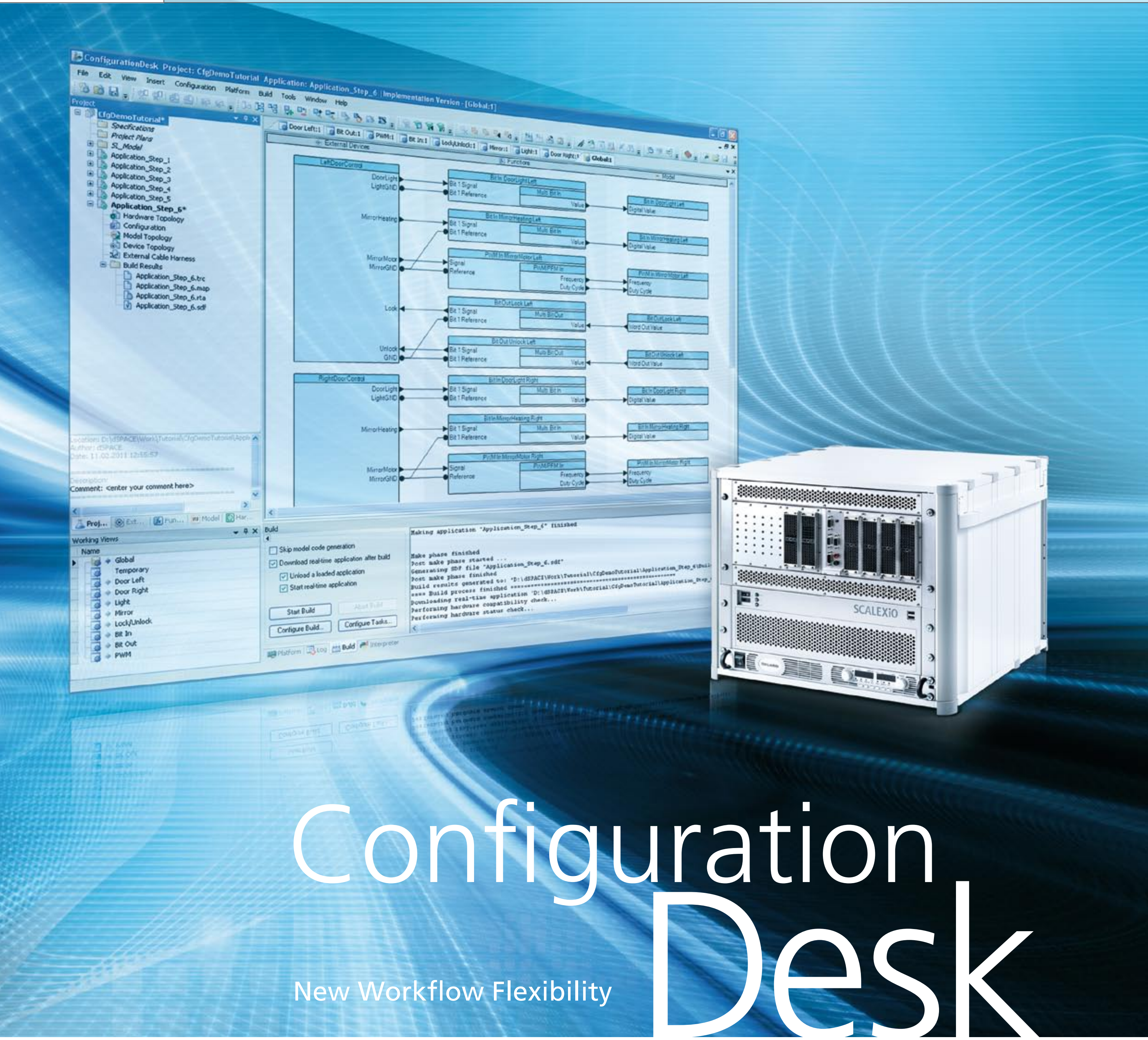


Location: D:\dSPACE\Work\Tutorial\CfgDemoTutorial\Appli
Author: dSPACE
Date: 11.02.2011 12:55:57
Description
Comment: <enter your comment here>

☐ Skip model code generation
☑ Download real-time application after build
☑ Unload a loaded application
☑ Start real-time application

