The automobile industry sees its future in the continuous development of innovative, networked in-vehicle systems. One answer to the growing complexity this will bring is to perform systematic standardization based on AUTOSAR. Strategies for introducing the new standard are being investigated in three reference projects at DENSO CREATE. The company is intensively comparing current procedures and methods with those required by AUTOSAR, with the aim of solving possible conflicts and ensuring that the introduction of AUTOSAR at DENSO runs smoothly.

Process Requirements

DENSO CREATE, a 100% Japanese subsidiary of the DENSO CORPORATION with responsibility for the areas of IT and software development, has carried out a process optimization project for introducing AUTOSAR. AUTOSAR provides a way of describing software in great detail to make it easy to reuse. The various stages of an AUTOSAR-compliant development process produce descriptions of the software architecture, the overall system, and the system configurations for individual electronic control units (ECUs). However, there are also conventional development projects running. These have other design steps that must be integrated seamlessly and without conflicts. Three aspects are particularly important:

- It must be possible to define the function architecture with the necessary level of abstraction.
- A control engineer using tools such as Simulink® or dSPACE AutoBox to develop prototypes must have full freedom to design and must not be restricted by AUTOSAR.
- An ECU supplier must retain the ability to optimize the implementation of a software architecture in specific ways.

One Step at a Time

In three reference projects, DENSO CREATE is examining a sequence of major steps and methods which make up an AUTOSAR-compliant development process (figure 1). Architecture modeling and function algorithm modeling are treated separately in these steps:

Function Architecture Design: Define and visualize the necessary function blocks and signals. This step produces a formalized description of essential requirements and can be in the nature of a whiteboard. Early consistency checks can be performed with tool support.

Control Model Design: Add algorithms to the defined functions to complete them. This classic form of function modeling is performed with MATLAB®/Simulink® and TargetLink.

Network Topology Design: Define the ECUs and how they are networked.

Function Mapping & Communication Design: When the functions have been mapped to the ECUs, define the local and global communication.

Software Structure Design: Define the software structure to be implemented on the ECUs. It may be necessary to transform a structure so that software development meets the given requirements.

Implementable Model Design: Adapt the model to the selected software structure and refine it for the purpose of production code

DENSO CREATE investigates strategies for introducing the AUTOSAR standard

Talking AUTOSAR

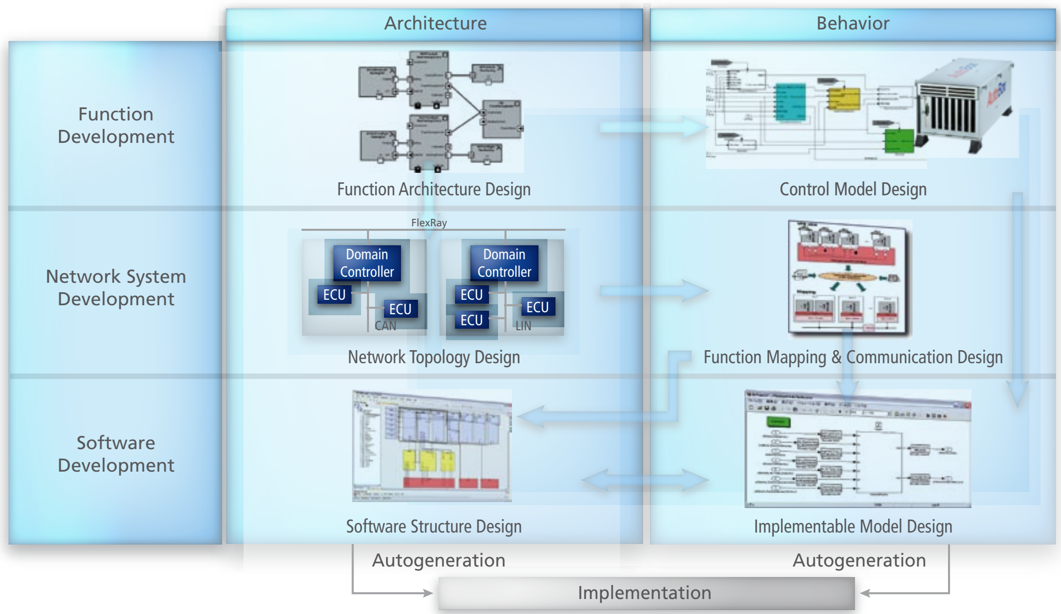

Figure 1: Sequence of major steps and methods in an architecture-based development process.

generation with TargetLink. Example: Add scaling information and link it to measurement and calibration variables.

Autocode Generation: When the design phase has been completed, generate the software components of the application layer from the implementation models by autocoding with TargetLink. Also generate the run-time environment (RTE), and configure and generate the basic software. Example: Generate the communication drivers from the network description.

