device, ControlDesk opens a dialog for you to specify whether to use the current value

or to upload the corresponding value from the ECU for each of the settings concerned.

Online calibration is started for the selected platform/device according to the configured start online calibration behavior. ControlDesk opens the Calibration Memory Dialog, if necessary. For further information, refer to Basics on Starting Online Calibration on page 17.

If you start online calibration for a measurement or calibration device, the memory page specified by the Initial page setting during device configuration is activated in ControlDesk immediately after you start online calibration.

If you start online calibration for a platform, its application is loaded to the hardware if no application is loaded yet. You can let the loaded application start automatically when you start online calibration.

When you start online calibration, ControlDesk automatically detects whether an application has been terminated. For details, refer to Detecting terminated applications (ControlDesk Platform Management 📖).

If switching to the online mode is not possible for the selected platform/device, ControlDesk displays the Device State Overview dialog (unless you disable the display of that dialog on the Platform Management page of the ControlDesk Options dialog).

Calibration Memory dialog   To get identical parameter values in ControlDesk's mirrored memory and on the platform/device.

Platform/Device/Page   Lists the platforms/devices (and their memory pages) whose parameter values on the connected dSPACE real-time hardware or ECU differ from those in ControlDesk's mirrored memory.

Data Set CRC   Lists the CRC checksum values of the memory pages in the mirrored memory of ControlDesk (if possible). For more information, refer to How to Customize Calibration Memory Segments (ControlDesk Platform Management 📖).

ECU CRC   Lists the CRC checksum values of the memory pages on the connected dSPACE real-time hardware or ECU (if possible). For more information, refer to How to Customize Calibration Memory Segments (ControlDesk Platform Management 📖).

Command   Lets you specify for each platform/device and memory page whether to upload parameter values from the connected hardware to ControlDesk, or download them from ControlDesk to the hardware. The available commands depend on the selected platform/device type and the ECU's memory page concept.

Command	Description
Upload[1]	ControlDesk uploads the parameter values from the hardware to ControlDesk.
Download	ControlDesk downloads the parameter values from ControlDesk to the hardware.
Upload connected variables[1]	ControlDesk uploads the values of all the parameters and writable measurements that are currently selected for calibration from the hardware to the host PC. These are the

Command	Description
	parameters and writable measurements that are currently connected to an instrument and visualized on a layout. They are shown in the **Variables** controlbar with a red chain symbol. **Note** For a platform/device without a working data set, selecting the 'upload connected variables' option the first time online calibration is started prevents ControlDesk from creating a working and a reference data set. However, ControlDesk needs a working data set to save calibration data in ControlDesk. Use the 'upload' option for the first online transition instead.
Download connected variables	ControlDesk downloads the values of all the parameters and writable measurements that are currently selected for calibration from ControlDesk to the hardware. These are the parameters and writable measurements that are currently connected to an instrument and visualized on a layout. They are shown in the **Variables** controlbar with a red chain symbol.

[1] If data sets are selected for downloading on calibration start, their parameter values are always downloaded. Refer to How to Select Additional Data Sets for Downloading on Online Calibration Start on page 58.

Measurement Only Leaves the parameter values in ControlDesk's mirrored memory and on the hardware inconsistent, and closes the **Calibration Memory** dialog. As a result,

- ControlDesk connects to the hardware.
- Online calibration is not started.
- Measuring variables is possible.

Platform/Device State Overview dialog

To get details on the configuration and connection state for platforms/devices in the active experiment.

State Displays a symbol representing the current configuration and connection state individually for each platform/device. The symbols have the following meanings:

Symbol	Description
✔	The platform/device hardware or VEOS is connected and configured correctly. Starting online calibration for the platform/device is possible. [1]
✖	The platform/device hardware or VEOS is either not connected or not configured correctly. Starting online calibration for the platform/device will fail. [2]

[1] For an ECU Diagnostics device containing more than one logical link selected for use this means that at least one of the logical links is connected and configured correctly, and that online calibration is possible for these logical links. The **Project** controlbar displays status information individually for each logical link.

Symbol	Description
2)	For an ECU Diagnostics device this means that none of the contained logical links selected for use is connected or configured correctly, and that starting online calibration will fail for each logical link.

Platform/Device   Lists all the platforms/devices in the currently active experiment.

Description   Displays the current state.

Details   Lets you get a complete list of error messages for the platforms/devices for which starting online calibration currently is not possible.

Cancel   Lets you close the dialog without starting online calibration. This lets you reconfigure the platforms/devices of your experiment or connect missing hardware to the host PC, for example, before you try to start online calibration again.

Related topics

Basics

HowTos

References

Start Proposed Calibration

Access   This command is available only if online calibration has been started. It is available for platforms/devices. You can access it via:

Ribbon	Home – Calibration – Proposed Calibration
Context menu of	None
Shortcut key	None
Icon	▽

Purpose

To start proposed calibration for all the platforms/devices of the active experiment.

Description

ControlDesk defers the download of modified parameters to the platform/device hardware until you select the Apply Proposed Calibration or Stop Proposed Calibration commands. Refer to Apply Proposed Calibration on page 73 and Stop Proposed Calibration on page 88.

> **Note**
>
> If you activate another data set (see How to Activate the Working or Reference Page on page 35) after you start proposed calibration, ControlDesk automatically stops proposed calibration. All parameter modifications you made since you started proposed calibration will get lost.

Related topics

HowTos

References

Measurement/Recording-Related Commands (ControlDesk Measurement and Recording 📖)

Stop Online Calibration (for Single Platform/Device)

Access

This command is available only if online calibration has been started globally. Online calibration must have been started for the selected platform/device, but no measurement must be running for it. You can access it via:

Ribbon	None
Context menu of	Project ⑦ controlbar – platform/device
Shortcut key	None
Icon	None

Purpose

To stop online calibration for the selected platform/device.

Description

You can stop online calibration for a single platform/device. The statuses of the other platforms/devices in the current experiment are not affected.

To restart online calibration for the selected platform/device later on, refer to Start Online Calibration (for Single Platform/Device) on page 83.

Result

ControlDesk stops communication with the selected platform/device hardware. After you stop online calibration, the hardware remains in the "connected" state.

For details on the possible platform/device states, refer to Basics of Platform/Device States (ControlDesk Platform Management 📖).

> **Tip**
>
> To stop online calibration globally for all platforms/devices in the current experiment in one step, refer to Go Offline on page 77.

Related topics

HowTos

References

Stop Proposed Calibration

Access

This command is available only if proposed calibration has been started. It is available for platforms/devices. You can access it via:

Ribbon	Home – Calibration – Proposed Calibration
Context menu of	None
Shortcut key	None
Icon	

Purpose

To stop proposed calibration for all platforms/devices of the active experiment.

Description	ControlDesk downloads modified parameters immediately after you modified them.

> **Note**
>
> If there are modified parameters that have not yet been applied, ControlDesk applies them before the proposed calibration is stopped.

Related topics

HowTos

Data Set Management

Where to go from here

Information in this section

Activate Application Image

Access	This command is available only for platforms (except for the Multiprocessor System and VEOS platforms). You can access this command via:

Ribbon	None
Context menu of	Project ⎘ controlbar – working data set
Shortcut key	None
Icon	None

Purpose	To activate the additional parameter values added to a real-time application by creating a new application image.

Result	The additional parameter values are active the next time the real-time application is run.

Description	When you create a new application image, its real-time application contains the parameter values of the selected data set as new parameter values (in addition to the original parameter values from the Simulink model). You can toggle between the original and the additional values via the context menu of the working data set, by activating / deactivating the application image for the next start of the real-time application.

Related topics	HowTos

References

Close (Data Set)

Access	This command is only available for opened data sets which are not connected to a reference or working page. You can access this command via:

Ribbon	None
Context menu of	Project ⎘ controlbar – data set

Shortcut key	None
Icon	None

Purpose   To close a data set which is not connected to a reference or working page.

Result   The selected data set is closed.

Description   Data sets can allocate an enormous amount of memory, especially when you work with dSPACE real-time hardware. You can reduce memory consumption by closing data sets.

A disadvantage of closing a data set is that it costs time to reopen it, if you need it again, for example, in a comparison task. In the decision of closing data sets you must therefore weigh up performance and memory aspects.

Related topics

Basics

References

Copy As New

Access   You can access this command via:

Ribbon	None
Context menu of	Project ⓘ controlbar – data set
Shortcut key	None
Icon	None

Purpose   To create a new writable data set.

Result   A writable copy of the selected data set is pasted into the active experiment.

Description

Each data set in an experiment, whether writable or not, can form the basis for a new writable data set. The selected data set is copied and can then be defined as the new working data set.

Related topics

Basics

HowTos

Create and Flash Application Image

Access

This command is available only for MicroAutoBox II/III and the DS1007 PPC Processor Board. Online calibration must be stopped. You can access this command via:

Ribbon	None
Context menu of	Project ⓘ controlbar – data set
Shortcut key	None
Icon	None

Purpose

To change the application image that belongs to the currently active variable description and load it to the flash memory of the platform.

Description

The changed application image contains the parameter values of the selected data set as new parameter values of the real-time application (in addition to the original parameter values specified in the Simulink model). The next time you start online calibration, the changed real-time application is loaded to the flash memory and the RAM of the real-time hardware.

> **Note**
>
> The following preconditions must be met for creating an application image via ControlDesk:
> - The real-time application must be built using an RTI version from dSPACE Release 5.1 or later.
> - When the real-time application is built, the Enable data set storage in application option in the Simulink® Coder™ must be enabled. For details, refer to RTI general build options page (RTI and RTI-MP Implementation Reference 📖).

Related topics

HowTos

References

Create and Load Application Image

Access

This command is available only for MicroAutoBox II/III and the DS1007 PPC Processor Board. Online calibration must be stopped. You can access this command via:

Ribbon	None
Context menu of	Project ⎘ controlbar – data set
Shortcut key	None
Icon	None

Purpose

To change the application image that belongs to the currently active variable description and use the additional parameter values when online calibration is started.

Description

The changed application image contains the parameter values of the selected data set as new parameter values of the real-time application (in addition to the original parameter values specified in the Simulink model). The next time you start online calibration, the changed real-time application is loaded to the RAM

of the real-time hardware and you can use the new parameter values in the experiment.

> **Note**
>
> If you want to create a new application image without loading the real-time application to the real-time hardware, refer to Create Application Image on page 96.

You can use Deactivate Application Image on page 104 and Activate Application Image on page 92 to toggle between the original parameter values (from the Simulink model) and the additional parameter values (from the ControlDesk data set).

> **Note**
>
> The following preconditions must be met for creating an application image via ControlDesk:
> - The real-time application must be built using an RTI version from dSPACE Release 5.1 or later.
> - When the real-time application is built, the **Enable data set storage in application** option in the Simulink® Coder™ must be enabled. For details, refer to RTI general build options page (RTI and RTI-MP Implementation Reference 📖).

Related topics

HowTos

References

Create Application Image

Access

This command is available only for MicroAutoBox II/III and the DS1007 PPC Processor Board. You can access this command via:

Ribbon	None
Context menu of	Project ↗ controlbar – data set

Shortcut key	None
Icon	None

Purpose
To create a new application image with an additional set of parameter values.

Description
Opens a dialog for you to select a destination folder and a name for a new application image. The created application image contains the parameter values of the selected data set as new parameter values of the real-time application (in addition to the original parameter values specified in the Simulink model). You can use the real‑time application with the new parameter values by selecting the variable description (SDF) file of the new application image.

> **Tip**
>
> If you want to change the application image that belongs to the currently active variable description, you can speed up the process by selecting one of the following commands:
> - If you want to use the additional parameter values immediately, refer to Create and Load Application Image on page 95.
> - If you also want to load the modified application image with activated additional parameter values to the flash memory of the dSPACE hardware, refer to Create and Flash Application Image on page 94.

When the SDF file is saved under a new name, the associated MAP, TRC and PPC files are automatically named after the SDF file, and the references to the PPC and TRC files contained in the SDF file are automatically adapted to the new file names.

You can use Activate Application Image on page 92 and Deactivate Application Image on page 104 to toggle between the original parameter values (from the Simulink model) and the additional parameter values (from the ControlDesk data set).

> **Note**
>
> The following preconditions must be met for creating an application image via ControlDesk:
> - The real-time application must be built using an RTI version from dSPACE Release 5.1 or later.
> - When the real-time application is built, the **Enable data set storage in application** option in the Simulink® Coder™ must be enabled. For details, refer to RTI general build options page (RTI and RTI-MP Implementation Reference 📖).

Related topics

Create Data Set

Access

This command is available for:

- Variable descriptions of platforms that have no data set (online calibration is running)
- Variable descriptions of platforms that have no data set but contain initial data (online calibration is not running)
- Variable descriptions of devices that have no memory segment definition

You can access this command via:

Ribbon	None
Context menu of	- Project ⧉ controlbar – variable description of platforms - Project ⧉ controlbar – variable description of devices without memory segment definition
Shortcut key	None
Icon	None

Purpose

To add an initial data set to a platform/device.

Result

ControlDesk adds an initial data set to the variable description and makes it the working data set.

Description

The reasons for the absence of initial data sets and solutions to the problem are different for platforms and devices. Refer to:

- Creating data sets for a dSPACE platform
- Creating data sets for a device

Creating data sets for a dSPACE platform	**Online calibration is running** You can select Create Data Set to create a data set. The parameter values are read from the connected hardware.

Online calibration is not running   If the variable description (SDF) file contains initial data, which requires a specific setting for the RTI build process, you can select Create Data Set to create a data set. For details on the required RTI build option, refer to Basics of Data Sets on page 44.

A variable description of a dSPACE platform might have no data sets for the following reasons:

- You have added a new variable description to a dSPACE platform. The default is for it to have no initial data sets. You can change this default setting on the Data Set Management page of the ControlDesk Options dialog.
- You have removed all the data sets of a dSPACE platform manually.