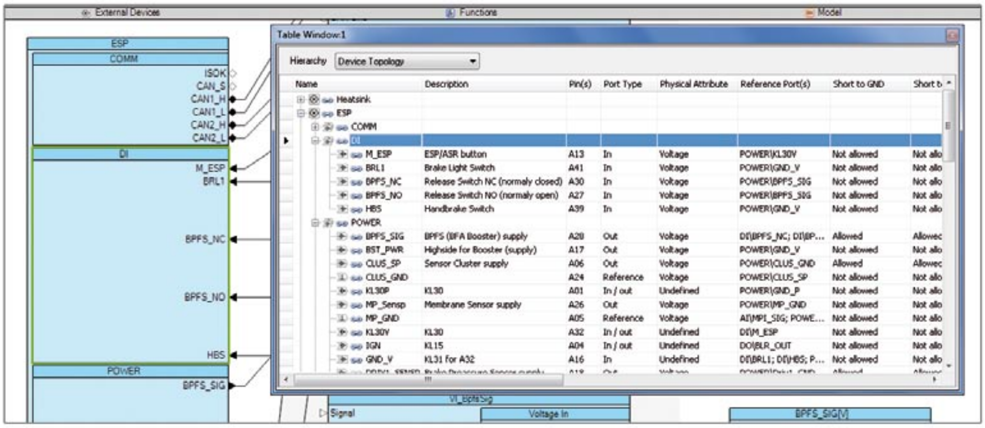
```
Making application "Application_Step_6" finished


Make phase finished
Post make phase started ...
Generating SDF file "Application_Step_6.sdf"
Post make phase finished
Build results generated to: "D:\dSPACE\Work\Tutorial\CfgDemoTutorial\Application_Step_6\Buil
==== Build process finished ===========================================
Downloading real-time application 'D:\dSPACE\Work\Tutorial\CfgDemoTutorial\Application_Step_
Performing hardware compatibility check...
Performing hardware status check...
```

ConfigurationDesk

New Workflow Flexibility

ConfigurationDesk is the configuration software for the new SCALEXIO simulator. Together, ConfigurationDesk and SCALEXIO are setting new standards in flexibility and reusability for hardware-in-the-loop simulation.



Hierarchy: Device Topology

Name	Description	Pin(s)	Port Type	Physical Attribute	Reference Port(s)	Short to GND	Short t
Heatsink							
ESP							
COMM							
DI							
M_ESP	ESP/ASR button	A13	In	Voltage	POWER\KL30V	Not allowed	Not allo
BRL1	Brake Light Switch	A41	In	Voltage	POWER\GND_V	Not allowed	Not allo
BPFS_NC	Release Switch NC (normaly closed)	A30	In	Voltage	POWER\BPFS_SIG	Not allowed	Not allo
BPFS_NO	Release Switch NO (normaly open)	A27	In	Voltage	POWER\BPFS_SIG	Not allowed	Not allo
HBS	Handbrake Switch	A39	In	Voltage	POWER\GND_V	Not allowed	Not allo
POWER							
BPFS_SIG	BPFS (BFA Booster) supply	A28	Out	Voltage	DI\BPFS_NC; DI\BP...	Allowed	Allowec
BST_PWR	Highside for Booster (supply)	A17	Out	Voltage	POWER\GND_V	Not allowed	Not allo
CLUS_SP	Sensor Cluster supply	A06	Out	Voltage	POWER\CLUS_GND	Allowed	Allowec
CLUS_GND		A24	Reference	Voltage	POWER\CLUS_SP	Not allowed	Not allo
KL30P	KL30	A01	In / out	Undefined	POWER\GND_P	Not allowed	Not allo
MP_Sensp	Membrane Sensor supply	A26	Out	Voltage	POWER\MP_GND	Not allowed	Not allo
MP_GND		A05	Reference	Voltage	AI\MPI_SIG; POWE...	Not allowed	Not allo
KL30V	KL30	A32	In / out	Undefined	DI\M_ESP	Not allowed	Not allo
IGN	KL15	A04	In / out	Undefined	DO\BLR_OUT	Not allowed	Not allo
GND_V	KL31 for A32	A16	In	Undefined	DI\BRL1; DI\HBS; P...	Not allowed	Not allo

Figure 1: Interface description for the connected devices.

Changed Requirements

In the field of hardware-in-the-loop (HIL) simulation, almost all companies now have specialized departments that develop and maintain different parts of projects (models, I/O configuration, etc.). These subprojects have to be combined flexibly to form a complete project, and the resulting HIL system has to be fully documented. The documentation must cover the connected devices i.e., electronic control units (ECUs) and real loads, the cable harness between the HIL system and the ECUs, the HIL hardware components with their configurations, and the environment model used for testing.

Project Management with ConfigurationDesk

dSPACE designed ConfigurationDesk® with an intuitive, graphical user interface that supports the above requirements. The new Implementation Version is the successor to the well-known Real-Time Interface (RTI) for configuring hardware and implementing real-time applications on SCALEXIO®, dSPACE's new HIL technology. The SCALEXIO hardware is completely configured with ConfigurationDesk, in other words, no settings have to be made by jumpers or any other hardware. In addition, ConfigurationDesk documents the entire HIL system, consisting of external devices, the cable harness, the dSPACE hardware with its configurations, and the environment model with all the interfaces to the I/O.

