

# Systematic AUTOSAR Migration

- Audi developing AUTOSAR software for shock absorber control
- Setting up an efficient tool chain
- Modeling and generating AUTOSAR-compliant software with TargetLink

The AUTOSAR standard offers new solutions for mastering increasing complexity as the number of different technologies, functions, and most of all, variants continues to grow. In addition, it ensures that software components are easy to reuse. Audi implemented a shock absorber control that complies with the AUTOSAR concept. This project provided important insights on setting up a tool chain in which TargetLink, dSPACE's production code generator, is used for modeling and generating AUTOSAR-compliant ECU software.

## The Challenge of Complexity

There are several ways of handling the complexity of ECU development. Approaches based on suitable software architectures and standardized hardware platforms look promising. The AUTOSAR standard's role here is as a manufacturer-independent starting point for designing software architectures. OEMs' concentration on designing AUTOSAR software components (SWCs) will enhance the quality of software due to the ease with which they can be reused.

## Easy to Reuse

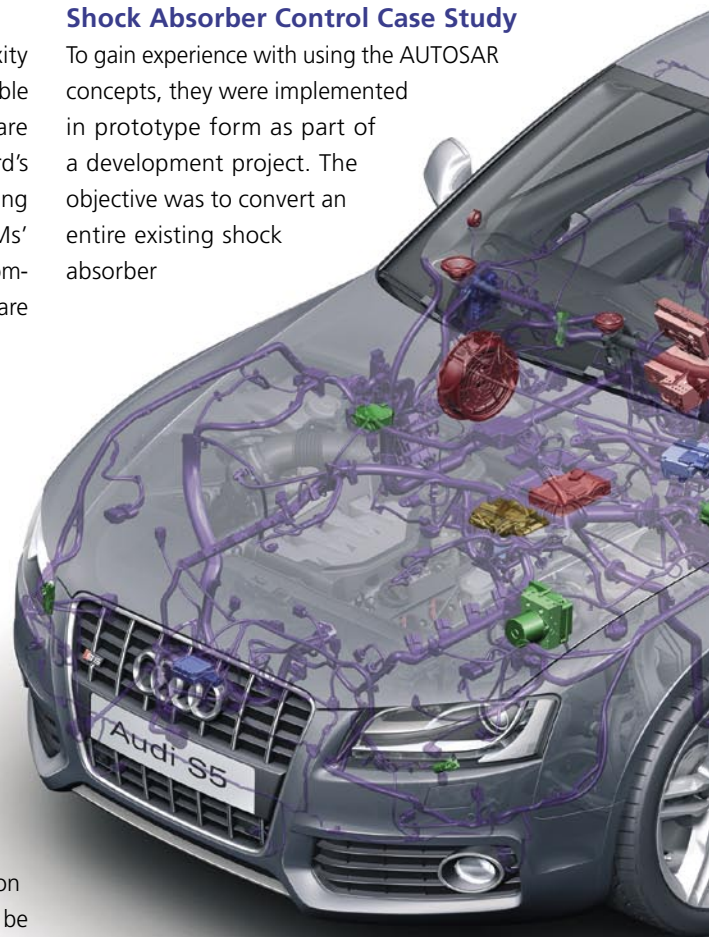
Because all function logics are encapsulated in AUTOSAR-compliant software components, items such as the processing algorithms for sensor data can be reused in different projects and with reference to different processor-specific requirements, without the code having to be adapted. This means that automobile manufacturers not only save costs, they also save time when launching new projects.

## Systematic Abstraction

Systematic abstraction of the system architecture means that explicit decisions on how the application software will be deployed on ECUs do not have to be made until a late stage of a project. This means that the logical software architecture can be designed at a very early stage, independently of hardware architecture. Individual functions can be moved from ECU to ECU at will, giving developers the freedom to perform function distribution or integration whenever they want: For example, they can delay it until after an intensive internal testing or exploration phase.

## Shock Absorber Control Case Study

To gain experience with using the AUTOSAR concepts, they were implemented in prototype form as part of a development project. The objective was to convert an entire existing shock absorber



control system into AUTOSAR-compatible software components by means of TargetLink, and to test it on a production-close vehicle as the prototype development platform. The control consists of four body acceleration sensors, four distance sensors, and four continuously

controlled shock absorbers. A central ECU evaluates the sensor signals and calculates the shock absorber control, taking into account further vehicle dynamics variables such as the steering angle, yaw rate, brake signal, lateral acceleration, vehicle speed, and engine torque. The ECU receives these variables from the vehicle's CAN bus. The ECU communicates with the active shock absorbers via a FlexRay bus.