Creating data sets for a device

If the variable description of a device does not contain a definition of memory segments, no initial data sets are created for the device. If you start online calibration, you can select only Upload connected variables, which allows you to calibrate values. As no data sets are created, you cannot save calibration data.

To store calibration data when there are no memory segments, you can select Create Data Set in the variable description's context menu.

A device might have no memory segments for the following reasons:

- You have added a new variable description to a device that does not contain a definition of memory segments.
- You have removed all the memory segments of a device manually.

Related topics

Basics

Create Data Set from Open Layouts

Access

This command is available only for variable descriptions of dSPACE platforms that have at least one data set. You can access this command via:

Ribbon	None
Context menu of	Project ⧉ controlbar – variable description, data set, data set group
Shortcut key	None
Icon	None

Purpose	To create a new sub data set that contains all the parameters that are connected to instruments on currently opened layouts.

Result	The experiment contains a new sub data set. It is named after the selected data set with a From_open_Layouts extension.

Description	This command works like a filter that checks whether each parameter in the selected data set is connected to an instrument on an opened layout. All the parameters for which this is true are written to the new sub data set. The values of the new sub data set are copied from the selected data set.
	If you select this command from the context menu of a variable description, the currently active data set (working or reference data set) is used as the basis for the new sub data set.

Related topics	**Basics**

Create ECU Image

Access	This command is available only for measurement and calibration devices. You can access this command via:

Ribbon	None
Context menu of	Project ⍰ controlbar – data set
Shortcut key	None
Icon	None

Purpose	To create a new ECU Image file in the HEX, MOT or S19 file format.

Description	Creates an ECU Image file ⍰. You can select one of the following file formats for the ECU Image file.

Extension	Description
HEX	Intel file format for ECU Image files.
MOT	Motorola file format for ECU Image files.
S19	Motorola file format for ECU Image files.

The content of the file depends on the ECU Image file that was initially loaded. It can contain the code of the ECU application, or the data of the parameters within the application, or both.

The created ECU Image file can be used for ECU flash programming ⟁.

Related topics

HowTos

Data Set Grouping

Access

You can access this command via:

Ribbon	None
Context menu of	Project ⟁ controlbar – variable description
Shortcut key	None
Icon	None

Purpose

Lets you enable or disable the grouping of data sets.

Description

Groups are structuring elements that are only displayed in the project tree but not stored in the file system. If you enable data set grouping, the **Data Sets** folder is displayed and you can add groups and subgroups to it. Refer to Add Group (ControlDesk Project and Experiment Management 📖).

If you disable data set grouping, all data sets are displayed on the same level. ControlDesk does not save the grouping information. If you enable data set grouping again, you have to create new groups.

Related topics

Basics

Project Structure and File Storage (ControlDesk Project and Experiment Management 📖)

References

Add Group (ControlDesk Project and Experiment Management 📖)

Data Set Management Page

Access

This page is part of the ControlDesk Options dialog.

The dialog can be opened via the **Options Command** (ControlDesk User Interface Handling 🕮).

Purpose

To specify general settings of the data set management.

Dialog settings

Generate initial data sets for dSPACE variable descriptions (*.SDF)   Lets you specify whether to generate initial data sets for platforms when you add a variable description. Avoiding the generation of initial data sets is the default setting after you install ControlDesk. If you avoid the generation of initial data sets, ControlDesk creates "complete" data sets the first time you start online calibration.

Check limits on import data sets   Lets you specify whether to check the variable limits when you import a data set. If activated, the parameter values are checked against the hard limits specified in the variable description. If a hard limit is exceeded, the parameter value is set to the hard limit. For variables with conversion tables (e.g. CompuVTabs), an undefined value is always changed to a defined one.

> **Note**
>
> If you want to deactivate this option to make the import faster, make sure that the values to be imported do not exceed any hard limits.

Handle writable measurements as parameters   Lets you add writable measurements to the reference data set and working data set, and all new data sets you create. Refer to Adding Writable Measurement Variables to Data Sets on page 50.

> **Note**
>
> If you activate or deactivate the handling of writable measurements as parameters, follow the rules below. Otherwise data sets might become incomplete or inconsistent.
> - Stop online calibration before you change the handling of writable measurements.
> - With exception to data sets with the MMF file name extension, remove all the data sets that are already assigned to the variable description.

Data set handling on variable description reload   Lets you specify whether to remove or to restore data sets automatically after reloading or replacing a variable description.

Option	Description
Prompt user	ControlDesk asks you whether to remove or restore data sets each time you reload or replace a variable description.
Remove data sets	ControlDesk automatically removes the data sets of the reloaded or replaced variable description from the project tree. The data sets are not deleted in the file system. If you want to restore them later, you can import them.
Restore data sets	ControlDesk maintains the data set assignment and tries to reestablish all parameter connections. If parameters cannot be found they are removed from the data set. ControlDesk then saves the remaining parameters in a sub data set that can be reused in the experiment.

> **Note**
>
> Reestablishing parameter connections takes time. If you want to restore extensive data sets, remove unnecessary ones from the variable description before you start reloading or replacing.

ASAM Calibration Data Format version   Lets you specify the ASAM Calibration Data Format version for saving or exporting a data set.

Option	Description
CDF v2.0.0	ControlDesk saves the data set to a file with the ASAM Calibration Data Format version 2.0.
CDF v2.1.0	ControlDesk saves the data set to a file with the ASAM Calibration Data Format version 2.1.

Importing a CDFX file is independent of this setting. You can always import both file format versions.

Related topics

Basics

References

Options Command (ControlDesk User Interface Handling 📖)

Deactivate Application Image

Access

This command is available only for platforms (except for the Multiprocessor System platform). You can access this command via:

Ribbon	None
Context menu of	Project ⬀ controlbar – working data set
Shortcut key	None
Icon	None

Purpose

To deactivate the additional parameter values added to a real-time application by creating a new application image.

Result

The original parameter values (specified in the Simulink model) are active, the next time the real-time application is run.

Description

When you create a new application image, its real-time application contains the parameter values of the selected data set as new parameter values (in addition to the original parameter values from the Simulink model). You can toggle between the original and the additional values via the context menu of the working data set, by activating/deactivating the application image for the next start of the real-time application.

Related topics

HowTos

References

Export Data Sets / Export

Access

You can access this command via:

Ribbon	Home – Calibration – Data Sets
Context menu of	Project ⎘ controlbar – variable description
Shortcut key	None
Icon	None

Purpose

To export a data set.

Result

ControlDesk opens a standard dialog to save the data set file independently of the calibration project.

If you have selected Export Data Sets via the Home ribbon, ControlDesk opens the Export Data Set dialog (see below) which lets you select the data set you want to export.

Description

You can select one of the following formats:

- ASAM Calibration Data Format (CDFX)
- ASAM Calibration Data Format (XML)

The CDFX format is the standard format for a ControlDesk data set.

You can specify the ASAM Calibration Data Format version (2.0 or 2.1) on the Data Set Management page of the ControlDesk Options dialog.

> **Note**
>
> When you export a data set to a format other than CDFX, parameters without initial data are removed from the exported data set.

Export Data Set dialog

Lists all the data sets of the active experiment and lets you select the one you want to export.

OK   Opens a standard dialog to save the data set file independently of the calibration project.

Related topics

HowTos

References

Import Data Set(s)

Access

You can access this command via:

Ribbon	Home – Calibration – Data Sets
Context menu of	Project ⬚ controlbar – variable description, **Data Sets** folder or group
Shortcut key	None
Icon	None

> **Note**
>
> This command is not available in operator mode.

Purpose

To import a data set to the active experiment.

Result

ControlDesk opens a standard dialog to browse for a data set file. The selected data set is added to the active experiment.

If you have selected **Import Data Sets** via the **Home** ribbon, ControlDesk opens the **Import Data Set** dialog (see below) which lets you select the active variable description to which the data sets are to be imported.

Note

By default, the limits of the parameter values of the imported data sets are checked against the hard limits specified in the variable description. If you are sure that none of the parameter values to be imported exceeds the hard limits, you can deactivate this check to make the import faster. You can change this default setting on the Data Set Management page of the ControlDesk Options dialog.

Description

You can select files with one of the following formats:

- ASAM Calibration Data Format (CDFX)
- ASAM Calibration Data Format (XML)
- HEX file (HEX, S19, MOT)

Note

If you import a CDFX file, ControlDesk uses only those elements that it needs for its data sets, such as the variable name, type, and value. Other information such as information on the level of maturity is not needed by ControlDesk.

If you save or export this data set later on, the resulting CDFX file contains only these elements.

Tip

If you import an ECU Image file (HEX, S19, MOT), ControlDesk performs an EPK check to verify the consistency of the A2L file and the imported data set, if possible. For details, refer to Consistency Checks (EPK Checks) (ControlDesk Platform Management 📖).

For data sets derived from an ECU Image file, you can initiate a consistency check manually. Refer to Check EPK (ControlDesk Platform Management 📖). Data sets derived from an ECU Image file have the MMF file extension.

Import Data Set dialog

Lists all the variable descriptions that are currently active and lets you select the one to which the data sets are to be imported.

OK   Opens a standard dialog to browse for data set files. The selected data sets are added to the selected variable description of the active experiment.

Rename or Override dialog

The dialog opens if the data set you want to import has the same name as one that is already available in the active experiment.

Rename   Lets you rename the data set to be imported.

Override   (Available only if an overwriting of the data set is possible) Lets you overwrite the data set in the experiment.

An overwriting might be impossible for different reasons, see the following examples:

- The data set is assigned to an inactive variable description.
- The data set has the same name as one that is stored in the experiment's `Data Sets` folder but not visible in the experiment. To navigate to the `Data Sets` folder, right-click the experiment node in the Project controlbar and select Explore Folder.

Related topics

Basics

References

Make Read Only

Access

You can access this command via:

Ribbon	None
Context menu of	Project 🗗 controlbar – data set or sub data set, data set group
Shortcut key	None
Icon	None

Purpose

To make a data set or sub data set read-only.

Description

For platforms/devices, you can make data sets and sub data sets read-only. For devices, you can also define one data set as the reference data set (for comparisons in instruments).

Each time you copy a data set or sub data set, the initial status of the new sub data set is writable.

Via the context menu of a data set group, you can make all the data sets within the group read-only.

Related topics

Basics

HowTos

Make Reference

Access

The command is available only for devices. You can access the command via:

Ribbon	None
Context menu of	Project ⬀ controlbar – data set
Shortcut key	None
Icon	None

> **Note**
>
> This command is not available in operator mode.

Purpose

To define a data set as the new reference data set and write it to the reference page (if available on the hardware and online calibration is started).

Result

The data set is written to the reference page of the mirrored memory and activated. Its parameter values are set to read-only.

Description

The reference data set is the counterpart of the working data set. The working data set is writable and lets you calibrate parameter values. The reference data set is not writable and functions as a reliable data set with proven parameter values. It can be used as a backup in problem situations or for comparisons with other data sets.

Devices with one memory page do not have a reference page. However, ControlDesk needs a reference data set to compare its values with the values of the working data set in instruments, for example to display tendency symbols, or to show reference values in instruments.

The reference page for a device with one memory page cannot be activated. If you want to work with its parameter values, you must copy it and define it as the working data set.

Related topics

Basics

HowTos

Make Working

Access

The command is available if online calibration is stopped. You can access the command via:

Ribbon	None
Context menu of	Project ⬚ controlbar – data set
Shortcut key	None
Icon	None

Purpose

To define a data set as the new working data set.

Result

The data set is written to the mirrored memory on the host PC.

Description

Usually the working page of an ECU is writable and you can download the working data set to calibrate parameter values online. Some ECU's do not have a writable working page and you cannot download the working data set. In these cases you can use Make Working for offline calibration.

> **Note**
>
> If you define a sub data set as the working data set, a new complete data set is created and the missing values are taken from the currently active working data set. The name of the new data set is a combination of the name of the currently active working data set and the name of the sub data set.

Related topics

Basics

HowTos

Make Working / Download

Access

The command is available if online calibration is started. You can access this command via:

Ribbon	None
Context menu of	Project ⓘ controlbar – data set
Shortcut key	None
Icon	None

Purpose

To define a data set as the new working data set.

Result

The data set is written to the working page. Its parameter values remain writable.

Description

The working data set is writable and lets you calibrate parameter values. For devices, it is the counterpart of the reference data set. The reference data set is not writable and functions as a reliable data set with proven parameter values. It can be used as a backup in problem situations or for comparisons with other data sets.

Some ECU's do not have a writable working page and you cannot download the working data set. In these cases you can use Make Working for offline calibration.

> **Note**
>
> If you define a sub data set as the working data set, a new complete data set is created and the missing values are taken from the currently active working data set. The name of the new data set is a combination of the name of the currently active working data set and the name of the sub data set.

Related topics

Basics

HowTos

References

Open (Data Set)

Access

This command is only available for closed data sets. You can access this command via:

Ribbon	None
Context menu of	Project ⍐ controlbar – data set, data set group
Shortcut key	None
Icon	None

Purpose

To open a data set.

Description

A closed data set is indicated by a gray symbol next to the data set name.

Only the reference and the working data set are opened automatically when you activate an experiment.

Use the context menu of a data set group to open all the data sets within the group.

Related topics

Basics

HowTos

References

Read from Hardware

Access

The command is available if online calibration is started and no working data set is available. You can access this command via:

Ribbon	None
Context menu of	Project ⬚ controlbar – data set (except for working or reference data set)
Shortcut key	None
Icon	None

Purpose

To read the parameter values of a data set or sub data set directly from the platform or device hardware without creating a working data set.

Result

The values of the selected data set or sub data set are read directly from the hardware.

Description

In most cases, you will use a working data set to calibrate parameter values on the hardware. If no working data set is available, you can read the values of a data set or a sub data set directly from the hardware. For this approach, online calibration must be started.