Interface Description of the Connected Devices

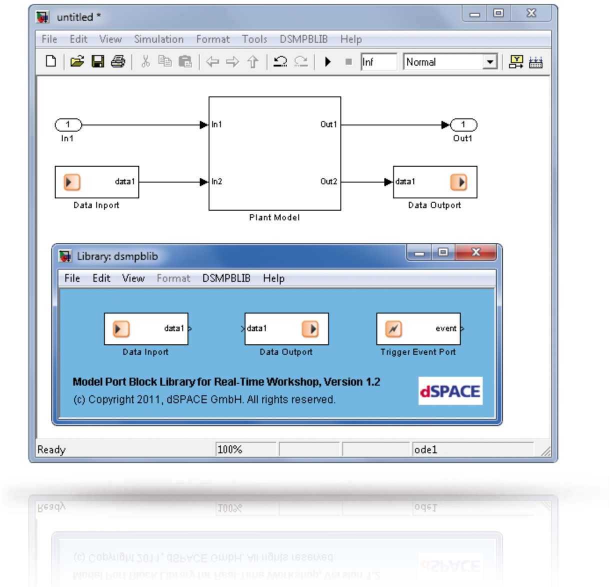
Any HIL project begins with defining the electrical interface of the ECU under test. This requires information on the number of connectors and pins, and also more abstract data such as logical signal names with descriptions, the signal direction (the ECU's input or output), physical signal properties (voltage or current signal), and information on whether a specific electrical failure can be switched to an ECU pin and which loads the ECU expects on that pin. This data can be entered in the tool manually (Figure 1) or imported via a Microsoft®Excel® list.

Preparing the Environment Models

The environment models are usually available as MATLAB®/Simulink® models. To connect the I/O to a model, dSPACE provides a Simulink library containing what are called model port blocks for input, output and trigger signals (Figure 2). These blocks can be inserted on any level in the model. If all the signals for I/O connection are on the top model level, standard Simulink In blocks and Out blocks can be used as an alternative. Initially, the signal chain ends at these interface blocks, meaning that no hardware-specific information is placed in the Simulink model. This makes it very easy to reuse the same model in different HIL projects.

From the ECU to the Model

All the components are brought together in ConfigurationDesk to form an overall HIL project. In an easy-to-read three-column view, the inter-



Model Port Block Library for Real-Time Workshop, Version 1.2

(c) Copyright 2011, dSPACE GmbH. All rights reserved.

Figure 2: Simulink model interface.

faces of the connected devices (left column) are linked to the HIL system's I/O functionality (center column), which in turn is linked to the model's interfaces (right column). ConfigurationDesk can make suggestions on suitable I/O functionalities for the connected devices. For example, when an ECU pin provides a voltage signal, the tool suggests an HIL I/O function for voltage measurement. You can also select I/O functions from a library. The tool also warns you if there are any wrong connections such as ECU outputs that are accidentally connected to HIL outputs instead of to HIL inputs.

The I/O functions are connected to the model interfaces (right column) in the next work step. There are two different ways to do this: Either ConfigurationDesk generates a model interface description that fits the I/O functions and exports it to a Simulink model, or you yourself prepare the model in Simulink by inserting model port blocks. A model analysis is then performed to transfer the interface description to ConfigurationDesk. Initially, the I/O functions define only the functionalities that the simulator has to execute (for example, PWM signal generation), and not the SCALEXIO hardware that will provide that functionality. In most cases, the same functionality can be provided on different hardware channels or channel types. The necessary hardware channels are therefore assigned to the I/O functions in a separate configuration step called hardware resource assignment. This can be done manually, or automatically by ConfigurationDesk. If signal properties such as the maximum power consumption cannot be handled by a single hardware channel, ConfigurationDesk determines the number of channels required for the channel type and assigns them to the I/O function.

Calculating the Cable Harness

ConfigurationDesk not only computes the data for connecting the HIL system and the ECU, but can also calculate data for connecting the HIL system to other external devices such as real loads. All the pins to be connected are listed on an Excel spreadsheet, together with the connections between the simulator pins or direct connections between the ECUs and the real loads, making it very easy to assemble the actual cable harness.

Controlling the Build Process

All the phases of the build process are controlled from ConfigurationDesk. The model C code is generated by MATLAB/Simulink. For the I/O functionality, ConfigurationDesk generates extra code parts. It is also possible to switch off individual phases. For example, if only the I/O functionality has been modified, generation time can be saved by deactivating model code generation and reusing “old” model code. When all the C code parts have been generated, they are compiled and linked, and a real-time application is generated from them. The application can then be loaded to a SCALEXIO system for execution.

Flexible Work Steps

The sequence of work steps described above is only one of many possible workflows. Moreover, not all the elements of the signal chain are required in every case. Some examples: Information on the connected devices is not necessary for HIL simulation itself. A build process can be carried out even if only the I/O functions and the behavior model are available. The left column is primarily necessary for computing the external cable harness, and for documentation.