### A Prototype Development Environment

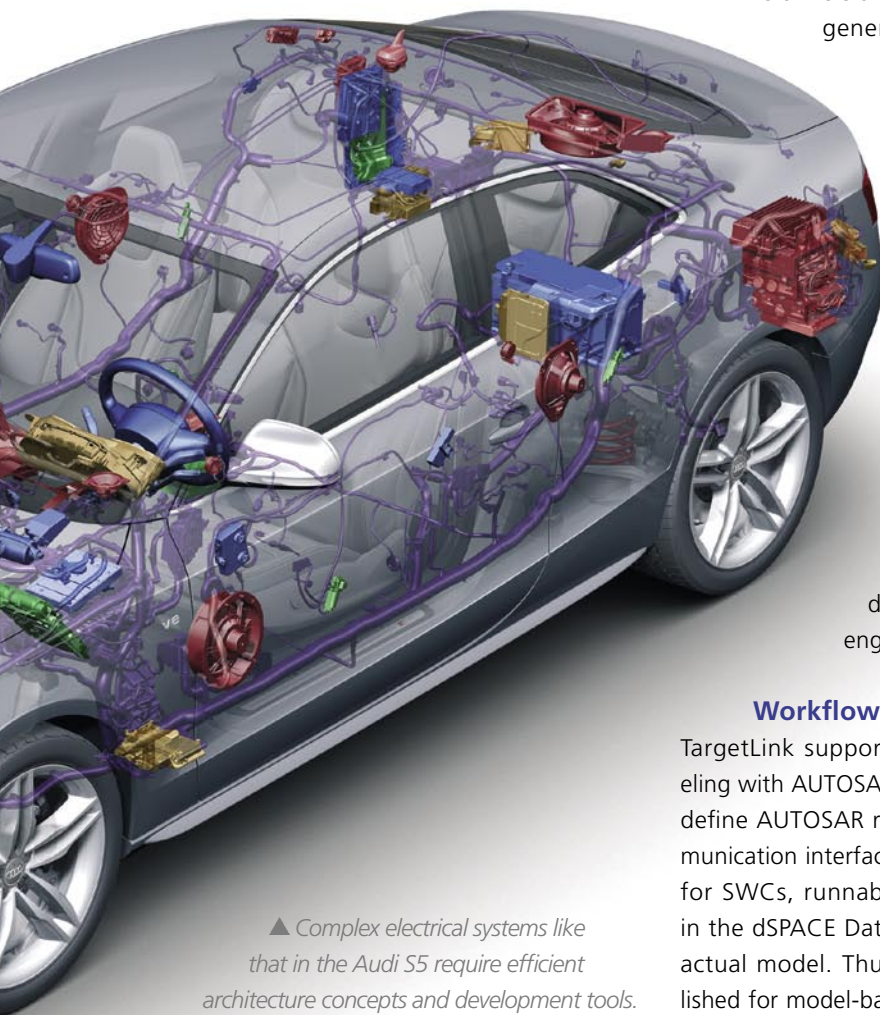
The tool chain currently in use is based on AUTOSAR Release 2.0. Standardized file formats allow

- Elektrot's EB tresos® for configuring the AUTOSAR-compliant basic software (e.g. OS) and generation of the RTE
- Configuration tools for the FlexRay stack
- Production-close ECU prototypes

### Model-Based Development of SWCs

From the point of view of function developers, the most important tool in an AUTOSAR tool chain is the modeling tool. It lets them implement and model their ideas simply, using tried and tested processes.

As of Version 2.2, dSPACE's production code generator TargetLink supports the design of AUTOSAR software components as well as conventional model-based development, and allows the corresponding target code to be generated automatically. Because communication and hardware connections are abstracted, function developers can concentrate fully on the actual application. To take production aspects into account from the beginning, it is advisable to give the function developers support from design engineers.



▲ *Complex electrical systems like that in the Audi S5 require efficient architecture concepts and development tools.*