Related topics

Basics

References

Read from Working Data Set

Access

The command is available only for devices. You can access the command via:

Ribbon	None
Context menu of	Project ⎘ controlbar – data set or sub data set
Shortcut key	None
Icon	None

Purpose

To read the parameter values of a data set or sub data set from the working data set.

Result

All the parameter values of a data set or sub data set are read from the working data set.

Related topics

HowTos

References

Remove All Data Sets

Access

This command is available only for variable descriptions of dSPACE platforms that have at least one data set. You can access this command via:

Ribbon	None
Context menu of	Project ⬀ controlbar – variable description (SDF file) of a platform
Shortcut key	None
Icon	None

Purpose

To remove all the data sets of a platform's variable description.

Result

ControlDesk removes all the data sets of the selected variable description (including the working data set).

> **Tip**
>
> To create a new initial data set, refer to Create Data Set on page 98.

Related topics

Basics

References

Rename (Data Set)

Access

You can access this command via:

Ribbon	None
Context menu of	Project ⬀ controlbar – data set
Shortcut key	None
Icon	None

Purpose	To rename a data set.
Result	Opens the Data Set Name dialog, see below.
Data Set Name dialog	Lets you enter a name for the data set. The name must be unique in the active experiment.
Related topics	Basics

Save (Data Set)

Access

You can access this command via:

Ribbon	None
Context menu of	Project ⓘ controlbar – data set, data set group
Shortcut key	None
Icon	None

Purpose

To save a data set within the active experiment.

Via the context menu of a data set group, you can save all changed data sets within the group.

Related topics

Basics

References

Save Data Set As

Access

You can access this command via:

Ribbon	None
Context menu of	Project ⧉ controlbar – data set
Shortcut key	None
Icon	None

Purpose

To save the data set as new data set within the active experiment.

Related topics

Basics

References

Write to Hardware

Access

The command is available if online calibration is started. You can access this command via:

Ribbon	None
Context menu of	Project ⧉ controlbar – data set (except for working or reference data set)
Shortcut key	None
Icon	None

Purpose

To write the parameter values of a data set or sub data set directly to the platform or device hardware without creating a working data set.

Result

The values of the selected data set or sub data set are written directly to the hardware.

Description	In most cases, you will use a working data set to calibrate parameter values on the hardware. If no working data set is available, you can write the values of a data set or a sub data set directly to the hardware. For this approach, online calibration must be started.

Related topics

Basics

References

Write to Working Data Set

Access

You can access this command via:

Ribbon	None
Context menu of	Project ⬔ controlbar – data set or sub data set
Shortcut key	None
Icon	None

Purpose

To write the values of a data set or sub data set to the working data set.

Result

All the parameter values of a data set or sub data set are written to the working data set. Parameters that are not in a sub data set keep their values.

Related topics

HowTos

References

Automation

Where to go from here

Information in this section

Programming ControlDesk Automation

Where to go from here

Information in this section

Information in other sections

Tool Automation Demos (ControlDesk Automation 📖)
Demonstrate how to automate ControlDesk and use ControlDesk events.

Automating Parameter Calibration

Introduction

In ControlDesk, you can calibrate parameters by changing their values in instruments on a layout. ControlDesk offers specialized instruments for editing variable types, for example, the Variable Array for scalar parameters and the Table Editor for nonscalar parameters.

Activating working and reference data sets

The following listing shows how to make the third data set of the active experiment the working data and the second data set the reference data set (excerpts from the `DataSetHandling.py` demo script).

```python
class MainDemoController(object):
    (...)
    def Initialize(self):
        (...)
        # Start ControlDesk.
        self.ControlDeskApplication = Dispatch("ControlDeskNG.Application")
        # Define Enums object.
        self.Enums = Enums(self.ControlDeskApplication)
        # Add XCP on CAN platform.
        self.DemoPlatform = self.ControlDeskApplication.ActiveExperiment.Platforms.Add(
            self.Enums.PlatformType.XCPonCAN)
        (...)
```

```
    def DefineWorkingAndReferenceSet(self):
        # Get the last created data set
        newDataSet = self.DemoPlatform.ActiveVariableDescription.DataSets.Item(2)
        # Set data set to working set
        newDataSet.MakeWorking()
        # Activate the working data set.
        newDataSet.Activate()
        # Set the second data set as the reference data set.
        self.DemoPlatform.ActiveVariableDescription.DataSets.Item(1).MakeReference()
```

Changing parameter values

The following listing shows how to change the values of parameters that are connected to a Variable Array (excerpts from the `LayoutAndInstrumentHandling.py` demo script).

```
# Get the root path
SCRIPTPATH = os.path.abspath(os.path.curdir)
if len(sys.argv) > 0:
    SCRIPTPATH = os.path.dirname(sys.argv[0])
# Define CalDemo .a2l image path.
A2LIMAGEPATH = os.path.abspath(os.path.join(SCRIPTPATH, "..\..\..\CalDemo\CalDemo.a2l"))
# Define CalDemo .mot image path.
MOTIMAGEPATH = os.path.abspath(os.path.join(SCRIPTPATH, "..\..\..\CalDemo\CalDemo.mot"))
(...)
# A list with string representations of some variables used in this demo.
VARIABLECONNECTIONSTRINGS = ["%s()://DummyOmega", "%s(5ms)://control_out",
                              "%s(OnChange)://f_Ki_1", "%s()://abs_sinp2_cosp2_table",
                              "%s()://airmass_x_table", "%s()://MeasureVector_MASK_0x0100[2]"]
(...)
# Name of the Variable Array instrument.
VARIABLE_ARRAY_INSTRUMENT_NAME = "TestVariableArray"
(...)
class MainDemoController(object):
    (...)
    def AddPlatformAndConnectVariableToKnobInstrument(self):
        (...)
        # Add XCP on CAN platform.
        self.DemoPlatform = self.ControlDeskApplication.ActiveExperiment.
                            Platforms.Add(self.Enums.PlatformType.XCPonCAN)
        self.DemoPlatform.VariableDescriptions.AddWithImage(A2LIMAGEPATH, MOTIMAGEPATH)
    (...)
    def EditVariableArray(self):
        # Get Variable Array instrument.
        variableArrayInstrument = self.ControlDeskApplication.LayoutManagement.ActiveLayout.Instruments.Item(
            VARIABLE_ARRAY_INSTRUMENT_NAME)
        (...)
        # Add new rows associated with variables.
        for index in range(3):
            # Add new row
            newRow = variableArrayInstrument.Rows.Add()
            # Add variable to row.
            newRow.MainVariable = VARIABLECONNECTIONSTRINGS[index] % self.DemoPlatform.Name
            # Set value.
            newRow.Value = 3.0 * index
```

> **Note**
>
> When you try to set the value of a parameter outside the parameter's value range, these changes are discarded without notice. However, you can implement a notification yourself by reading back the parameter value from the instrument.
>
> This applies only when you calibrate a parameter *by changing its value in an instrument on a layout*. It does not apply when you use the *direct variable access feature (refer to Automating Direct Variable Access (ControlDesk Automation* 📖 *))*.
>
> For information on parameter range limits, refer to Handling Parameter Range Limits on page 32.

Related topics

Basics

Automating Data Set Management

Introduction

A data set is a complete set of the parameters of an ECU. It reflects the current status of ECU parameter calibration. ControlDesk's Data Set Management allows you to create new data sets, export and import data sets, etc.

Unless otherwise indicated, the program listings below consist of excerpts from the `DataSetHandling.py` demo script.

Creating a new data set

The following listing shows how to create a new data set by copying an existing one.

```python
# Get the root path
SCRIPTPATH = os.path.abspath(os.path.curdir)
if len(sys.argv) > 0:
    SCRIPTPATH = os.path.dirname(sys.argv[0])
# Define CalDemo .a2l image path.
A2LIMAGEPATH = os.path.abspath(os.path.join(SCRIPTPATH, "..\..\..\CalDemo\CalDemo.a2l"))
# Define CalDemo .mot image path.
MOTIMAGEPATH = os.path.abspath(os.path.join(SCRIPTPATH, "..\..\..\CalDemo\CalDemo.mot"))
(...)
class MainDemoController(object):
    def Initialize(self):
        (...)
        # Create the enums object.
        self.Enums = Enums(self.ControlDeskApplication)
        (...)
        # Add XCP on CAN platform.
        self.DemoPlatform = self.ControlDeskApplication.ActiveExperiment.Platforms.Add(self.Enums.PlatformType.XCPonCAN)
        (...)
    def CreateAndEditNewDataSet(self):
        # Add variable description.
        self.DemoPlatform.VariableDescriptions.AddWithImage(A2LIMAGEPATH, MOTIMAGEPATH)
        # Access first data set.
        firstDataSet = self.DemoPlatform.ActiveVariableDescription.DataSets.Item(0)
        # Activate data set.
        firstDataSet.Activate()
        # Create new data set from copy.
        newDataSet = firstDataSet.Copy("CopiedDataSet")
```

Defining and activating a reference or working data set

The following listing shows how to select a data set and to make it the working or reference data set.

```python
# Get the root path
SCRIPTPATH = os.path.abspath(os.path.curdir)
if len(sys.argv) > 0:
    SCRIPTPATH = os.path.dirname(sys.argv[0])
# Define CalDemo .a2l image path.
A2LIMAGEPATH = os.path.abspath(os.path.join(SCRIPTPATH, "..\..\..\CalDemo\CalDemo.a2l"))
# Define CalDemo .mot image path.
MOTIMAGEPATH = os.path.abspath(os.path.join(SCRIPTPATH, "..\..\..\CalDemo\CalDemo.mot"))
(...)
```

```
class MainDemoController(object):
    def Initialize(self):
        (...)
        # Create the enums object.
        self.Enums = Enums(self.ControlDeskApplication)
        (...)
        # Add XCP on CAN platform.
        self.DemoPlatform = self.ControlDeskApplication.ActiveExperiment.Platforms.Add(self.Enums.PlatformType.XCPonCAN)
        (...)
    def CreateAndEditNewDataSet(self):
        # Add variable description.
        self.DemoPlatform.VariableDescriptions.AddWithImage(A2LIMAGEPATH, MOTIMAGEPATH)
        (...)
    def DefineWorkingAndReferenceSet(self):
        # Get the last created data set (created with "CreateAndEditNewDataSet")
        newDataSet = self.DemoPlatform.ActiveVariableDescription.DataSets.Item(2)
        # Set data set to working set
        newDataSet.MakeWorking()
        # Activate the working data set.
        newDataSet.Activate()
        # Set the second data set as the reference data set.
        self.DemoPlatform.ActiveVariableDescription.DataSets.Item(1).MakeReference()
```

Exporting and importing a data set

The following listing shows how to export the active data set and then to import it to a new variable description.

```
# Get the root path
SCRIPTPATH = os.path.abspath(os.path.curdir)
if len(sys.argv) > 0:
    SCRIPTPATH = os.path.dirname(sys.argv[0])
# Create new root path in the temporary system folder.
PROJECTROOTPATH = os.path.join(tempfile.gettempdir(), "DemoRoot")
# Define CalDemo .a2l image path.
A2LIMAGEPATH = os.path.abspath(os.path.join(SCRIPTPATH, "..\..\..\CalDemo\CalDemo.a2l"))
(...)
class MainDemoController(object):
    def Initialize(self):
        (...)
        # Create the enums object.
        self.Enums = Enums(self.ControlDeskApplication)
        (...)
        # Add XCP on CAN platform.
        self.DemoPlatform = self.ControlDeskApplication.ActiveExperiment.Platforms.Add(PlatformType.XCPonCAN)
    (...)
    def ExportAndImportDataSets(self):
        # Create export path.
        exportPath = os.path.join(PROJECTROOTPATH, "ExportDataSet.CDFX")
        # Get active data set.
        activeDataSet = self.DemoPlatform.ActiveVariableDescription.DataSets.ActiveDataSet
        (...)
        # Export data set.
        activeDataSet.Export(exportPath)
        # Create new variable description.
        newVarDes = self.DemoPlatform.VariableDescriptions.Add(A2LIMAGEPATH)
        # Import data set.
        newVarDes.DataSets.Import(exportPath)
```

> **Tip**
>
> Before importing a data set, you can check it for inconsistent parameters.
> See the following example:
>
> ```
> activeVariableDescription = self.DemoPlatform.\
>                 ActiveVariableDescription
> parameterList = activeVariableDescription.DataSets.\
>                 CheckForInconsistentParameters("E:\DataSets\
>                 CalDemo XCP-1-2.CDFX")
> ```
>
> After execution, `parameterList` contains all the inconsistent parameters.

Related topics

Basics

Data Set Handling

Data Set-Related Interfaces

Introduction   A data set is a complete set of the parameters of an ECU. It reflects the current status of ECU parameter calibration. ControlDesk's Data Set Management allows you to create new data sets, compare and merge different data sets.

Description   The Data Set Management object implements the *IXaDataSetManagement* interface. The properties of this interface can be used to manipulate the settings of the Data Set Management.

The *DataSets* property of an active variable description returns an object which implements the *IXaDataSets* collection. The properties of this collection can be used to get several objects representing single data sets. Each object implements an *IXaDataSet* object.

Related interfaces

Interface	Description
IXaDataSets (refer to DataSets / IXaDataSets <<Collection>> (ControlDesk Automation 📖))	Access to the collection of data sets in a platform.
IXaDataSet (refer to DataSet / IXaDataSet <<Interface>> (ControlDesk Automation 📖))	This interface is to access a single data set.
IXaDataSetManagement (refer to DataSetManagement / IXaDataSetManagement <<Interface>> (ControlDesk Automation 📖))	This interface is to access the data set management object.

Related documentation

Topic	Description
Automating Data Set Management on page 122	A data set is a complete set of the parameters of an ECU. It reflects the current status of ECU parameter calibration. ControlDesk's Data Set Management allows you to create new data sets, export and import data sets, etc.

Troubleshooting

Where to go from here	Information in this section

Problem When Starting Online Calibration for Devices Without Initial Data Set

Problem

For devices without an initial data set, calibration data cannot be saved on the host PC if "Upload connected variables" is selected the first time you start online calibration.

