

Multi-Level System Integration based on AUTOSAR

Ulrich Freund

ETAS GmbH

Borsigstrasse 14

70469 Stuttgart

+49/711/896-61-416

ulrich.freund@etas.de

ABSTRACT

The design of distributed embedded real-time system is a challenging task. Besides solving the control-engineering issues, one has to consider real-time scheduling, reliability and production requirements w.r.t. production cost of the electronic control unit (ECU). This has a considerable impact on the employed software design techniques. These design techniques are well known in the automotive software industry, but are applied with different flavors at each vehicle manufacturer and their suppliers. This situation has changed considerably with the results of the AUTOSAR development partnership, which unifies the flavors of automotive software design.

Automotive software design is embedded in the so-called V-Cycle of embedded automotive system development[1]. It starts with the requirements analysis which results later on in a model of the control algorithm. The control algorithm is tested against a vehicle model and establishes the topmost level in system integration. The second system integration level is the adaptation of the control algorithm to be run on a rapid-prototyping system. The rapid-prototyping system is integrated into an existing E/E-architecture. The E/E-architecture consists of the ECUs connected by networks like CAN or FlexRay and gateways. Sensors- and actuators being used by several control algorithms are coupled to an ECU, which might propagate signals to other ECUs via a vehicle network. From the software point of view, the control-algorithm has now to respect real-time scheduling and the quantization of the sensor- and actuator signals, no matter whether these signals are generated on the rapid-prototyping system or exchanged via the bus with other ECUs.

Further development steps in the V-cycle are the software implementation, the ECU- and the network integration. The integrated ECUs and networks are tested against vehicle models running on Hardware-in-the-loop (HiL) systems. If this works fine, the ECUs are integrated in the real vehicle for calibration. The software implementation and the ECU integration are deeply influenced by AUTOSAR.

AUTOSAR is a development partnership of all stakeholders in the automotive software development (e.g. vehicle manufacturers and their suppliers) which unifies several software implementation techniques[2]. It describes a common ECU software architecture[3] consisting of configurable basic software modules (BSW), a runtime environment (RTE) and a software component description[4]. The software component description describes the interfaces for data-exchange as well as the access points for the RTE. The basic idea of AUTOSAR is to establish a so-called virtual functional bus (VFB)

consisting of interconnected software components which are later mapped to an E/E-architecture. Currently, the VFB structure of AUTOSAR software architectures is mainly driven by the next generation E/E-architectures. The VFB structure forms the third system integration level.

The next integration step into an AUTOSAR environment is the integration of the rapid-prototyping tested control algorithm to AUTOSAR software components. Since most VFB descriptions use fixed-point interfaces, the control algorithm has to be transformed to fixed-point arithmetic. If the control algorithm is modelled in tools like ASCET, this conversion can be achieved by code-generation. The same holds true for control algorithms already being used in E/E-architectures without an AUTOSAR software architecture. The VFB-description with the control-algorithms forms the fourth system integration level, which can be simulated with plant-models on a PC by tools like INTECRIO-VP.

The fifth system integration level is given by the mapping process as defined in the AUTOSAR methodology[5] and requires the configuration of the RTE and the BSW modules for a single ECU. At this integration level, one can perform HiL testing and calibration in the same way as for non-AUTOSAR systems.

Several evaluation projects, e.g. [6] and [7], have shown that the multi-level integration approach is feasible to guide the configuration capabilities of AUTOSAR software architectures.

Categories and Subject Descriptors

J.7 COMPUTERS IN OTHER SYSTEMS, Process control; Real time, D.1.2 Automatic Programming, D.1.m Miscellaneous

General Terms

Design

Keywords: Automatic Code-Generation, Automotive, Embedded Software, Rapid Prototyping

REFERENCES

- [1] Schäuffele, J., Zurawka, T.: Automotive Software Engineering, Vieweg Verlag, Wiesbaden, 2003
- [2] AUTOSAR: Technical Overview, www.autosar.org
- [3] AUTOSAR: The Layered Software Architecture, www.autosar.org
- [4] AUTOSAR: The Software Component Template, www.autosar.org
- [5] AUTOSAR: The Methodology, www.autosar.org
- [6] Freund, U., Möstel, A.: Model-based OEM Software Branding of AUTOSAR ECUs, 5th AUTOSAR Premium-Member Conference, Brussels 2006
- [7] Frey, P., Freund U., AUTOSAR Compliant Reengineering of an Engine Management System, In Proceedings of the 8th Stuttgart International Symposium "Automotive and Engine Technology", March 2008

Copyright is held by the author/owner(s).

ICSE'08, May 10–18, 2008, Leipzig, Germany.

ACM 978-1-60558-079-1/08/05.