Gentle Belts

Seat belts tightened the AUTOSAR way at Autoliv

In a crash, seat belts have to respond fast, without impacting too heavily on their wearers. The solution is to use pretensioners, which tighten the belts during the very first fractions of a second. Autoliv's active seat belt goes one step further by using a gentle pre-pretensioner that optimally softens belt impact. Its controllers use AUTOSAR-compliant software.

Seat Belt Pre-Crash Systems

At its location in Cergy near Paris, automotive supplier Autoliv is developing an electric seat belt pre-pretensioning system for production use by two different automobile manufacturers. This mechatronics system consists of a seat-belt retractor, an ECU and an electric motor (figure 1). As soon as emergency conditions are detected that are a potential pre-crash situation, such as panic braking or the vehicle being oversteered or understeered, the seat belt is electrically tightened to restrain its wearer before the accident. Then if an accident actually does occur, each vehicle occupant is ideally positioned in his or her seat, and the entire passive safety system, comprising pyrotechnical seat-belts and airbags, can deliver the best protection. If there is no accident, the belt is simply released. Not only does the device help reduce injuries in the event of an accident, it also delivers a warning to a driver who gets too close to safe driving limits, thereby acting as an active prevention device.

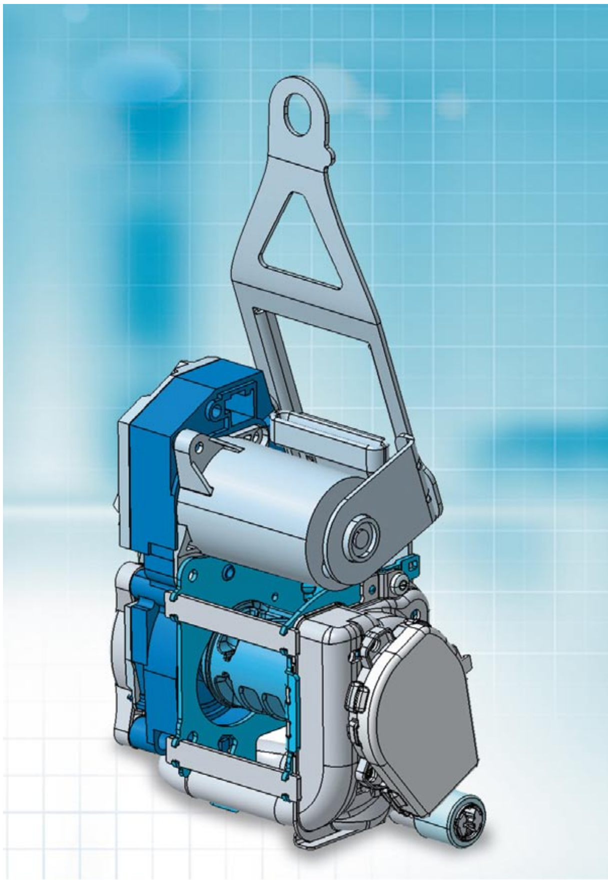
Additionally, the pre-pretensioner's comfort functions make the seat belt easy to put on and release, and dynamically adjust its tension to different driving situations.

The ECU processes data on the environment and on vehicle dynamics received from other ECUs via the vehicle's CAN bus, and provides control output to the electric motor for belt tensioning. The hardware platform for the ECU is a 32-bit microcontroller.

OEM-Specific Processes

The requirements for the ECU under development can be categorized as either functional or system-specific. The functional requirements for the application, in this case the belt tensioning logic, are implemented by Autoliv. They can take different forms: textual specifications, models, or initial software component descriptions in the ECU extract (see textbox

Figure1: Autoliv's seat belt pre-pretensioner holds the wearer in a better position before an accident occurs.



The AUTOSAR Workflow). To ensure smooth interaction with other ECUs installed in the vehicle, the OEM provides the system-specific requirements, for example, for ECU communication via the in-vehicle bus system (in this case, CAN). Depending on the OEM that Autoliv is working for, these requirements are also provided in the ECU extract obtained from the AUTOSAR system description, or they can take the form of a communication matrix (in this case, a DBC file describing the CAN communication).