Description

To save calibration data on the host PC, a working data set is required.
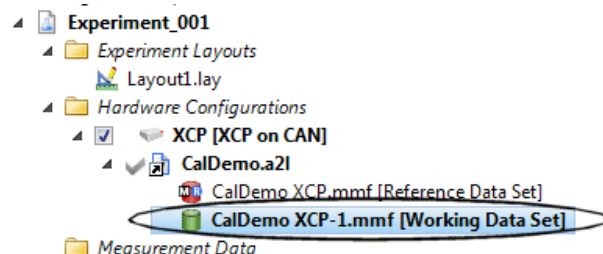
For measurement and calibration devices, the initial data set is generated from the variable description and the ECU Image file. However, if you add a variable description but no ECU Image file to a device, ControlDesk creates the device but does not generate the initial data set. Instead, the working and reference data sets are generated from the parameter values that are uploaded from the connected hardware when online calibration is started for the first time. Since a working data set is a complete data set, ControlDesk must upload all the parameters from the connected hardware to create the data sets.

If you select "Upload connected variables" the first time online calibration is started, an incomplete set of parameters is uploaded from the hardware. Since sub data sets cannot be used as working or reference data sets, ControlDesk cannot create a working data set from them. After online calibration is started, you can perform online calibration, but ControlDesk will not be able to save the calibrated parameter values.

Solution

Add an ECU Image file to the device when adding its variable description. ControlDesk initially fills the device's mirrored memory with the contents of the ECU Image file, and then fills the working and reference data sets with the parameter values of the mirrored memory. As a result, you can save calibration data right from the first time online calibration is started.

If you do not have an ECU Image file, select "Upload" the first time online calibration is started. ControlDesk then uploads all the parameter values from the hardware and creates a working data set from their values. After the working data set is created, ControlDesk can save calibration data.

Related topics

Basics

Problem When Downloading ECU Parameters with RESERVED Entries Outside of Memory Segments

Problem

When online calibration is started, downloading a complex ECU parameter, such as a map or curve, that is located outside of memory segments does not work as expected if the parameter has one or more RESERVED entries in the A2L file. The parameter is uploaded instead.

Description

The problem occurs only if all of the following applies:

- The parameter is complex, i.e., it consists of several components.
- The record layout of the complex parameter contains one or more RESERVED entries e.g., as padding between NO_AXIS_PTS and AXIS_PTS or between AXIS_PTS and FNC_VALUES components.
- The complex parameter is located outside of a calibration memory segment.
- The ECU Image file that was imported together with the variable description does not cover the entire memory range of the affected parameter.
- Start Online Calibration Behavior is set to Download or Download Connected.

When you start online calibration, ControlDesk checks whether the entire memory range of a complex parameter exists in the mirrored ECU memory in ControlDesk before it starts a download or upload.

If a parameter is not completely available in the mirrored memory, ControlDesk ignores a Download or Download Connected command for this parameter. Instead, it uploads the parameter from the ECU memory to to equalize the ECU memory with the mirrored memory. This is a default ControlDesk behavior.

Due to the RESERVED entries of the affected complex parameter, ControlDesk erroneously determines the parameter as not completely available in the mirrored memory. As a consequence, the parameter is uploaded and not downloaded.

Solution

As a workaround, calibrate the affected parameters after online calibration is started. In this state, the values of the parameters are written to the ECU memory as expected.

You can calibrate the values of the affected parameters in the following ways:

- In an instrument
- Via tool automation
- By selecting a data set that contains the desired parameter values and writing it to the working data set

Related topics

Basics

Limitations

Where to go from here

Information in this section

Limitations for Calibration

Introduction

There are some limitations for calibrating variables in ControlDesk:

Limitations for writable measurement variables

No calibration via the Slider and the Knob   You cannot calibrate writable measurement variables using the Slider and the Knob.

Calibration only if initial value is available   You can calibrate a writable measurement variable only if an initial value is available. This means that before you can calibrate the writable measurement variables, you have to create or open an experiment and then start a measurement.

Limitations for Data Set Management

Introduction

There are some limitations for managing data sets with ControlDesk:

Limitation for creating a new data set from open layouts	You can create a new data set from open layouts. For example, if online calibration has not started yet and if no initial parameter values are available, the values of parameters visualized in instruments on currently open layouts are mistakenly set to "0.0" in the created data set although no parameter values are available.
Limitation for importing and exporting CDFX files	If you import a CDFX file, ControlDesk uses only those elements that it needs for its data sets, such as the variable name, type, and value. Other information such as information on the level of maturity is not needed by ControlDesk. If you save or export this data set later on, the resulting CDFX file contains only these elements.
Limitation for exporting a data set to a format other than CDFX	When you export a data set to a format other than CDFX, parameters without initial data are removed from the exported data set.

Glossary

Introduction	Briefly explains the most important expressions and naming conventions used in the ControlDesk documentation.

Where to go from here

Information in this section

Numerics

3-D Viewer   An instrument for displaying items in a 3-D environment.

A

A2L file   A file that contains all the relevant information on measurement and calibration variables in an ECU application ⃞ and the ECU's communication interface(s). This includes information on the variables' memory addresses and conversion methods, the memory layout and data structures in the ECU as well as interface description data (IF_DATA) ⃞ .

Acquisition   An object in the Measurement Configuration ⃞ controlbar that specifies the variables to be measured and their measurement configuration.

Active variable description   The variable description that is currently active for a platform/device. Multiple variable descriptions can be assigned to one platform/device, but only one of them can be active at a time.

Additional write variable   A scalar parameter or writable measurement variable that can be connected to an instrument in addition to the main variable ⃞ . When the value of the main variable changes, the changed value is also applied to all the additional write variables connected to the instrument.

Airspeed Indicator   An instrument for displaying the airspeed of a simulated aircraft.

Altimeter   An instrument for displaying the altitude of a simulated aircraft.

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

Application image   An image file that contains all the files that are created when the user builds a real-time application. It particularly includes the variable

description (SDF) file. To extend a real-time application, ControlDesk lets the user create an updated application image from a data set. The updated application image then contains a real-time application with an additional set of parameter values.

Artificial Horizon   An instrument displaying the rotation on both the lateral and the longitudinal axis to indicate the angle of pitch and roll of a simulated aircraft. The Artificial Horizon has a pitch scale and a roll scale.

Automatic Reconnect   Feature for automatically reconnecting to platform/device hardware, for example, when the ignition is turned off and on, or when the physical connection between the ControlDesk PC and the ECU is temporarily interrupted.

If the feature is enabled for a platform/device and if the platform/device is in the 'unplugged' ⃝ state, ControlDesk tries to re-establish the logical connection to the platform/device hardware. After the logical connection is re-established, the platform/device has the same state as before the unplugged state was detected. A measurement started before the unplugged state was detected is resumed.

Automation   A communication mechanism that can be used by various programming languages. A client can use it to control a server by calling methods and properties of the server's automation interface.

Automation script   A script that uses automation to control an automation server.

Axis point object   Common axis ⃝

B

Bar   An instrument (or a value cell type of the Variable Array ⃝) for displaying a numerical value as a bar deflection on a horizontal or vertical scale.

Bitfield   A value cell type of the Variable Array ⃝ for displaying and editing the source value of a parameter as a bit string.

Bookmark   A marker for a certain event during a measurement or recording.

Browser   An instrument for displaying HTML and TXT files. It also supports Microsoft Internet Explorer© plug-ins that are installed on your system.

Bus communication replay   A feature of the Bus Navigator ⃝ that lets you replay logged bus communication data from a log file. You can add replay nodes

to the Bus Navigator tree for this purpose. You can specify filters to replay selected parts of the logged bus communication ⬚ .

Bus configuration   A configuration of all the controllers, communication matrices, and messages/frames/PDUs of a specific communication bus such as CAN. ControlDesk lets you display and experiment with bus configurations in the Bus Navigator ⬚ .

Bus connection   A mode for connecting dSPACE real-time hardware to the host PC via bus. The list below shows the possible bus connections:

- dSPACE real-time hardware installed directly in the host PC
- dSPACE real-time hardware installed in an expansion box connected to the host PC via dSPACE link board

Bus Instrument   An instrument available for the Bus Navigator ⬚ . It can be configured for different purposes, for example, to display information on received messages (RX messages) or to manipulate and transmit messages (TX messages). The instrument is tailor-made and displays only the message- and signal-specific settings which are enabled for display and/or manipulation by ControlDesk during run time.

Bus logging   A feature of the Bus Navigator ⬚ that lets you log raw bus communication data. You can add logger nodes on different hierarchy levels of the Bus Navigator tree for this purpose. You can specify filters to log filtered bus communication. The logged bus communication can be replayed ⬚ .

Bus monitoring   A feature of the Bus Navigator ⬚ that lets you observe bus communication. You can open monitoring lists and add monitor nodes on different hierarchy levels of the Bus Navigator tree for this purpose. You can specify filters to monitor filtered bus communication.

Bus Navigator   A controlbar ⬚ for handling bus messages, such as CAN messages, LIN frames, and Ethernet packets.

Bus statistics   A feature of the Bus Navigator ⬚ that lets you display and log statistical information on the bus load during bus monitoring ⬚ .

Bypassing   A method for replacing an existing ECU function by running a new function.

C

Calculated variable   A scalar variable that can be measured and recorded, and that is derived from one or more *input variables*.

The following input variable types are supported:

- Measurement variables ⬚
- Single elements of measurement arrays ⬚ or value blocks ⬚
- Scalar parameters ⬚ , or existing calculated variables

The value of a calculated variable is calculated via a user-defined *computation formula* that uses one or more input variables.

Calculated variables are represented by the ⇕ symbol.

CalDemo ECU   A demo program that runs on the same PC as ControlDesk. It simulates an ECU on which the Universal Measurement and Calibration (XCP ⧉) protocol and the Unified Diagnostic Services (UDS) protocol are implemented.

The CalDemo ECU allows you to perform parameter calibration, variable measurement, and ECU diagnostics with ControlDesk under realistic conditions, but without having to have a real ECU connected to the PC. Communication between the CalDemo ECU and ControlDesk can be established via XCP on CAN or XCP on Ethernet, and UDS on CAN.

> **Tip**
>
> If communication is established via XCP on Ethernet, the CalDemo ECU can also run on a PC different from the PC on which ControlDesk is running.

The memory of the CalDemo ECU consists of two areas called memory page ⧉. Each page contains a complete set of parameters, but only one page is accessible by the CalDemo ECU at a time. You can easily switch the memory pages of the CalDemo ECU to change from one parameter ⧉ to another in a single step.

Two ECU tasks run on the CalDemo ECU:

- ECU task #1 runs at a fixed sample time of 5 ms. In ControlDesk's **Measurement Configuration**, ECU task #1 is related to the time-based **5 ms**, **10 ms**, **50 ms** and **100 ms** measurement rasters of the CalDemo ECU.

- ECU task #2 has a variable sample time. Whenever the CalDemo ECU program is started, the initial sample time is 5 ms. This can then be increased or decreased by using the **dSPACE CalDemo** dialog.

    ECU task #2 is related to the **extEvent** measurement raster of the CalDemo ECU.

The CalDemo ECU can also be used to execute diagnostic services and jobs, handle DTCs and perform measurement and calibration via ECU diagnostics.

The CalDemo ECU program is run by invoking `CalDemo.exe`. The file is located in the `.\Demos\CalDemo` folder of the ControlDesk installation.

Calibration   Changing the parameter ⧉ values of real-time application ⧉s or ECU application ⧉s.

Calibration memory segment   Part of the memory of an ECU containing the calibratable parameters. Memory segments can be defined as `MEMORY_SEGMENT` in the A2L file. ControlDesk can use the segments to evaluate the memory pages of the ECU.

ControlDesk lets you perform the calibration of:

- Parameters inside memory segments
- Parameters outside memory segments
- Parameters even if no memory segments are defined in the A2L file.

CAN Bus Monitoring device   A device that monitors the data stream on a CAN bus connected to the ControlDesk PC.

The CAN Bus Monitoring device works, for example, with PC-based CAN interfaces such as the DCI-CAN2 or the DCI-CAN/LIN1.

The device supports the following variable description file types:

- DBC
- FIBEX
- AUTOSAR system description (ARXML)

CANGenerator   A demo program that simulates a CAN system, that is, it generates signals that can be measured and recorded with ControlDesk. The program runs on the same PC as ControlDesk.

The CANGenerator allows you to use the CAN Bus Monitoring device ⬚ under realistic conditions, but without having to have any device hardware connected to the PC.

The CAN (Controller Area Network) protocol is used for communication between the CANGenerator and ControlDesk. However, since the CANGenerator runs on the same PC as ControlDesk, ControlDesk does not communicate with the device via a real CAN channel, but via a *virtual CAN channel* implemented on the host PC.

You can start the CAN generator program by running `CANGenerator.exe`. The file is located in the `.\Demos\CANGenerator` folder of the ControlDesk installation.

Capture   A data packet of all the measurement variables assigned to a measurement raster ⬚. The packet comprises the data that results from a single triggering of the raster.

CCP   Abbreviation of CAN Calibration Protocol. This protocol can be implemented on electronic control units (ECUs) and allows users to access ECUs with measurement and calibration systems (MCS) such as ControlDesk.

The basic features of CCP are:

- Read and write access to the ECU memory, i.e., providing access for calibration
- Synchronous data acquisition
- Flash programming for ECU development purposes

The CCP protocol was developed by ASAM e.V. (Association for Standardization of Automation and Measuring Systems e.V.). For the protocol specification, refer to http://www.asam.net.

The following device supports ECUs with an integrated CCP service:

- CCP device ⬚

CCP device   A device that provides access to an ECU with CCP connected to the ControlDesk PC via CAN, for example, for measurement and calibration purposes via CCP (CAN Calibration Protocol) ⬚.