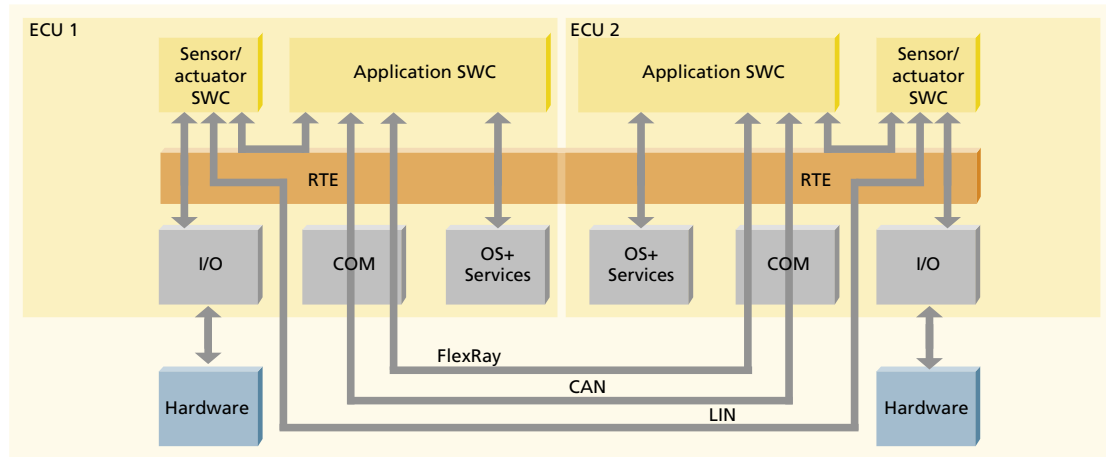
dedicated tools to be created. These make it easy to analyze other aspects of the overall system, such as time behaviors in the ECU. The following tools were used:

- TargetLink for model-based development and automatic production code generation for AUTOSAR SWCs

### Workflow with TargetLink

TargetLink supports AUTOSAR-compliant modeling with AUTOSAR blocks, which make it easy to define AUTOSAR runnables and model the communication interfaces. The AUTOSAR-specific data for SWCs, runnables, interfaces, etc., is stored in the dSPACE Data Dictionary and linked to the actual model. Thus, the entire workflow established for model-based design with TargetLink can also be applied to developing AUTOSAR software. Once an AUTOSAR function model has been designed, it can be simulated and tested with TargetLink at the model (MIL) and the software (SIL) level.

As well as generating the AUTOSAR-compliant code, TargetLink automatically produces the AUTOSAR software component descriptions.

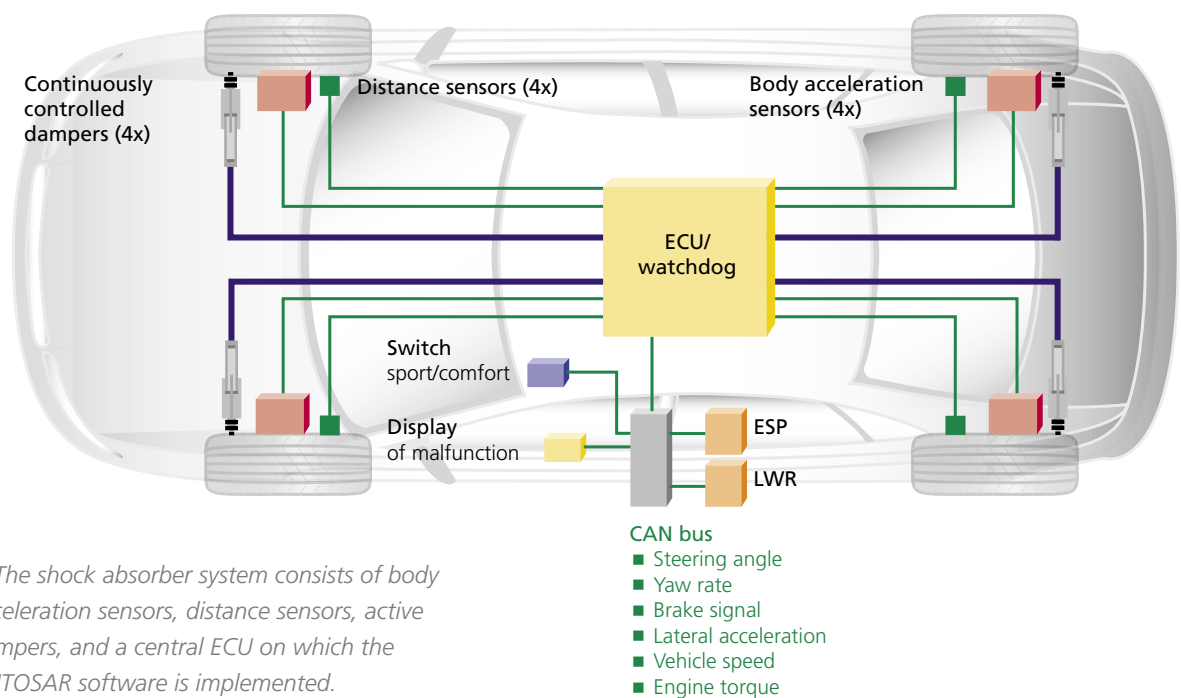


▲ Basic AUTOSAR structure. Several SWCs are distributed on two ECUs. The RTE allows the SWCs to communicate independently of the specific I/O hardware and bus that are used.

