DS100x, DS110x, MicroAutoBox II, MicroLabBox

Software Getting Started

Release 2021-A – May 2021

dSPACE

How to Contact dSPACE

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

How to Contact dSPACE Support

If you encounter a problem when using dSPACE products, contact your local dSPACE representative:

- Local dSPACE companies and distributors: http://www.dspace.com/go/locations
- For countries not listed, contact dSPACE GmbH in Paderborn, Germany.
  Tel.: +49 5251 1638-941 or e-mail: support@dspace.de

You can also use the support request form: http://www.dspace.com/go/supportrequest. If you are logged on to mydSPACE, you are automatically identified and do not need to add your contact details manually.

If possible, always provide the relevant dSPACE License ID or the serial number of the CmContainer in your support request.

Software Updates and Patches

dSPACE strongly recommends that you download and install the most recent patches for your current dSPACE installation. Visit http://www.dspace.com/go/patches for software updates and patches.

Important Notice

Contents

About This Document

Content

This document shows how to get started with the following dSPACE systems in connection with ControlDesk:

- DS1006
- DS1007
- DS1104
- MicroAutoBox II
- MicroLabBox

Note

- This document does not show the hardware installation of the dSPACE systems listed above: Your specific dSPACE system must already be installed before you can perform the steps described in this document. For more information, refer to the installation and configuration documentation of your system.
- This document does not show the initial working steps for the following dSPACE systems:
  - SCALEXIO

    For information on the initial working steps with SCALEXIO, refer to ConfigurationDesk Getting Started 🕮 and SCALEXIO – Hardware and Software Overview 🕮.
  - MicroAutoBox III

    For information on the initial working steps with MicroAutoBox III, refer to MicroAutoBox III - Getting Started 🕮.

Target audience

This document is intended for users who have no experience with dSPACE systems. Knowledge in handling computer hardware and Microsoft Windows operating systems is assumed.

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.

Examples:

- Where you find terms such as `rti<XXXX>` replace them by the RTI platform support you are using, for example, `rti1007`.
- Where you find terms such as `<model>` or `<submodel>` in this document, replace them by the actual name of your model or submodel. For example, if the name of your Simulink model is `smd_1007_sl.slx` and you are asked to edit the `<model>_usr.c` file, you actually have to edit the `smd_1007_sl_usr.c` file.

RTI block name conventions All I/O blocks have default names based on dSPACE's board naming conventions:

- Most RTI block names start with the board name.
- A short description of functionality is added.
- Most RTI block names also have a suffix.

Suffix	Meaning
B	Board number (for PHS-bus-based systems)
M	Module number (for MicroAutoBox II)

Suffix	Meaning
C	Channel number
G	Group number
CON	Converter number
BL	Block number
P	Port number
I	Interrupt number

A suffix is followed by the appropriate number. For example, DS2201IN_B2_C14 represents a digital input block located on a DS2201 board. The suffix indicates board number 2 and channel number 14 of the block. For more general block naming, the numbers are replaced by variables (for example, DS2201IN_Bx_Cy).

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.

Quick Start for Working with a dSPACE System

Introduction

After checking the preconditions, you can start work on typical use-cases.

Where to go from here

Information in this section

Required Software and Hardware Installation

Introduction

The dSPACE system must be installed before you can perform the steps described in this document.

Required software installation

To perform the first work steps, the installation of the following software is required:

- The MATLAB software must be installed with at least the following components:

- Simulink
- Simulink® Coder™
- Necessary toolboxes, refer to
  http://www.dspace.com/goto?toolboxes

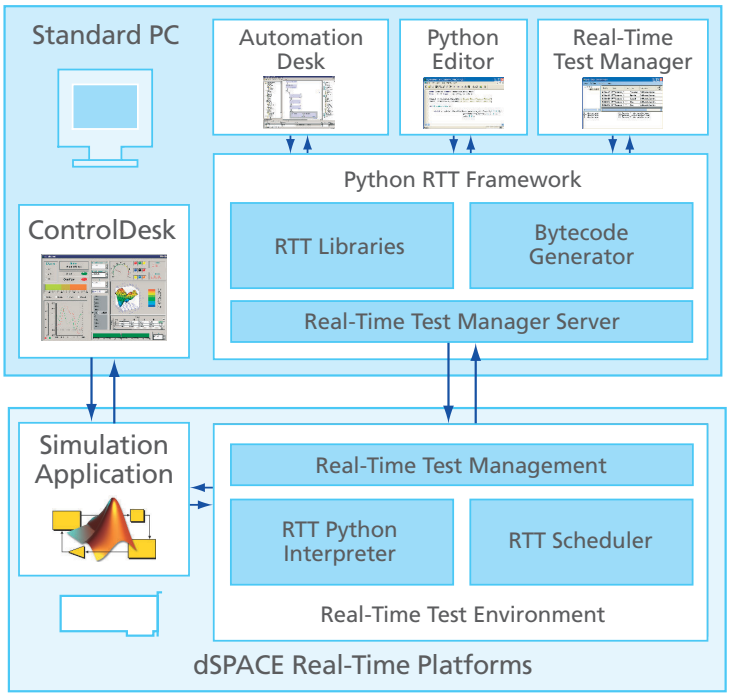
  The MATLAB version must match the available dSPACE Release. For compatibility information, refer to http://www.dspace.com/goto?compatibility.
- The following dSPACE software must be installed:
  - RTI (Real-Time Interface)
  - A C code cross compiler
  - ControlDesk

  Refer to Installing dSPACE Software 🕮.

Required hardware installation

To perform the first work steps, dSPACE hardware must be installed:

- dSPACE hardware is typically installed in an expansion box outside the PC. The expansion box must be connected to the host PC either via bus interface (using dSPACE's link boards) or via Ethernet.



- A DS1104 Controller Board is installed directly in the PC (PCI slot).
- MicroAutoBox II and MicroLabBox are connected to the host PC via Ethernet.

dSPACE boards and their installations The table below lists the different dSPACE boards and their installations:

dSPACE Board	Installation/Connection	Refer to
DS1006	The DS1006 is installed in an expansion box[1].	▪ DS1006 Hardware Installation and Configuration Guide 🕮 ▪ PHS Bus System Hardware Reference 🕮
DS1007	The DS1007 is installed in an expansion box connected to the host PC via Ethernet.	▪ DS1007 Hardware Installation and Configuration Guide 🕮 ▪ PHS Bus System Hardware Reference 🕮
DS1104	The DS1104 is installed in the host PC.	DS1104 Hardware Installation and Configuration 🕮
MicroAutoBox II	MicroAutoBox II is connected to the host PC via Ethernet.	MicroAutoBox II Hardware Installation and Configuration Guide 🕮
MicroLabBox	MicroLabBox is connected to the host PC via Ethernet.	MicroLabBox Hardware Installation and Configuration 🕮

[1] An expansion box can be connected to the host PC via bus interface or Ethernet.

How to Create a Real-Time Application

Objective

First you generate a real-time application for your dSPACE system.

Workflow

To generate a real-time application, you have to perform the following steps:

Step	Description
Step 1: Creating a Simulink model	You have to create a Simulink model using MATLAB and Simulink. Instead of programming C code manually, you implement the algorithm graphically using Simulink blocks. The models are saved as SLX files.
Step 2: Specifying RTI I/O interfaces	To connect the simulation model to the physical world, you need to introduce I/O interfaces into the model. These allow you to replace parts of your simulated model with real hardware. dSPACE's RTI (Real-Time Interface) blocks provide I/O interfaces for accessing dSPACE hardware.
Step 3: Generating C code	You can build the model created with Simulink and RTI blocks using the Simulink® Coder™. The Simulink® Coder™ generates C code from the model automatically.
Step 4: Compiling and linking the real-time application	The cross compiler environment compiles the generated C code and links the object files and libraries into an executable application for the real-time processor.

Precondition

Before you can start the build and download procedure, you have to ensure that your dSPACE hardware is registered correctly in ControlDesk. Refer to How to Register a dSPACE System on page 26.

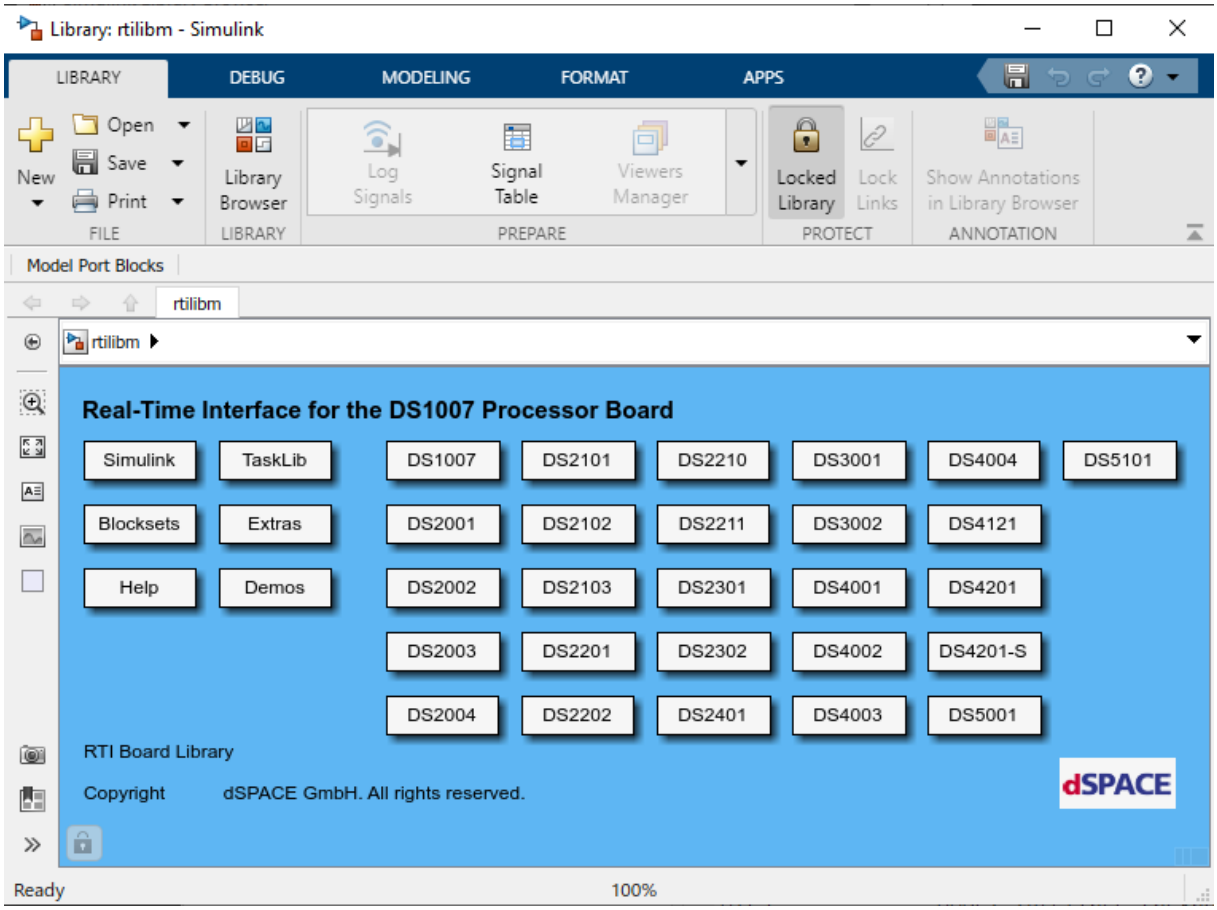
Method

To generate a real-time application

1 Start MATLAB.

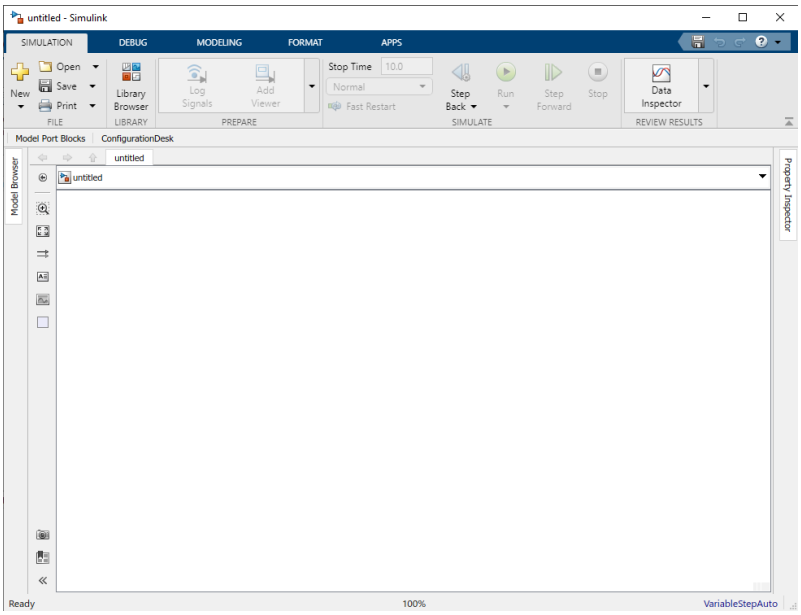
2 In the MATLAB Command Window, enter `rti<Name of the Library>`, for example, `rti1007`.

The corresponding RTI library opens.



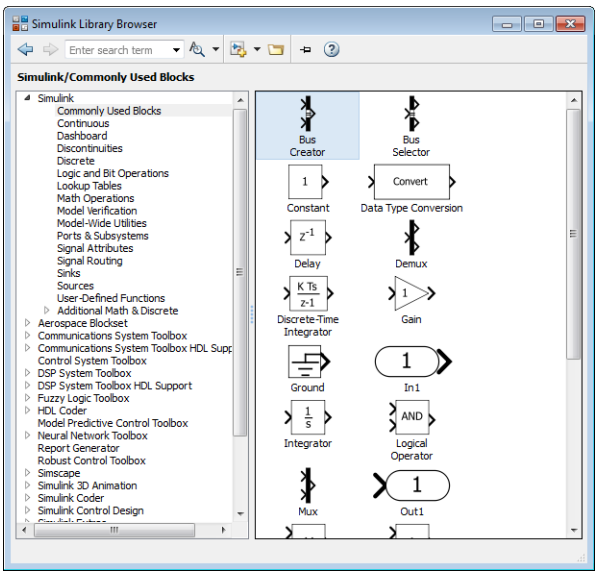
3 From the LIBRARY ribbon, select FILE – New – Model.

A new model opens.