Check Button   An instrument (or a cell type of the Variable Array ⬚) for displaying whether the value of a connected variable matches predefined values or for writing a predefined value to a connected variable.

cmdloader   A command line tool for handling applications without using the user interface of an experiment software.

Common axis   A parameter ⧉ that consists of a 1-dimensional array containing axis points. A common axis can be referenced by one or more curves ⧉ and/or map ⧉s. Calibrating the data points of a common axis affects all the curves and/or maps referencing the axis.

Common axes are represented by the ⊞ symbol.

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

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

or

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

Computation method   A formula or a table that defines the transformation of a source value into a converted value (and vice versa). In addition to the computation methods defined in the variable description file, ControlDesk provides the *__Identity* computation method which means the converted and the source value are equal.

Connected   A platform/device state defined by the following characteristics:

- A continuous logical connection is established between ControlDesk and the platform/device hardware.
- A platform/device must be in the 'connected' state before it can change to the 'measuring/recording' or 'online calibration started' state.
- Online calibration is impossible. ControlDesk did not yet adjust the memory segments containing calibration data in the platform/device and on the corresponding hardware. Offline calibration is possible.
- Platform/device configuration is not possible. However, you can invoke platform/device configuration for a platform/device that is in the connected state. ControlDesk temporarily sets the platform/device to the disconnected state.

The 'connected' platform/device state is indicated by the 🖴 icon.

Connection mode   dSPACE real-time systems can be installed within the host PC or connected to the host via a bus interface and/or via Ethernet. When the Ethernet is being used, different network clients might exist. The connection type being used and, in the case of Ethernet, the network client being used, determine the dSPACE systems that can be accessed.

Control primitive   A special diagnostic communication object for changing communication states or protocol parameters, or for identifying (ECU) variants.

Controlbar   A window or pane outside the working area. Can be docked to an edge of the main window or float in front of it. A controlbar can contain a

document, such as a layout, or a tool, such as the Bus Navigator. It can be grouped with other controlbars in a window with tabbed pages.

ControlDesk   The main version of ControlDesk for creating and running experiments, and for accessing dSPACE real-time hardware and VEOS. The functionality can be extended by optional software modules.

ControlDesk - Operator Version   A version of ControlDesk that provides only a subset of functionality for running existing experiments. The functionality can be extended by optional software modules.

ControlDesk Bus Navigator Module   An optional software module for ControlDesk for handling bus messages, such as CAN, LIN, and FlexRay messages, frames, and PDUs and Ethernet packets.

ControlDesk ECU Diagnostics Module   An optional software module for ControlDesk that facilitates the calibration and validation of ECU diagnostic functions.

ControlDesk ECU Interface Module   An optional software module for ControlDesk for calibration and measurement access to electronic control units (ECUs). The module is also required for calibration and measurement access to virtual ECUs (V-ECUs) used in SIL testing scenarios.

ControlDesk Signal Editor Module   An optional software module for ControlDesk for the graphical definition and execution of signal generators for stimulating model variables of real-time/offline simulation applications.

Controller board   Single-board hardware computing the real-time application. Contains a real-time processor for fast calculation of the model and I/O interfaces for carrying out the control developments.

Conversion table   A table that specifies the value conversion ⓘ of a source value into a converted value. In the case of verbal conversion ⓘ, the converted value is a string that represents one numerical value or a range of numerical values.

Conversion type   The type of a computation method ⓘ, for example a linear function or a verbal computation method.

Curve   A parameter ⓘ that consists of

- A 1-dimensional array containing the axis points for the x-axis. This array can also be specified by a reference to a common axis ⓘ.
- Another 1-dimensional array containing data points. The curve assigns one data point to each axis point.

Curves are represented by the 📊 symbol.

D

DAQ module   A hardware module for the acquisition of physical quantities

Data Cursor   One or two cursors that are used to display the values of selected chart positions in a Time Plotter ⬚ or an Index Plotter ⬚ .

Data logger   An object in the Measurement Configuration ⬚ controlbar that lets you configure a data logging ⬚ .

Data logger signal list   A list that contains the variables to be included in subsequent data loggings ⬚ on real-time hardware.

Data logging   The recording of data on dSPACE real-time hardware that does not require a physical connection between the host PC and the real-time hardware. In contrast to flight recording ⬚ , data logging is configured in ControlDesk.

Data set   A set of the parameters and their values of a platform/device derived from the variable description of the platform/device. There are different types of data sets:

- Reference data set ⬚
- Sub data set ⬚
- Unassigned data set ⬚
- Working data set ⬚

DCI-CAN/LIN1   A dSPACE-specific interface between the host PC and the CAN/CAN FD bus and/or LIN bus. The DCI-CAN/LIN1 transfers messages between the CAN-/LIN-based devices and the host PC via the universal serial bus (USB).

DCI-CAN2   A dSPACE-specific interface between the host PC and the CAN bus. The DCI-CAN2 transfers CAN and CAN FD messages between the CAN-based devices and the host PC via the universal serial bus (USB).

DCI-GSI2   Abbreviation of *dSPACE Communication Interface - Generic Serial Interface 2*. A dSPACE-specific interface for ECU calibration, measurement and ECU interfacing.

DCI-GSI2 device   A device that provides access to an ECU with DCI-GSI2 connected to the ControlDesk PC for measurement, calibration, and bypassing purposes via the ECU's debug interface.

DCI-KLine1   Abbreviation of *dSPACE Communication Interface - K-Line Interface*. A dSPACE-specific interface between the host PC and the diagnostics bus via K-Line.

Debug interface   An ECU interface for diagnostics tasks and flashing.

Default raster   A platform-/device-specific measurement raster ⬚ that is used when a variable of the platform/device is connected to a plotter ⬚ or a recorder ⬚ , for example.

Deposition definition   A definition specifying the sequence in which the axis point values of a curve or map are deposited in memory.

Device   A software component for carrying out calibration ⬚ and/or measurement ⬚ , bypassing ⬚ , ECU flash programming ⬚ , or ECU diagnostics ⬚ tasks.

ControlDesk provides the following devices:

- Bus devices:
  - CAN Bus Monitoring device 🗐
  - Ethernet Bus Monitoring device 🗐
  - LIN Bus Monitoring device 🗐
- ECU Diagnostics device 🗐
- GNSS device 🗐
- Measurement and calibration devices:
  - CCP device 🗐
  - DCI-GSI2 device 🗐
  - XCP on CAN device 🗐
  - XCP on Ethernet device 🗐

Each device usually has a variable description 🗐 that specifies the device's variables to be calibrated and measured.

Diagnostic interface   Interface for accessing the fault memory 🗐 of an ECU.

Diagnostic job   (often called Java job) Programmed sequence that is usually built from a sequence of the diagnostic service 🗐. A diagnostic job is either a single-ECU job or a multiple-ECU job, depending on whether it communicates with one ECU or multiple ECUs.

Diagnostic protocol   A protocol that defines how an ECU communicates with a connected diagnostic tester. The protocol must be implemented on the ECU and on the tester. The diagnostics database 🗐 specifies the diagnostic protocol(s) supported by a specific ECU.

ControlDesk's ECU Diagnostics device supports CAN and K‑Line as the physical layers for communication with an ECU connected to the ControlDesk PC. For information on the supported diagnostic protocols with CAN and K‑Line, refer to Basics of ECU Diagnostics with ControlDesk (ControlDesk ECU Diagnostics 📖).

Diagnostic service   A service implemented on the ECU as a basic diagnostic communication element. Communication is performed by selecting a service, configuring its parameters, executing it, and receiving the ECU results. When a service is executed, a defined request is sent to the ECU and the ECU answers with a specific response.

Diagnostic trouble code (DTC)   A hexadecimal index for the identification of vehicle malfunctions. DTCs are stored in the fault memory 🗐 of ECUs and can be read by diagnostic testers.

Diagnostics database   A database that completely describes one or more ECUs with respect to diagnostics communication. ControlDesk supports the ASAM MCD-2 D ODX database 🗐 format, which was standardized by ASAM e.V. (Association for Standardisation of Automation and Measuring Systems e.V.). For the format specification, refer to http://www.asam.net.

Proprietary diagnostics database formats are not supported by ControlDesk.

Diagnostics Instrument   An instrument for communicating with an ECU via the diagnostic protocol using diagnostic services 🗗, diagnostic jobs 🗗, and control primitives 🗗.

Disabled   A platform/device state defined by the following characteristics:

- No logical connection is established between ControlDesk and the platform/device hardware.
- When a platform/device is disabled, ControlDesk does not try to establish the logical connection for that platform/device. Any communication between the platform/device hardware and ControlDesk is rejected.
- Online calibration is impossible. Offline calibration is possible.
- Platform/device configuration is possible.

The 'disabled' platform/device state is indicated by the 🔒 icon.

Disconnected   A platform/device state defined by the following characteristics:

- No logical connection is established between ControlDesk and the platform/device hardware.
- When a platform/device is in the disconnected state, ControlDesk does not try to re-establish the logical connection for that platform/device.
- Online calibration is impossible. Offline calibration is possible.
- Platform/device configuration is possible.

The 'disconnected' platform/device state is indicated by the 🖥❌ icon.

Display   An instrument (or a value cell type of the Variable Array 🗗) for displaying the value of a scalar variable or the text content of an ASCII variable.

Documents folder   A standard folder for user-specific documents.

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

DS1006 Processor Board platform   A platform that provides access to a DS1006 Processor Board connected to the host PC for HIL simulation and function prototyping purposes.

DS1007 PPC Processor Board platform   A platform that provides access to a single multicore DS1007 PPC Processor Board or a DS1007 multiprocessor system consisting of two or more DS1007 PPC Processor Boards, connected to the host PC for HIL simulation and function prototyping purposes.

DS1104 R&D Controller Board platform   A platform that provides access to a DS1104 R&D Controller Board installed in the host PC for function prototyping purposes.

DS1202 MicroLabBox platform   A platform that provides access to a MicroLabBox connected to the host PC for function prototyping purposes.

DsDAQ service   A service in a real-time application 🗗 or offline simulation application (OSA) 🗗 that provides measurement data from the application to the

host PC. Unlike the host service ⧉ , the DsDAQ service lets you perform, for example, triggered measurements with complex trigger conditions.

The following platforms support applications that contain the DsDAQ service:

- DS1007 PPC Processor Board platform ⧉
- DS1202 MicroLabBox platform ⧉
- MicroAutoBox III platform ⧉
- SCALEXIO platform ⧉
- VEOS platform ⧉
- XIL API MAPort platform ⧉

dSPACE Calibration and Bypassing Service   An ECU service for measurement, calibration, bypassing, and ECU flash programming. The dSPACE Calibration and Bypassing Service can be integrated on the ECU. It provides access to the ECU application and the ECU resources and is used to control communication between an ECU and a calibration and/or bypassing tool.

With the dSPACE Calibration and Bypassing Service, users can run measurement, calibration, bypassing, and flash programming tasks on an ECU via the DCI-GSI2. The service is also designed for bypassing ECU functions using dSPACE prototyping hardware by means of the RTI Bypass Blockset in connection with DPMEM PODs. The dSPACE Calibration and Bypassing Service allows measurement, calibration, and bypassing tasks to be performed in parallel.

dSPACE Internal Bypassing Service   An ECU service for on-target prototyping. The dSPACE Internal Bypassing Service can be integrated in the ECU application. It lets you add additional functions to be executed in the context of the ECU application without the need for recompiling the ECU application.

dSPACE Log   A collection of errors, warnings, information, questions, and advice issued by all dSPACE products and connected systems over more than one session.

dSPACE system   A hardware system such as a MicroAutoBox III or SCALEXIO system on which the real-time application ⧉ runs.

Duration trigger   A trigger ⧉ that defines a duration. Using a duration trigger, you can, for example, specify the duration of data acquisition for a measurement raster ⧉ . A duration trigger can be used as a stop trigger ⧉ .

E

ECU   Abbreviation of *electronic control unit*.

ECU application   A sequence of operations executed by an ECU. An ECU application is mostly represented by a group of files such as ECU Image files ⧉ , MAP files, A2L files ⧉ and/or software module description files.

ECU calibration interface   Interface for accessing an ECU by either emulating the ECU's memory or using a communication protocol (for example, XCP on CAN).

ECU diagnostics   Functions such as:
- Handling the ECU fault memory: Entries in the ECU´s fault memory can be read, cleared, and saved.
- Executing diagnostic services and jobs: Users can communicate with an ECU via a diagnostic protocol using diagnostic services, diagnostic jobs, and control primitives.

ControlDesk provides the ECU Diagnostics device ⬚ device to access ECUs for diagnostic tasks. Communication is via diagnostic protocol ⬚s implemented on the ECUs.

ECU diagnostics with ControlDesk are completely based on Open Diagnostic Data Exchange (ODX), the ASAM MCD-2 D diagnostics standard.

ControlDesk provides the Fault Memory Instrument ⬚ and the Diagnostics Instrument ⬚ for ECU diagnostics tasks.

ECU Diagnostics device   A device that provides access to ECUs connected to the ControlDesk PC via CAN or K-Line for diagnostics or flash programming purposes.

ControlDesk provides the *ECU Diagnostics v2.0.2* device, which supports the ASAM MCD-3 D V2.0.2 standard.

ControlDesk supports the following ODX database standards:
- ASAM MCD-2 D V2.0.1
- ASAM MCD-2 D V2.2.0 (ISO 22901-1)

ECU flash programming   A method by which new code or data is stored in ECU flash memory.

ECU Image file   A binary file that is part of the ECU application ⬚. It usually contains the code of an ECU application and the data of the parameters within the application. It can be stored as an Intel Hex (HEX) or Motorola S-Record (MOT or S19) file.

EESPort Configurations controlbar   A controlbar ⬚ for configuring error configuration ⬚s.

Electrical error simulation   Simulating electrical errors such as loose contacts, broken cables, and short-circuits, in the wiring of an ECU. Electrical error simulation is performed by the failure simulation hardware of an HIL simulator.