Modeling the ECU Application with SystemDesk and TargetLink

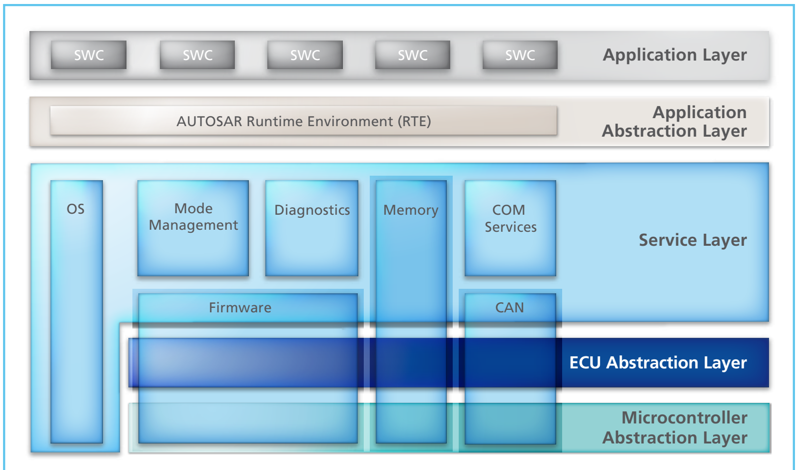
The model-based development of application software and correct configuration of AUTOSAR basic software, and the integration of all the software on the target platform, are essential steps in the AUTOSAR development process.

For the pre-pretensioner controller development, these tasks are being carried out by a combination of tools: the architecture software SystemDesk® and the code generator TargetLink® from dSPACE, and the configuration editor EB tresos® Studio from Elektrobit (EB).

To develop the AUTOSAR software of the ECUs, Autoliv introduced the design flow depicted in Figure 3. Working in SystemDesk, the software architect imports the requirements from the ECU extract as a starting point for creating a detailed software architecture. The descriptions of the AUTOSAR services are also read in as further building blocks. Autoliv utilizes all the available design options for the software architecture to maximize reusability across different projects. The descriptions of the individual software components are then passed to the function developers in AUTOSAR format.

The function developers import the AUTOSAR descriptions into TargetLink, where they can generate a frame model with the predefined inputs and outputs. The frame model is then filled either by reusing existing submodels or by modeling from scratch. The well-established model-in-the-loop (MIL) and software-in-the-loop (SIL) simulation methods are used for testing and validation. Then the AUTOSAR-compliant production code is generated automatically by TargetLink. The entire function development process is distributed across several different Autoliv locations.

Figure 2: Layered structure of the AUTOSAR software architecture.



The AUTOSAR Workflow

The Automotive Open System Architecture, or "AUTOSAR" for short, is a standard created by car manufacturers, ECU suppliers and tool providers to meet the increased requirements on the functionality and quality of in-vehicle software, while at the same time shortening development times. The standard AUTOSAR workflow begins with the OEM defining the system parameters, which comprise the network topology and ECU communication (messages, signals) as well as the distribution of application software to different ECUs in the network. The resulting system description is made available to the ECU manufacturers either as an AUTOSAR System Description file or as an "ECU extract from System Description" which contains only the data that are relevant for the specific ECU.

The OEM can specify the software architecture in advance with different granularities, after which the supplier develops single software components for it. Implementations with the desired behaviors are then produced for the individual software components, either by reusing existing software components or by developing new functionalities. For new developments, a model-based process can be used.

The descriptions of the software components provide a frame for developing a MATLAB®/Simulink® model from which the AUTOSAR-compliant production code is generated. The supplier configures the AUTOSAR basic software modules in accordance with the contracted AUTOSAR system description and also generates the code for these modules, which together with the code for the applications makes up the overall ECU software (figure 2).