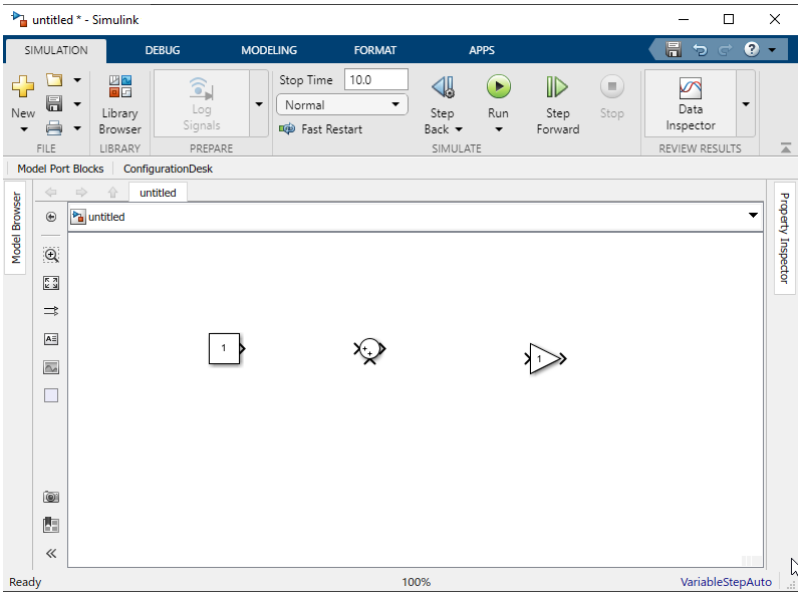


4 From the SIMULATION ribbon in the model, select LIBRARY – Library Browser.

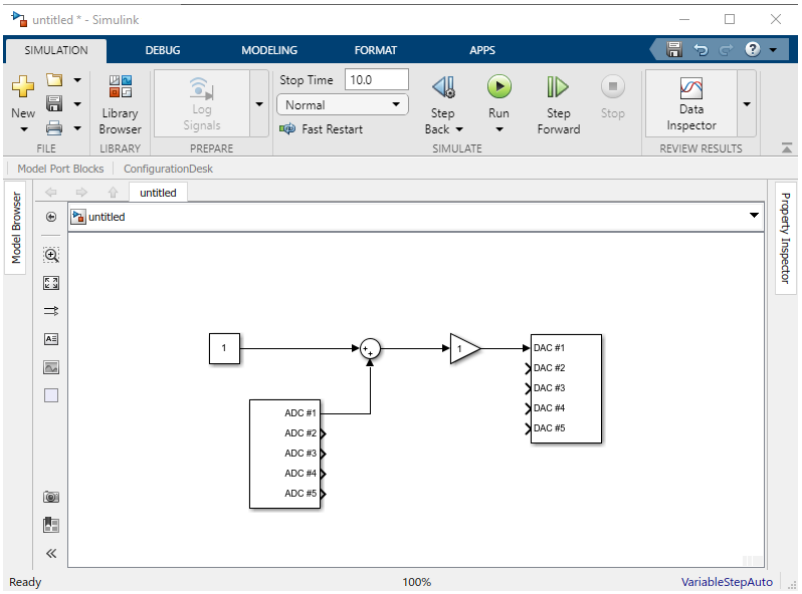
The Simulink Library Browser opens.



5 Add blocks from the **Simulink Library Browser** to the model via drag & drop.



6 Add RTI blocks from the RTI library to the model via drag & drop, and connect the blocks as shown in the following illustration.



7 Save the model as an SLX file.

8 Press **Ctrl** + **B**.

The Simulink® Coder™ generates C code for the model. The RTI build process compiles the generated C code, links the object files and libraries into an executable application and downloads the application to the real-time hardware directly after the compilation (build). The build status is displayed in the MATLAB Command Window.

Result

You have generated a real-time application consisting of the following files:

File	Description
MAP	Map file with address information of variables
PPC, x86 or RTA	Real-time application to be downloaded to the dSPACE real-time hardware
SDF	System description file to be used by ControlDesk. It contains references to the PPC/x86/RTA, MAP, and TRC file.
TRC	Variable description file
TRZ	If real-time testing is enabled for the generated real-time application, the build process generates a TRZ file containing the MAP file and the TRC file.

Next step

You can now experiment with the real-time application with ControlDesk. Refer to How to Experiment with a Real-Time Application in ControlDesk on page 15.

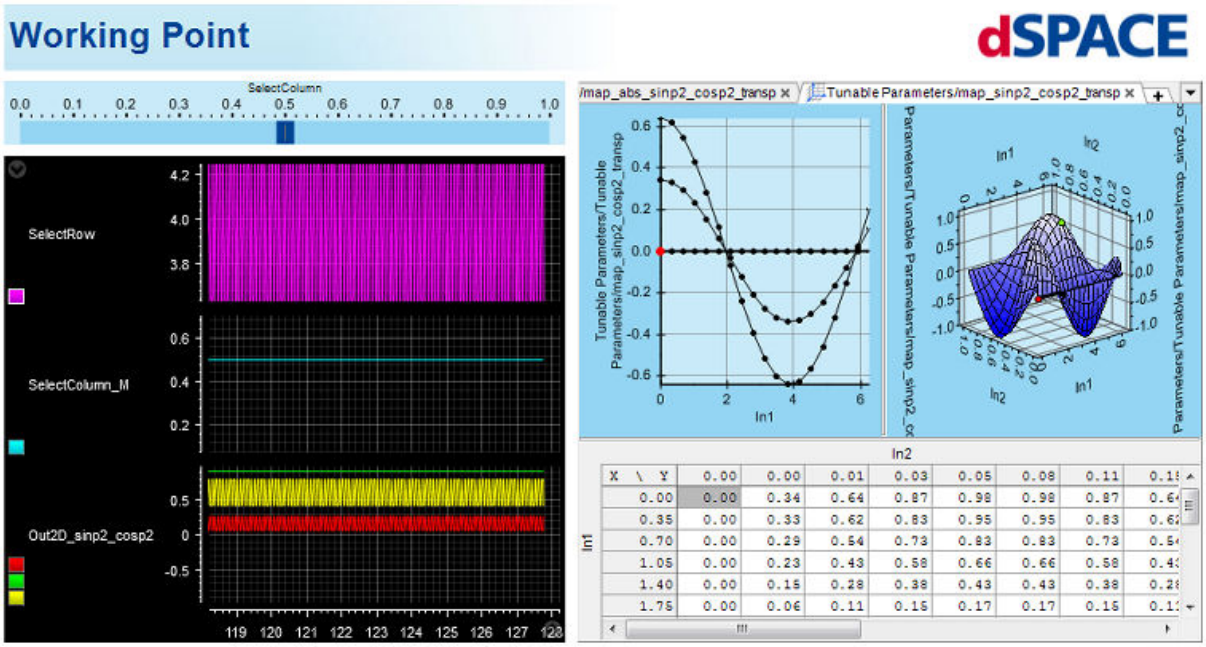
How to Experiment with a Real-Time Application in ControlDesk

Objective

To experiment with the real-time application, you can open a platform-specific ControlDesk demo.

Experimenting with a real-time application

The Real-Time application demos provided by ControlDesk let you quickly start experimenting with a real-time application. They also let you familiarize yourself with the ControlDesk instruments for measuring signals and changing parameter values.



Workflow

To experiment with a real-time application in ControlDesk, perform the following steps:

1. Register your dSPACE real-time hardware as shown in Part 1.

   The registration of a DS1007 is shown as an example. The registration process for other platforms may differ. In this case, perform the steps described in How to Register a dSPACE System on page 26.
2. Open a demo and experiment with the real-time application as shown in Part 2.

Part 1

To register a DS1007

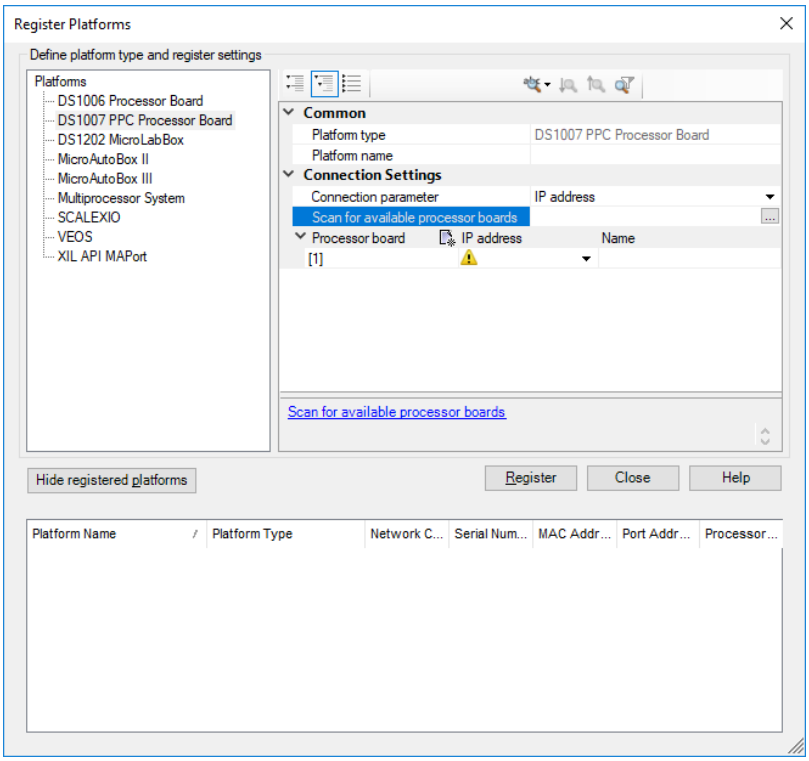
1 Start ControlDesk.

2 On the Platform ribbon, click Register Platform.

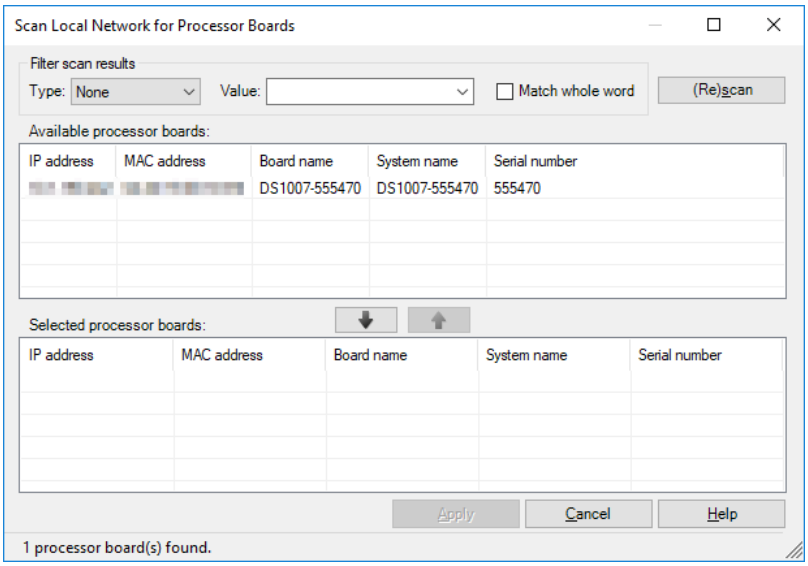
The Register Platforms dialog opens.

3 In the Register Platforms dialog, select DS1007 PPC Processor Board.

4 Click Scan for available processor boards as shown in the following illustration.



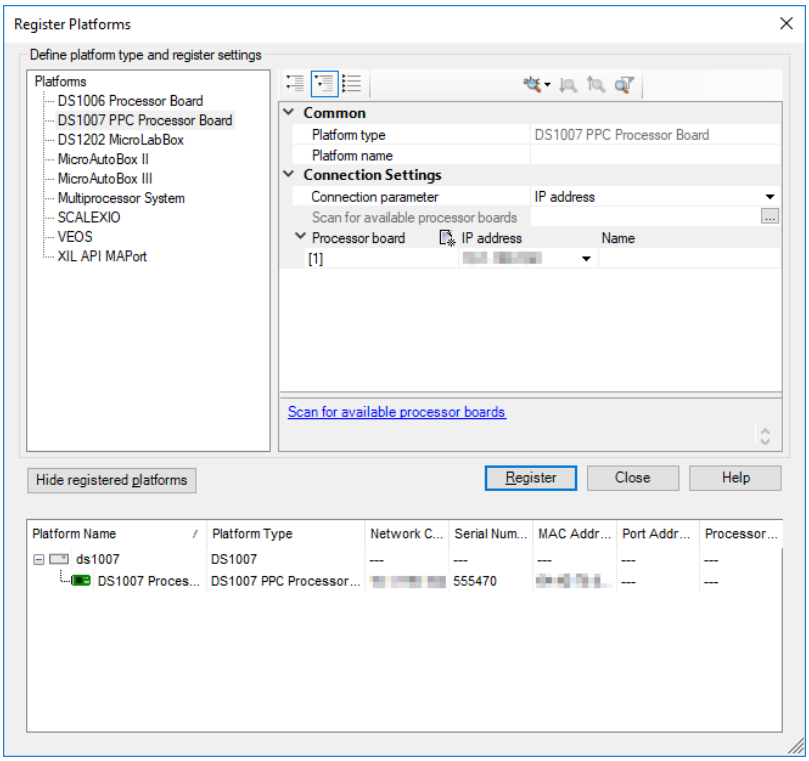
The Scan Local Network for Processor Boards dialog opens.



5 In the Scan Local Network for Processor Boards dialog, select the DS1007, and click the down arrow  to move it to the Selected processor boards area.

Click Apply to return to the Register Platforms dialog.

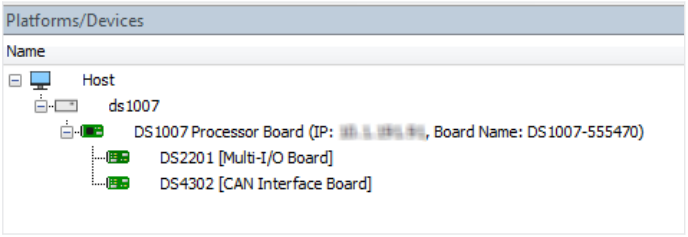
6 In the Register Platforms dialog, click Register. The DS1007 is now registered as shown in the following illustration.



Close the Register Platforms dialog.

Interim result

The following illustration shows the registered DS1007 in the Platforms/Devices controlbar:

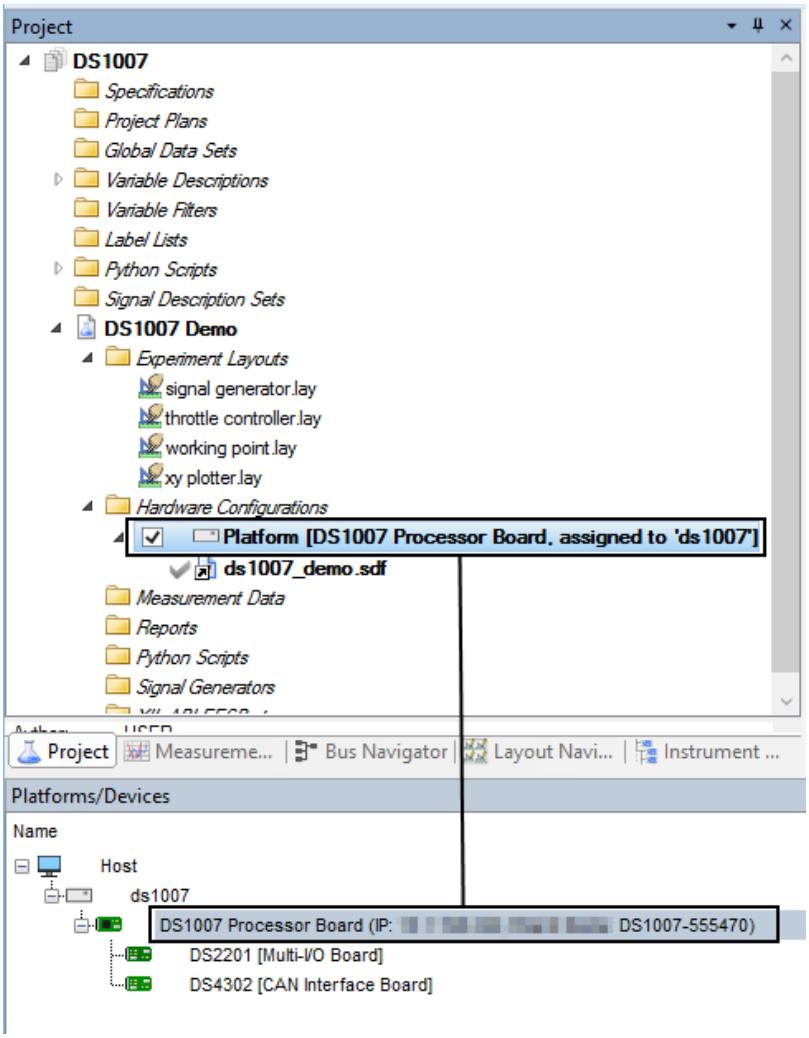


Part 2

To open a demo and experiment with the real-time application

1 On the File ribbon, click Open – Open Project + Experiment to open the DS1007 Demo experiment.

ControlDesk opens the experiment and automatically assigns the DS1007 registered in Part 1 to the DS1007 platform displayed in the Project Manager.