Electrical Error Simulation port (EESPort)   An *Electrical Error Simulation port* (EESPort) provides access to a failure simulation hardware for simulating electrical errors in an ECU wiring according to the ASAM AE XIL API standard.

The configuration of the EESPort is described by a hardware-dependent *port configuration* and one or more *error configurations*.

Environment model   A model that represents a part or all of the ECU's environment in a simulation scenario.
The environment model is a part of the simulation system ⎘ .

Environment VPU   The executable of an environment model ⎘ built for the VEOS platform. An environment VPU is part of an offline simulation application (OSA).

Error   An electrical error that is specified by:
- An error category
- An error type
- A load type

Error category   The error category defines how a signal is disturbed. Which errors you can create for a signal depends on the connected failure simulation hardware.

Error configuration   An XML file that describes a sequence of errors you want to switch during electrical error simulation. Each error configuration comprises error sets with one or more errors.

Error set   An error set is used to group errors (pin failures).

Error type   The error type specifies the way an error category – i.e., an interruption or short circuit of signals – is provided. The error type defines the disturbance itself.

Ethernet Bus Monitoring device   A device that monitors the data stream on an Ethernet network connected to the ControlDesk PC.
The device supports the following variable description file type:
- AUTOSAR system description (ARXML)

Ethernet connection   A mode for connecting dSPACE real-time hardware to the host PC via Ethernet. The list below shows the possible Ethernet connections:
- dSPACE real-time hardware installed in an expansion box connected to the host PC via Ethernet.
- MicroAutoBox II/III and MicroLabBox connected via Ethernet.

Ethernet decoding   A feature of the Bus Navigator ⎘ that lets you view protocol data and raw data of an Ethernet frame.

Event   An event that is triggered by an action performed in ControlDesk.

Event context   The scope of validity of event source ⎘ s and event ⎘ s. There is one event handler ⎘ code area for each event context.

Event handler   Code that is executed when the related event ⎘ occurs.

Event management   Functionality for executing custom code according to actions triggered by ControlDesk.

Event source   An object providing and triggering event ⎘ s.
LayoutManagement is an example of an event source.

Event state   State of an event ⧉. ControlDesk provides the following event states:

- No event handler ⧉ is defined
- Event handler is defined and enabled
- Event handler is defined and disabled
- Event handler is defined, but no Python code is available
- Event handler is deactivated because a run-time error occurred during the execution of the Python code

Expansion box   A box that hosts dSPACE boards. It can be connected to the host PC via bus connection or via network.

Experiment   A container for collecting and managing information and files required for a parameter calibration and/or measurement task. A number of experiments can be collected in a project but only one of them can be active.

Extension script   A Python script (PY or PYC file) that is executed each time ControlDesk starts up. An extension script can be executed for all users or user-specifically.

F

Failure insertion unit   Hardware unit used with dSPACE simulators to simulate failures in the wiring of an ECU, such as broken wire and short circuit to ground.

Fault memory   Part of the ECU memory that stores diagnostic trouble code (DTC) entries with status and environment information.

Fault Memory Instrument   An instrument for reading, clearing, and saving the content of the ECU's fault memory ⧉.

Firmware update   An update for the firmware installed in the board's flash memory. Firmware should be updated if it is older than required by the real-time application to be downloaded.

Fixed axis   An axis with data points that are not deposited in the ECU memory. Unlike a common axis ⧉, a fixed axis is specified within a curve ⧉ or map ⧉. The parameters of a fixed axis cannot be calibrated.

Fixed parameter   A parameter ⧉ that has a fixed value during a running simulation. Changing the value of a fixed parameter does not immediately affect the simulation results. The affect occurs only after you stop the simulation and

start it again. A fixed parameter is represented by an added pin in its symbol, for example: ▣.

Flash job   A specific diagnostic job for flashing the ECU memory. A flash job implements the process control for flashing the ECU memory, such as initialization, security access, writing data blocks, etc.

Flight recording   The recording of data on dSPACE real-time hardware that does not require a physical connection between the host PC and the real-time hardware. In contrast to data logging ⬧, flight recording is not configured in ControlDesk but via RTI and RTLib.

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

G

Gauge   An instrument for displaying the value of the connected variable by a needle deflection on a circular scale.

Gigalink module   A dSPACE board for connecting several processor boards in a multiprocessor system. The board allows high-speed serial data transmission via fiber-optic cable.

GNSS data   Positioning and timing data that is transmitted by a Global Navigation Satellite System (GNSS), such as GPS, GLONASS, or Galileo. GNSS receivers use this data to determine their location.

GNSS device   A device that provides positioning data from a GNSS receiver (e.g., a serial GPS mouse) in ControlDesk.

ControlDesk provides the *GNSS (GPS, GLONASS, Galileo, ...)* device that supports various global navigation satellite systems.

GPX file   An XML file that contains geodata, such as waypoints, routes, or tracks. In ControlDesk, you can import GPX files to visualize GNSS positioning data in a Map instrument.

Group   A collection of variables that are grouped according to a certain criterion.

H

Heading Indicator   An instrument displaying the heading direction of a simulated aircraft on a circular scale.

Host service   A service in a real-time application ⍰ that provides measurement data from the application to the host PC.

The following platforms support applications that contain the host service:

- DS1006 Processor Board platform ⍰
- DS1104 R&D Controller Board platform ⍰
- MicroAutoBox platform ⍰
- Multiprocessor System platform ⍰

|

Index Plotter   A plotter instrument ⍰ for displaying signals that are measured in an event-based raster (index plots).

Input quantity   A measurement variable that is referenced by a common axis and that provides the input value of that axis.

Instrument   An on-screen representation that is designed to monitor and/or control simulator variables interactively and to display data captures. Instruments can be arranged freely on layout ⍰ s.

The following instruments can be used in ControlDesk:

- 3-D Viewer ⍰
- Airspeed Indicator ⍰
- Altimeter ⍰
- Animated Needle ⍰
- Artificial Horizon ⍰
- Bar ⍰
- Browser ⍰
- Bus Instrument ⍰
- Check Button ⍰
- Diagnostics Instrument ⍰
- Display ⍰
- Fault Memory Instrument ⍰
- Frame ⍰
- Gauge ⍰
- Heading Indicator ⍰
- Index Plotter ⍰
- Invisible Switch ⍰
- Knob ⍰
- Multistate Display ⍰
- Multiswitch ⍰
- Numeric Input ⍰
- On/Off Button ⍰

- Push Button ⓘ
- Radio Button ⓘ
- Selection Box ⓘ
- Slider ⓘ
- Sound Controller ⓘ
- Static Text ⓘ
- Steering Controller ⓘ
- Table Editor ⓘ
- Time Plotter ⓘ
- Variable Array ⓘ
- XY Plotter ⓘ

Instrument Navigator   A controlbar ⓘ that displays a tree with all the instrument ⓘs of the active layout ⓘ and all the variables that are connected to them. The Instrument Navigator's main function is easy selection of instruments in complex layouts.

Instrument script   A Python script used to extend the functionality of an instrument ⓘ.

Instrument Selector   A controlbar ⓘ that provides access to ControlDesk's instrument ⓘs. The instruments can be placed on a layout ⓘ via double-click or drag & drop.

Interface description data (IF_DATA)   An information structure, mostly provided by an A2L file ⓘ, describing the type, features and configuration of an implemented ECU interface.

Internal Interpreter   ControlDesk's built-in programming interface for editing, running and importing Python scripts. It contains an Interpreter controlbar ⓘ where the user can enter Python commands interactively and which displays output and error messages of Python commands.

Interpreter controlbar   A controlbar ⓘ that can be used to execute line-based commands. It is used by the Internal Interpreter ⓘ to print out Python standard error messages and standard output during the execution or import of Python scripts.

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

IOCNET   IOCNET (I/O carrier network) is a dSPACE-specific high-speed serial communication bus that connects all the real-time hardware in a SCALEXIO system. IOCNET can also be used to build a multiprocessor system that consists of multiple SCALEXIO processor hardware components.

K

Knob   An instrument for displaying and setting the value of the connected variable by means of a knob on a circular scale.

L

Label list   A list of user-defined variables that can be used for saving connected variables, etc.

Layout   A window with instrument ⬚s connected to variables of one or more simulation models.

Layout Navigator   A controlbar ⬚ that displays all opened layout ⬚s. It can be used for switching between layouts.

Layout script   A Python script used to extend the functionality of a layout ⬚.

Leading raster   The measurement raster ⬚ that specifies the trigger ⬚ settings for the Time Plotter ⬚ display. The leading raster determines the time range that is visible in the plotter if a start and stop trigger is used for displaying the signals.

LIN Bus Monitoring device   A device that monitors the data stream on a LIN bus connected to the ControlDesk PC.
The LIN Bus Monitoring device works, for example, with PC-based LIN interfaces.
The device supports the following variable description file types:
- LDF
- FIBEX
- AUTOSAR system description (ARXML)

Load type   The load type specifies the option to disturb a signal with or without load rejection.

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>`

Logical link   A representation of an ECU specified in the diagnostics database. A logical link contains information on the ECU itself, and all the information required for accessing it, such as the diagnostic protocol ⬚ used for

communication between the ECU and ControlDesk. Each logical link is represented by a unique short name in the ODX database ⧉ .

Look-up table   A look-up table maps one or more input values to one output value. You have to differentiate between the following look-up table types:
- A 1-D look-up table maps one input value to one output value.
- A 2-D look-up table maps two input values to one output value.
- An n-D look-up table maps multidimensional table data with 3 or more input values to one output value.

Look-up table is a generic term for curves ⧉ and maps ⧉ .

M

Main variable   A scalar variable that is visualized in an instrument that can be used to change parameter values. In addition to the main variable, additional write variable ⧉ s can also be connected to (but not visualized in) the same instrument. When you change the value of the main variable in an instrument, the changed value is also applied to all the additional write variables connected to that instrument.

Map   A parameter ⧉ that consists of
- A 1-dimensional array containing the axis points for the x-axis. This array can also be specified by a reference to a common axis ⧉ .
- A 1-dimensional array containing the axis points for the y-axis. This array can also be specified by a reference to a common axis ⧉ .
- A 2–dimensional array containing data points. The map assigns one data point of the array to each pair of x-axis and y-axis points.

Maps are represented by the ⊞ symbol.

Map file   A file that contains symbols (symbolic names) and their physical addresses. It is generated during a build process of an ECU application.

Map instrument   A customized Browser ⧉ instrument. It uses an instrument script to open a web map and connect positioning data to the map. The Map instrument offers prepared connection nodes to connect variables with GNSS data ⧉ .

Measurement   Viewing and analyzing the time traces of variables ⧉ , for example, to observe the effects of ECU parameter changes.
ControlDesk provides various instruments ⧉ for measuring variables.

Measurement (variable type)   A scalar variable that can be measured, including individual elements of a measurement array.

Measurement variables are represented by the ⬛ symbol.

Measurement array   A 1-, 2-, or 3-dimensional array of measurement variables. In variable lists, ControlDesk displays entries for the measurement array itself and for each array element.

Measurement arrays are represented by the ⊞ symbol.

Measurement buffer   A ring buffer that buffers measurement data at the start of a measurement ⧉. The measurement buffer size determines the amount of data that can be buffered. Earlier values are overwritten by later values when the buffer capacity is exceeded (buffer overflow).

Measurement Configuration   A controlbar ⧉ that allows you to configure measurement ⧉, recording ⧉ and data logging ⧉.

Measurement Data API   Application programming interface for accessing measurement data. The API lets the user access measurement data without having to use ControlDesk.

Measurement Data Pool   A controlbar ⧉ that provides access to measurement data recorded in measurement data files.

Measurement raster   Specification of how often a value of a variable ⧉ is updated during a measurement ⧉. A measurement raster can be derived from a measurement service ⧉.

Measurement service   The generic term for the following services:
- CCP ⧉ service
- DsDAQ service ⧉
- Host service ⧉
- XCP ⧉ service

Measurement signal list   A list containing the variables to be included in subsequent measurements and recording. The list is global for all platforms/devices of the current experiment. The measurement signal list is available in the configuration area of the **Measurement Configuration** ⧉ controlbar.

Measurement variable   Any variable type that can be measured but not calibrated.

Measuring/recording   A platform/device state defined by the following characteristics:
- A continuous logical connection is established between ControlDesk and the platform/device hardware.
- Online calibration is possible. Parameter values can be changed directly on the platform/device hardware.
- A measurement (or recording) is running.
- Platform/device configuration is not possible.

The 'measuring' / 'recording' platform/device state is indicated by the ▶👓 icon.

Memory page   An area of a calibration memory. Each page contains a complete set of parameters of the platform/device hardware, but only one of the pages is "visible" to the microcontroller of the ECU or the real-time processor (RTP) of the platform hardware at a time.

ControlDesk supports platform/device hardware with up to two memory pages. These are usually the working page ⁇ and the reference page ⁇ . The parameter values on the two memory pages usually are different. ControlDesk lets you switch from one page to the other, so that when parameters are changed on one page, the changes can be made available to the ECU or prototyping hardware via a single page switch.

Messages controlbar   A controlbar ⁇ displaying a history of all error and warning messages that occur during work with ControlDesk.

MicroAutoBox III platform   A platform that provides access to a MicroAutoBox III connected to the host PC for function prototyping purposes such as Bypassing ⁇ .

MicroAutoBox platform   A platform that provides access to a MicroAutoBox II connected to the host PC for function prototyping purposes such as bypassing.

Mirrored memory   A memory area created by ControlDesk on the host PC that mirrors the contents of the available memory pages of calibration and prototyping hardware. For hardware with two memory pages, the mirrored memory is divided into a reference and a working page, each of them containing a complete set of parameters. When a calibration or prototyping platform/device is added to an experiment, ControlDesk initially fills the available memory pages of the mirrored memory with the contents of the ECU Image file ⁇ (initial filling for calibration devices) or with the contents of the SDF file (initial filling for platforms).

