



# Real Tests for virtual ECUs

The trend is clear: Use the ECU models from the development process for simulation, and save a lot of time, money and work by early testing and validation. dSPACE is helping developers do this by introducing the simulation of virtual ECUs.

*Dr. Krügel, what are virtual ECUs?*  
Nowadays, the application software and basic software for an ECU are available in an early phase of its development. If you integrate these two levels, bring them together, you get a virtual electronic control unit or V-ECU. A V-ECU is a realistic ECU model that can be executed on a simulation platform such as a PC. Thanks to the AUTOSAR standard, integration like this is much easier today than it would be without stan-

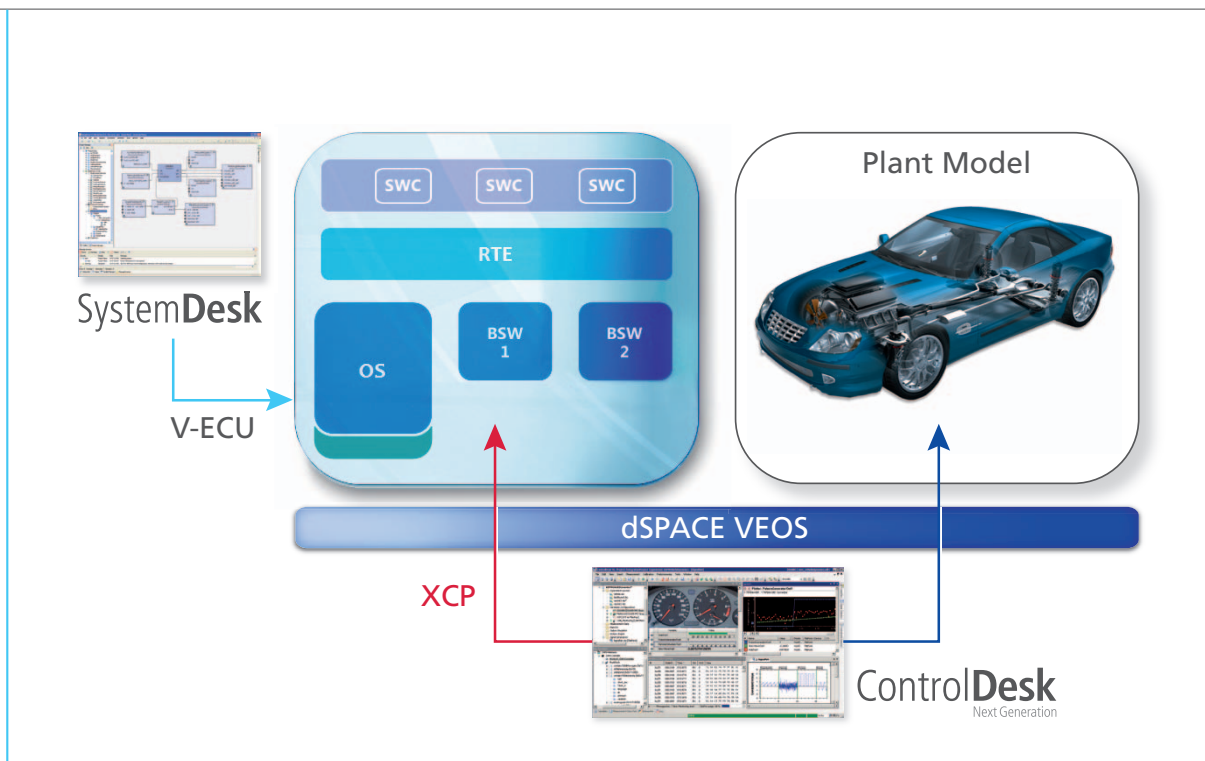
dardization. More and more customers were asking us for this form of integration and other potential uses.

#### *How do customers benefit from using virtual ECUs?*

Today, ECU architects can easily lose track of things in a complex system. But by generating a V-ECU, they can obtain a realistic representation of the future ECU as early as the development phase, and simulate it offline on a PC.

They can perform consistency tests, check the plausibility of interfaces, and verify task scheduling very early on. And because ECU functionality can be tried out at an early stage, it can be verified even before the first ECU prototypes are available.