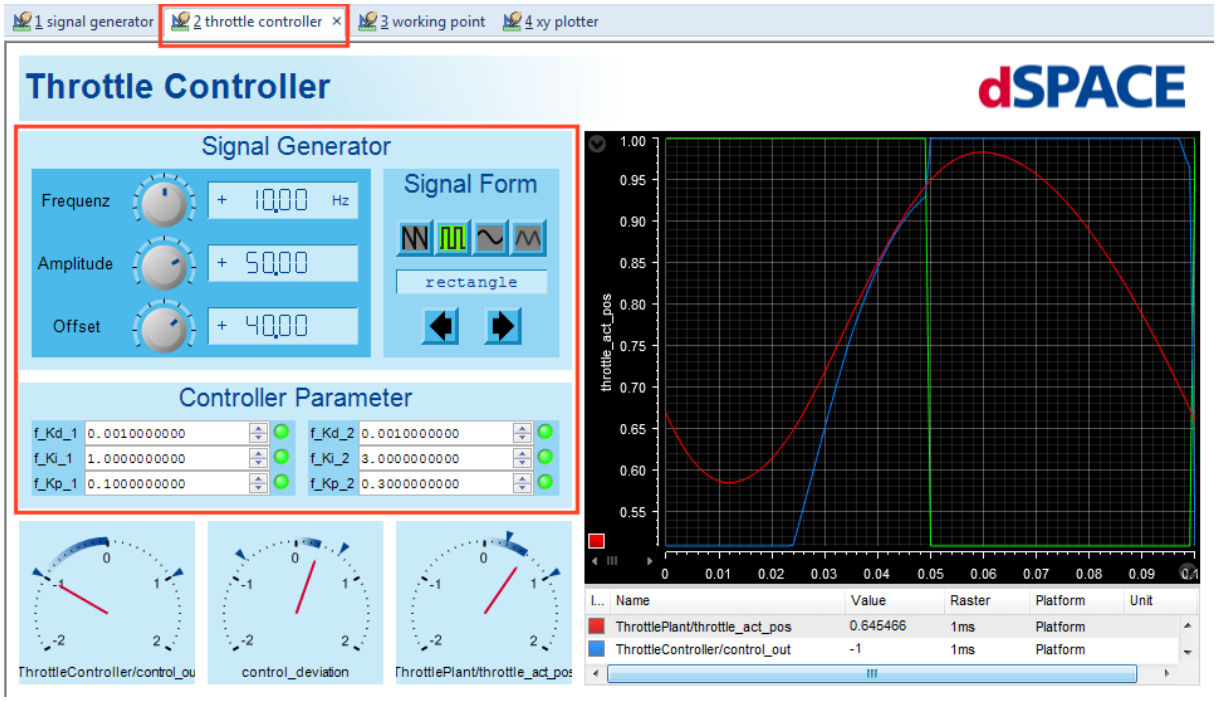


2 On the Home ribbon, click Status Control – Start Measuring.

ControlDesk automatically downloads the experiment's application to the hardware.

3 Change to the throttle controller layout.

4 Experiment with the real-time application, e.g., by changing parameter values on the layouts.



Result

You experimented with the real-time application in ControlDesk and finished the quick start using dSPACE hardware. If you need a detailed description of how to experiment with ControlDesk, refer to Experimenting with ControlDesk on page 57.

Related topics

Basics

Real-Time Application Demos (ControlDesk Introduction and Overview 🕮)

HowTos

How to Register a dSPACE System.....26

References

Open Project + Experiment (ControlDesk Project and Experiment Management 🕮)
Register Platforms (ControlDesk Platform Management 🕮)
Start Measuring (ControlDesk Measurement and Recording 🕮)

Detailed Description for Working with a dSPACE System

Introduction

If the quick start instructions do not match your requirements, this document also provides a detailed description for working with your dSPACE system.

Where to go from here

Information in this section

Accessing the Hardware with ControlDesk

Introduction

You can manage dSPACE real-time hardware with the ControlDesk Platforms/Devices controlbar.

Where to go from here

Information in this section

Basic Steps When Accessing the Hardware

Introduction

An overview of the workflow and an introduction to the graphical user interface of ControlDesk help you to access the hardware.

Where to go from here

Information in this section

Workflow for Accessing the Hardware

Workflow steps

To access dSPACE hardware, you can use ControlDesk, which lets you experiment with dSPACE systems. For example, you can use it to register dSPACE boards, download applications and manage experiments.

The first time you access dSPACE real-time hardware, you have to perform the following workflow steps:

1. Starting ControlDesk

   ControlDesk is installed on the host PC during software installation. Refer to How to Start ControlDesk on page 23.
2. Registering dSPACE boards

   ControlDesk provides the **Platforms/Devices** controlbar that lets you register dSPACE real-time hardware. Refer to How to Register a dSPACE System on page 26.
3. Checking the configuration of dSPACE boards

   When the dSPACE board is registered you can check the board configuration. Refer to Checking the Configuration of the dSPACE Boards on page 30.
4. Updating the firmware

   Your board usually contains the latest firmware. If you install a new dSPACE Release it may contain newer firmware. In this case the firmware should be updated. Refer to Updating the Firmware on page 32.

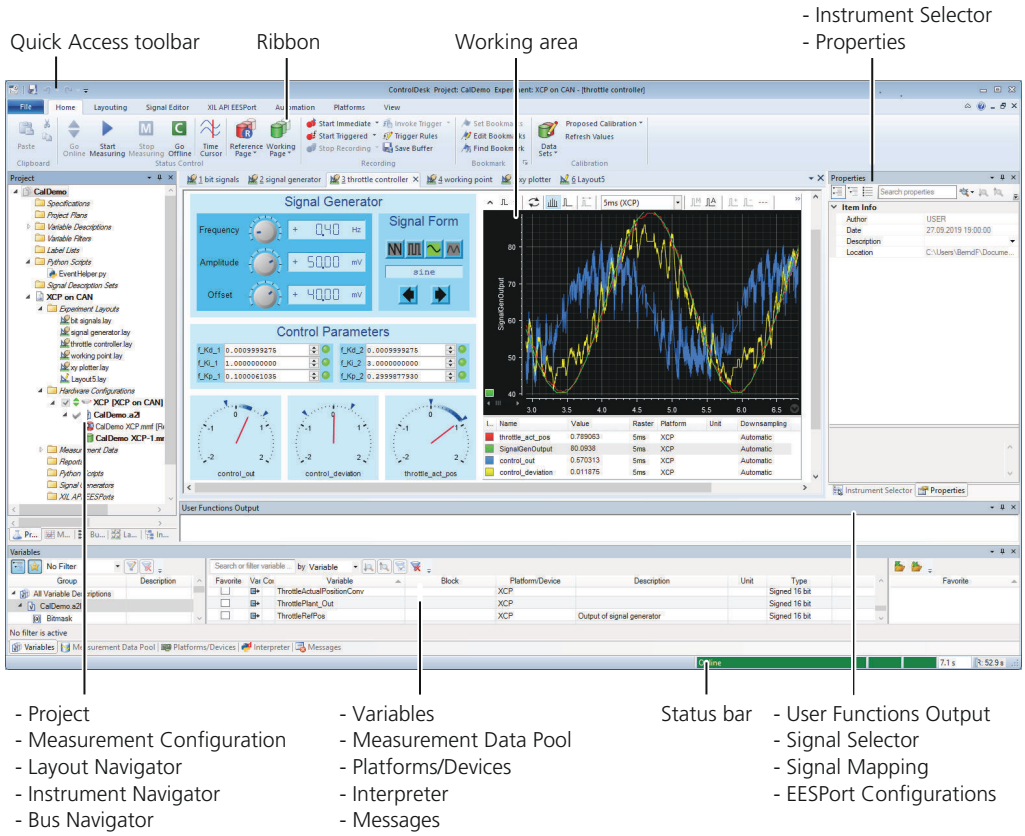
How to Start ControlDesk

Objective

After installation of the dSPACE software, ControlDesk is available in the Windows **Start** menu and as an icon on the desktop.

User interface

The following illustration shows the user interface of ControlDesk.

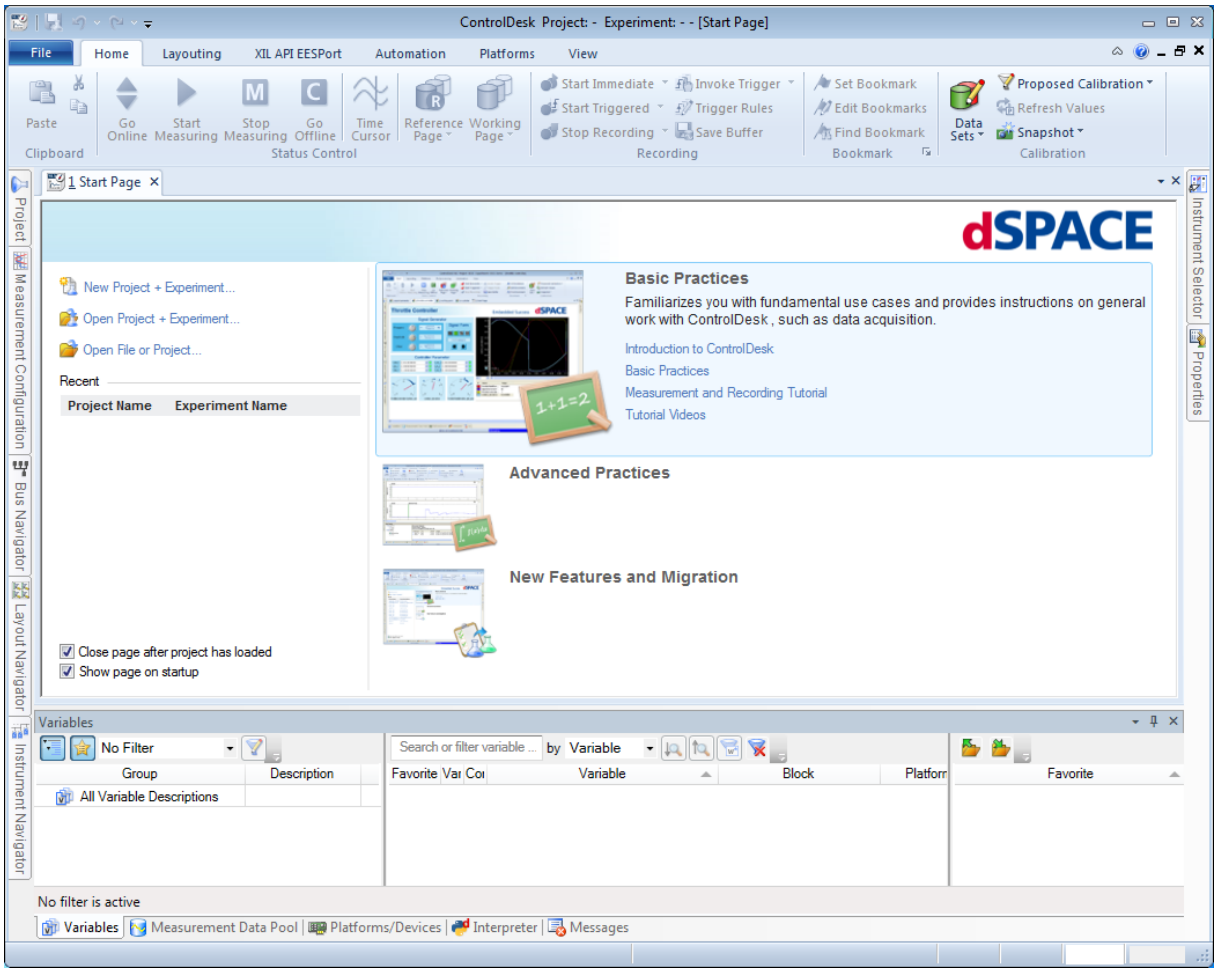


Method

To start ControlDesk

1 In the Windows Start menu, select Programs – dSPACE ControlDesk 7.4.

ControlDesk starts, see the following illustration.



Result

ControlDesk is ready to register your dSPACE board.

Related topics

Basics

Introduction to ControlDesk (ControlDesk Introduction and Overview 🕮)

How to Register a dSPACE System

Objective

After installing dSPACE real-time hardware, you have to make it known to ControlDesk.

Tip

You do not need to register the DS1104 since it supports the plug & play feature. The DS1104 is registered automatically.

Preconditions

- To register a platform to access dSPACE real-time hardware, the hardware must be connected to the host PC.
- MicroAutoBox II: Before you register MicroAutoBox II, you should configure it using the `DS1401ConfigGUI.exe` utility, for example, to change MicroAutoBox II's default IP address. The utility is located in `<RCP_HIL_InstallationPath>\Exe`. For configuration details, refer to Connecting the MicroAutoBox II to the Host PC via Ethernet (MicroAutoBox II Hardware Installation and Configuration Guide 🕮).

Tip

You can add the `DS1401ConfigGUI.exe` tool as a user function. Refer to How to Add External Programs or Scripts as User Functions to RTI (ControlDesk Customization 🕮).

Method

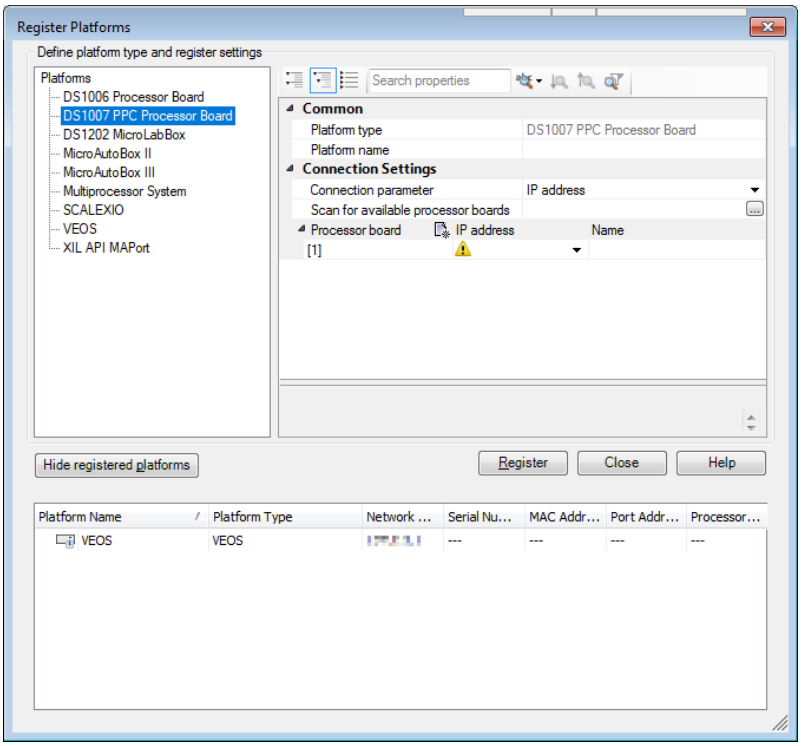
To register a dSPACE system

1. On the Platforms ribbon, click Platform Management - Register Platforms.

   The Register Platforms dialog opens.

2. From the Platforms list, select the type of the platform you want to register.

The registration settings vary according to the selected platform type. The following illustration shows an example for the DS1007 platform type.



3 Specify the registration settings for the dSPACE hardware you want to register. The number and kinds of properties depend on the selected platform type.