- Mirrored memory for offline calibration

  Parameter values can even be changed offline ⁇ . Changes to parameter values that are made offline affect only the mirrored memory.

- Offline-to-online transition for online calibration

  For online calibration, an offline-to-online transition must be performed. During the transition, ControlDesk compares the memory page ⁇ s of the hardware of each platform/device with the corresponding pages of the mirrored memory. If the pages differ, the user has to equalize them by uploading them from the hardware to the host PC, or downloading them from the host PC to the hardware.

- Mirrored memory for online calibration

  When ControlDesk is in the online mode, parameter value changes become effective synchronously on the memory pages of the hardware and in the mirrored memory. In other words, parameter values on the hardware and on the host PC are always the same while you are performing online calibration.

Modular system   A dSPACE processor board and one or more I/O boards connected to it.

Multi-capture history   The storage of all the capture ⁇ s acquired during a triggered measurement ⁇ . The amount of stored data depends on the measurement buffer.

Multi-pin error   A feature of the SCALEXIO concept for electrical error simulation that lets you simulate a short circuit between three or more signal

channels and/or bus channels. The channels can be located on the same or different boards or I/O units. You can simulate a short circuit between:

- Channels of the same signal category (e.g., four signal generation channels)
- Channels of different signal categories (e.g., three signal generation channels and two signal measurement channels)
- Signal channels and bus channels (e.g., two signal generation channels, one signal measurement channel, and one bus channel)

Multiple electrical errors A feature of the SCALEXIO concept for electrical error simulation that lets you switch electrical errors at the same time or in succession. For example, you can simulate an open circuit for one channel and a short circuit for another channel at the same time, without deactivating the first error.

Multiprocessor System platform A platform that provides access to:

- A multicore application running on a multicore DS1006 board
- A multiprocessor application on a multiprocessor system consisting of two or more DS1006 processor boards interconnected via Gigalink.

ControlDesk handles a multiprocessor/multicore system as a unit and uses one system description file (SDF file) to load the applications to all the processor boards/cores in the system.

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

Multistate LED A value cell type of the Variable Array ⓐ for displaying the value of a variable as an LED state.

Multiswitch An instrument for changing variable values by clicking sensitive areas in the instrument and for visualizing different states depending on the current value of the connected variable.

N

Numeric Input An instrument (or a value cell type of the Variable Array ⓐ) for displaying and setting the value of the connected variable numerically.

O

Observing variables Reading variable values cyclically from the dSPACE real-time hardware and displaying their current values in ControlDesk, even if no measurement ⓐ is running. Variable observation is performed without using a measurement buffer, and no value history is kept.

For platforms that support variable observation, variable observation is available for parameters ⍰ and measurement variables ⍰ that are visualized in single-shot instruments ⍰ (all instruments except for a plotter ⍰). If you visualize a variable in a single-shot instrument, the variable is not added to the measurement signal list ⍰. Visualizing a parameter or measurement variable in a plotter automatically adds the variable to the measurement signal list.

ControlDesk starts observing variables if one of the following conditions is true:

- Online Calibration is started ⍰ for the platform.

  All the parameters and measurement variables that are visualized in single-shot instruments are observed.

- Measurement is started ⍰ for the platform.

  All the visualized parameters and measurement variables that are not activated for measurement in the measurement signal list are observed. Data of the activated parameters and measurement variables is acquired using measurement rasters.

ODX database   Abbreviation of Open Diagnostic Data Exchange, a diagnostics database ⍰ that is the central ECU description for working with an ECU Diagnostics device ⍰ in ControlDesk. The ODX database contains all the information required to perform diagnostic communication between ControlDesk and a specific ECU or set of ECUs in a vehicle network. ControlDesk expects the database to be compliant with ASAM MCD-2 D (ODX).

Offline   State in which the parameter values of platform/device hardware in the current experiment cannot be changed. This applies regardless of whether or not the host PC is physically connected to the hardware.

The mirrored memory ⍰ allows parameter values to be changed even offline.

Offline simulation   A PC-based simulation in which the simulator is not connected to a physical system and is thus independent of the real time.

Offline simulation application (OSA)   An offline simulation application (OSA) file is an executable file for VEOS. After the build process with a tool such as the VEOS Player, the OSA file can be downloaded to VEOS.

An OSA contains one or more VPUs ⍰, such as V-ECUs and/or environment VPUs.

On/Off Button   An instrument (or a value cell type of the Variable Array ⍰) for setting the value of the connected parameter to a predefined value when the button is pressed (On value) and released (Off value).

Online calibration started   A platform/device state defined by the following characteristics:

- A continuous logical connection is established between ControlDesk and the platform/device hardware.
- Online calibration is possible. Parameter values can be changed directly on the platform/device hardware.
- Platform/device configuration is not possible.

Before starting online calibration, ControlDesk lets you compare the memory page ⍰s on the platform/device hardware with the corresponding pages of the mirrored memory ⍰. If the parameter values on the pages differ, they must be

equalized by uploading the values from the hardware to ControlDesk, or downloading the values from ControlDesk to the hardware. However, a page cannot be downloaded if it is read-only.

The 'online calibration started' platform/device state is indicated by the ♦⬮ symbol.

Operation signal   A signal ⬚ which represents the result of an arithmetical operation (such as addition or multiplication) between two other signals.

Operator mode   A working mode of ControlDesk in which only a subset of the ControlDesk functionality is provided. You can work with existing experiments but not modify them, which protects them from unintentional changes.

Output parameter   A parameter ⬚ or writable measurement ⬚ whose memory address is used to write the computed value of a calculated variable ⬚ to.

P

Parameter   Any variable type that can be calibrated.

Parameter (variable type)   A scalar parameter ⬚, as well as the individual elements of a value block ⬚.

Scalar parameters are represented by the $\boxed{\text{P}}$ symbol.

Parameter limits   Limits within which parameters can be changed. Parameters have hard and weak limits.

- Hard limits

  Hard limits designate the value range of a parameter that you *cannot* cross during calibration.

  The hard limits of a parameter originate from the corresponding variable description ⬚ and cannot be edited in ControlDesk.

- Weak limits

  Weak limits designate the value range of a parameter that you *should not* cross during calibration. When you cross the value range defined by the weak limits, ControlDesk warns you.

  In ControlDesk, you can edit the weak limits of a parameter within the value range given by the parameter's hard limits.

PHS (Peripheral High Speed) bus   A dSPACE-specific bus for communication between a processor board and the I/O boards in a modular system. It allows direct I/O operations between the processor board (bus master) and I/O boards (bus slaves).

PHS-bus-based system   A modular dSPACE system consisting of a processor board such as the DS1006 Processor Board and I/O boards. They communicate with each other via the PHS (Peripheral High Speed) bus ⬚.

Pitch variable   A variable connected to the pitch scale of an Artificial Horizon ⧉ .

Platform   A software component representing a simulator where a simulation application is computed in real-time (on dSPACE real-time hardware) or in non-real-time (on VEOS).

ControlDesk provides the following platforms:

- DS1006 Processor Board platform ⧉
- DS1007 PPC Processor Board platform ⧉
- DS1104 R&D Controller Board platform ⧉
- DS1202 MicroLabBox platform ⧉
- MicroAutoBox platform ⧉
- MicroAutoBox III platform ⧉
- Multiprocessor System platform ⧉
- SCALEXIO platform ⧉
- VEOS platform ⧉
- XIL API MAPort platform ⧉

Each platform usually has a variable description ⧉ that specifies its variables.

Platform trigger   A trigger ⧉ that is available for a platform ⧉ and that is evaluated on the related dSPACE real-time hardware or VEOS.

Platforms/Devices controlbar   A controlbar ⧉ that provides functions to handle devices ⧉, platforms ⧉, and the applications ⧉ assigned to the platforms.

Plotter instrument   ControlDesk offers three plotter instruments with different main purposes:

- The Index Plotter ⧉ displays signals in relation to events.
- The Time Plotter ⧉ displays signals in relation to measurement time.
- The XY Plotter ⧉ displays signals in relation to other signals.

Port configuration   To interface the failure simulation hardware, an EESPort needs the hardware-dependent *port configuration file* (PORTCONFIG file). The file's contents must fit the connected HIL simulator architecture and its failure simulation hardware.

Postprocessing   The handling of measured and recorded data by the following actions:

- Displaying measured or recorded data
- Zooming into measured or recorded signals with a plotter ⧉
- Displaying the values of measurement variables and parameters as they were at any specific point in time

Processor board   A board that computes real-time applications. It has an operating system that controls all calculations and communication to other boards.

Project   A container for collecting and managing the information and files required for experiment/calibration/modification tasks in a number of experiments ⏍. A project collects the experiments and manages their common data.

Project controlbar   A controlbar ⏍ that provides access to projects and experiments and all the files they contain.

Project root directory   The directory on your file system to which ControlDesk saves all the experiments and documents of a project ⏍. Every project is associated with a project root directory, and several projects can use the same project root directory. The user can group projects by specifying several project root directories.

ControlDesk uses the Documents folder ⏍ as the default project root directory unless a different one is specified.

Properties controlbar   A controlbar ⏍ providing access to the properties of, for example, platforms/devices, layouts/instruments, and measurement/recording configurations.

Proposed calibration   A calibration mode in which the parameter value changes that the user makes do not become effective on the hardware until they are applied. This allows several parameter changes to be written to the hardware together. Being in proposed calibration mode is like being in the offline calibration mode temporarily.

Push Button   An instrument (or a value cell type of the Variable Array ⏍) for setting the value of the connected parameter by push buttons.

Python Editor   An editor for opening and editing PY files.

Q

Quick start measurement   A type of measurement in which all the ECU variables configured for measurement are measured and recorded, starting with the first execution of an ECU task. ControlDesk supports quick start measurements on ECUs with DCI-GSI2, CCP, and XCP (except for XCP on Ethernet with the TCP transmission protocol).

Quick start measurement can be used to perform cold start measurements. Cold start means that the vehicle and/or the engine are cooled down to the temperature of the environment and then started. One reason for performing cold start measurements is to observe the behavior of an engine during the warm-up phase.

R

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

Real-time application   An application that can be executed in real time on dSPACE real-time hardware. A real-time application can be built from a Simulink model containing RTI blocks, for example.

Record layout   A record layout is used to specify a data type and define the order of the data in the memory of the target system (ECU, for example). For scalar data types, a record layout allows you to add an address mode (direct or indirect). For structured (aggregated) data types, the record layout specifies all the structure elements and the order they appear in.

The RECORD_LAYOUT keyword in an A2L file is used to specify the various record layouts of the data types in the memory. The structural setup of the various data types must be described in such a way that a standard application system will be able to process all data types (reading, writing, operating point display etc.).

Record layout component   A component of a record layout. A structured record layout consists of several components according to the ASAP2 specification. For example, the AXIS_PTS_X component specifies the x-axis points, and the FNC_VALUES component describes the function values of a map or a curve.

Recorder   An object in the Measurement Configuration ⓘ controlbar that specifies and executes the recording ⓘ of variables according to a specific measurement configuration.

Recorder signal list   A list that contains the variables to be included in subsequent recordings ⓘ.

Recording   Saving the time traces of variables to a file. Both measurement variables and parameters can be recorded. Recorded data can be postprocessed ⓘ directly in ControlDesk.

A recording can be started and stopped immediately or via a trigger:

- Immediate recording

  The recording is started and stopped without delay, without having to meet a trigger condition.

- Triggered recording

  The recording is not started or stopped until certain trigger conditions are met. These conditions can be defined and edited in ControlDesk.

Reduction data   Additional content in an MF4 file that allows for visualizing the MF4 file data depending on the visualization resolution. Reduction data therefore improves the performance of the visualization and postprocessing of measurement data.

Reference data set   A read-only data set assigned to the reference page of a device that has two memory page ⓘ s. There can be only one reference data set for each device. The reference data set is read-only.

Reference page   Memory area containing the parameters of an ECU. The reference page contains the read-only reference data set ⧉ .

> **Note**
>
> Some platforms/devices provide only a working page ⧉ . You cannot switch to a reference page in this case.

Resynchronization   Mechanism to periodically synchronize the drifting timers of the platform/device hardware ControlDesk is connected to. Resynchronization means adjustment to a common time base.

Roll variable   A variable connected to the roll scale of an Artificial Horizon ⧉ .

S

Sample count trigger   A trigger ⧉ that specifies the number of samples in a data capture.

A sample count trigger can be used as a stop trigger ⧉ .

SCALEXIO platform   A platform that provides access to a single-core, multicore or multiprocessor SCALEXIO system ⧉ connected to the host PC for HIL simulation and function prototyping purposes.

SCALEXIO system   A dSPACE hardware-in-the-loop (HIL) system consisting of at least one processing hardware component, I/O boards, and I/O units. They communicate with each other via the IOCNET ⧉ . In a SCALEXIO system, two types of processing hardware can be used, a DS6001 Processor Board or a real-time industry PC as the SCALEXIO Processing Unit. The SCALEXIO system simulates the environment to test an ECU. It provides the sensor signals for the ECU, measures the signals of the ECU, and provides the power (battery voltage) for the ECU and a bus interface for restbus simulation.

SDF file   The system description file that describes the files to be loaded to the individual processing units of a simulation platform. It also contains the variable description of the relevant simulation application ⧉ .

The SDF file is generated automatically when the TRC file ⧉ is built.

Segment   The minimum part a segment signal ⧉ can consist of.
There are different kinds of segments to be used in segment signals:

- Segments to form synthetic signal shapes (sine, sawtooth, ramp, etc.)
- Segments to perform arithmetical operations (addition, multiplication) with other segments
- Segments to represent numerical signal data (measured data)

Segment signal   A signal ⧉ consisting of one or more segment ⧉s.

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

Signal

- Representation of a variable ⧉ measured in a specific measurement raster ⧉.
- Generic term for segment signal ⧉s and operation signal ⧉s.