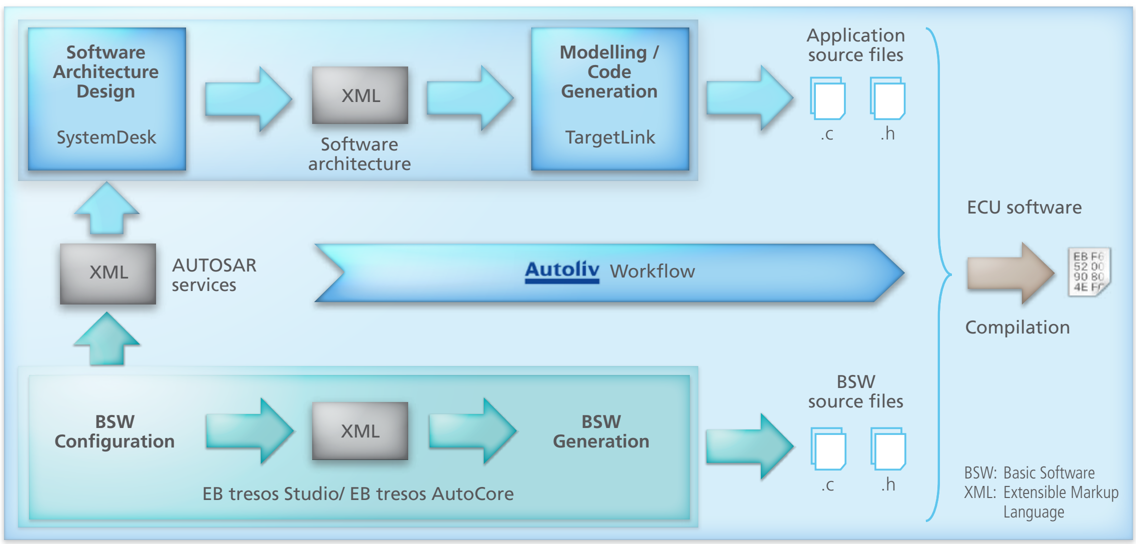
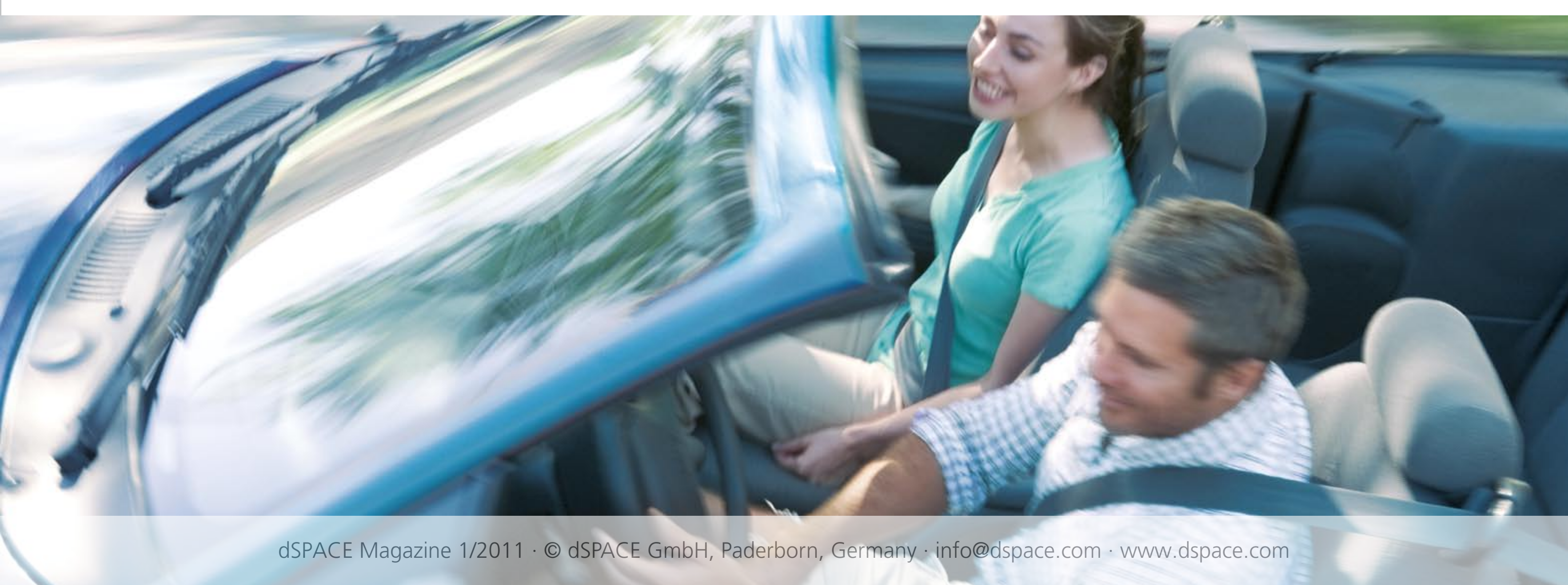


Figure 3: AUTOSAR design flow used at Autoliv.