Property	Description / Refer to
Common Properties	
Multiprocessor type	Common Properties (ControlDesk Platform Management 🕮)
Platform name	Lets you specify a unique name for the selected platform. After registration, the name is displayed in the ControlDesk Platforms/Devices controlbar. The valid characters are 'a ... z', 'A ... Z', '0 ... 9', '_', '-' and ' '. The name must not start or end with an underline, hyphen or blank. If you do not specify a platform name, ControlDesk displays a default name in the Platforms/Devices controlbar.
Platform type	Common Properties (ControlDesk Platform Management 🕮)
Topology check	General Settings Properties (ControlDesk Platform Management 🕮)
Connection Settings Properties	
Alias name	Assignment Properties (ControlDesk Platform Management 🕮)
Board name	Assignment Properties (ControlDesk Platform Management 🕮)
Connection parameter	Lets you select one of the following connection parameters for registration: ▪ Alias name ▪ Board name ▪ IP address ▪ MAC address

Property	Description / Refer to
	The selected parameter is used to register the member processing units/processor boards (SCALEXIO), member processor boards (DS1007), member platforms (DS1202 MicroLabBox, MicroAutoBox III), or simulator (VEOS).
Connection type	Assignment Properties (ControlDesk Platform Management 🕮)
IP address	Assignment Properties (ControlDesk Platform Management 🕮)
MAC address	Assignment Properties (ControlDesk Platform Management 🕮)
Network client	Assignment Properties (ControlDesk Platform Management 🕮)
Platform	Lets you specify the platform belonging to the DS1202 MicroLabBox or MicroAutoBox III platform, or the simulator (local host PC or remote PC) for the VEOS platform. The platform has an edit field to specify its connection parameter value.
Port address	Assignment Properties (ControlDesk Platform Management 🕮)
Processor board	Lets you specify the processor boards belonging to the DS1007 platform. Click  to add a processor board, or click to delete the selected processor board. **Note** You cannot subsequently add processor boards to a DS1007 system that is already registered. Each processor board has an edit field to specify its connection parameter value and an edit field to specify a unique name for it. After registration, the name is displayed in the ControlDesk **Platforms/Devices** controlbar. The valid characters are 'a ... z', 'A ... Z', '0 ... 9', '_', '-' and ' '. The name must not start or end with an underline, hyphen or blank. If you do not specify a custom name for a processor board, ControlDesk displays a default name for it in the **Platforms/Devices** controlbar.
Processing units	Lets you specify the processing units or processor boards belonging to the SCALEXIO platform. Click  to add a processing unit or processor board, or click to delete the selected processing unit or processor board. **Note** You cannot subsequently add processing units or processor boards to a SCALEXIO system that is already registered. Each processing unit or processor board has an edit field to specify its connection parameter value and an edit field to specify a unique name for it. After registration, the name is displayed in the ControlDesk **Platforms/Devices** controlbar. The valid characters are 'a ... z', 'A ... Z', '0 ... 9', '_', '-' and ' '. The name must not start or end with an underline, hyphen or blank. If you do not specify a custom name for a processing unit or processor board, ControlDesk displays a default name for it in the **Platforms/Devices** controlbar.
Scan for available processor boards/ processing units/ platforms	Lets you scan the local network for connected platform hardware or simulators (VEOS). Depending on the platform type to be registered, ControlDesk opens the **Scan Local Network for Processor Boards**, the **Scan Local Network for Processing Units** or the **Scan Local Network for Platforms** dialog and displays all the platform hardware found in the network. Refer to Scan Local Network for Processor Boards/ Processing Units /Platforms dialog (ControlDesk Platform Management 🕮).

Property	Description / Refer to
MAPort Implementation Properties	
MAPort Implementation	Lets you select the MAPort implementation for the XIL API MAPort platform. The list displays all the supported XIL API MAPort implementations. The XIL API MAPort platform analyzes the IMF files in the standard XIL API folder (`ProgramData\ASAM\XIL\Implementation`) to determine the installed XIL API MAPort implementations.
Product name	MAPort Implementation Properties (ControlDesk Platform Management 🕮)
Product version	MAPort Implementation Properties (ControlDesk Platform Management 🕮)
Vendor name	MAPort Implementation Properties (ControlDesk Platform Management 🕮)
XIL API version	MAPort Implementation Properties (ControlDesk Platform Management 🕮)
Multiprocessor Configuration Properties	
Network client	Assignment Properties (ControlDesk Platform Management 🕮)
Processors	Lets you specify the number of processors belonging to the multiprocessor system. Click to add a processor, or click to delete the selected processor. **Note** You cannot subsequently add members to a multiprocessor system that is already registered.
Processor name	Common Properties (ControlDesk Platform Management 🕮)
Port address	Assignment Properties (ControlDesk Platform Management 🕮)

4 Click **Register** to complete the registration.

The registered platform is displayed with its registration settings in the **Registered platforms** list.

Tip

If you register a platform with corrupted boot firmware or with hardware components containing different firmware versions, a message box containing warning messages about the detected firmware problem is displayed. The warning messages are also displayed in the **Message Viewer**. In the **Platforms/Devices** controlbar, the affected hardware components are marked with the ⚠ symbol. Its tooltip also provides information on the detected firmware problem. You should check the entries and perform firmware updates, if necessary. Refer to Update Firmware (ControlDesk Platform Management 🕮).

5 Click **Close** to close the **Register Platforms** dialog.

Result

You have registered a dSPACE system.

Related topics

References

Register Platforms (ControlDesk Platform Management 🕮)

Checking the Configuration of the dSPACE Boards

Introduction

After registration, you can check the properties of the dSPACE system with ControlDesk.

Where to go from here

Information in this section

How to Check the Configuration

Objective

To check whether the system is ready to experiment with real-time applications.

Check application and related ControlDesk experiment

To check whether your dSPACE system works correctly, it is recommended that you load the system-specific *check application*. For each check application, there is a ZIP file with a backup of a ControlDesk experiment referencing the check application. To check the configuration of your specific system, you open the system-specific ZIP file in ControlDesk and load the contained real-time application to the system.

System	File	Location[1)]
DS1006	`smd_1006_ch.zip`	`\Demos\DS1006\Check\`
	`pipt1_dual1006_ch.zip`[2)]	`\Demos\DS1006mp\Dual1006\Check\`
DS1007	`smd_1007_ch.zip`	`\Demos\DS1007\Check\`
DS1104	`smd_1104_ch.zip`	`\Demos\DS1104\Check\`
MicroLabBox	`smd_1202_ch.zip`	`\Demos\DS1202\Check\`
MicroAutoBox II	`smd_1401_ch.zip`	`\Demos\DS1401\Check\`

1) Folder relative to the folder of the RCP & HIL installation
2) For a multiprocessor system consisting of two DS1006 boards.

Method

To check the configuration

1. Start ControlDesk.
2. On the File ribbon, click Open - Project + Experiment from Backup.
   ControlDesk opens a standard Open dialog.
3. In the Open dialog, specify the ZIP file that matches your system type and click Open. For the location of the ZIP file, see the table above (refer to Check application and related ControlDesk experiment).
   ControlDesk opens the project and experiment that belong to the check application.
4. From the context menu of the experiment's platform, select Configure Platform/Device to configure the platform according to the dSPACE real-time hardware connected to the host PC. For example, you have to specify the IP address of the hardware.
5. Press F5 to start a measurement.
6. If no error message is displayed in the Message Viewer, the board is installed correctly.

Result

You have checked the configuration.

Related topics

References

Configure Platform/Device (ControlDesk Platform Management 🕮)

How to View the Properties of a dSPACE System

Objective

After registration, you can view the properties of the dSPACE system.

Preconditions

The dSPACE board is registered.

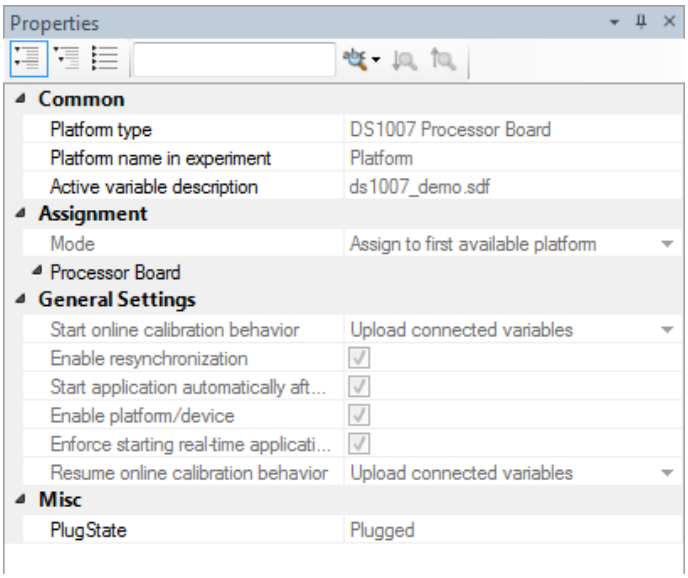
Method

To view the properties of a dSPACE system

1. On the View ribbon, click Controlbar - Switch Controlbars - Properties to open the Properties controlbar.

2 In the **Project Manager**, select the corresponding board.

The platform information is displayed in the **Properties** controlbar. The illustration below shows the properties of a DS1007 as an example.



Result

You have viewed the properties of your dSPACE system. For a detailed description of the properties, refer to Platform/Device-Related Properties (ControlDesk Platform Management 🕮).

Related topics

References

Platform/Device-Related Properties (ControlDesk Platform Management 🕮)

Updating the Firmware

Introduction

Usually your board contains the latest firmware. If you install a new dSPACE Release it may contain newer firmware. In this case the firmware can be updated.

Where to go from here

Information in this section

Basics on Firmware 33
Gives you information on the different kinds of firmware.

Basics on Firmware

Introduction

You can execute a real-time application on dSPACE real-time hardware only if the different kinds of firmware are available. The loaded firmware version has to provide the functionality implemented in the real-time application.

Firmware features

The firmware for a hardware component provides basic functionality that is stored in a nonvolatile memory. For example, it includes functions for the communication between the host PC and the hardware, and can also provide I/O functions such as CAN or LIN protocol support, or complex I/O functions for an FPGA component.

The firmware archives provides all the relevant firmware components that are required for your hardware.

Details on the firmware archives

The following firmware archives are available.

DS1006FwArchive.arc The firmware archive for a modular system based on a DS1006 Processor Board contains the following firmware components:

- DS1006 boot firmware
- DS1006 FPGA Code
- DS2202 CAN slave firmware
- DS2210 CAN slave firmware
- DS2211 CAN slave firmware
- DS4302 CAN slave firmware
- DS4330 LIN slave firmware
  (see the note above)
- DS4342 firmware

DS1007FwArchive.arc The firmware archive for a modular system based on a DS1007 PPC Processor Board contains the following firmware components:

- HCN Boot Firmware
- HCN Firmware
- CN Boot Firmware
- CN Firmware

- CN CPU Configuration
- System FPGA Firmware
- DS2202 CAN slave firmware
- DS2210 CAN slave firmware
- DS2211 CAN slave firmware
- DS4302 CAN slave firmware
- DS4330 LIN slave firmware
  (see the note above)
- DS4342 firmware

DS1104FwArchive.arc The firmware archive for a modular system based on a DS1104 R&D Controller Board contains the following firmware components:

- DS1104 boot firmware
- DS1104 Slave DSP firmware

MABXFwArchive.arc (for MicroAutoBox II) The firmware archive for MicroAutoBox II contains the following firmware components:

- DS1401 boot firmware
- DS1401 System PLD firmware
- DS1401 Host IF PLD firmware
- DS1401 Host IF firmware
- ADC TYPE 4 PLD firmware
- DIO TYPE 3 PLD firmware
- DIO TYPE 4 PLD firmware
- FPGA TYPE 1 PLD firmware
- AIO TYPE 1 PLD firmware
- CAN TYPE 1 firmware
- DS4342 firmware

Note

To program the firmware that supports the RTI DS1552 I/O Extension blockset for the DS1552 Multi-I/O module, you have to use the **`DS1401UpdateExtIO`** command, which is described in the MicroAutoBox II RTLib Reference 🕮.

DS1403FwArchive.arc (for MicroAutoBox III) The firmware archive for MicroAutoBox III contains the following firmware components:

- ADC Type4 UserFpga
- AIO Type1 UserFpga
- CAN Type1 UserFirmware
- DIO Type3 UserFpga
- DIO Type4 UserFpga
- DS1403 CnFirmware
- DS1403 CnIpl
- DS1403 UserCpld

- DS1403 UserFirmware
- DS1403 UserFpga
- DS1403 UserIpl
- DS1514 FPGA Base Board IoCpld
- DS1521 UserFpga
- DS4342 CAN FD Interface Module UserFpga
- FPGA Type1 IoFpga (DS1552)
- FPGA Type1 IoFpga (DS1554)

DS1202FwArchive.arc (for MicroLabBox) The firmware archive for MicroLabBox contains the following firmware components:

- HCN Boot Firmware
- HCN Firmware
- CN Boot Firmware
- CN Firmware
- CN CPU Configuration
- System FPGA Firmware
- CAN Type 1 firmware
- I/O clock buffer configuration
- I/O CPLD firmware
- I/O FPGA firmware

SCALEXIOFwArchive.arc As of dSPACE Release 2020-B, this is the default firmware archive for SCALEXIO systems. It provides the Linux-based firmware components and the KVM Hypervisor firmware.

The firmware archive for SCALEXIO systems with a Linux operating system contains the following firmware components:

- DSx86_32 UserFirmware
- DSx86_32 FactoryFirmware
- DS2502 UserFpga
- DS2551 UserFpga
- DS2601 UserFirmware and DS2601 UserFpga
- DS2621 UserFirmware and DS2621 UserFpga
- DS2642 UserFirmware and DS2642 UserFpga
- DS2655 UserFirmware and DS2655 UserFpga
- DS2655M1 UserFpga
- DS2655M2 UserFpga
- DS2656 UserFirmware and DS2656 UserFpga
- DS2671 UserFirmware and DS2671 UserFpga
- DS2672 UserFirmware and DS2672 UserFpga
- DS2680 UserFirmware, DS2680 UserFpga, and DS2680 IoFpga1 … 3
- DS2690 UserFirmware and DS2690 UserFpga
- DS2907 UserFirmware and DS2907 UserFpga
- DS6001 UserFirmware, DS6001 UserIplFirmware, and DS6001 UserFpga
- DS6051 UserFpga

- DS6071 UserFirmware
- DS6072 UserFirmware
- DS6073 UserFirmware
- DS6101 UserFirmware, DS6101 UserIplFirmware, and DS6101 UserFpga
- DS6121 UserFpga
- DS6201 UserFpga
- DS6202 UserFpga
- DS6221 UserFpga
- DS6241 UserFpga
- DS6301 UserFpga
- DS6311 UserFpga
- DS6321 UserFpga
- DS6333-CS UserFpga
- DS6333-PE UserFpga
- DS6335-CS UserFpga
- DS6341 UserFpga
- DS6342 UserFpga
- DS6351 UserFpga
- DS6601 UserFpga
- DS6602 UserFpga
- DS6651 UserFpga
- Dsx86_32 HypervisorFirmware

SCALEXIOQNXFwArchive.arc Up to and including dSPACE Release 2020-A, this was the default archive for SCALEXIO systems. To use the QNX firmware archive, you must explicitly select it.

The firmware archive for SCALEXIO systems with a QNX operating system contains the following firmware components:

- DSx86_32 UserFirmware
- DSx86_32 FactoryFirmware
- DS2502 UserFpga
- DS2551 UserFpga
- DS2601 UserFirmware and DS2601 UserFpga
- DS2621 UserFirmware and DS2621 UserFpga
- DS2642 UserFirmware and DS2642 UserFpga
- DS2655 UserFirmware and DS2655 UserFpga
- DS2655M1 UserFpga
- DS2655M2 UserFpga
- DS2656 UserFirmware and DS2656 UserFpga
- DS2671 UserFirmware and DS2671 UserFpga
- DS2672 UserFirmware and DS2672 UserFpga
- DS2680 UserFirmware, DS2680 UserFpga, and DS2680 IoFpga1 … 3
- DS2690 UserFirmware and DS2690 UserFpga
- DS2907 UserFirmware and DS2907 UserFpga

- DS6001 UserFirmware, DS6001 UserIplFirmware, and DS6001 UserFpga
- DS6051 UserFpga
- DS6071 UserFirmware
- DS6072 UserFirmware
- DS6073 UserFirmware
- DS6101 UserFirmware, DS6101 UserIplFirmware, and DS6101 UserFpga
- DS6121 UserFpga
- DS6201 UserFpga
- DS6202 UserFpga
- DS6221 UserFpga
- DS6241 UserFpga
- DS6301 UserFpga
- DS6311 UserFpga
- DS6321 UserFpga
- DS6333-CS UserFpga
- DS6333-PE UserFpga
- DS6335-CS UserFpga
- DS6341 UserFpga
- DS6342 UserFpga
- DS6351 UserFpga
- DS6601 UserFpga
- DS6602 UserFpga
- DS6651 UserFpga
- Dsx86_32 HypervisorFirmware

Note

- The archive format for DS1007 and MicroLabBox changed with Firmware Archives 2.0 contained in dSPACE Release 2015-B. To open an archive in the new format, you must use Firmware Manager 2.0 or later.
- The archive format for SCALEXIO changed with Firmware Archives 2.1 contained in dSPACE Release 2016-A. To open an archive in the new format, you must use Firmware Manager 2.1 or later.