The shock absorber control is subdivided into several AUTOSAR software components, each comprising several runnables. In this specific case, communication between interrunnables was used intensively, though the AUTOSAR standard currently supports this only for scalar variables. Thanks to an Audi-specific TargetLink extension, however, the function developers can also work with vector signals, which TargetLink converts into code patterns for scalar variables. This simplifies the modeling process and also ensures compliance with AUTOSAR.

### Implementing the AUTOSAR Software

As part of SWC implementation on the ECU, the AUTOSAR operating system was configured and the run-time environment (RTE) was generated by EB tresos. To generate the RTE, the software component descriptions generated by TargetLink were imported into EB tresos, and the RTE was generated on the basis of the information in them. The last step was to configure the FlexRay drivers, which are in the basic software. The Infineon TriCore microcontroller family was used to set up the prototype hardware because of its good performance and broad-based connec-

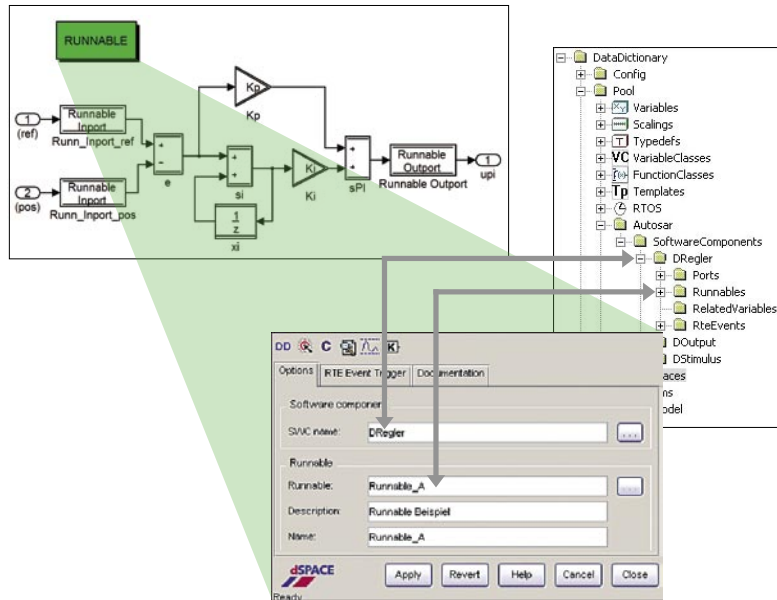


▲ The shock absorber system consists of body acceleration sensors, distance sensors, active dampers, and a central ECU on which the AUTOSAR software is implemented.



tions to typical automotive peripherals. These so-called “engineering device” microcontroller derivatives provide a powerful data interface for the necessary tests and for instrumentation tasks.

Successful function implementation on the production-close ECU was verified both in test drives and in tests on the simulator. These also clearly showed that the AUTOSAR-compliant code generated by TargetLink meets the production requirements on size and run-time behavior.



▲ With TargetLink, the AUTOSAR specifications stored in the dSPACE Data Dictionary can be linked directly to the model.

**Tool Chain Know-How**

The experience gained is being invested to set up an AUTOSAR tool chain for function developers to develop new vehicle features close-to-production and with high efficiency.

*“The support given to the AUTOSAR standard by dSPACE tools and close cooperation with dSPACE enable us to introduce AUTOSAR successfully.”*

**Dr. Karsten Schmidt, AUDI AG**

As a tool for model-based development and automatic code generation, TargetLink addresses the design of individual, clearly defined software components. The SystemDesk architecture tool is an obvious choice for overall system design, as it closes the gap in the systematic design of software systems. In addition, modeling and analyzing the temporal relations in single ECUs and in the overall system looks likely to grow in importance.

**The Charm of the AUTOSAR Idea**

As the advantages described here show, the AUTOSAR idea offers numerous options for increasing the efficiency of function development. A production-close framework can be used at an early stage in the development process, thereby minimizing potential

difficulties in cooperation between OEMs and suppliers. Interaction between different tools is the key to successful implementation of the AUTOSAR idea. dSPACE provides an excellent basis for this in the form of TargetLink and SystemDesk, together with defined file formats and open interfaces.

*Dr. Karsten Schmidt  
Frank Gesele  
AUDI AG  
Germany*



*Dr. Karsten Schmidt is responsible for the further development of the AUTOSAR tool chain in the chassis section at Audi.*



*Frank Gesele is team leader for vertical dynamics and proprietary software development in the chassis electronics section at Audi.*