Implementation: Finally, translate the C source code into object code, link the objects, and load the software to the ECU.

Each step has its own subset of AUTOSAR description elements specified for it. This ensures that only the really necessary elements are defined in each step.

Roles and Responsibilities for Optimum Efficiency

Efficiency is a major criterion for the new process. The individual steps were studied closely from this point of view to determine the necessary work methods, costs, and volumes. This made it possible to identify responsibilities, define roles, and optimize the procedures for work groups. It was found that control developments, for example, could be carried out in largely the same way as before, without the responsible engineers having an in-depth understanding of AUTOSAR. AUTOSAR-compliant modeling and implementation needs to be performed by a specialist team equipped with suitable development tools.

New Design Environments with Available Solutions

Having gained a thorough knowledge of the new process, engineers next investigated a tool chain for implementing it. User groups met several times to make basic decisions, after which DENSO CREATE was able to set up a design environment based on dSPACE products and supplemented by commercially available solutions.
For the fundamental design of the function architecture, DENSO and DENSO CREATE used SystemDesk from dSPACE. MATLAB/Simulink/dSPACE RTI and AutoBox were used to verify control functions. SystemDesk was then used to derive the software architecture from the function architecture. At this point, EB tresos® from Elektrobit was used to configure the basic software – including RTE generation for connecting the application software to the basic software. The AUTOSAR-compatible application software was developed with TargetLink. First implementation information was added to the controller models, and then TargetLink converted them into efficient, AUTOSAR-compliant code.

> "SystemDesk gave ample support to the most crucial area, linking control algorithm development and the AUTOSAR Basic Software on a model basis, giving us an overview of our desired ECU design development process."
>
> *Masahiro Goto, DENSO CORPORATION*

Nobuhide Kobayashi, DENSO CREATE Inc.
Nobuhide Kobayashi is Project Manager of Software Development Department at DENSO CREATE in Aichi, Japan.

Yasuo Tatematsu, DENSO CREATE Inc.
Yasuo Tatematsu is Project Manager of Software Development Department at DENSO CREATE in Aichi, Japan.

Various Application Practices

DENSO CREATE carried out three subprojects in total, each with its own focus (figure 2), with applications from different vehicle domains.

- The first subproject focused on proving that the tools could be integrated seamlessly.
- The second subproject looked at how to reuse existing software. Information was gathered on the methodical development and integration of such software, for example, on the issue of whether sensor and actuator functions should be developed as software components.
- The third subproject studied the software in an existing climate conditioning ECU and converted it to AUTOSAR.

Looking Toward the Future

DENSO CREATE will continue to develop the process, with a special focus on making the tool chain seamless. This will cover issues such as automatically transforming specifications between process steps and a mechanism for synchronizing the various development tools in distributed development work. Last but not least, the performance of verification features such as the new SystemDesk Simulation Module and ways of integrating them into the process have to be studied in detail. This will optimize the process for future and complex tasks and further enhance its efficiency.

Nobuhide Kobayashi, Yasuo Tatematsu, DENSO CREATE Inc.
Masahiro Goto, DENSO CORPORATION

> "We are extremely grateful to dSPACE, not only for providing AUTOSAR-compliant tools, but also for working with us on-site in considering how to apply AUTOSAR appropriately. We are also grateful for the full training environment and for the efficient way in which tools were introduced."
>
> *Nobuhide Kobayashi, DENSO CREATE Inc.*

Figure 2: Three subprojects with applications from different vehicle domains.

		Trial #1 Vehicle Dynamics System	Trial #2 Display System (Dashboard)	Trial #3 Air Conditioning System
	Purpose of Trial	Perform architecture based development	Establish how to reuse legacy software	Establish how to adapt production program
Development Size	Number of SW-C/Runnables	6/11	26/299	41/141
Development Size	Number of DataElements/ Communication messages	42/5 (trans.), 5 (recv.)	56/3 (trans.), 18 (recv.)	59/5 (trans.), 14 (recv.)
Development Size	Basic software (BSW) module used	Mainly COM Stack (partially MCAL)	Mainly MCAL (legacy COM is used)	COM Stack, ECUM, MCAL etc.
Result	Learning AUTOSAR specification	■ VFB, RTE, COM, ECUM	■ VFB, RTE, COM, ECUM ■ MCAL	■ VFB, RTE, COM, ECUM ■ MCAL
Result	Set up engineering environment (Development environment, Design direction)	■ Resolve inconsistency among tools ■ Experience debug know-how (Understand BSW structure etc.)	■ Plan the direction how to implement legacy software into AUTOSAR structure ■ Experience debug know-how	■ Execute with experiences of trial #1 and #2 (Evaluate tool chain and design direction)
	Man-months	9.3 man-months	13.4 man-months	18.0 man-months