Special firmware

The firmware archives installed with your dSPACE software provide the standard firmware type. There might be other firmware types to be managed with the Firmware Manager.

Custom firmware The Firmware Manager allows you to install custom firmware that dSPACE provides for solutions or engineering projects.

User firmware

Note

When you use a SCALEXIO system, this term is used for the standard firmware. The following restrictions refer only to customized firmware.

User firmware is a firmware that is based on dSPACE firmware but extended with your own functionality. The Firmware Manager does not support loading user firmware.

Note

dSPACE accepts no liability for incorrect operation or property damage when using user firmware with dSPACE hardware.

Default factory firmware MicroAutoBox II, MicroAutoBox III, MicroLabBox, DS1007, and SCALEXIO are providing a secured mode for using the default factory firmware. If firmware is corrupted, the hardware, automatically reboots, if necessary, and loads the default factory firmware that lets you access the board and retry the firmware update.

For further information, refer to:

- Using MicroAutoBox II: How to Start MicroAutoBox II to Secured Mode (MicroAutoBox II Hardware Installation and Configuration 🕮)
- Using MicroAutoBox III: How to Force a Start with Factory Firmware (MicroAutoBox III Hardware Installation and Configuration 🕮)
- Using MicroLabBox: How to Force a Restart with Factory Firmware (MicroLabBox Hardware Installation and Configuration 🕮)
- Using modular system based on DS1007: How to Start the DS1007 to Secured Mode (DS1007 Hardware Installation and Configuration Guide 🕮)

Note

Note the following restriction if you use DS1007, MicroAutoBox II, or MicroLabBox.

In secured mode, for example, caused by an interrupted firmware repair or update process, you cannot repair or update the firmware components of the I/O components of the board (connected I/O boards, internal I/O modules, or the I/O FPGA). You firstly have to repair or update the firmware components of the base board and then reboot the board to leave the secured mode. Error messages regarding to the repair or update process of the I/O components can be ignored. After reboot, you can continue the repair or update process for the firmware of the I/O components.

If your real-time hardware does not provide a secured mode for using the default factory firmware, you have to repair the board's boot firmware. You can do this via the command line interface of the firmware management or via the Platform Manager. To repair the corrupted boot firmware via the command line interface, refer to Examples of Script-Based Firmware Management (Firmware Manager Manual 🕮).

Related topics

Basics

Basics on the Firmware Manager (Firmware Manager Manual 🕮)

How to Prepare the Firmware Update

Objective

The preparation of a firmware update consists of specifying some general firmware settings.

Preconditions

The following preconditions must be fulfilled for configuring the general firmware settings:

- The real-time hardware must be connected to the host PC.
- The real-time hardware must be switched on.
- The required firmware archive must be available.

  You can find the latest firmware archives on the dSPACE website at http://www.dspace.com/go/firmware.
- If a real-time application is loaded to the board's flash memory, it is recommended to clear the flash before starting the update process to avoid unpredictable output signals.

  If a real-time application is running, it is stopped by the firmware management.
- If you have registered a multiprocessor system, you can update only one processor at a time.
- If you have registered a multicore system with additional I/O boards, you have to select the core to which the I/O boards are connected for the update of the entire system. The other cores will be updated, too.

Note

- The archive format for DS1007 and MicroLabBox changed with Firmware Archives 2.0 contained in dSPACE Release 2015-B. To open an archive in the new format, you must use Firmware Manager 2.0 or later.
- The archive format for SCALEXIO changed with Firmware Archives 2.1 contained in dSPACE Release 2016-A. To open an archive in the new format, you must use Firmware Manager 2.1 or later.

Method

To prepare the firmware update

1. Open the Platform Manager.
2. If no real-time hardware is displayed in the Platform Manager, register the real-time hardware that you want to update.

3 Choose **Update Firmware** in the platform's context menu to open the **Update Firmware Wizard**.

The wizard starts with the **Select Mode** dialog.

4 Select the firmware update mode.

By default, the **Update** mode is set to update all firmware components of your real-time hardware with later firmware. With the **Repair** mode enabled, you can select the firmware components to be repaired.

To switch to the repair mode, select **Firmware repair mode** in the **Select Mode** dialog.

5 Click **Next** to continue with the **Select Firmware Archive** dialog.

The latest firmware archive for the selected platform is automatically set. Optionally, browse for another firmware archive. This might be useful if you want to update to a firmware version other than the latest or repair user firmware, for example.

6 Click **Next** to continue with the **Select Firmware Components** dialog.

Result

You have configured the settings which are required for a firmware update process in update or repair mode.

Related topics

HowTos

How to Repair Firmware (Firmware Manager Manual 🕮)
How to Update Firmware..... 40

How to Update Firmware

Objective

Gives you the instructions for the firmware *update* mode.

Preconditions

The firmware update process has to be prepared with the **Update Mode** specified as described in How to Prepare the Firmware Update on page 39.

Safety precautions

⚠ WARNING

Risk of injury and/or material damage

Updating the firmware can cause uncontrolled movements of connected devices.

- Disconnect actuators and sensors from the associated real-time hardware before you start the update process.

NOTICE

Interrupting the update process disables the hardware

If the firmware update is interrupted, for example, by switching off the power, you have to restart the update process.

Note

Follow the instructions of the firmware management tool to correctly finish the firmware update process. For example, in some cases the hardware has to be rebooted to complete the firmware update.

Method

To update firmware

1 In the Select Firmware Components dialog, click Update to start the firmware update process.

In the Update column, the firmware components to be updated are marked and red. The components are not marked for update if the version of the currently installed firmware is identical to or later than the firmware available in the specified firmware archive.

If there are updatable firmware components, the update process starts. You can see the progress in the Status column. The initial '--' entry is replaced by a percentage. If the progress information cannot be detected continuously, only the states 50% and 100% are displayed. If the process successfully finished, an OK is shown, otherwise an error message is displayed.

If the firmware update will require more than 40 minutes, an estimate of the time is displayed. Then you can decide whether to start the process. Interrupting a running firmware update process is not possible.

Note

You must not switch off the hardware during the firmware update process. This will cause a corrupted firmware.

Follow the given instructions to complete the firmware update. For example, some firmware components require a hardware restart.

Result

You have updated the firmware components of your hardware.

Related topics

HowTos

How to Prepare the Firmware Update....39
How to Repair Firmware (Firmware Manager Manual 🕮)

Implementing Models via Simulink/RTI or via Handcoding

Introduction

The first step is to implement your model. You can either embed the blocks provided by dSPACE's Real-Time Interface (RTI) in a Simulink model or use RTLib's functions to handcode your application directly in C.

Where to go from here

Information in this section

How to Start RTI

Objective

dSPACE Real-Time Interface (RTI) is the interface between Simulink and the dSPACE systems: In connection with MATLAB®/Simulink®, the C code for the real-time model for a specific dSPACE system is generated by the Simulink® Coder™.

Preconditions

- Ensure that MATLAB, Simulink, Simulink® Coder™, and Real-Time Interface (RTI) for your dSPACE system are properly installed.
- The required licenses must be available and activated. For information on handling the license mechanism, refer to Installing dSPACE Software 🕮.

Method

To start RTI

1. Start MATLAB.
2. In the MATLAB Command Window, enter one of the following commands to open the RTI block library of the corresponding dSPACE system:

dSPACE System	Command
DS1006	`rti1006`
DS1007	`rti1007`

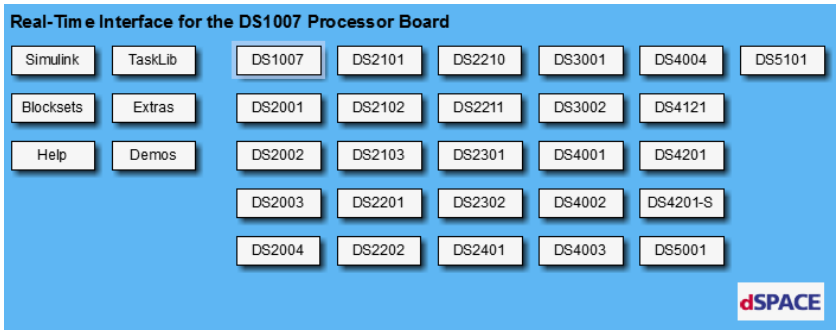
dSPACE System	Command
DS1104	`rti1104`
MicroAutoBox II	`rti1401`
MicroLabBox	`rti1202`
Multiprocessor system	`rtimp`

Result

You started RTI and opened the RTI block library of a specific dSPACE system.

Example

In the MATLAB Command Window, enter `rti1007` for a DS1007-based modular system. The following RTI block library is opened:



To switch to another platform such as MicroLabBox, enter `rti1202`.

Next step

Now you can work with a Simulink model. Refer to How to Implement a Model via Simulink and RTI on page 43.

Related topics

HowTos

How to Implement a Model via Simulink and RTI.....43

How to Implement a Model via Simulink and RTI

Objective

A demo shows you how to work with a Simulink model.

Demo model

The `smd_<xxxx>_sl` demo model is an example of how to use the RTI library (`<xxxx>` is a placeholder for the platform). The demo simulates a damped spring-mass system stimulated by a square-wave signal. It is a ready-to-use example that does not require any I/O hardware.

Precondition

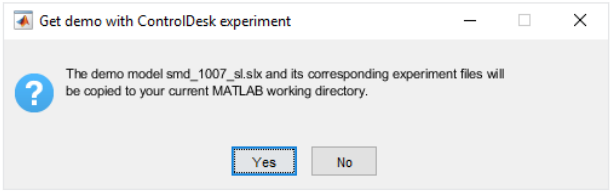
Before you can start the build and download procedure, you have to ensure that your dSPACE hardware is registered correctly in ControlDesk. Refer to How to Register a dSPACE System on page 26.

Method

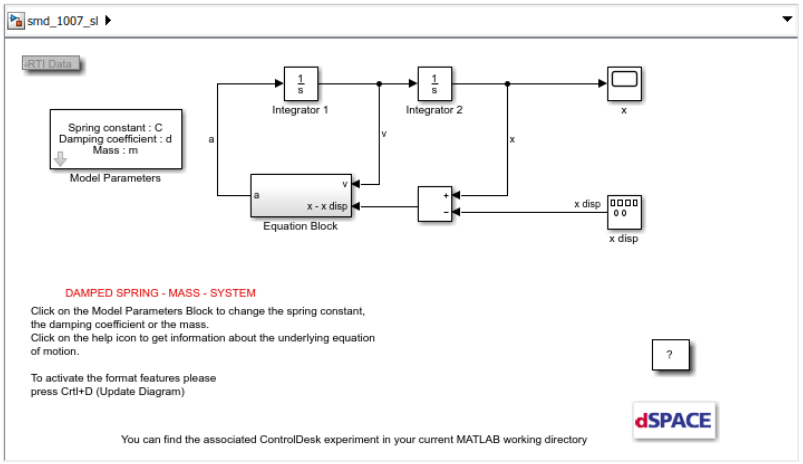
To implement a model via Simulink and RTI

1. In the Library: rtilib<xxxx> window, double-click Demos.
2. Double-click Spring - Mass - Damper in the Library: rti<xxxx>demolib window.

   The Get demo with ControlDesk experiment dialog opens.
3. In the Get demo with ControlDesk experiment dialog, click Yes to copy the demo model and the experiment files to your current MATLAB working directory.



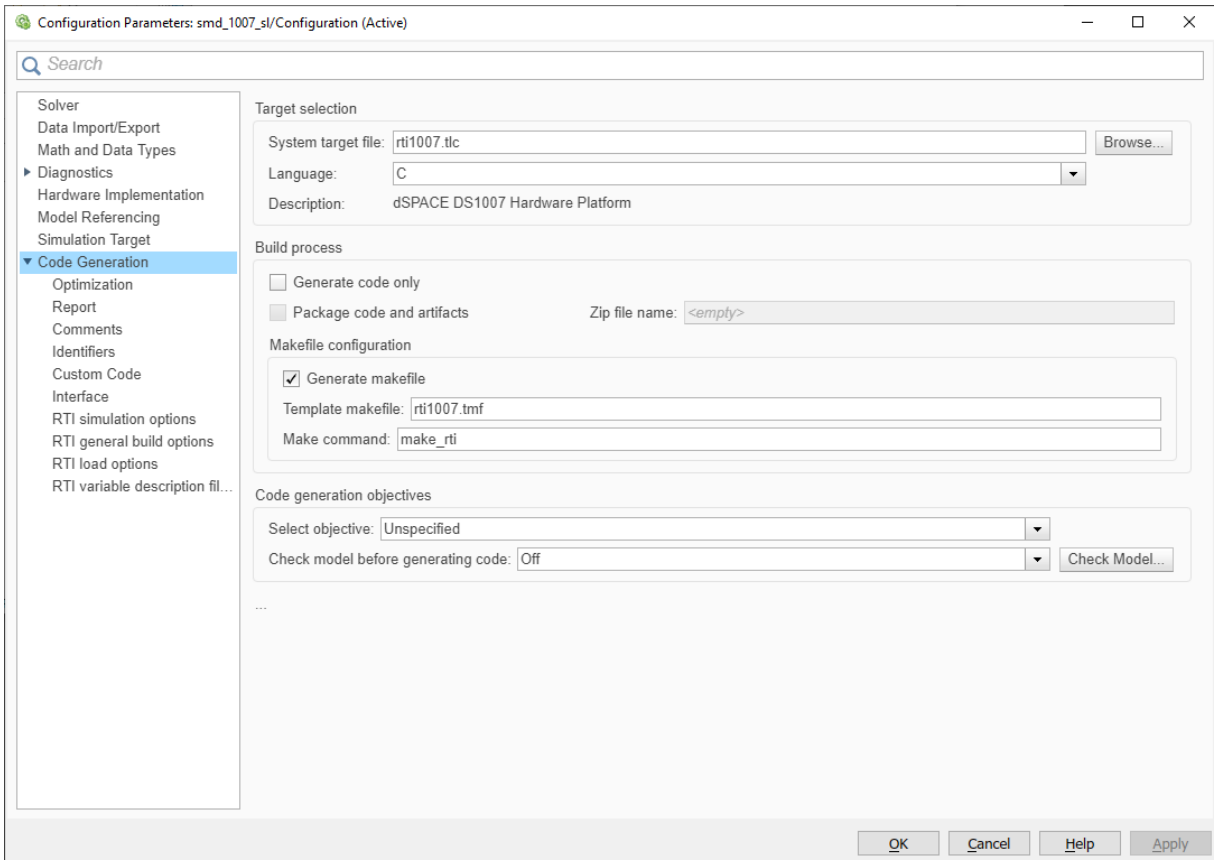
The Simulink model is then opened.



4. To examine the model, double-click its blocks.

5 Press **Ctrl** + **E**.

The Configuration Parameters dialog opens.



6 On the Code Generation page, check that the system target file (**rti<xxxx>.tlc**) and the template makefile (**rti<xxxx>.tmf**) match your dSPACE system, and that the make command **make_rti** is specified. If necessary, change the entries.

7 Press **Ctrl** + **B** to build the real-time application and download it to the dSPACE system.

The **smd_<xxxx>_sl.sdf** system description file and the real-time application are generated. The real-time application is loaded to your dSPACE system and started automatically.