Project Clarity

Any HIL project rapidly accumulates large quantities of signal chains, so the ability to concentrate on individual aspects is a great help. ConfigurationDesk therefore gives you options for creating different views. For example, with just a few clicks you can display only signal chains that contain a voltage measurement, or create a view of the signal chains that are connected to one specific ECU connector. A global view containing all the elements of a project is also always available.

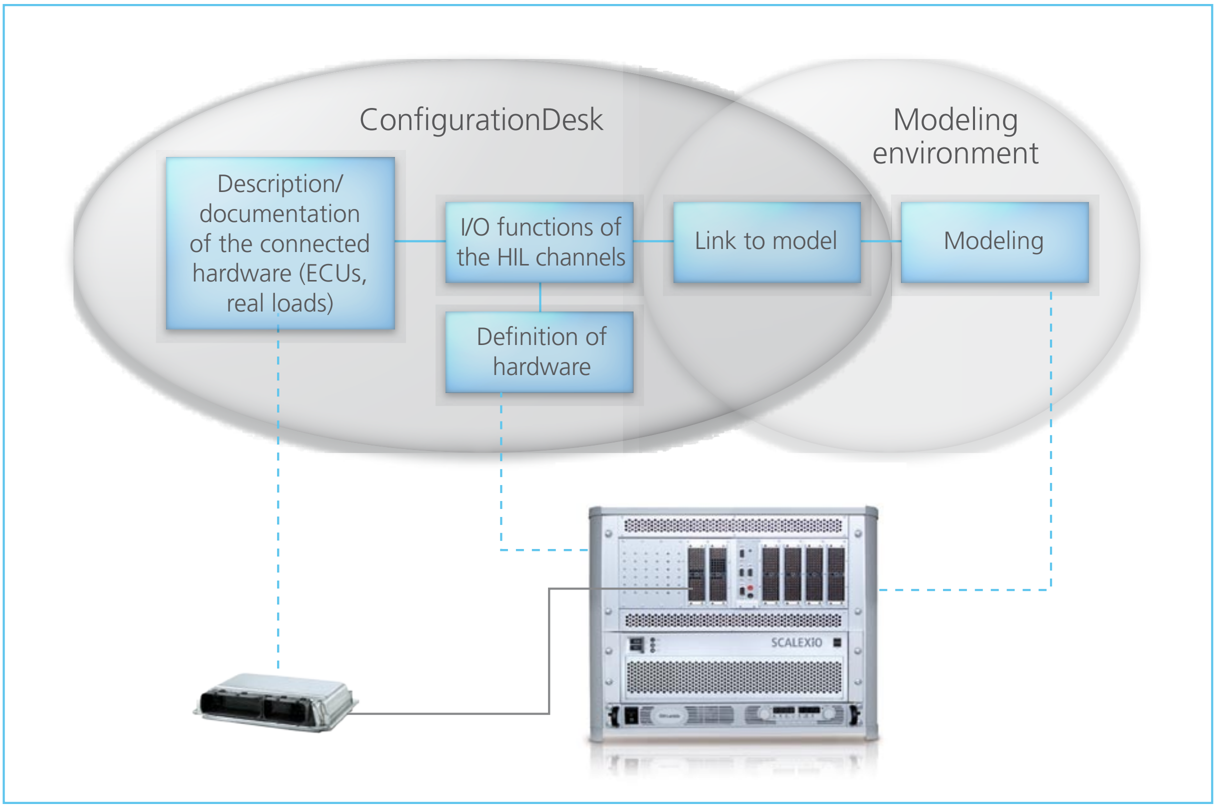
Reusing Subcomponents

The subcomponents of a project are very easy to reuse (Figure 3). For example, an ECU description can be imported into a new project and adapted. The same Simulink model can easily be used for different projects. Because the model does not contain details of the I/O hardware, completely different I/O functionalities can be connected to the same model interfaces. All subcomponents are saved to separate XML files so that they can be managed in a version control tool. ■

The Benefits at a Glance

- ConfigurationDesk makes working in distributed teams very easy: For example, Simulink modeling can be performed separately from (and independently of) hardware configuration.
- The modular design of ConfigurationDesk makes it easy to reuse subcomponents.
- The flexible SCALEXIO hardware can be optimally utilized because the functionality and the hardware are configured separately.
- ConfigurationDesk provides comprehensive documentation on the entire HIL project.

Figure 3: Signal chain on the HIL system and assigned subcomponents in ConfigurationDesk.


ConfigurationDesk
Modeling environment
Description/ documentation of the connected hardware (ECUs, real loads)
I/O functions of the HIL channels
Link to model
Modeling
Definition of hardware
SCALEXIO