HIL system users benefit in other ways: They can use existing V-ECUs to test the actual HIL tests in advance and to optimize HIL simulator utilization – for example, by running



Use case for VET: Integration and virtual testing of ECUs. SystemDesk is used to model an ECU architecture and to integrate the SWCs and BSWCs that were implemented. Together with the controlled system model, the behavior of the ECU software can be simulated with the offline simulator dSPACE VEOS. The offline simulation is visualized and controlled in ControlDesk Next Generation.

## “Bring your ECU functions to life and test them early – it’s the way of the future!”

Dr. Karsten Krügel, dSPACE GmbH

numerous variant tests beforehand. When virtual electronic control units are used for validation in these application cases and others, we call this “virtual ECU testing”, or VET for short.

### What dSPACE products are used for VET?

We believe that V-ECUs will play an important role from beginning to end of the development process, so we are integrating almost our entire tool chain. In the first phase, TargetLink® is the production code generator, and our SystemDesk® AUTOSAR architecture tool is used to generate V-ECUs. The controlled system models required for closed-loop scenarios are provided by dSPACE ASMs. ControlDesk® Next Generation enables visualization and experimentation. Our offline simulator dSPACE VEOS is used for PC-based simulation. And because all of our

solutions are open, third-party products can be integrated at numerous points in the development process. We are already supporting numerous use scenarios for virtual ECUs, and more will be added, drawing on our wealth of experience in the HIL and rapid control prototyping fields. The initial response from our customers shows that we are on the right road.

### What other developments are planned?

We see enormous potential in automating the testing of virtual ECUs. Like HIL tests, these should run autonomously overnight or on weekends. Seamless use and reuse of these tests are also key issues. The same tests that were executed on a PC must be reused later on the HIL simulator. This also applies to models, data sets and layouts. And we will continue to extend the indi-

vidual tools and optimize how they interact in VET application scenarios.

Dr. Krügel, thank you for talking to us.

### Dr. Karsten Krügel

Dr. rer. nat. Karsten Krügel is a product manager at dSPACE GmbH and responsible for topics such as offline simulation and virtual validation.



# Creating and Simulating Virtual ECUs

## Introduction

With the dSPACE tool chain, architectures and function models are created according to the AUTOSAR standard, and then brought together and integrated to form a virtual ECU (V-ECU). The resulting V-ECU is connected to a model of the controlled system to form a simulation system that is executed in offline simulation on a PC, and visualized and tested with experiment tools.

The following example of an adaptive cruise control (ACC) system shows the workflow with the dSPACE products TargetLink®, SystemDesk®, the dSPACE offline simulator VEOS and ControlDesk® Next Generation.

## ACC System

The ACC system used as a reference example here consists of the function models for distance and speed control, which will be integrated on the ACC ECU later, and also of the controlled system model. This represents the vehicle environment for the ACC ECU, so it contains elements such as models for transmission control, look-up tables representing the engine, and a radar sensor for distance measurement. An additional part of the controlled system model is a preceding reference vehicle that drives at different speeds in different driving scenarios.

## Creating AUTOSAR-Compliant Model Code

In this example of a development process, the ACC system's two controller models, Distance and Speed, are developed with TargetLink. The function developers model the interfaces and runnables for them according to AUTOSAR. AUTOSAR-compliant model code and a corresponding variable description (ASAP2) are generated for each TargetLink model. To bring the model parts together in a software architecture in SystemDesk, they are exported as AUTOSAR software components (SWCs) in a container that contains the generated code and ASAP2 files plus the associated AUTOSAR description files (ARXML).

## Modeling AUTOSAR Software Architectures

After export from TargetLink, the containers for the Distance and Speed controller models can be directly imported into SystemDesk. With this bottom-up procedure, the SWCs and all the describing AUTOSAR elements such as ports, interfaces and data types and their internal behaviors are now available in SystemDesk. The compatible interfaces of the Distance and Speed components are then connected in a software architecture.

## Generating Virtual ECUs

The software architecture and the SWCs it contains are mapped to a system with one ECU that was modeled in SystemDesk. OS tasks are created during ECU configuration, and the runnables modeled in TargetLink are assigned to the tasks. The ECU system can be created and configured manually, or alternatively, the process can be automated via SystemDesk. Once the ECU has

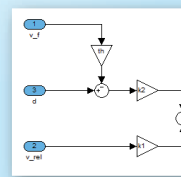
been created, the AUTOSAR run-time environment (RTE) is generated to implement the connections between the application's components to the basic software.