If the real-time application — stored in the current MATLAB working folder — is stopped, you can reload it to the dSPACE system using ControlDesk. No new build process is needed if you do not change the Simulink model in the meantime.

Result

The real-time application is built and loaded to the dSPACE system.

Next steps

You can use ControlDesk to stop and restart the real-time application. Refer to Handling Real-Time Applications with ControlDesk on page 48.

The `smd_<xxxx>_sl.zip` file contains a ControlDesk demo experiment for experimenting with the MATLAB/Simulink-based demo application. The ZIP file is located in the `\Demos\DS<xxxx>\GettingStarted\Simulink` folder in your RCP & HIL installation.

Related topics

HowTos

How to Implement a Model via Handcoding....46
How to Start RTI....42

How to Implement a Model via Handcoding

Objective

A demo model shows you how to work with a C-coded model.

C-coded demo model

The C-coded example model `smd_<xxxx>_hc.c` demonstrates how to proceed without MATLAB and Simulink (`<xxxx>` is a placeholder for the platform). This model does not require any I/O hardware. It simulates a damped spring-mass system stimulated by a square-wave signal. The parameters are spring, mass and damper.

down<xxxx> utility

To compile and link the real-time application for your dSPACE system, you can use the `down<xxxx>` utility in a Command Prompt window. The utility downloads, starts and checks the application as well.

Method

To implement a model via handcoding

1. Extract the file
   `<RCP_HIL_InstallationPath>\Demos\DS<xxxx>\GettingStarted\HandCode\smd_<xxxx>_hc.zip`
   to a folder for which you have write permission.
2. Open the file `.\smd_<xxxx>_hc\Source Files\smd_<xxxx>_hc.c` in a text editor of your choice to inspect the code.
3. Open the `Command Prompt for dSPACE RCP and HIL` in the `dSPACE RCP and HIL` start menu folder.
4. In the `Command Prompt for dSPACE RCP and HIL`, change to the folder you extracted the ZIP file to.
5. Enter the following command:
   `down<xxxx> smd_<xxxx>_hc`
   For example, when you enter `down1007 smd_1007_hc` for the DS1007, the `smd_1007_hc.rta` real-time application file is generated.

Result

The `smd_<xxxx>_hc` real-time application is generated, downloaded and started.

Next step

You can use ControlDesk to stop and restart the real-time application, refer to Handling Real-Time Applications with ControlDesk on page 48.

The `smd_<xxxx>_hc.zip` file contains a ControlDesk demo experiment for experimenting with the MATLAB/Simulink-based demo application. The ZIP file is located in the `\Demos\DS<xxxx>\GettingStarted\HandCode\` folder in your RCP & HIL installation.

Related topics

HowTos

How to Implement a Model via Simulink and RTI....43
How to Start RTI....42

Handling Real-Time Applications with ControlDesk

Introduction

This section describes how to handle applications on real-time processors (RTP).

Where to go from here

Information in this section

How to Load an Application to the Program Memory and Start the Real-Time Processor 49
To execute the application, it must be downloaded to the program memory.

How to Stop a Real-Time Application 50
You can use the ControlDesk **Platforms/Devices** controlbar to stop an application running on a real-time processor.

How to Reload a Real-Time Application 50
You can use the ControlDesk **Platforms/Devices** controlbar to reload an application running on a real-time processor.

How to Download an Application to the Flash Memory and Start the Real-Time Processor 51
Some dSPACE boards have a flash memory. This allows them to be used as a stand-alone system without a connection to the host PC.

How to Clear an Application from the Flash Memory 53
If an application is loaded to the flash memory, the dSPACE board starts the application automatically after reboot. If you want to avoid this, you have to clear the flash memory.

How to Download and Start an Application via Command Line 54
To download and start applications on the real-time hardware, you can use the `cmdloader` tool.

Information in other sections

Implementing Models via Simulink/RTI or via Handcoding 42
The first step is to implement your model. You can either embed the blocks provided by dSPACE's Real-Time Interface (RTI) in a Simulink model or use RTLib's functions to handcode your application directly in C.

How to Load an Application to the Program Memory and Start the Real-Time Processor

Objective

You can load real-time applications to the program memory and start the real-time processor in the ControlDesk Platforms/Devices controlbar.

Method

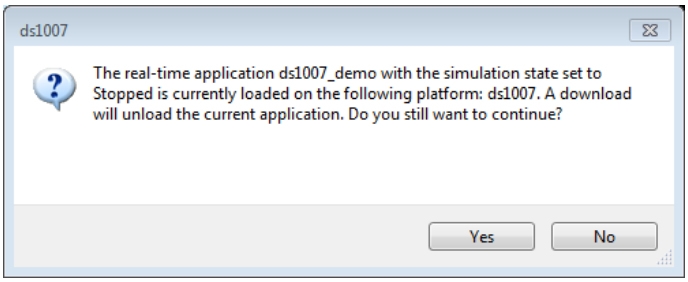
To download an application to the program memory and start the Real-Time Processor

1 In the Platforms/Devices controlbar, open the context menu of your platform icon (or multiprocessor icon) and select Real-Time Application - Load.

The Select Real-Time Application dialog opens.

2 In the Select Real-Time Application dialog, select an SDF file, or a PPC, RTA, or x86 file (depending on the hardware type).

3 If a previously loaded application is still running on your system, a dialog similar to the following opens:



If you click Yes, the running application is stopped before the new application is downloaded and started.

Result

You have downloaded an application to the program memory and started the real-time processor.

Related topics

HowTos

How to Download an Application to the Flash Memory and Start the Real-Time Processor 51

References

Real-Time Application - Load (ControlDesk Platform Management 🕮)

How to Stop a Real-Time Application

Objective

You can use the ControlDesk **Platforms/Devices** controlbar to stop an application running on a real-time processor.

Method

To stop a real-time application

1 In the **Platforms/Devices** controlbar, open the context menu of your platform. Depending on your platform, select **Stop** or **Stop RTP**.

Result

You have stopped the real-time application.

Related topics

HowTos

How to Reload a Real-Time Application..... 50

How to Reload a Real-Time Application

Objective

You can use the ControlDesk **Platforms/Devices** controlbar to reload an application running on a real-time processor.

Restrictions

- You cannot reload a multiprocessor application. You have to download it again. Refer to How to Load an Application to the Program Memory and Start the Real-Time Processor on page 49.
- You cannot reload an application for a modular system containing DS230x boards. You have to download it again. Refer to How to Load an Application to the Program Memory and Start the Real-Time Processor on page 49.

Method

To reload a real-time application

1 In the **Platforms/Devices** controlbar, open the context menu of your board icon and select **Real Time Application - Reload**.

Tip

For some platforms, the command is available in the context menu of the application to be reloaded.

Result

You have reloaded a real-time application.

Related topics

HowTos

How to Stop a Real-Time Application....50

How to Download an Application to the Flash Memory and Start the Real-Time Processor

Objective

A flash memory is used to load a real-time application automatically after power-up. You must first download the real-time application to the flash memory.

Flash memory

Flash memory of dSPACE boards Several dSPACE boards are equipped with a flash memory for real-time applications. The flash memory can be used to load a real-time application automatically after power-up of the board.

The following table shows which dSPACE boards have a flash memory for autobooting a real-time application.

dSPACE Board	Description
DS1006	Flash memory on CompactFlash card
DS1007	Flash memory
DS1104[1)]	Flash memory
MicroAutoBox II	Flash memory
MicroLabBox	Flash memory

1) The DS1104 cannot detect a reboot of the host PC. As a consequence, a real-time application in the DS1104 flash memory is not started automatically when the host PC is rebooted. To start a real-time application from the flash memory of the DS1104, disconnect the host PC from the mains for a short time by switching off the power supply, or by disconnecting the power connector.

Loading the application on power up On power-up, the dSPACE board always starts executing the bootstrap loader contained in the flash memory. The loader checks for an application program currently stored in the flash memory. If it finds one, the application is started. If it does not detect an application in the flash memory, the loader enters the idle state and waits for commands from the connected host PC.

Method

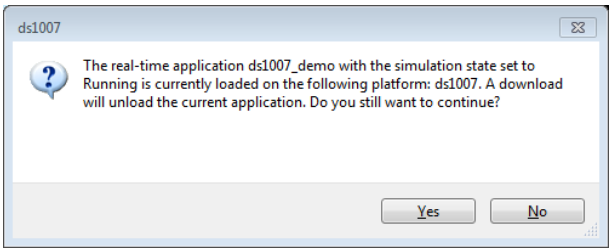
To download a real-time application to the flash memory and start the real-time processor

1 In the Platforms/Devices controlbar, open the context menu of your platform and select **Real-Time Application - Load to Flash (DS1006/DS1104/MicroAutoBox II)** (ControlDesk Platform Management 🕮) or **Real-Time Application - Load to Flash (DS1007/MicroLabBox/MicroAutoBox III/SCALEXIO)** (ControlDesk Platform Management 🕮).

The Select Real-Time Application dialog opens.

2 In the Select Real-Time Application dialog, select an SDF file, or a PPC, RTA, or x86 file (depending on the hardware type).

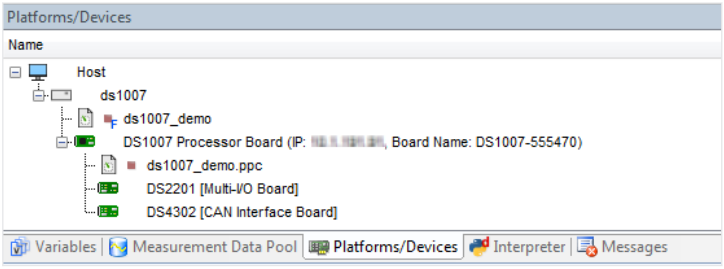
3 If a previously loaded application is still running on your system, a dialog similar to the following opens:



If you click Yes, the running application is stopped before the new application is downloaded and started.

Result

You have downloaded an application to the flash memory and started the real-time processor. The F next to the real-time application icon in the Platforms/Devices controlbar indicates that the running application is loaded from the flash memory.



Related topics

HowTos

How to Load an Application to the Program Memory and Start the Real-Time Processor....49

How to Clear an Application from the Flash Memory

Objective

If you want to prevent the system from booting a flash application, you have to clear the application from the flash memory.

DS1006 CompactFlash card removal

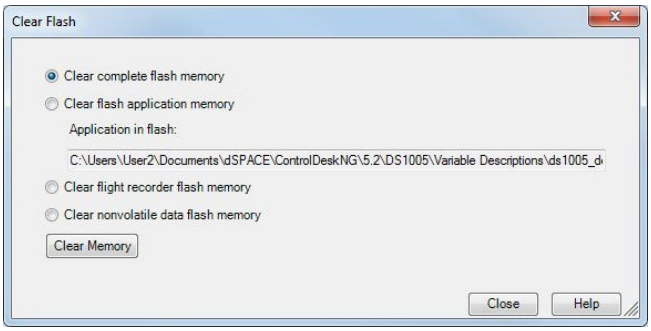
DS1006: A DS1006 uses the flash memory of a CompactFlash card, which is removable. You can remove the CompactFlash card to avoid an automatic start. Remove it only while the expansion box is switched off.

Method

To clear an application from the flash memory

1 In the Platforms/Devices controlbar, open the context menu of your platform icon and select Clear Flash.

The Clear Flash dialog opens.



2 In the Clear Flash dialog, select one of the following options to clear the memory (the available options depend on the selected platform type):

- Clear complete flash memory to clear the whole flash memory.
- Clear application flash memory to clear only the loaded application from the application flash memory. The currently loaded application is displayed in the Application in flash field.
- Clear nonvolatile data flash memory to clear only data that is stored as nonvolatile. Flight recorder data is stored in the nonvolatile memory and therefore also cleared.

3 Click Clear Memory to clear the flash memory according to your selection.

Result

You have cleared an application from the flash memory.

Related topics

HowTos

How to Download an Application to the Flash Memory and Start the Real-Time Processor 51

How to Download and Start an Application via Command Line

Objective

To download and start applications on the real-time hardware, you can use the `cmdloader` tool.

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

Using the cmdloader tool

The `cmdloader` tool is located in the `.\Main\bin` folder of your ControlDesk installation and in the `<RCP_HIL_InstallationPath>\Exe\CmdLoader\bin` folder of your RCP & HIL installation. When you use the tool, you have to specify the entire path to it. If the path contains blanks, you have to enclose the path in quotation marks.

Example For example, to display the `cmdloader` help, enter

```
"C:\Program Files\dSPACE ControlDesk 7.4\Main\bin\cmdloader" -?
```

in a Command Prompt window.

Tip

To use the `cmdloader` tool in a Command Prompt window without having to specify the path to the tool, select dSPACE ControlDesk 7.4 - Platform Management Loader on the Windows Start menu.
This opens a Command Prompt window. You do not have to specify the path as long as this Command Prompt window is open.

For details on `cmdloader` and its parameters, refer to Handling Applications via Command Line (ControlDesk Platform Management 🕮).

Preconditions

You must have registered the dSPACE system(s) since the `cmdloader` tool requires registration information.

Method

To download and start an application via command line

1 Select All Programs - dSPACE ControlDesk 7.4 - Utilities - Platform Management Loader from the Start menu.
This opens a DOS window. You do not have to specify the path as long as this DOS window is open.

2 In the DOS window, enter `cmdloader` with valid parameters as shown below:

```
cmdloader <{application [-suppress_start]} | -start | -stop>
                <-p platform_name | {-rnc net_client -t platform_type}>
                [-go seconds] [-fl] [-ol output_level] [-q] [-ra] [-?]
```

The table below shows the most important command parameters:

Parameter	Description
`application`	The application to download to the selected platform. You must specify the file name extension (SDF, PPC, RTA or x86). You also have to specify the relative or absolute path to the application. If the path contains blanks, you have to enclose the path in quotation marks, e.g.,: `"C:\Program Files\dSPACE ControlDesk 7.4\Demos\RTApplications\DS1007\ds1007_demo.sdf"`
`-p platform_name`[1), 2)]	To scan the recent hardware configuration for the platform (specified via `platform_name`), to try to register the platform, and to load the application (specified via the `application` parameter) to the platform.
`-fl`	To load the application (specified via the `application` parameter) to the flash memory of the platform (specified via the `-p platform_name` parameter).
`-ra`	To scan the recent hardware configuration for platforms that are not yet registered and to try to register them.

1) Mandatory parameter

2) The platform name as displayed in the Properties controlbar:

The platform can be a dSPACE single- or multiprocessor system.

Result

You have downloaded and started an application via command line.

Examples

The following examples show how to use the `cmdloader` tool for batch operations.

- **`cmdloader -p ds1104 ds1104_demo.sdf`**

  The `cmdloader` tool loads the `ds1104_demo.sdf` application to a platform named `ds1104` and starts the application.

- **`cmdloader -p ds1401 -fl ds1401_demo.sdf`**

  The `cmdloader` tool loads the `ds1401_demo.sdf` application to the flash memory of a platform named `ds1401` and starts the application.

- **`cmdloader -ra`**

  The `cmdloader` tool scans the recent hardware configuration for platforms that are not yet registered and tries to register them.

- **`cmdloader -ra -p ds1007 ds1007_demo.sdf`**

  This example combines the use of the following parameters:

  - `-ra`
  - `-p platform_name`
  - `application`

Related topics

Basics

Handling Applications via Command Line (ControlDesk Platform Management 🕮)

HowTos

How to Download an Application to the Flash Memory and Start the Real-Time Processor....51
How to Load an Application to the Program Memory and Start the Real-Time Processor....49

Experimenting with ControlDesk

Introduction

To experiment with real-time applications, use ControlDesk.

Where to go from here

Information in this section

Demo Projects/Experiments and Related Files

Introduction

You can use demo projects/experiments for your platform to experiment with ControlDesk features. The Real-Time application demo projects allow you to work with ControlDesk and use all of its features with dSPACE real-time hardware connected to the host PC.

Accessing the demo projects/experiments

ControlDesk provides a demo project/experiment and related files for each platform.