    A signal is part of a signal description set ⧉ which can be displayed and edited in the working area.

Signal description set   A group of one or more signals ⧉.
A signal description set and its signals can be edited in the working area by means of the Signal Editor ⧉. Each signal description set is stored as an STZ file ⧉ either in the **Signal Description Sets** folder or in the **Signal Generators** folder.

Signal Editor   A software component to create, configure, display, and manage signals ⧉ in signal description sets ⧉.

Signal file   A file that contains the wiring information of a simulator and that is part of the standard dSPACE documentation of dSPACE Simulator Full-Size. Normally, dSPACE generates this file when designing the simulator. Before using a failure simulation system, users can adapt the signal file to their needs.

Signal generator   An STZ file containing a signal description set ⧉ and optional information about the signal mapping ⧉, the description of variables, and the real-time platform.
The file is located in the **Signal Generators** folder and used to generate, download, and control Real-Time Testing sequences, which are executed on the real-time platform to stimulate ⧉ model variables in real time.

Signal Mapping   A controlbar ⧉ of the Signal Editor ⧉ to map model variables to signals ⧉ and variable aliases ⧉ of a signal generator ⧉.

Signal Selector   A controlbar ⧉ of the Signal Editor ⧉. The Signal Selector provides signals ⧉ and segments ⧉ for arranging and configuring signal description sets ⧉ in the working area.

SIL testing   Abbreviation of *software-in-the-loop testing*.
Simulation and testing of individual software functions, complete virtual ECUs (V‑ECUs ⧉), or even V‑ECU networks on a local PC or highly parallel in the cloud independently of real-time constraints and real hardware.

Simulation application   The generic term for offline simulation application (OSA) ⧉ and real-time application ⧉.

Simulation system   A description of the composition of V-ECU models, environment models, real ECUs, and their interconnections required for simulating the behavior of a system. A simulation system is the basis for the generation of a simulation application ⧉ for a given simulator platform.

Simulation time group   Group of platforms/devices in an experiment whose simulation times are synchronized with each other. If resynchronization ⧉ is enabled, ControlDesk synchronizes a simulation time group as a whole, not the single members of the group individually.

Simulator   A system that imitates the characteristics or behaviors of a selected physical or abstract system.

Single-processor system   A system that is based on one dSPACE processor or controller board.

Single-shot instrument   An instrument ⧉ that displays an instantaneous value of a connected variable without keeping a value history. In ControlDesk, all instruments except for a plotter ⧉ are single-shot instruments. For platforms ⧉ that support the variable observer ⧉ functionality, you can use single-shot instruments to observe variables.

Slave application   An application assigned to the slave DSP ⧉ of a controller or I/O board. It is usually loaded and started together with the real-time application ⧉ running on the corresponding main board.

Slave DSP   A DSP subsystem installed on a controller or I/O board. Its slave application ⧉ can be loaded together with the real-time application ⧉ or separately.

Slider   An instrument (or a value cell type of the Variable Array ⧉) for displaying and setting the value of the connected variable by means of a slide.

Sound Controller   An instrument for generating sounds to be played.

Standard axis   An axis with data points that are deposited in the ECU memory. Unlike a common axis ⧉, a standard axis is specified within a curve ⧉ or map ⧉. The parameters of a standard axis can be calibrated, which affects only the related curve or map.

Start trigger   A trigger ⧉ that is used, for example, to start a measurement raster ⧉. A platform trigger ⧉ can be used as a start trigger.

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

Steering Controller   An instrument for changing variable values using a game controller device such as a joystick or a steering wheel.

Stimulation   Writing signals to variables in real-time models during a simulation run.

Stop trigger   A trigger ⧉ that is used, for example, to stop a measurement raster ⧉.

String   A text variable in ASCII format.

Strings are represented by the ▦ symbol.

Struct   A variable with the struct data type. A struct 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 struct arrays as elements.

Structs are represented by the ⊞ symbol.

Struct array   An array of homogeneous struct ⓘ variables.

Struct arrays are represented by the ⊞ symbol.

STZ file   A ZIP file containing signal descriptions in the STI format. The STZ file can also contain additional MAT files to describe numerical signal data.

Sub data set   A data set that does not contain the complete set of the parameters of a platform/device.

Symbol   A symbolic name of a physical address in a MAP file. A symbol can be associated to a variable in the Variable Editor, for example, to support an address updates.

System variable   A type of variable that represents internal variables of the device or platform hardware and that can be used as measurement signals in ControlDesk to give feedback on the status of the related device or platform hardware. For example, an ECU's power supply status or the simulation state of a dSPACE board can be visualized via system variables.

T

Table Editor   An instrument for displaying and setting values of a connected curve, map, value block, or axis in a 2-D, 3-D, and grid view. The Table Editor can also display the values of a measurement array.

The Table Editor can be used for the following variable types:

- Common axis ⓘ (⊞)
- Curve ⓘ (⊞)
- Map ⓘ (⊞)
- Measurement array ⓘ (⊞)
- Value block ⓘ (⊞)

Time cursor   A cursor which is visible at the same time position in the following instruments:

- In all Time Plotters ⓘ
- In all XY Plotters ⓘ
- In all bus monitoring lists ⓘ

You can use the time cursor to view signal values at a specific point in time. If you move the time cursor, all measured signals and the respective parameters are

updated. Instruments and bus monitoring lists display the values that are available at the selected time position.

Time Plotter   A plotter instrument ⓘ for displaying signals that are measured in a time-based raster (time plots).

Topology   A description of the processor boards belonging to a multiprocessor system and their interconnections via Gigalinks. The topology also contains information on which Gigalink port of each processor board is connected to the Gigalink ports of other processor boards in the multiprocessor system.

Topology information is contained in the real-time application (PPC/x86/RTA) files of the multiprocessor system's processor boards.

TRC file   A variable description file with information on the variables available in an environment model ⓘ running on a dSPACE platform ⓘ.

Trigger   A condition for executing an action such as starting and stopping a measurement raster ⓘ or a recorder ⓘ.

The generic term for the following trigger types:

- Duration trigger ⓘ
- Platform trigger ⓘ
- Sample count trigger ⓘ

Trigger condition   A formula that specifies the condition of a trigger ⓘ mathematically.

Triggered measurement   The measurement of a measurement raster ⓘ started by a platform trigger ⓘ. The data flow between the dSPACE real-time hardware or VEOS and the host PC is not continuous.

U

Unassigned data set   A data set that is assigned neither to the working page nor to the reference page of a platform/device. An unassigned data set can be defined as the new working or reference data set. It then replaces the "old" working or reference data set and is written to the corresponding memory page, if one is available on the platform/device.

Unplugged   A platform/device state defined by the following characteristics:

- The logical connection between ControlDesk and the hardware was interrupted, for example, because the ignition was turned off or the ControlDesk PC and the hardware were disconnected.
- Before the state of a platform/device changes to 'unplugged', the platform/device was in one of the following states:
  - 'Connected'
  - 'Online calibration started'
  - 'Measuring' / 'Recording'

> **Tip**
>
> A device for which the connection between ControlDesk and the device hardware currently is interrupted is also set to the 'unplugged' state when you start online calibration if both the following conditions are fulfilled:
> - The device's Start unplugged property is enabled.
> - The Start online calibration behavior property is set to 'Ignore differences'.
>
> This is possible for CCP and XCP devices. For details on the two properties listed above, refer to General Settings Properties (ControlDesk Platform Management 📖).

- If the Automatic Reconnect feature is enabled for a platform/device and if the platform/device is in the 'unplugged' state, ControlDesk periodically tries to re-establish the logical connection for that platform/device.
- Online calibration is impossible. Offline calibration is possible.
- Platform/device configuration is possible.

The 'unplugged' platform/device state is indicated by the ⚠ icon.

Untriggered measurement   The measurement of a measurement raster ⓘ not started by a platform trigger ⓘ. The data flow between the dSPACE real-time hardware or VEOS and the host PC is continuous.

User function   An external function or program that is added to the ControlDesk user interface for quick and easy access during work with ControlDesk.

User Functions Output   A controlbar ⓘ that provides access to the output of external tools added to the Automation ribbon.

V

Value block   A parameter ⓘ that consists of a 1- or 2-dimensional array of scalar parameters ⓘ.

In variable lists, ControlDesk displays entries for the value block itself and for each array element.

Value blocks are represented by the ⊞ symbol.

Value conversion   The conversion of the original *source values* of variables of an application running on an ECU or dSPACE real-time hardware into the corresponding scaled *converted values*.

Variable   Any parameter ⓘ or measurement variable ⓘ defined in a variable description ⓘ. ControlDesk provides various instrument ⓘs to visualize variables.

Variable alias   An alias name that lets the user control the property of a segment ⓘ by a model parameter of a real-time application.

Variable Array   An instrument for calibrating parameters and displaying measurement variable values.

The Variable Array can be used for the following variable types:

- Measurement ⓘ (🖃)
- Measurement array ⓘ (🖃)
- String ⓘ (🆎)
- Struct ⓘ (🔳)
- Struct array ⓘ (🔳)
- Value ⓘ (P)
- Value block ⓘ (Ⅲ)

Variable connection   The connection of a variable ⓘ to an instrument ⓘ. Via the variable connection, data is exchanged between a variable and the instrument used to measure or calibrate the variable. In other words, variable connections are required to visualize variables in instrument.

Variable description   A file describing the variables in a simulation application, which are available for measurement, calibration, and stimulation.

Variable Editor   A tool for viewing, editing, and creating variable descriptions in the ASAM MCD-2MC (A2L) file format. The Variable Editor allows you to create A2L files from scratch, or to import existing A2L files for modification.

Variable Filter   A variable filter contains the filter configuration of a combined filter, which is used to filter the variable list in the **Variables** controlbar using a combination of filter conditions.

Variables controlbar   A controlbar ⓘ that provides access to the variables of the currently open experiment.

V-ECU   Abbreviation of *virtual ECU*.

ECU software that can be executed in a software-in-the-loop (SIL) testing ⓘ environment such as a local PC or highly parallel in the cloud independently of real-time constraints and real ECU hardware.

Vehicle information   The ODX database ⓘ can contain information for one or more vehicles. Vehicle information data is used for vehicle identification purposes and for access to vehicles. It references the access paths (logical links) to the ECUs.

VEOS   A simulator ⓘ which is part of the PC and allows the user to run an offline simulation application (OSA) ⓘ without relation to real time.

VEOS Player is the graphical user interface for VEOS.

VEOS platform   A platform that configures and controls the offline simulation application (OSA) ⓘ running in VEOS ⓘ and that also provides access to the application's environment VPU ⓘ.

VEOS Player   An application running on the host PC for editing, configuring and controlling an offline simulation application (OSA) ⓘ running in VEOS.

Verbal conversion   A conversion ⧉ in which a conversion table ⧉ is used to specify the computation of numerical values into strings. The verbal conversion table is used when you switch the value representation from source to converted mode and vice versa.

Verbal conversion range   A conversion ⧉ in which a conversion table ⧉ is used to specify the computation of a range of numerical values into strings. The verbal conversion range table is used when you switch the value representation from source to converted mode and vice versa.

View set   A named configuration of the controlbar ⧉s of ControlDesk. A view set has a default state and a current state that can differ from the default state. The configuration includes the geometry, visibility, and docking or floating state of controlbars.

Visualization   The representation of variable ⧉s in instrument ⧉s:

- Measurement variable ⧉s are visualized in instruments to view and analyze their time traces.
- Calibration parameters ⧉ are visualized in instruments to change their values.

VPU   Abbreviation of *virtual processing unit*. A VPU is part of an offline simulation application in VEOS. Each VPU runs in a separate process of the PC.

VPU is also the generic term for:

- V-ECUs
- Environment VPUs
- Controller VPUs
- Bus VPUs

W

Working data set   The data set currently residing in the memory of a platform/device hardware. There can be only one working data set for each calibration platform/device. The working data set is read/write.

Working page   Memory area containing the parameters of an ECU or prototyping hardware (memory page ⧉). The working page contains the read/write working data set ⧉.

If the platform/device also provides a reference page ⧉, ControlDesk lets you switch between both pages.

Writable measurement   A scalar variable that can be measured and calibrated.

X

XCP   Abbreviation of *Universal Measurement and Calibration Protocol*. A protocol that is implemented on electronic control units (ECUs) and provides access to ECUs with measurement and calibration systems (MCS) such as ControlDesk.

XCP is based on the *master-slave principle*:

- The ECU is the slave.
- The measurement and calibration system is the master.

The "X" stands for the physical layers for communication between the ECU and the MCS, such as CAN (Controller Area Network) and Ethernet.

The basic features of XCP are:

- ECU parameter calibration (CAL)
- Synchronous data acquisition (DAQ)
- Synchronous data stimulation (STIM), i.e., for bypassing
- ECU flash programming (PGM)

The XCP protocol was developed by ASAM e.V. (Association for Standardisation of Automation and Measuring Systems e.V.). For the protocol specification, refer to http://www.asam.net.

The following ControlDesk devices support ECUs with an integrated XCP service:

- XCP on CAN device ⬚
- XCP on Ethernet device ⬚

XCP on CAN device   A device that provides access to an ECU with XCP connected to the ControlDesk PC via CAN. Using the XCP on CAN device, you can access the ECU for measurement and calibration purposes via XCP (*Universal Measurement and Calibration Protocol*).

XCP on Ethernet device   A device that provides access to an ECU or V-ECU ⬚ with XCP connected to the ControlDesk PC via Ethernet. The XCP on Ethernet device provides access to the ECU/V-ECU via XCP (*Universal Measurement and Calibration Protocol*) for measurement and calibration purposes.

XIL API EESPort   Electrical Error Simulation port (EESPort) ⬚

XIL API MAPort platform   A platform that provides access to a simulation platform via the ASAM XIL API implementation that is installed on your host PC.

XY Plotter   A plotter instrument ⬚ for displaying signals as functions of other signals.