To prepare the ECU for closed-loop simulation with a controlled system model, the open interfaces of the SWCs are defined as the ECU's inputs and outputs, and then the V-ECU is generated.

## Function models

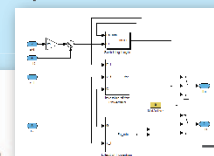
### TargetLink

#### Distance



ARXML .c, .a2l

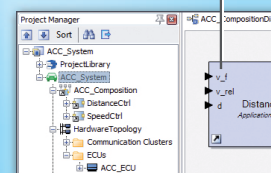
### Speed



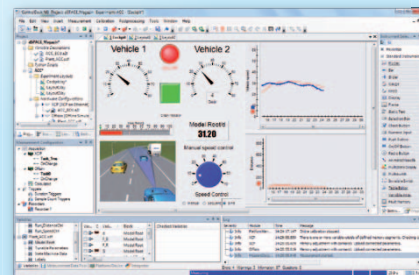
ARXML .c, .a2l

## Architecture/ ECU system

### SystemDesk



## ControlDesk Next Generation

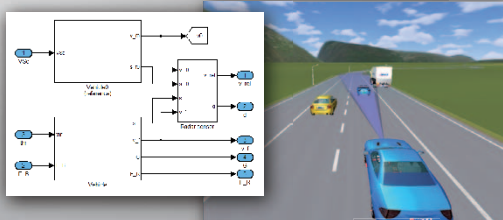


.a2l

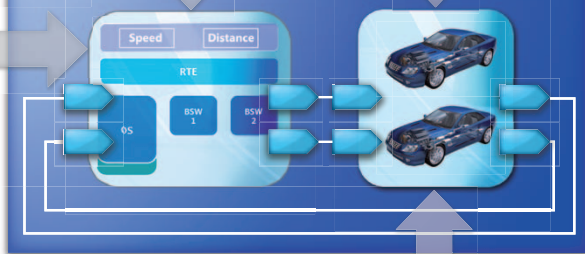
.trc

## Controlled system model

### Simulink



### VEOS



### Simulation system

#### Controlled System Model and Creation of the Simulation System

A controlled system model mirroring the physical behavior of the two vehicles and the radar sensor is used to simulate the overall system. Thus, the ACC control in the V-ECU can be tested in advance virtually, and its behavior in the overall system can be visualized with ControlDesk Next Generation. The controlled system model is

created with MATLAB®/Simulink®/Stateflow® and can be reused later in other verification steps such as HIL simulation.

Users can connect the V-ECU and the controlled system model via their defined inputs and outputs to create the simulation system. Because the interfaces have the same names, the connections can be defined quickly with one click.

#### dSPACE VEOS

Ideally, the PC computation uses simulation technology that can handle the same tools, models, transmission protocols and variable descriptions that have been firm fixtures in the real-time world for many years. dSPACE VEOS is precisely the product that is needed here, providing seamless transitions between PC offline simulations and real-time simulations on a HIL simulator. The same variable descriptions are used for online and offline application cases, i.e., ASAP2 for ECUs and TRC for controlled system models. With VEOS, a complex overall system can be simulated on a developer PC for early validation in the development process.

#### Experimenting and Testing

ControlDesk Next Generation lets users work interactively with the virtual overall system, providing completely transparent access to variables in the controlled system model and the V-ECU. The V-ECU's measurement and calibration variables (such as internal ACC control parameters) and the controlled system model's variables (such as the distance between the two vehicles and their current speeds) can be visualized and adjusted via numerous instrument types: switches, displays, sliders, etc. Users can create photorealistic layouts with these in order to assess intuitively during offline simulation whether the simulated software components are working correctly. The resulting project data, e.g., layouts, measurement data and data sets, can easily be reused later on a HIL simulator.

#### What Next?

As of the end of 2012, it will be possible to access VEOS directly from AutomationDesk®, so that automated tests of virtual ECUs can be created and executed efficiently. It will also be possible to simulate virtual ECUs with dSPACE SCALEXIO®. This will enable users to test and validate virtual ECUs in combination with real ECUs. In addition, Real-Time Testing and the Signal Editor in ControlDesk Next Generation will support offline simulation.