For instructions on opening demo projects, refer to Opening a demo project (ControlDesk Introduction and Overview 🕮).

Related files

Demo projects/experiments contain the following files:

PPC/x86/RTA file The real-time application file to be downloaded to the hardware.

TRC file Provides information on available variables and how they are grouped. Trace files are either generated by RTI/Simulink or hand-coded.

MAP file Maps names of variables to addresses of the physical memory. It is generated by the C compiler.

SDF file The system description (SDF) file specifies which executable is downloaded to which processor.

Related topics

Basics

Real-Time Application Demos (ControlDesk Introduction and Overview 🕮)

How to Measure Variable Values

Objective

To measure the variable values of a running real-time application, you have to connect an instrument to the variables.

Tip

A successful measurement also confirms that your ControlDesk installation is working correctly.

Working without the demo project/experiment

To show you how to set up a project/experiment and perform a measurement, the instructions below show all the steps from defining a project/experiment to starting a measurement. For this reason, the instructions do *not* use the prepared demo project/experiment, but only the variable description file contained in the demo.

Preconditions

The board must be registered. Refer to How to Register a dSPACE System on page 26.

Method

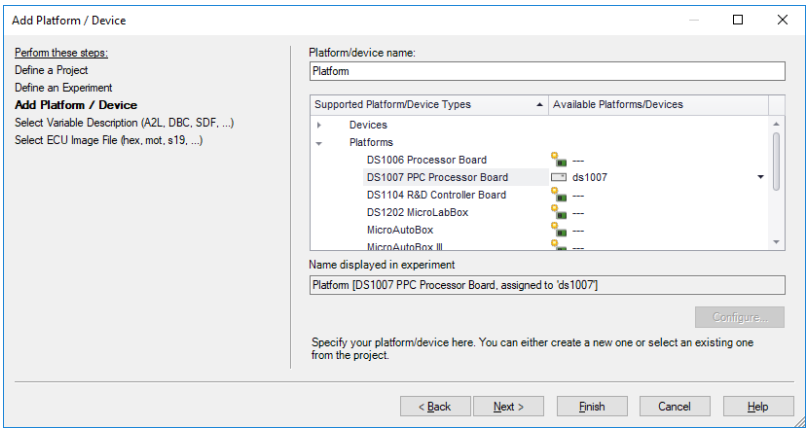
To measure variable values

1. Start ControlDesk.
2. On the File ribbon, click New - Project + Experiment to define a new project/experiment.

   The Define a Project dialog opens.
3. In the Define a Project dialog, enter a name for the new project.

   Click Next > to open the Define an Experiment dialog.
4. In the Define an Experiment dialog, enter a name for the new experiment and click Next > to open the Add Platform / Device dialog.

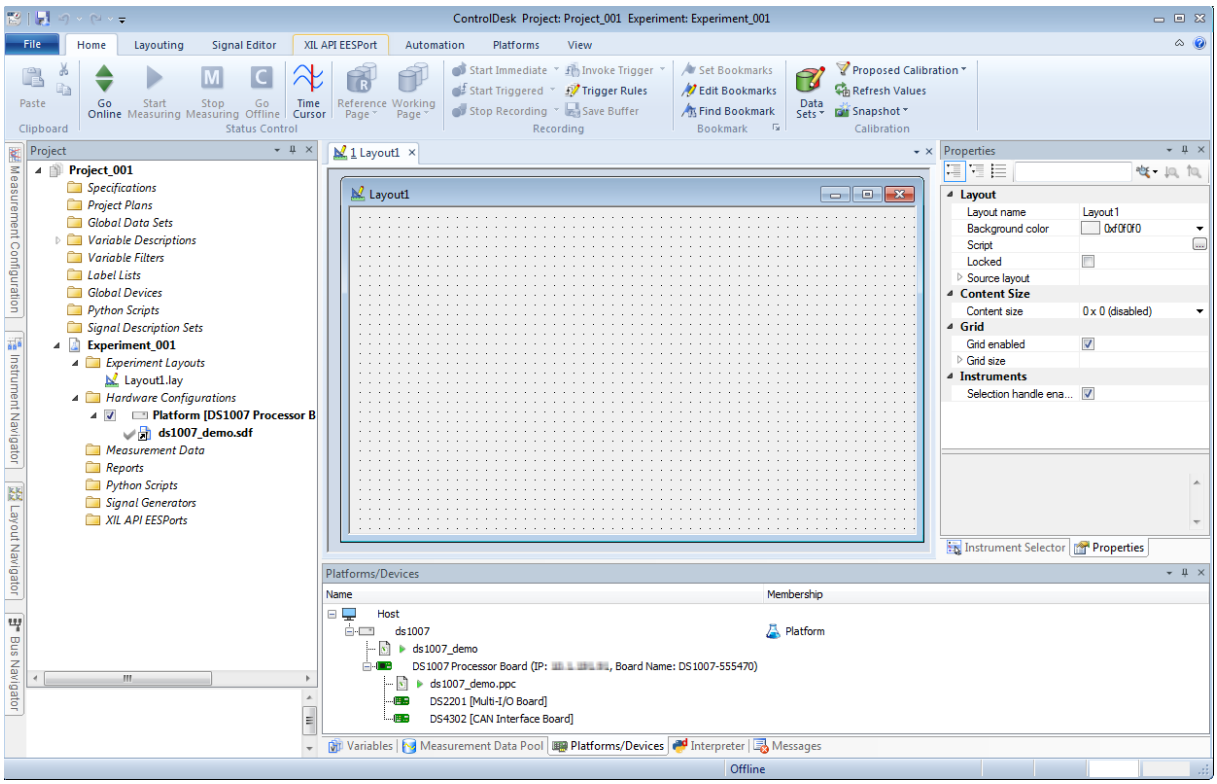
5 In the Add Platform / Device dialog, add the desired platform or device and click Next > to open the Select Variable Description dialog.



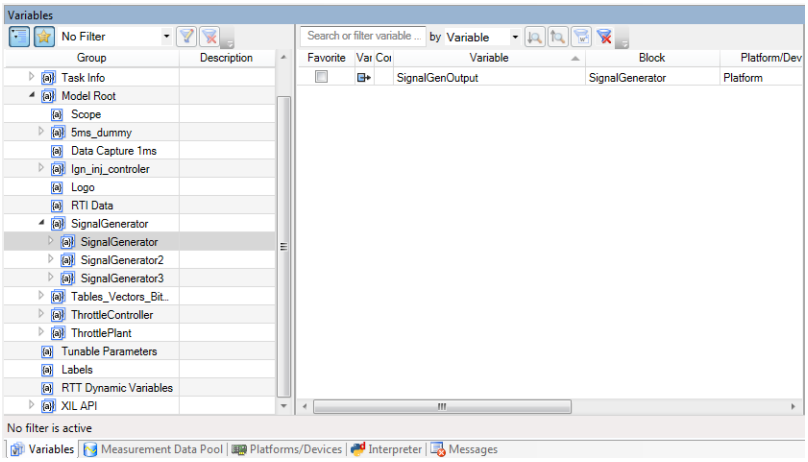
6 In the Select Variable Description dialog, click Import from file to navigate to the installation folder of your ControlDesk installation, select the `<platform>_demo.sdf` variable description file for your real-time hardware from the `\Demos\RTApplications\<platform>` folder and click Open. For example, select `../Demos/RTApplications/DS1007/ds1007_demo.sdf` for a DS1007.

7 Click Finish.

The new project/experiment opens.

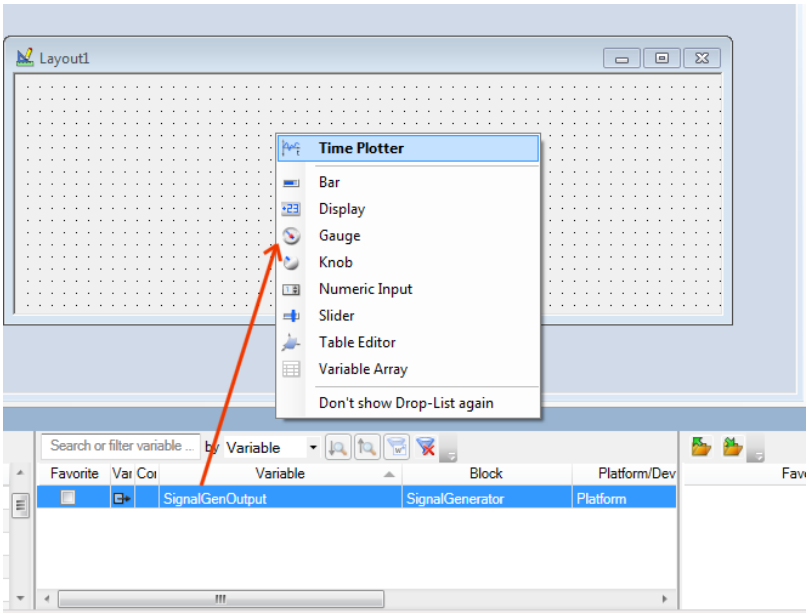


8 In the tree view of the Variables controlbar, navigate to `DS1007_demo.sdf/Model Root/SignalGenerator/SignalGenerator`.



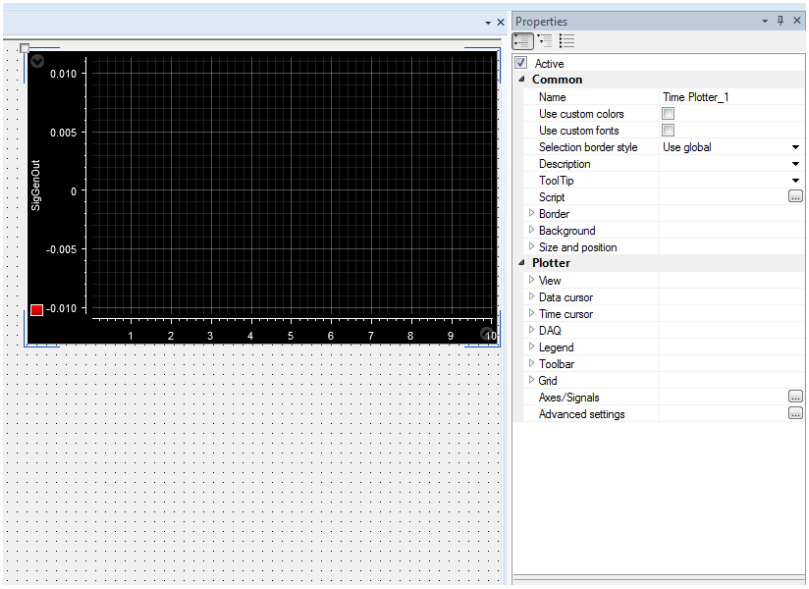
9 In the Variable list, select the SignalGenOutput variable and drag it to the new layout.

10 In the Instrument Type list, click Time Plotter.



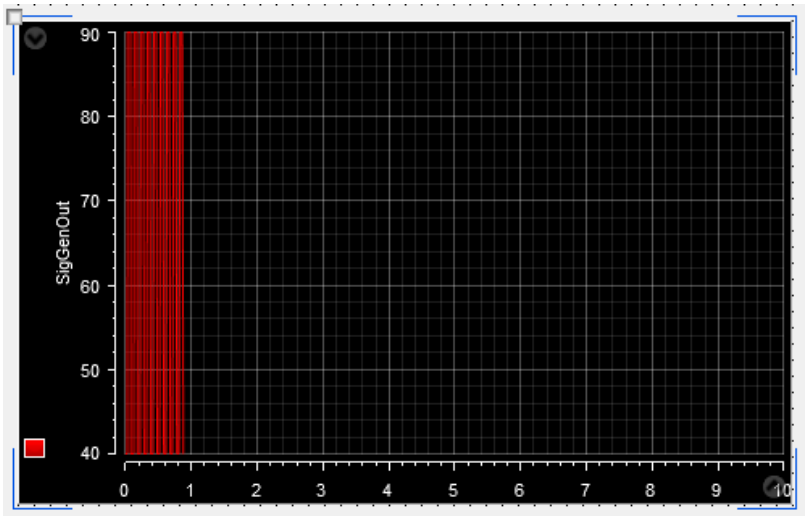
The Time Plotter opens. You have connected the SignalGenOutput variable to the Time Plotter.

11 To display or change the properties of the **Time Plotter**, click the instrument. The **Properties** controlbar shows the plotter properties.



12 Press **F5** to measure the variable values.

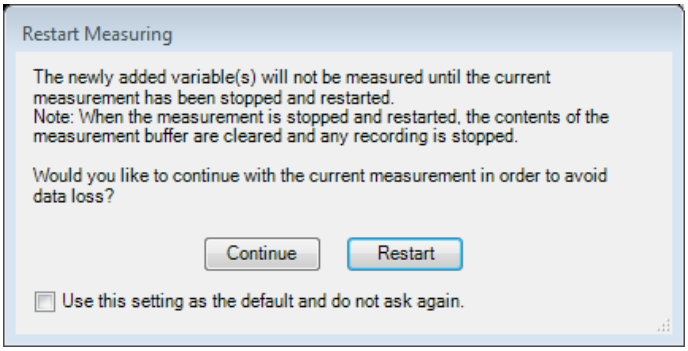
The illustration below shows a running measurement.



13 To add another variable to your measurement, change to the **Throttle Controller** variable group in the tree view of the **Variables** controlbar and select the **throttle valve position [%]** variable in the **Variable list**.

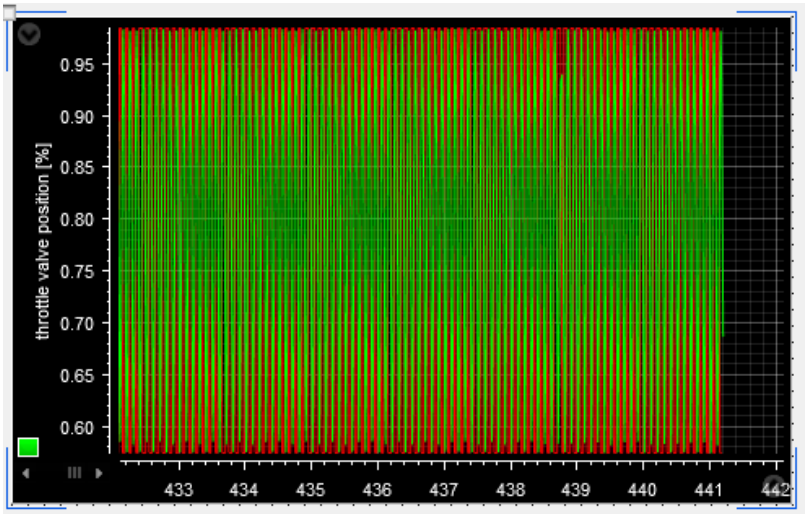
14 Drag it to the **Time Plotter**.

15 The **Restart Measuring** dialog opens and asks you either to keep the current measurement or to stop it and start a new one, because the added variable cannot be measured until a restart is performed.



Click Restart to perform the measurement again.

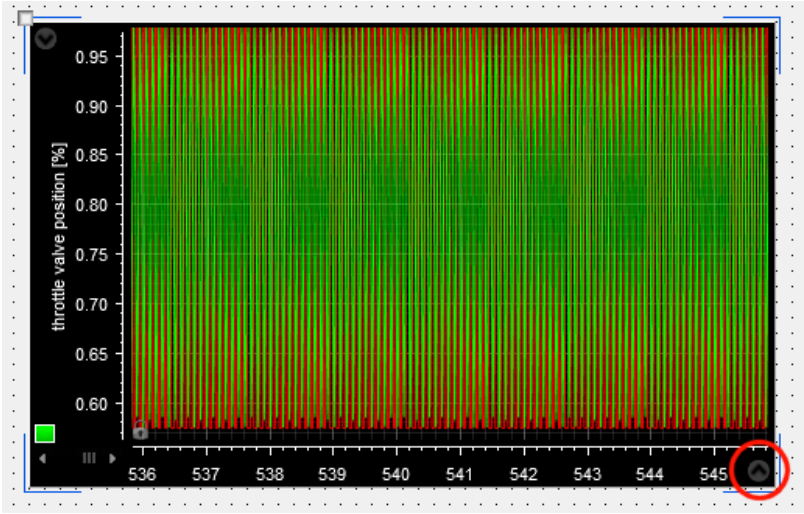
You have added another variable to your measurement.