Configuring the AUTOSAR Basic Software with EB tresos Studio

Autoliv's software integrator starts work in parallel to the application software development. The developer configures and generates the ECU's basic software with EB tresos Studio. Different OEMs specify the basic software to be used in different ways, using the EB tresos AutoCore as the basis. This contains the AUTOSAR modules, including the AUTOSAR run-time environment (RTE) and the hardware-specific microcontroller abstraction layer (MCAL). Different OEM-specific modules are added to the modules in the EB tresos AutoCore and integrated to create complete, OEM-specific basic software. One of AUTOSAR's main advantages comes into play here, enabling Autoliv to integrate the seat belt pre-pretensioner application into ECUs with different basic software implementations. It is essential that the software

> "Working with SystemDesk greatly improved software reusability across projects."
>
> *Claude Redon, Autoliv*



Strength in Unity

"Cooperate on standards, compete on implementation" is the philosophy behind the AUTOSAR consortium's standardization activities. Since 2006 cooperation is also a driving force for dSPACE and EB, both Premium Members in the consortium right from the beginning. Their joint efforts focus on their software tools, and on ensuring that they work together smoothly according to the AUTOSAR design methodology. dSPACE's contribution is its established software architecture and simulation tool SystemDesk and its production-proven code generator TargetLink, while Elektrobit provides production-ready AUTOSAR basic software with its products EB tresos AutoCore and EB tresos Studio. Together, these form a complete, well-coordinated tool chain covering AUTOSAR design methods from the software architecture description to model-based application development, to the configuration and generation of the electronic control unit's (ECU's) basic software on the target platform, thereby covering all the layers of the AUTOSAR standard software architecture as shown in figure 1. This extremely practical solution for ECU development benefits OEMs and their Tier 1 suppliers alike.

> "Early software verification was achieved by performing back-to-back tests with TargetLink."
>
> *Claude Redon, Autoliv*

components comply with the AUTOSAR standard (in this case, a mixture of AUTOSAR Release 3.0 and 3.1) and also with the interface specifications, so that they can be connected via the AUTOSAR run-time interface (RTE) with no problems.

From Architecture to Production Code

The AUTOSAR basic software is configured by reading the software architecture that was exported from SystemDesk, plus the ECU extract or the DBC file, into EB tresos Studio. This process results in a preconfigured RTE and preconfigured communication modules. To complete the RTE configuration, the data elements of the individual software components' ports are mapped to the signals in the bus communication (data-to-signal mapping), the runnables are assigned to operating system tasks (runnable-to-task mapping), and the service ports are connected to the appropriate ports on the software components (service port mapping).
Next, the integrator configures the parameters of the basic software modules in EB tresos Studio, partly using script automation. Then code is generated in EB tresos Studio to produce the complete source code for the AUTOSAR basic software. This is passed to the build environment together with the source code for the application components from TargetLink. The final result is the executable binary code, which is used to program the ECU. Figure 4 illustrates the resulting software layers and shows which tools are used for which steps.

Benefit from Cooperation

dSPACE's and EB's early and continued commitment to this demanding automobile industry standard is reflected in the tools that they supply. The tools combine perfectly to harness the two companies' core areas of expertise: the model-based development of application software and the creation of software run-time platforms for ECUs. The parts of the specification required for applying the AUTOSAR standard are mapped to the tools to make developers' work easier. The advantages of AUTOSAR for the reliable development of reusable software are combined with the model-based methods used to develop that software efficiently. The tools support the specification and generation of software, provide generators and automated steps, and help ensure seamless consistency and validation. The systematic use of standardized AUTOSAR descriptions ensures that the tools interact smoothly.
Wherever there are still specification gaps in the standard, the tool manufacturers consult to avoid incompatibilities. Pragmatic solutions are found and tested to prevent in advance any technical problems that might occur.
At Autoliv, the dSPACE and EB tresos products were supplemented by process consultation in introduction projects based on AUTOSAR. Experts from dSPACE and EB gave on-site support to ensure efficient tool use.