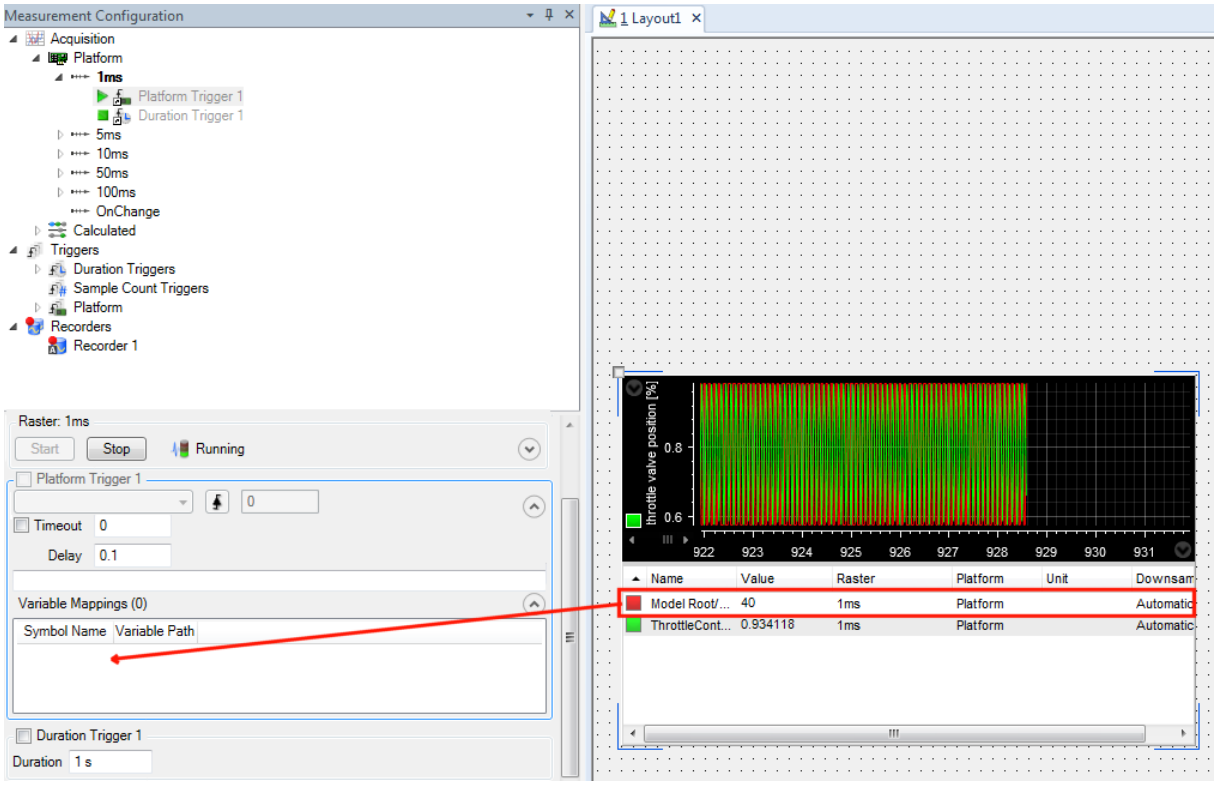
With ControlDesk, you can perform triggered measurements on dSPACE real-time hardware. Triggered measurement means that data capture with a measurement raster is not continuous, but started and stopped by triggers. In the following steps, you will specify a start trigger (indicated by a ▶ symbol), and a stop trigger (indicated by a ■ symbol) for the measurement raster.

16 In the Measurement Configuration controlbar, click Platform - 1ms - Platform Trigger 1. The trigger currently is grayed.

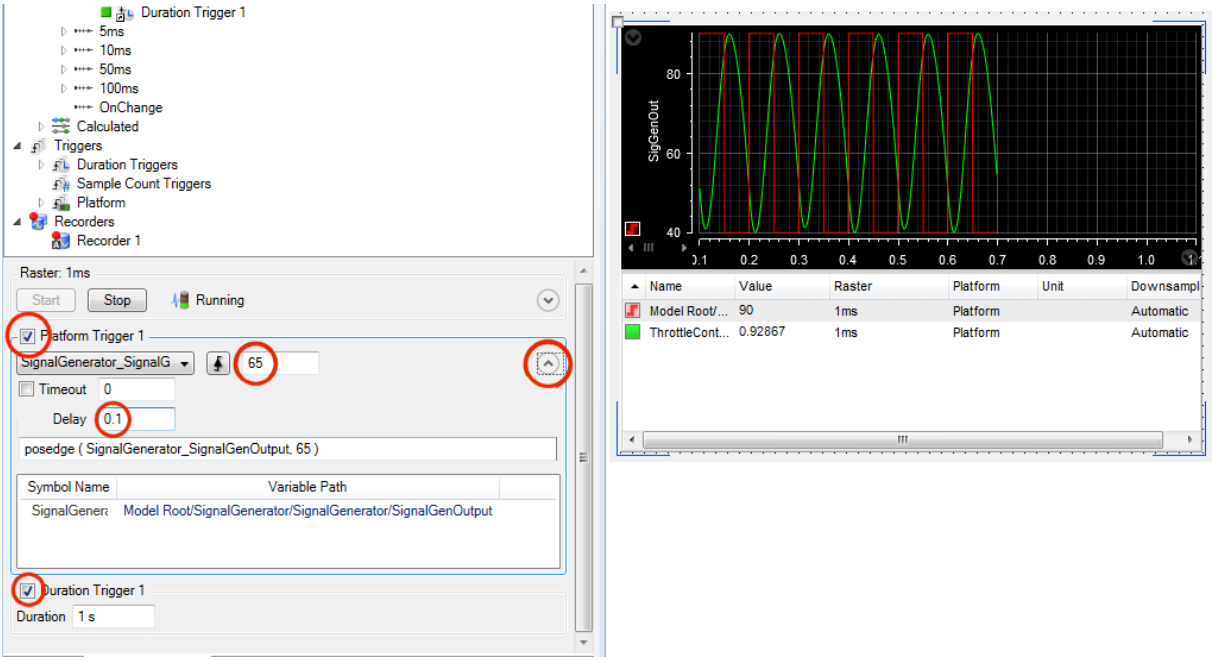
17 Expand the Time Plotter to get a list of all plotted variables by clicking the expand button in the lower right corner.



18 Drag the SignalGenOutput variable to the expanded Variable mappings field as shown below.



19 Make the specifications as shown in the following illustration:



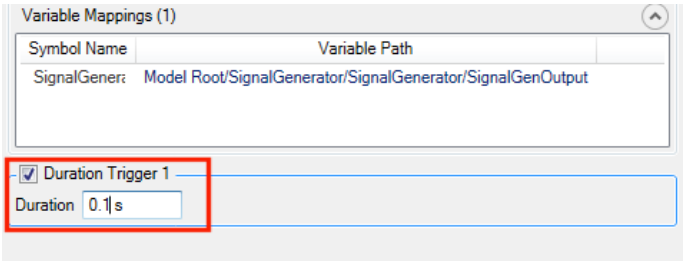
Setting	Value
Platform Trigger 1 checkbox	Selected
Reference Value edit field	`65`
Delay [s] edit field (click the expand button if hidden)	`0.1`

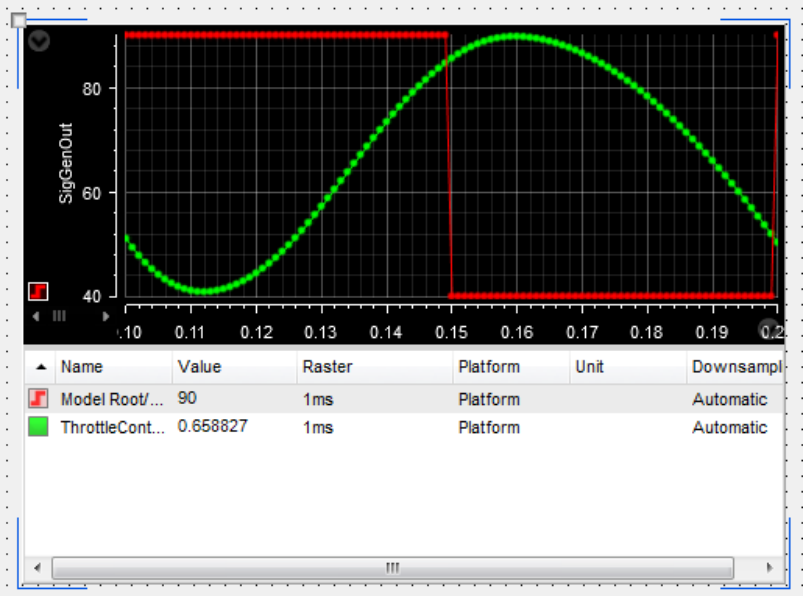
20 In the Measurement Configuration controlbar, click Platform - 1ms - Duration Trigger 1.

21 Make the specifications as shown in the following illustration:

Setting	Value
Duration Trigger 1 checkbox	Selected
Duration edit field	`0.1 s`

The plotter now displays the data stream of the variable values when triggered.





22 On the File ribbon, click Save Project or press **Ctrl** + **Shift** + **S** to save the project/experiment.

Result

You have measured the values of the variables.

Next steps

For instructions on changing the parameters of a running application with ControlDesk, refer to How to Change Parameter Values of a Running Application on page 65.

Related topics

Basics

Demo Projects/Experiments and Related Files.....57

HowTos

How to Change Parameter Values of a Running Application.....65

How to Change Parameter Values of a Running Application

Objective

You can change the parameter values of a running application with ControlDesk.

Demo project/experiment

For a short description of the demo project/experiment, refer to Demo Projects/Experiments and Related Files on page 57.

Preconditions

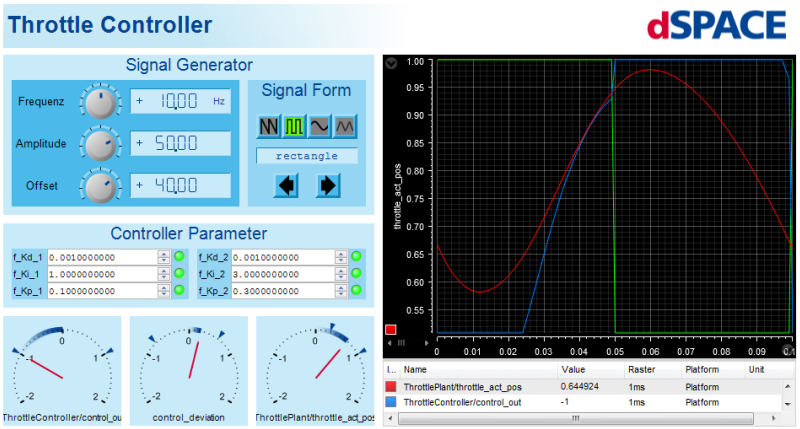
The board must be registered. Refer to How to Register a dSPACE System on page 26.

Method

To change parameter values of a running application

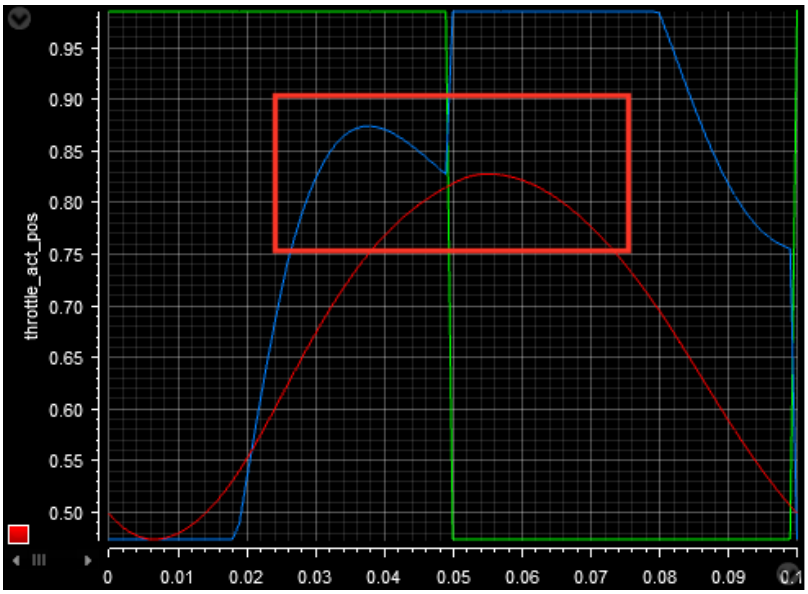
1. Start ControlDesk.
2. On the File ribbon, click Open – Open Project + Experiment to open the DS1007 Demo project/experiment.

   ControlDesk opens the project/experiment together with all the related files including the SDF file.
3. On the Home ribbon, click Status Control – Start Measuring.
4. Change to the throttle controller layout.



5. To change a parameter value while measuring, enter a new variable value: In the value field of the f_Kd_1 variable, change the value from `0.001` to `0.002`.

You can see that the plotter display changes slightly.



You have changed the parameter values during the measurement.

Result

You have changed parameter values of a running application.

Related topics

Basics

Demo Projects/Experiments and Related Files..... 57

HowTos

How to Measure Variable Values..... 58

Glossary

Introduction

Briefly explains the most important expressions and naming conventions used when you get started with dSPACE systems in connection with ControlDesk.

Where to go from here

Information in this section

A

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

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.

B

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

C

Calibration Changing the parameter values of real-time applications, ECU applications, or VPUs.

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

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.

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.

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.

D

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>`

dSPACE system A hardware system such as MicroAutoBox II or MicroLabBox on which a 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 only.

E

Environment model A model that represents a part or all of the ECU's environment in a simulation scenario.

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.

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.

F

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.

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.

G

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.

I

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 layouts.

Instrument Navigator A controlbar that displays a tree with all the instruments of the active layout and all the variables that are connected to them. The Instrument Navigator's main function is easy selection of instruments in complex layouts.

Instrument Selector A controlbar that provides access to ControlDesk's instruments. The instruments can be placed on a layout via double-click or drag & drop.

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.

L

Layout A window with instruments connected to variables of one or more simulation models.

Layout Navigator A controlbar that displays all opened layouts. It can be used for switching between layouts.

M

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 Configuration A controlbar that allows you to configure measurement and recording.

Measurement raster Specification of how often a value of a variable is updated during a measurement. A measurement raster is derived from a measurement service.

Messages controlbar A controlbar displaying a history of all error and warning messages that occur during work with ControlDesk.

MicroAutoBox platform A platform that provides access to a MicroAutoBox II connected to the host PC for function prototyping purposes such as bypassing.

Modular system A dSPACE processor board and one or more I/O boards connected to it.

O

Offline State in which the parameter values of hardware or a VPU in the current experiment cannot be changed. This applies regardless of whether or not the host PC is physically connected to the hardware.

P

Parameter Any variable type that can be calibrated.

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).

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.

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.

Properties controlbar A controlbar providing access to the properties of, for example, platforms/devices, layouts/instruments, and measurement/recording configurations.

R

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.

Recorder An object in the Measurement Configuration controlbar that specifies and executes the recording of variables according to a specific measurement configuration.

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.

S

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.

Single-processor system A system that is based on one dSPACE processor or controller board.

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.

Stop trigger A trigger that is used, for example, to stop a measurement raster.

T

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 services:

- Duration trigger
- Platform trigger

U

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.

V

Variable Any parameter or measurement variable defined in a variable description. ControlDesk provides various instruments to visualize variables.

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
- Value block

Variable description A file describing the variables in a simulation application, which are available for measurement, calibration, and stimulation.

Variables controlbar A controlbar that provides access to the variables of the currently open experiment.