Project Experience

Autoliv is fully committed to the AUTOSAR standard in these projects. The initial investments for introducing AUTOSAR are already being off-

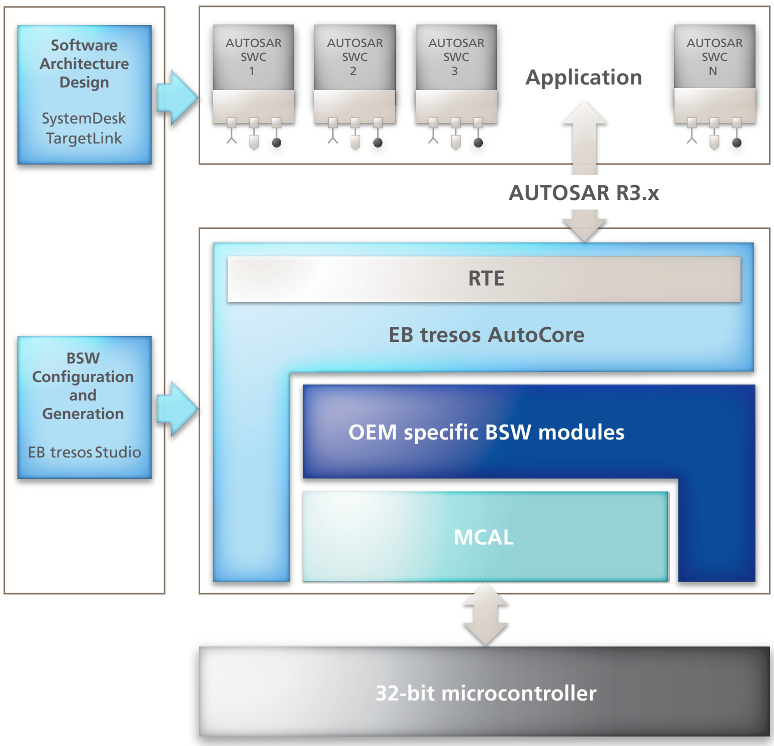


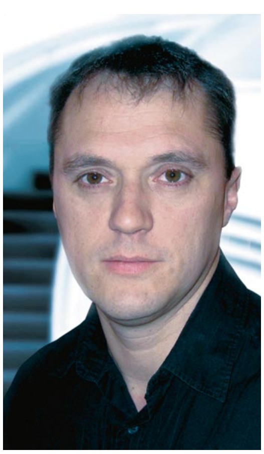
Figure 4: The structure of the final ECU software.

In Brief

For optimum safety, seat belt pre-pretensioners hold their wearers in a better position before an accident occurs. They also act as an active prevention device by delivering a warning to the driver who gets too close to safe driving limits. The controller for Autoliv's latest pre-pretensioner has been developed in compliance with the AUTOSAR standard. The design flow is based on the SystemDesk architecture software and the TargetLink code generator from dSPACE for developing the application layer. The basic software is configured with EB tresos Studio from Elektrobit. Easy software modularization and improved reuse of the application software are two of the benefits Autoliv experienced during development.

set by considerable improvements in software development. Two of the benefits are explicit software modularization and improved reuse of the application software. The systematic use of the standardized AUTOSAR exchange formats gives optimum support to all the individual process steps. The AUTOSAR methods are ideal for combining with model-based function development and automatic production code generation. The experience gathered and the results achieved in these projects show that the dSPACE and EB tools are a complete success in these very tool-supported developments. Continuous further development, based on new versions of the standard and on the requirements of ongoing projects, keeps the products up-to-date and guarantees proven tool interaction. ■

Peter Kirsch, Elektrobit
Claude Redon, Autoliv
Joachim Stroop, dSPACE



Pictured left to right are:

Claude Redon
Mr. Redon is a software architect with responsibility for the pre-pretensioner family at Autoliv in Cergy, France.

Joachim Stroop
Mr. Stroop is Lead Product Manager with responsibility for the SystemDesk tool at dSPACE GmbH in Paderborn, Germany.

Peter Kirsch
Mr. Kirsch is Product Manager with responsibility for the tool EB tresos Studio at Elektrobit Automotive GmbH, Erlangen, Germany.